ICT- 248577 C2POWER

D5.4 Version 2.0

Modelling and Simulation Tools to Evaluate Energy Efficiency Strategies in Cooperative Short-Range Networks – Final Version

Contractual Date of Delivery to the CEC:	December 2012
Actual Date of Delivery to the CEC:	January 2013
Editor:	CREATE-NET
Author(s):	Ayman Radwan, Firooz B. Saghezchi, Jonathan Rodriguez (IT) Riccardo Fedrizzi, Tinku Rasheed (CREATE-NET,) Jacek Kibilda, Michał Kucharzak (WCB), Peter Trapps (VBM)
Participant(s):	CREATE-NET, IT, WCB, VBM
Work-Package:	WP5
Est. person months:	24
Security:	PU
Nature:	R
Version:	1.0
Total number of pages:	105

Abstract: With the increase of energy demands in new generation wireless multi-standard mobile devices, there is a clear need for solutions to decrease the energy consumption of such devices to provide the promise of truly mobile experience, while offering advanced multimedia applications. The C2POWER project aims to investigate solutions to decrease the overall energy consumption of mobile devices, through exploiting two disruptive independent but complementary solutions, namely: short range cooperative communications in homogenous networks and cognitive automated handovers within heterogeneous environments. WP5 deliverables addresses the first of these two solutions. Moreover, this final deliverable presents the outcome of the C2POWER work related to the simulation of energy efficient strategies in cooperative short-range networks within homogeneous and heterogeneous networks. Firstly, a common simulation tool was developed in order to validate the proposed techniques, which is introduced in this deliverable in terms of the simulator architecture, implemented modules and simulated environments. The document also lists all the proposed cooperative algorithms which have been developed within C2POWER project. The details of each algorithm are introduced, along with the simulation results from the common system level simulator. The simulation environment was extended to cover short-range cooperative communications in heterogeneous environment (C2POWER scenario 3). The simulation results confirm the C2POWER concept of saving energy using short range low power cooperative technologies (such as WiMedia, or WiFi in adhoc mode). Different algorithms achieve different ranges of energy savings based on the cooperative criteria, considered technologies, and the context information used; which also affects the algorithm complexity. In general, C2POWER proposed cooperative algorithms showed gains in terms of energy savings in the region of up to 50%.

Keyword list: Energy savings, green communications, modeling, simulations, NS-2, NS-Miracle, cooperative communications, game theory, routing, relaying

Executive Summary

C2POWER main goal is reducing energy consumption of multi-standard mobile devices in heterogeneous environment. C2POWER project looks at two different approaches for improving the energy consumption in mobile devices: by means of cooperative short-range communication and employing energy efficient cognitive handovers. The first approach is considered in the scope of WP5 by investigating i) energy efficient cooperative strategies and policies in task T5.1, ii) energy efficient routing schemes in task T5.2 and iii) energy efficient relaying techniques in task T5.3. All previous three tasks of WP5 aim at reducing energy consumption in the multi-standard wireless devices considering multiple radio technologies. The remaining tasks in WP5 are basically, T5.4 on system level simulations of energy efficient techniques and strategies and T5.5 on the implementation of the most efficient proposed algorithm.

This deliverable is the last deliverable of WP5 and encloses the aspects related to the common simulation tool faced within the project. More precisely in this deliverable the following points are presented and discussed:

- Simulator design
- Simulator implementation
- Simulation scenarios
- Simulations results and discussion

In the first part of the deliverable we explain the choices made in the simulator design and the algorithm description that we implemented within the simulator. This part of the deliverable is intended to describe the general guidelines and the requirements for the C2POWER simulator implementation. Firstly, the baseline ideas for the C2POWER simulator architecture are described, then, the cooperative strategies implemented within the simulator are presented.

Having explained how the design of the simulator is structured, we explain the implementation of the common simulator by means of class diagrams and activity charts. The simulator is built upon the widely used NS2 simulator exploiting the NS-MIRACLE library extension to meet the requirement of multi-standard devices.

Within C2POWER two kind of scenario are considered for the evaluation of short-range cooperation namely scenario 1 (short range cooperation among homogeneous MTs) and scenario 3 (short range cooperation among heterogeneous MTs). The scenarios implemented are discussed together with the benchmark scenarios to use in order to validate the effectiveness of the proposed techniques.

Results emanating from the proposed cooperative short-range techniques show good gains in terms of energy efficiency which can amount up to 50%. However, energy gain is sensitive towards the scenario and simulation parameters. Those behaviors are collected and discussed in the conclusion of the deliverable in order to provide definite design guidelines towards candidate energy saving protocols for mobile devices for future networks.

Table of Contents

List of Figures

List of Tables

Document History

Date	Version	Status	Comments
11/09/2012	V 0.1	ToC	Created table of contents
11/10/2012	V0.3	Draft	First edit by editor
07/12/2012	V 0.5	Draft	Second edit by editor
17/12/2012	V 0.6	Draft	Third edit by editor
11/1/2013	V 0.8	Clean	Final edit by editor and submission to PTM
30/01/2013	V1.0	Clean	Revised version by PTM
31/01/2013	V2.0	Clean	Revised version by PC

List of Abbreviations and Acronyms

Term	Description
BS	Base Station
CBR	Constant Bit Rate
EAR	Energy Aware Routing
IDLE	Idle state of radio interface
IP	Internet Protocol layer
IE	Information Element
LLC	Logical link control layer
LR	Long Range
MT	Mobile Terminal
MAC	Medium Access Control layer
PHY	Physical layer
RAT	Radio Access Terminal
RDP	Route Discovery Packet
REER	Relay-based Energy Efficient Routing approach
RRP	Route Reply Packet
REP	Route Error Packet
RX	Receive state of radio interface
SR	Short Range
TX	Transmitting state of radio interface
WP	Work Package

1. Introduction

Future wireless mobile networks are expected to support high data rates and provide users with ubiquitous continuous connection, meaning being able to roam around freely while connected and not be bound to a single location. For mobile networks to support ubiquitous connectivity with high data rates, mobile devices are equipped with multi-standard interfaces (LTE, WiMAX, WiFi, Bluetooth, WiMedia, etc.). Multi-standard interfaces allow mobile devices to connect to various radio access technologies (RATs), but at the expense of higher energy consumption. Having multiple interfaces active simultaneously increases the energy consumption of mobile devices. Mobile terminals (MTs) depend on rechargeable batteries for their energy needs. Battery capacity is limited, especially with the limitation on the size of batteries installed in mobile devices. The evolution in battery industry is slow with no expectation of any breakthrough in the near future. There is a continuous growing gap between the energy requirement of mobile devices, to support advanced wireless applications, and the progress in the battery industry.

Hence, the expected advances in mobile networking applications are limited by their increasing energy requirements. There is a clear need for new disruptive strategies to address the aspects of energy efficiency in the operation of MTs, especially in the wireless communication part.

The C2POWER project aims to reduce the energy consumption at the mobile device by exploiting the opportunity to operate in multi-standard network based on using the cooperative paradigm together with context information. In Work Package 5, the project targets the goal of energy saving through short-range cooperation among different multi-standard MTs. More precisely, the goals of this work package addresses the following:

- Development of cooperative strategies and policies between nodes belonging to a cluster to optimize energy-efficiency.
- Development of energy-efficient protocols and routing schemes for cooperative networks.
- Development of a system level simulation environment to evaluate the realistic energy gain of cooperative strategies according to defined performance metrics and benchmarks.

Within this work package various algorithms were developed and will be presented in this deliverable. Developed algorithms comprises an energy efficient cluster technique, a node selection mechanism, cooperative relaying exploiting 2 hop cooperation and finally two routing schemes exploiting cooperation over more than two-hop routes. Moreover a common system-level simulator platform has been implemented to test the performance of the proposed techniques and study the algorithms behaviors in various scenarios.

The rest of the deliverable is organized as follows:

In section 2 we present the design and the requirements of the simulator. We first explain and motivate the basic idea behind the common framework as the enabler for testing energy efficiency mechanisms. Secondly we present the algorithms implemented in the simulator from a theoretical point of view. The first technique is the energy-efficient clustering technique. This algorithm is responsible for creating clusters within the C2POWER scenario to allow cooperation and is exploited by different algorithms. The second technique presented is a node selection algorithm based on a cooperative game where a combinatorial optimization method is used in order to find the best matching of source nodes and their corresponding relays. Then the energy efficient multi-radio cooperative relaying is presented which is based on the fact that multi-radio relaying in wireless networks can greatly increase capacity and performance by exploiting the best available links and network interfaces to a destination node. The above two techniques allow to save energy employing two-hop cooperation (i.e. the source MT transmits to a relay MT which will in turn transmit to the BS). C2POWER also exploit more than two hop cooperation employing routing techniques. Within C2POWER we will present two kind of routing techniques for energy saving: firstly energy aware routing which is based on an on-demand routing strategy where a route discovery is initiated when a MT in the network has a packet to send to a destination (BS); secondly a light relaying-based routing technique where routing decisions are taken on the fly at every MT and considering information about neighbours of neighbours.

In section 3 we present the implementation of the common simulator exploiting the NS-2 simulator. In order to simulate C2POWER scenarios, we need multi-interface devices. Because of this we used the NS-MIRACLE library extension for NS-2. In this section we explain the important classes, methods and messages which are involved in order to meet the concept introduced in section 2.

In order to test developed algorithms several parameters need to be considered in order to create a scenario to perform simulations. In section 4 we present the considerations that were taken in order to set up suitable simulation environment for both C2POWER scenario one and three.

In section 5 we present and discuss the results coming from the common simulator for each of the algorithm implemented. Firstly C2POWER scenario one is investigated in section 5.1 while in section 5.2 C2POWER scenario 3 results are presented. Finally we conclude the deliverable in section 6.

2. Simulator Design

In this section, we present the simulator design and the description of the algorithms, which were implemented on the C2POWER simulator. This part of the deliverable is intended to describe the general guidelines and the requirements for the C2POWER simulator implementation. In Section 2.1, the C2Power simulator architecture is described, while in Sections 2.2 to 2.6 the implementation of selected C2POWER algorithms are presented.

2.1 C2Power Simulator Design

The simulator consists of the simulation environment setup, simulator core and blocks of cooperation algorithms (derived from methods presented in [1]). In particular, the ns2 simulator has the following features:

- C2POWER layer along with adaptation layer that intercepts C2POWER control frames (encapsulated payload of technology-specific beacon frames) and either forwards data frames upwards or relays them to another radio interface (**skeleton**)
- Control frames to create neighbours' list, manage the cooperative cluster (**clustering**), perform node selection (**node selection**), and manage any cooperative algorithm (cooperative routing or relaying).
- A novel routing layer for the short range RAT (**routing**) and relaying (**relaying**). For sake of easiness, it is possible to implement the routing as part of the C2POWER layer, and then use a “no-routing” routing layer on the short range RAT, which lets the MAC specify the next hop.

The different features are integrated under the common simulator framework. The functional elements of the simulation platform and their relations are shown in Figure 2-1, where

1) The simulation environment defines the scenario setup, along with radio environment settings, such as the number of nodes, mobility pattern, traffic model, propagation conditions and user mobility.
2) The simulator core defines the common functionalities related with interaction between the C2POWER functions and network simulator (ns-miracle), as well as common framework for all C2POWER algorithms, e.g. addressing, interfaces, user settings, cooperation management.
3) C2POWER cooperative blocks define functionalities dedicated to the specific cooperative function (cooperation protocol, clustering, node selection, cooperative relaying, energy-saving routing and relay-based routing).

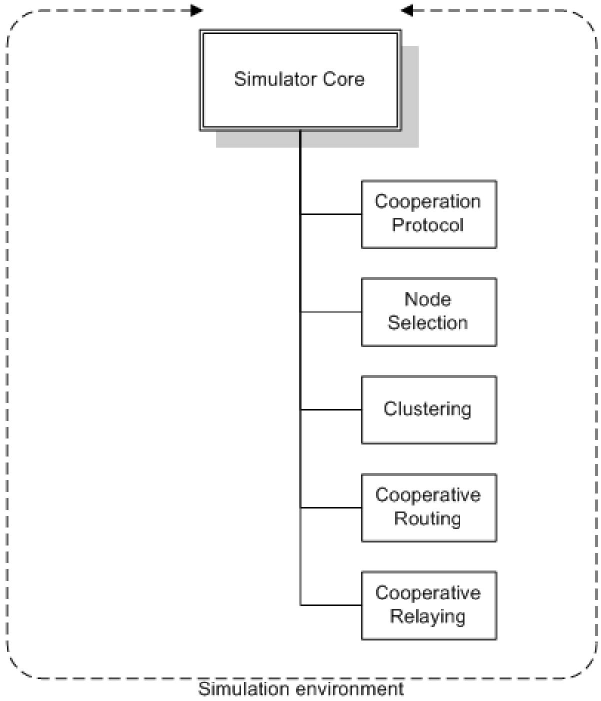


Figure 2-1. C2POWER simulation platform framework

2.1.1 C2POWER layer

From the above described simulation framework, C2POWER application can be extracted and discussed separately. The C2POWER application is a complete and stand-alone application that enables energy saving in multi-standard mobile nodes. The energy saving gain is realized through cooperative networking, in particular through the realization of cooperative techniques presented in [1].

The C2POWER application is envisioned as part of the so called C2POWER mobile device, where it integrates with the device's protocol stack as presented in Figure 2-2. Such a modular representation makes it easier to analyse and evaluate interoperability in C2POWER network devices, in particular the interoperability of C2POWER application functions with the existing protocol suite. From the figure, one can recognize the fact that in this particular setup C2POWER interconnects application layer with the networking (IP) layer, which characterizes most of today's heterogeneous networking solutions. As it can be seen, C2POWER itself is independent from the underlying radio technology. However, C2POWER functionality is very much related to the MAC layer in the sense that it requires that C2POWER understands what type of frame the device has received, and whether it is a data frame or a beacon frame, which determines the further processing of the frame. The above mentioned functions are performed by C2POWER adaptation layer, which is able to intercept the C2POWER frames and data frames along with their data rate and SNR. Thus, the adaptation layer separates C2POWER layer from the knowledge of underlying radio technologies specifics. In that sense, the adaptation layer is required to recognize specific radio interface and manage the transportation of C2POWER frames, also in the cases of no support for beaconing and IE congestions.

Apart from the main application with its adaptation layer, there is also a plug-in extension of C2POWER. The purpose of the plug-in is to extract various items of context information from other layers of the network device protocol stack. The context information is then made available to C2POWER, which utilizes it to decide about cooperation. For example, context information extracted from the lower layers is composed of SNR or data rate of the received packet, or from higher layers the monetary cost of service. As noted previously, the C2POWER application is independent from the underlying RAT, thus as a consequence C2POWER is capable of managing and having an interface to multiple radio technologies embedded in one device (see Figure 2-3). Such knowledge enables C2POWER to effectively employ and manage cooperative schemes, e.g. multi-radio cooperative relaying or multi-radio cooperative scanning. As required by the 802.11 standard [2], the underlying IEEE radio interfaces shall appear to the upper layers the same as IEEE 802 LAN standard. The presented layered model fulfils the context awareness and cooperative functions proposed in the C2POWER architecture [3].

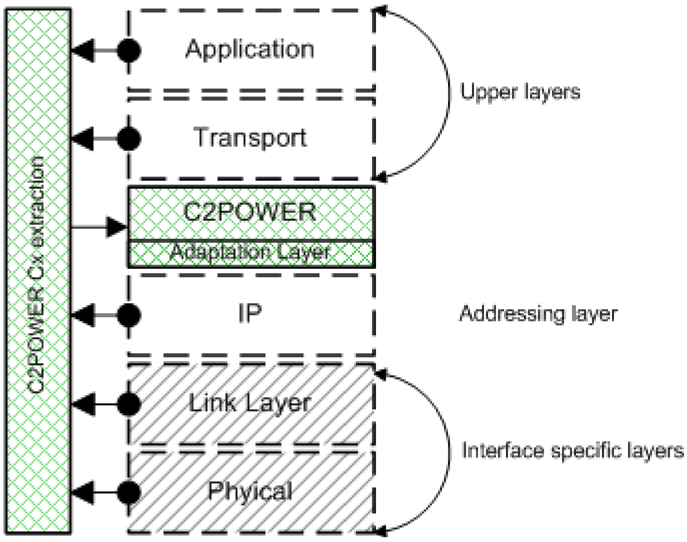


Figure 2-2. Protocol stack with C2POWER layer and plug-in

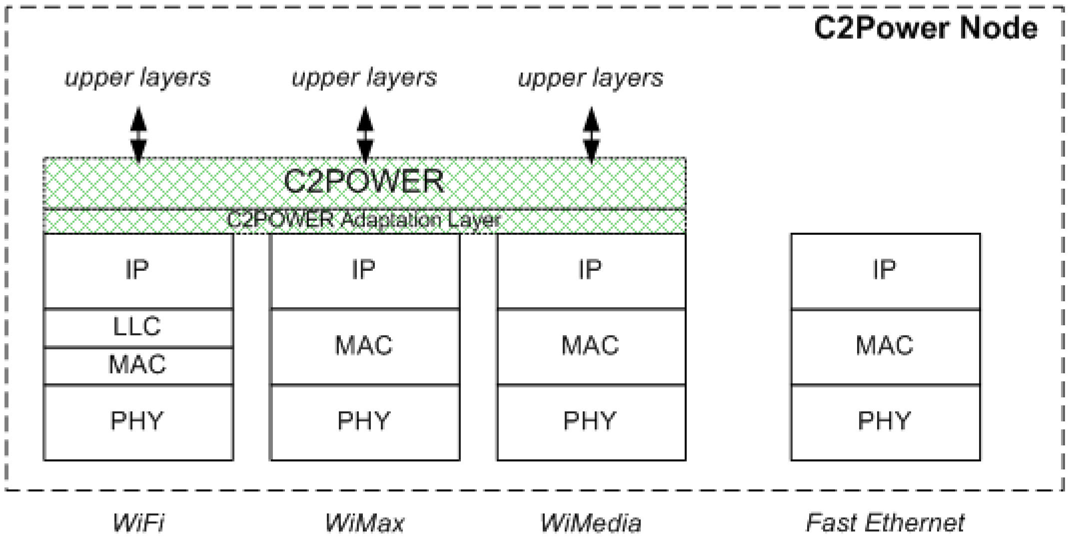


Figure 2-3. C2POWER network node with multiple radio interfaces

Based on the presented layered model, we now analyse the frame processing scheme if C2POWER is available in a particular device. At this point, we have to point out that each layer of protocol stack encapsulates the out-going packet with its own header, thereby connecting to its peer layer on other devices, which can subsequently expand the received packet and read the included information. Of course, the same scheme applies to C2POWER. We have to note however, that C2POWER does not generate its own control frames, instead it utilizes a vendor specific payload in the beacon frames—

already available in the technology. Although it does not destroy the general concept, it certainly breaks the typical rule of including the headers in a strict subsequent manner. Figure 2-4 presents a frame processing scheme for a device equipped with C2POWER capabilities.

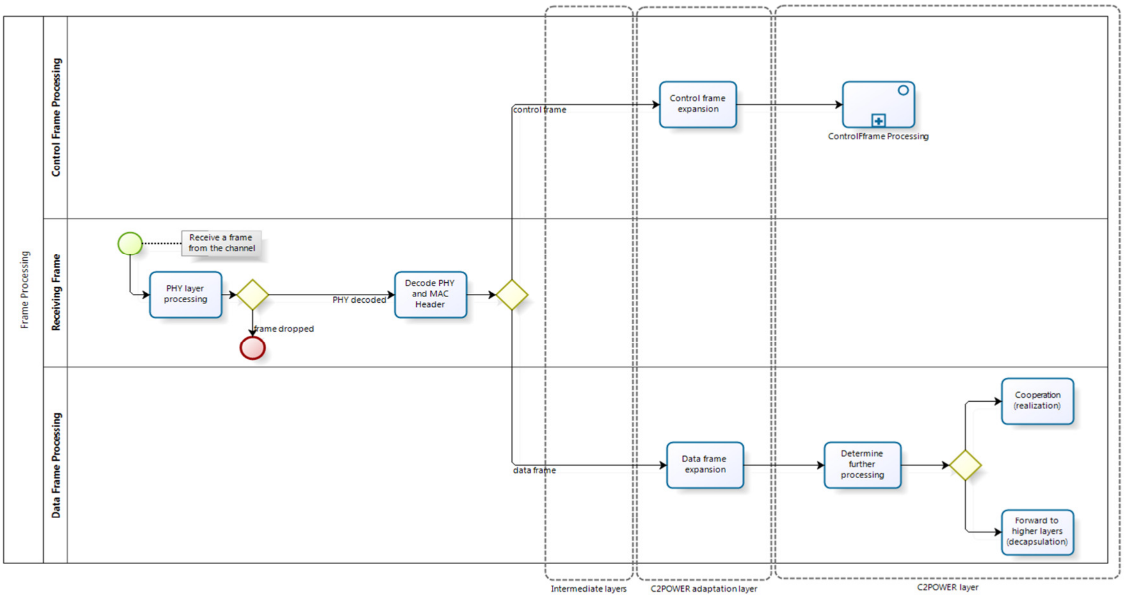


Figure 2-4. Frame processing within a C2POWER network device

The figure illustrates a set of key processing steps in a C2POWER network device, when a new frame is received from the channel. First, the frame is decoded in the PHY and MAC layers, which is outside of the C2POWER application. At this stage the frame can be dropped due to RSSI below receiving threshold, unrecognized type or unfixable errors. As an alternative, one can utilize Maximum Ratio Combining, which requires the incorrectly received frames to be stored and combined with others to benefit from multiplexing gain. At that point the frame is passed to C2POWER adaptation layer (we omit the processing in intermediate layers as it does not influence the C2POWER information), which determines whether it is a C2POWER control or data frame. This is done based on the availability of C2POWER header. If C2POWER header exists, thus control frame is assumed. The C2POWER layer is interested in capturing C2POWER control frames and relayed data frames. C2POWER layer needs to keep track of incoming data frames to be relayed. In the case that the frame does not contain C2POWER header nor is it to be relayed, then it is passed to upper layers for further processing. The captured relayed data frames are not passed to higher layers, but they are passed by C2POWER layer to appropriate radio interfaces for the next hop. In case the C2POWER application receives a control frame, it searches for a C2POWER encapsulated header. In its header, each C2POWER control frame contains information about its purpose. The main available frame type in C2POWER is <C2PowerBeacon>.

Figure 2-5 presents <C2PowerBeacon> processing, during which current state information about neighbour nodes in the vicinity are updated.

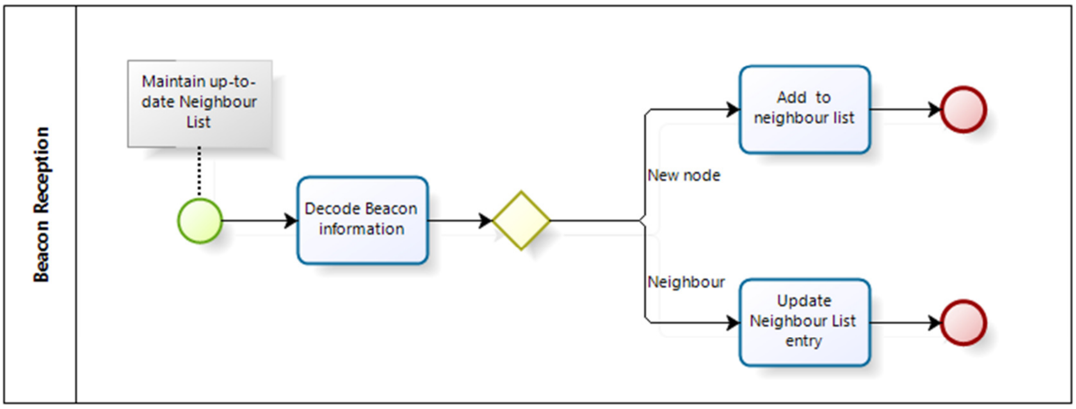


Figure 2-5. C2POWER Beacon reception and processing with follow up steps

In case the beacon is first time received, the sender of the beacon is added to a neighbour list. Every further beacon reception triggers an update of entries in the neighbour list. The information carried by the <C2PowerBeacon> may indicate available underlying technologies, channel conditions or currently supported data rates.

By sending the beacons, every node acknowledges its willingness to cooperation; thereby every C2POWER node with non-empty neighbour list may select any of its neighbours in order to perform routing or relaying. All C2POWER nodes in the one hop neighbourhood form a cluster. It is worth noting that, the neighbour lists (and thus, clusters) may be disjoint from each other for each node.

<C2PowerBeacons> are transmitted periodically, according to beacon frame transmission of the underlying protocol, to maintain the neighbour list up-to-date. Update Neighbour List entry (as presented in Figure 2-5) is also responsible for removing out-of-date entries. The neighbour is removed from the list after exceeding a given time threshold without receiving any beacon from the specific node.

Having understood the functionality and interaction between different functions forecasted for the C2POWER application, in the following subsections we describe the C2POWER algorithms designed to reduce the energy consumption of mobile devices through short-range cooperation.

2.2 Energy Efficient Clustering Technique

C2POWER cooperation protocol is realized through the exchange of C2POWER Distributed Management frames between neighbouring nodes. In particular, when possible, C2POWER is envisioned to utilize beacon frames available in specific radio technologies. This protocol is utilized to enable node discovery functionality, cluster formation, exchange of context information and cooperative strategy realization. In this subsection, we provide a specification of C2POWER Distributed Management frame, C2POWER cluster formation, as well as information distribution and resource reservation method for cooperative strategies.

2.2.1 C2POWER Distributed Management Frame

Neighbour discovery is the first step in cooperative networks to enable cooperative behaviours. For that purpose, mobile nodes are required to exchange information about their willingness to cooperate. This information is then used to form cooperative clusters, where cooperative strategies are further employed and cooperating mobile nodes are able to enjoy increased utility coming from cooperation. In C2POWER, we envision that nodes, which are willing to cooperate, advertise through exploitation of the application specific payload of beacon frames available in the underlying technology [1]. The reasons, which justify the choice of beacon frames, can be summarized as follows:

- Short-range technologies considered provide beaconing mechanism with application specific payload.
- Beacons are exchanged periodically at the priority of management frames of the specific radio technology.
- Beacons are not terminated at the MAC layer; due to the availability of different Information Elements (IEs), they are further passed to higher layers[1].

Respectively for different short-range systems, beacon frames contain in their payload various types of IEs. The IEs contain device and operation specific data that is being exchanged between different network entities. Each beacon frame consists of a number of IEs (some of them optional), with each class of IEs dedicated to a particular function. However, there is a particular class of IEs, namely vendor-specific IEs, that enable the manufacturer or application developer to customize its payload. Each of the short-range technologies considered in C2POWER provide such vendor-specific IEs:

- Vendor-specific IE in 802.11 [2], which enables utilization of *n* payload octets, $\{n: (n > 3) \wedge (n \leq 255)\}$.
- Application Specific IE in WiMedia [4], which enables utilization of *n* payload octets, $\{n:(n > 4) \wedge (n \leq 320\text{-}length\_of\_other\_IEs)\}$.

The position of the IE in a beacon frame is indicated by the Element ID (IEs are grouped in ascending order). Furthermore, each vendor-specific IEs has its own vendor ID[2], which is reserved with a particular standardization body. It is envisioned that C2POWER would use the same vendor ID regardless of the underlying technology[3], so that C2POWER ID could be recognized by the C2POWER application, once extracted by the MAC layer from the beacon and properly processed. Based on the above mentioned IEs, the C2POWER specific IE structure looks as shown in Table 2-1 and consists of control information, vendor ID and C2POWER frame.

[1] This property enables implementation of C2POWER control frames without the need to modify underlying radio technologies.

[2] In WiMedia vendor ID is named Specifier ID and it consists of 2Bytes. On contrary in 802.11 it is OUI (Organization Unique Identifier) and it consists of 3Bytes.

[3] Thus, vendor ID would be adapted to a specific technology for example by adding as a prefix FF octets.

Table 2-1. C2POWER IE

(2 + n) B		
1 B	1 B	n B**
Information Element ID*	Length	C2POWER frame

*If the frame is transmitted through a radio interface which does not support beacon frames, and the information needs to be conveyed over data frame, the Information Element ID field shall be omitted
**Dependent on the underlying technology, where *n* indicates the C2POWER frame length

The proposed Information Element contains a C2POWER Distributed Management frame, which is used as a cooperation-enabling mechanism between C2POWER enabled mobile nodes. The C2POWER frame is utilized for C2POWER distributed management purposes, for example:

- cluster formation;
- cluster maintenance;
- relay selection;
- strategy negotiation;
- call for cooperation.

Table 2-2. Proposed C2Power Distributed Management frame structure

2 B				8 B	n B
C2POWER frame header				Destination address	C2POWER frame payload
3 bits	1 bit	6 bits	6 bits	64 bits	8*n bits
Message type	Addressing type	Supported RATs	Supported cooperative strategies	MAC Address	Defined for corresponding cooperation strategy

The structure of the frame reflects the variety of purposes it is used for. Table 2-2 presents the structure of the C2POWER frame. The C2POWER frame consists of a header which defines: the message type, addressing type (either 1 for Unicast or 0 for Multicast), supported RATs (the technologies which are available to the mobile node) and supported cooperative strategies (the cooperative strategies available to the user of the mobile node). After the header, the C2POWER frame contains a Destination address field, which shall utilize the MAC address of the destination device or the multicast MAC address (agreed by the nodes in the cluster) of the cluster or the broadcast address. The rest of the C2POWER frame contains a payload specific to each of the cooperative strategies that are employed. The structure of the specific elements is specified in subsequent subsections of the deliverable.

C2POWER frame yields not only the exchange of information between neighbouring devices, but also enables creation of more complex networking forms such as clusters that enable application of energy saving schemes in cooperative manner. In the following sections, we will provide a description where C2POWER frame is utilized for these various purposes. Thus to distinguish the usage purposes, from now on, we will refer to C2POWER Distributed Management frame utilized for cluster maintenance, as C2POWER beacon and we will mark it with pseudo-code alias <C2PowerBeacon>.

C2POWER frame adaptation

Apart from the utilization of application specific payload of beacon frames in WiMedia and WiFi, there are situations where it might be beneficial to change the transport scheme of C2POWER frames based on the characteristics of underlying technology. First of all such situation occurs, if the application specific payload of beacon frame is already utilized by other applications or by excessive inclusion of Information Element (IE congestion). Second, C2POWER application might be utilized by a technology, which does not support beacon frame exchange, for example Bluetooth[4]. In both situations, C2POWER application is required to perform frame adaptation, so that C2POWER frame is exchanged through the payload of data frames. In this particular situation, C2POWER layer is required to provide mechanisms that enable users to choose which short-range technology is utilized to exchange C2POWER frames (by default WiMedia or 802.11 in such precedence are assumed if available). The adaptation function shall insert C2POWER frames into appropriate transport frames. If applicable, the adaptation function should also reserve data channel as well as read incoming C2POWER frames from the same technology. In case of IE congestion, C2POWER enabled device should either refrain from distributing C2POWER frames (there is no space in each of the consecutive beacon frames to include user specific data) or include a point in the beacon, that indicates that C2POWER Distributed Management frame will be included into the data frame payload. The pointer frame should have a separate Message Type and its value should replace the C2POWER frame payload.

2.2.2 C2POWER cluster

According to the derived requirements and provisioned algorithm for C2POWER, mobile nodes self organize into one-hop clusters to exploit the energy saving capabilities of short-range cooperation. The cooperation actually occurs whenever nodes determine that energy saving is feasible. In general, it provides an energy saving benefit without jeopardizing any required QoS. Nodes within the C2Power cooperative cluster communicate with each other using a common short-range interface. Any node, equipped with an enabled C2POWER application is required to:

- discover<C2PowerBeacon> distributed by neighbouring nodes with C2POWER application and;
- broadcast its own <C2PowerBeacon> to allow other nodes to read its C2POWER information.

Once this basic functionality is running, mobile nodes become aware of the mobile devices in their vicinity with enabled C2POWER application. Then, each node maintains its neighbourhood membership list, which contains information on the node addresses and their advertised C2POWER information. This information is used to form clusters based on the node capabilities, which can be further utilized for energy saving cooperative strategies. Figure 2-6 depicts such a network of nodes with enabled C2POWER capabilities and how they can organize themselves into clusters, for example based on the strategy they are employing (cooperative sensing).

[4] In fact, Bluetooth supports beacon frame exchange **Error! Reference source not found.**, however the concept is quite different from the assumptions taken in the document.

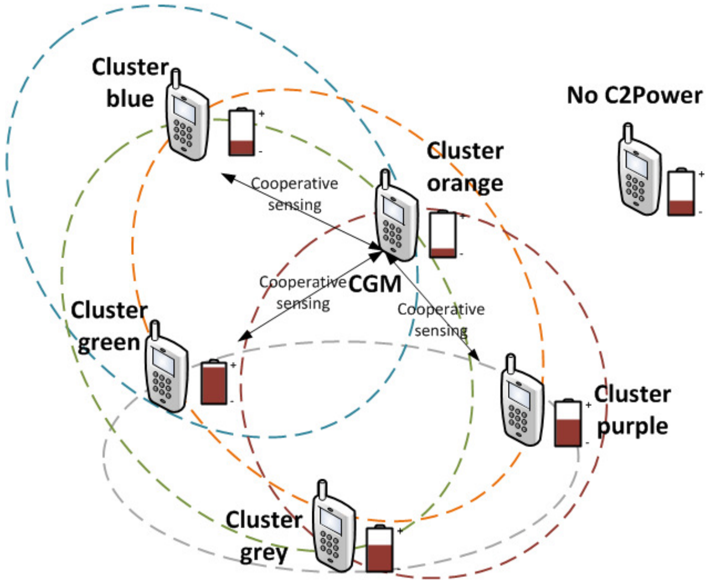


Figure 2-6. Network of mobile nodes with enabled C2POWER application

After the node collects a number of beacons corresponding to *MobilityThreshold* (*MobilityThreshold* is further explained later in the document) of beacons from each neighbour, it can select a cooperative strategy for the cluster and become a Cooperative Game Master (CGM). CGM can be seen as a node that is willing to exploit cooperative opportunities to save battery power.

Nodes with an active C2POWER application and sending <C2PowerBeacon> are willing to join a C2POWER cluster in the vicinity. Clusters are formed by nodes which receive *MobilityThreshold* number of a neighbour's <C2PowerBeacon> and the information contained in those frames matches the receiving node's capabilities or current user profile (e.g. supported cooperative protocols, available radio interfaces and operator supported payment plan). A cluster is formed once a node receives a number of valid <C2PowerBeacon> frames from at least one neighbouring node. The solution does not restrict the number of nodes in a cluster as long as they are close (one-hop) neighbours of the CGM. However, it is worth pointing out that the number of nodes in the cluster impacts the complexity of cooperative strategy calculations.

Two important thresholds (timers) are required to be defined that are integral part of the C2POWER clustering. The first threshold is *MobilityThreshold*, which reflects the minimum number of beacons received from one neighbour, before it is associated as a cluster member. The second threshold is *StabilityThreshold*, which reflects the maximum number of lost beacons before a cluster member is removed from the cluster.

Figure 2-7 depicts the activity diagram for the C2POWER cluster association and maintenance process. If a node has received a beacon from a neighbouring device, it checks whether it is a new neighbour. If it is from a new neighbour, it adds the node to the neighbour list. If the beacon is from a neighbour which is already on the neighbour list, it checks whether it has received *MobilityThreshold* number of beacons from this neighbour. In such a case, the neighbour becomes a member of the node's cluster for the supported cooperative protocol(s). The <C2PowerBeacon>received from a particular member are counted up to *StabilityThreshold*. Alternatively a node may not receive a <C2PowerBeacon> from a member for a certain period of time[5], then the node starts to decrement a counter corresponding to the number of received <C2PowerBeacon> frames. If the number falls below the first threshold, the member is no longer in the cluster. The application of the C2POWER clustering mechanism may lead to one of three cases:

- If the node is already a cluster member, its number of received beacons is incremented only to *StabilityThreshold* (*StabilityThreshold* ≥ *MobilityThreshold*);
- If the node is already a cluster member and beacon counter falls below *MobilityRestriction*, it is no longer considered part of the cluster;
- If the number of received beacons from a certain node falls to zero, it is excluded from the neighbour list.

The *MobilityThreshold* number of beacons to be collected reflects the instability of the mobile environment.

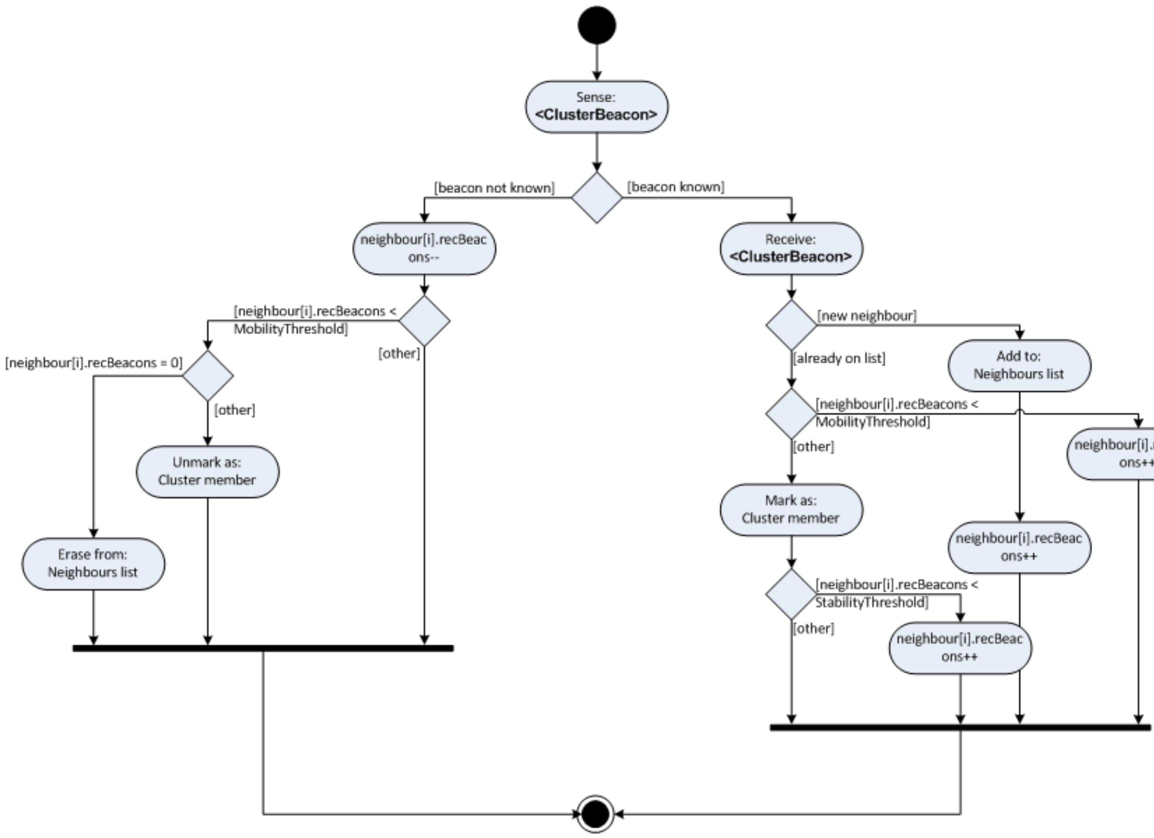


Figure 2-7. C2Power cluster association and maintenance

Additionally, each of the C2POWER nodes maintains a blacklist, of nodes which cannot be admitted to the cluster, due to “malicious behaviour” or due to technical impairments of the neighbour. The cellular network operator may indicate malicious behaviour of devices to users

[5] In WiMedia this period of time is defined by super frame timing and similarly such a scheme could be adopted for other TDMA-based technologies. In the case of non-TDMA technology, a maximum time between two consecutive beacons is implementation specific.

2.3 Energy efficient node selection Module

In this section, we present the cooperative relay selection game, introduced in D5.1 [1], using flowcharts and pseudo-codes. The cooperative game approach assumes that a cooperative cluster has already been formed and matches sources and relays in a way that the total energy saving of the cluster is maximized. The game assigns maximum one relay to each source node—sources communicate with the BS either over a direct link or over a cooperative two-hop link. The proposed approach also offers “core” solution for distributing the saved energy among the players (sources and relays) in a way that every player is satisfied and willing to cooperation. It also records the contribution of each MT (to the achieved energy saving) in a database to keep track of cooperative nodes for preventing cheating behaviour, billing purpose, etc.

In the following, we provide the algorithms that an MT and the BS needs to perform in the cooperative relay selection game. The algorithms cover the required activities from optimum relay-source matching to recording the share of each MT (from the saved energy) in the data base.

Cooperative Relay Selection Game

BS runs the node selection algorithm as the MTs suffer from the limited energy and the restricted computation power. It collects the energy cost (Joule/bit) for both short range and long range links and runs two linear programming (LP) problems: first problem determines the best relay-source matching (which maximizes the energy saving), while the second one determines the share of each MT from the saved energy. We assume that, for each link, there is an SINR threshold at the receiver and the transmitter adjusts its transmission power to deliver a signal with acceptable SINR at the receiver; we will use the term "power budget" to indicate this transmission power. The BS figures out the cost of a link by dividing the power budget of the link by the data rate of the same link. We assume that MTs have already formed a cooperative cluster, through exchanging some context information. Furthermore, we consider only the two-hop relay selection game (each source is looking for a single relay to reach the BS) in the uplink direction as our main concern is saving energy for MTs. Table 2-3 summarizes the definition of the abbreviations we use in the flowcharts or pseudo-codes in the rest of this section.

Table 2-3. Definition of abbreviations

Abbreviation	Definition
BLP	Binary Linear Programming
CSRCS	Cooperative Short Range Communication Session
ID	Identification
REQ	REQuest

The cooperative relay selection game algorithm has two parts: BS performs one part, and MTs perform the other part. Figure 2-8 and Figure 2-9 show the pseudo-codes for these two parts, respectively.

```
START
OBTAIN the power budget of the long range link to the BS
OBTAIN the data rate of the long range link to the BS
FOR any MT in the cooperative cluster
OBTAIN the power budget of the short range link to the MT
OBTAIN the data rate of the short range link to the MT
ENDFOR
SEND the power budgets and the date rates of the long and short range links to the BS
IF do not need to establish a CSRCS THEN
IF do not receive cooperation command from the BS DO
WAIT for K seconds
    GO TO START
ELSE
            RELAY the source node specified by the BS
    Until you receive CSRCS_END from BS
GO TO START
ELSE
SEND CSRCS_REQ to the BS
GET the ID of the relay from the BS
ESTABLISH a CSRCS with the specified relay
WHILE the CSRCS has not finished AND you have not received CSRCS_END from BS DO
IF have not received any command from the BS to change your relay THEN
KEEP your current CSRCS
ELSE
CLOSE your current CSRCS
SEND a CSRCS_END message to the BS
ESTABLISH a new CSRCS with the new relay introduced by the BS
ENDIF
ENDWHILE
SEND CSRCS_END message to the BS
ENDIF
GO TO START
END
```

Figure 2-8. Pseudo-code for the part of node selection algorithm performed by MTs

```
START
ON RECEIVE context information from any MT DO
STORE the context information and ID of MT
ENDDO
ON RECEIVE CSRCS_END from a source node DO
SEND CSRCS_END message to the relay
OBTAIN the core solution of the cooperative node selection game
UPDATE the energy credits for the cooperating nodes
ENDDO
ON RECEIVE CSRCS_REQ from a source node DO
     SOLVE the node selection BLP problem
DETERMINE the best source-relay matching
FOR any old matching that is dropped DO
SEND CSRCS_END message to the source
ENDFOR
FOR any new matching DO
COMMAND the relay to cooperate with the source
COMMAND the source to establish CSRCS with the relay
COUNT total number of bits relayed during the CSRCS
ENDFOR
ENDDO
GO TO START
END
```

Figure 2-9. Pseudo-code for the part of node selection algorithm performed by BS

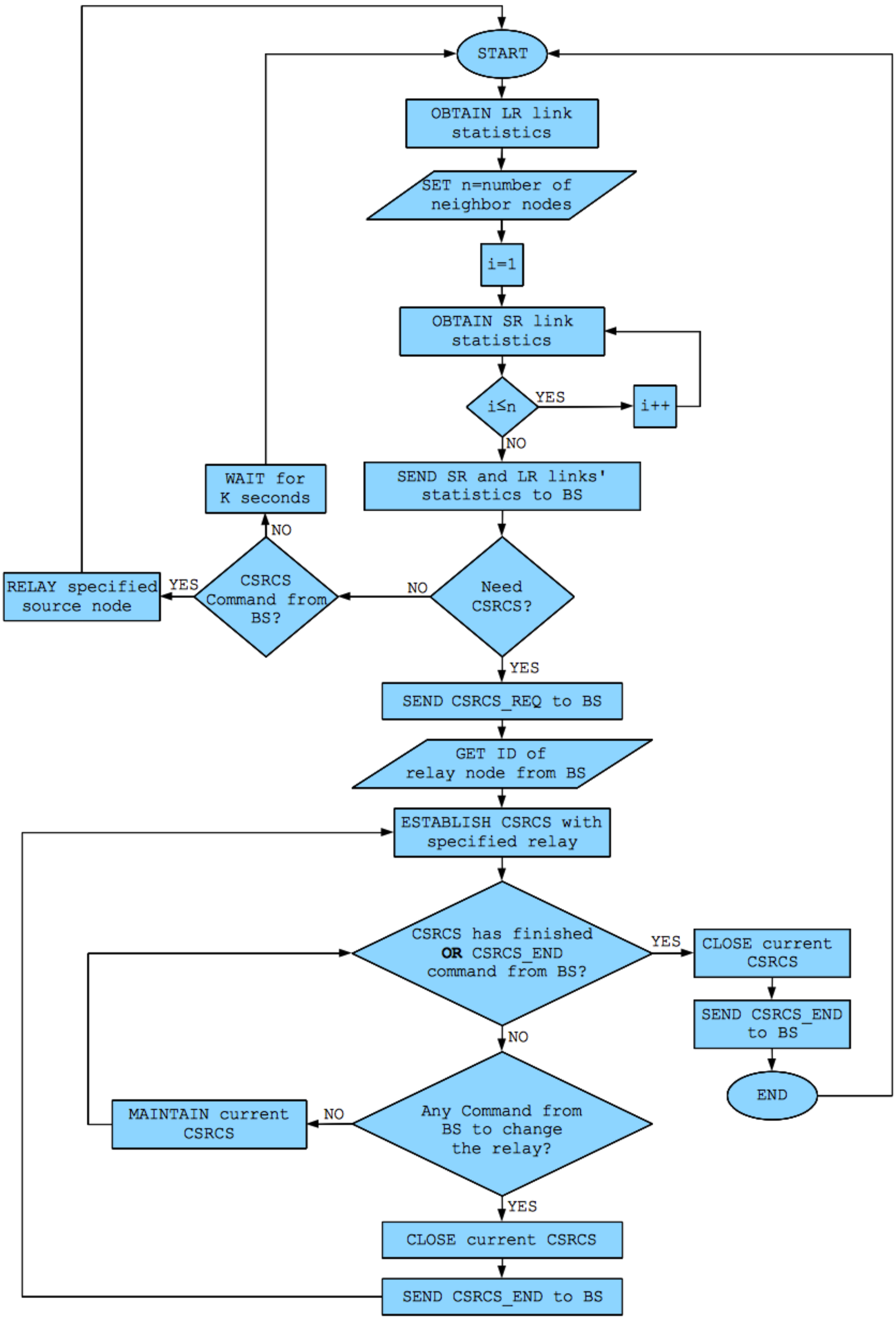


Figure 2-10. Flowchart of the part of the node selection algorithm that is performed by MTs

Figure 2-10 presents the flowchart for that part of the relay selection game algorithm that is performed by MTs. The part of the algorithm that is performed by the BS has three "ON RECEIVE" clauses in the pseudo-code. Each of these clauses means a different reaction by the BS. Figure 2-11 shows the state diagram of the node selection algorithm for the BS. There is only one state for the BS, which is the IDLE state. Whenever the BS receives context, cooperative short range communication session request (CSRCS_REQ), or cooperative short range communication session end (CSRCS_END) from any MT, it departs from the IDLE sate, performs the corresponding activity, and then returns back to the IDLE state.

In Figure 2-12 to Figure 2-14, we present three flowcharts, each of which corresponds to one of the three activities indicated by the rectangles in the state diagram of Figure 2-11: Saving context, Terminating CSRCS, and Initiating CSRCS. Figure 2-12 shows the algorithm that is performed by the BS while receiving context information from a MT; Figure 2-13 shows the algorithm that is followed by the BS after receiving a CSRCS_END message from a C2POWER MT; and finally, Figure 2-14 shows the part of the relay selection game algorithm that is performed by the BS after receiving a CSRCS_REQ from an MT.

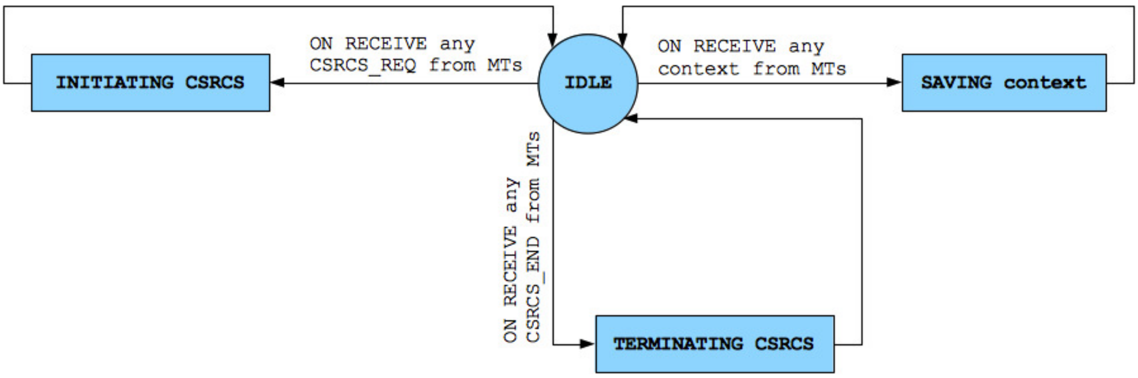


Figure 2-11. State diagram of the node selection algorithm for the BS

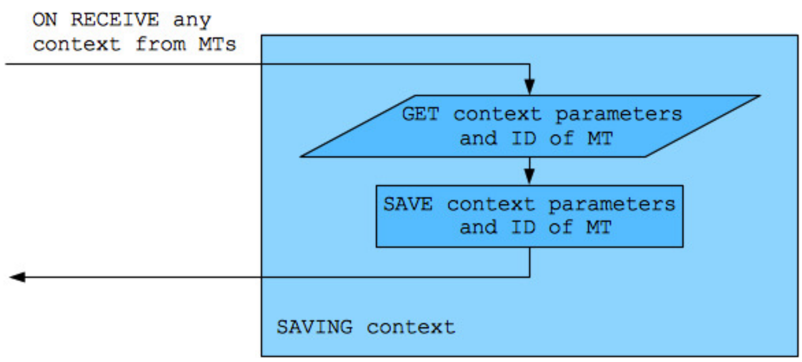


Figure 2-12. Flowchart for the part of the relay selection game algorithm that is performed by the BS when it receives context information from an MT

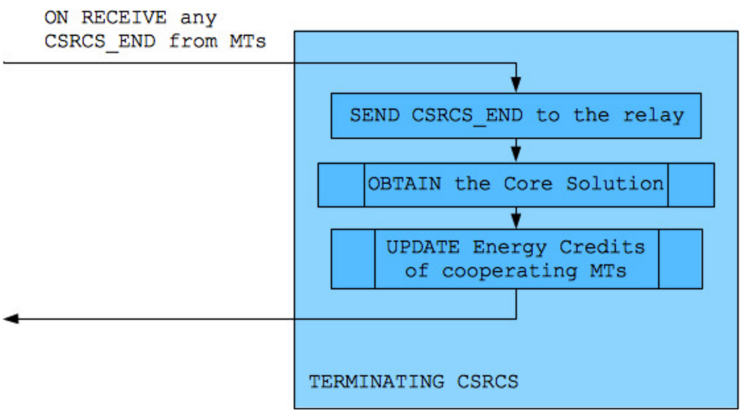


Figure 2-13. Flowchart for the part of the node selection game algorithm that is performed by the BS when it receives a CSRCS_END message from an MT

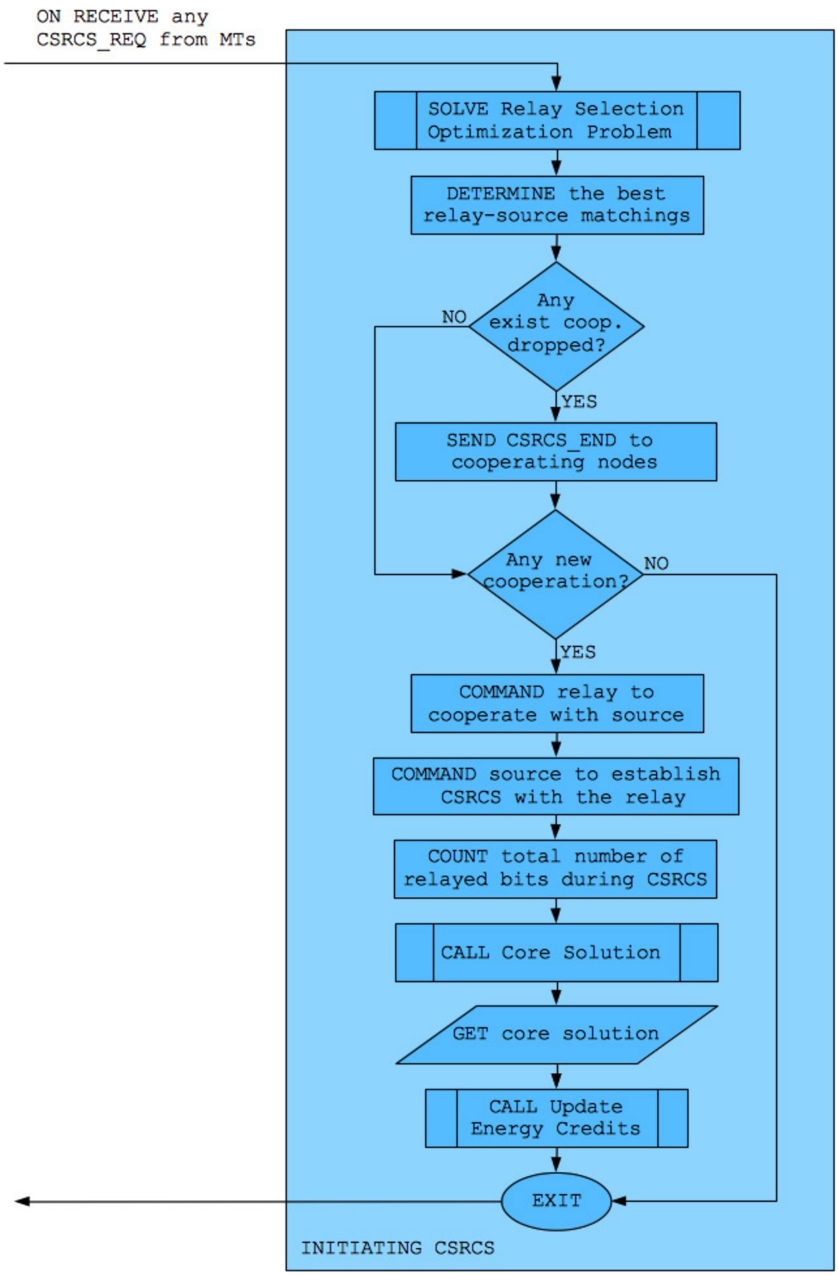


Figure 2-14. Flowchart for the part of the relay selection game algorithm that is performed by the BS when it receives a CSRCS_REQ message from an MT

Relay Selection Optimization Problem

Having collected power budgets as well as data rates of all (short/long range) links, the BS calculates the following terms: the cost of communication (Joule/bit) for each link (by dividing the power budget of the link by the data rate of the same link); the cost of relaying for each relay node, which is composed of two terms: the cost of receiving through short range link and the cost of transmitting the received data to BS over long range link; the value of each relay node for each of the source nodes, calculated by subtracting the cost of short range link from the cost of direct link (the value of a relay node to a source node is the amount of potential energy saving that the relay node can offer to the source node).

Source nodes compensate relay nodes for the energy that they spend to forward the traffic of their behalf. So, cooperation is beneficial when the energy saving that a relay offers to a source (the value of the relay to the source) is higher than the cost of relaying for the relay.

Finally, the BS calculates the worth of any mixed coalition—recall from D5.1 that any subset of nodes of a cluster is called a coalition and a subset that involves both types of nodes (relay and source) is called a mixed coalition—of one relay node and one source node. The worth of the coalition that is composed of relay node i and source node j is represented by a_{ij} and is given by the energy that these two nodes can save by their cooperation. In case that the cooperation of these two nodes does not lead to any energy saving, the source node prefers direct communication; in this case, worth of the coalition is set to zero.

We are now ready to compose the binary linear programming (BLP) problem for node selection. The unknown variables of this BLP problem are given by an MNx1 binary vector $\mathbf{x}$, the elements of which are the binary decision variables indicated by x_{ij}—value 1 for x_{ij} means that relay node i should relay the traffic of source node j, while value 0 means that relay i should avoid relaying source j. Figure 2-15 presents the pseudo-code of the node selection BLP problem and Figure 2-16 shows its flowchart.

START
FOR any source node in the cooperative cluster **DO**
CALCULATE the long range link cost for the source (e'_{jB})
ENDFOR
FOR any relay node in the cooperative cluster **DO**
CALCULATE the long range link cost for the relay (e_{iB})
CALCULATE the cost of relaying for the relay (c_i)
FOR any source node in the cooperative cluster **DO**
CALCULATE the cost of short range link between the source and the relay (e_{ij})
CALCULATE the value of the relay to the source ($h_{ij} = e'_{jB} - e_{ji}$)
CALCULATE the worth of mixed coalition of the relay and the source ($a_{ij} = \max\left[0, h_{ij} - c_i\right]$)
ENDFOR
ENDFOR
DEFINE $\mathbf{c} = \begin{bmatrix} a_{11} & \ldots & a_{1N} & a_{21} & \ldots & a_{2N} & \ldots & a_{M1} & \ldots & a_{MN} \end{bmatrix}$
DEFINE $\mathbf{x} = \begin{bmatrix} x_{11} & \ldots & x_{1N} & x_{21} & \ldots & x_{2N} & \ldots & x_{M1} & \ldots & x_{MN} \end{bmatrix}^T$
DEFINE matrix $\mathbf{A}$ as defined bellow in Equation (2.1)
DEFINE $\mathbf{b} = [\overbrace{1 \quad 1 \quad \ldots \quad 1}^{M+N}]^T$

SOLVE the following BLP problem:
$$\textit{Maximize } z = \mathbf{cx} \textit{ over the binay field}$$
$$\textit{Subject to}: \mathbf{Ax} \le \mathbf{b}$$
RETURN the answer $\mathbf{x}$
END

Figure 2-15. Pseudo-code of BLP node selection problem run by BS

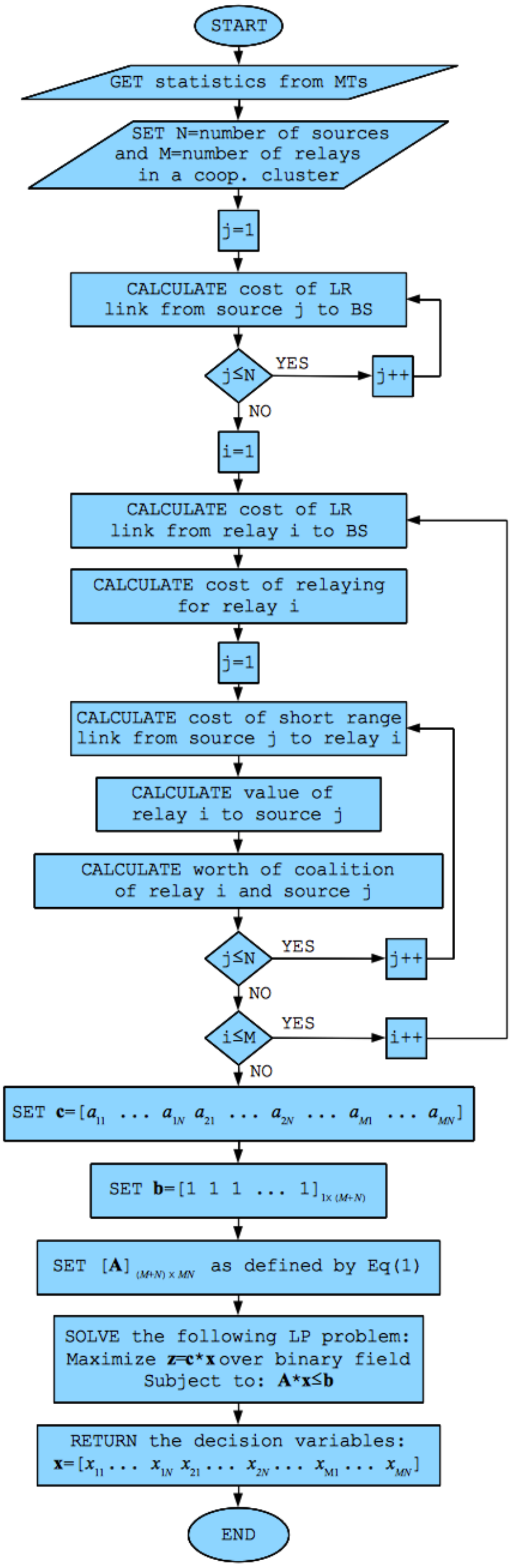


Figure 2-16. Flowchart of node selection BLP problem run by BS

where

$$
\mathbf{A}=
\begin{array}{r}
\\
M\left\{\begin{array}{c} \\ \\ \\ \\ \\ \end{array}\right.\\
N\left\{\begin{array}{c} \\ \\ \\ \\ \end{array}\right.
\end{array}
\begin{array}{c}
\overbrace{
\begin{array}{ccccc}
\overbrace{\hphantom{1\ 1\ \dots\ 1}}^{N} & \overbrace{\hphantom{0\ 0\ \dots\ 0}}^{N} & \overbrace{\hphantom{0\ 0\ \dots\ 0}}^{N} & \cdots & \overbrace{\hphantom{0\ 0\ \dots\ 0}}^{N}
\end{array}
}^{M}
\\
\begin{bmatrix}
1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0\\
0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0\\
0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & \dots & 0 & 0 & \dots & 0\\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots\\
0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & 1 & 1 & \dots & 1\\
1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & \dots & 1 & 0 & \dots & 0\\
0 & 1 & \dots & 0 & 0 & 1 & \dots & 0 & 0 & 1 & \dots & 0 & \dots & 0 & 1 & \dots & 0\\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots\\
0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 & \dots & 0 & 0 & \dots & 1
\end{bmatrix}_{(M+N)\times MN}
\end{array}
\qquad \textbf{(2.1)}
$$

Core Solution Algorithm

Core solution determines the share of each MT of a cooperative cluster from the distribution of the saved energy for sending one bit from each source node to the BS. Solution of the dual of node selection BLP problem gives the core solution of the cooperative node selection game[1]. We present the core solution by the vector variable $\mathbf{y}=[u_1,...,u_m,v_1,...,v_n]$, where u_i (i=1,…,m) indicates the share of relay node i and v_j (j=1,…,n) indicates the share of source node j for sending one bit to the BS during a cooperative short range communication session (CSRCS). The matrices **A**, **b**, and **c** are the same matrices that were defined in the previous subsection (the superscript T indicates the matrix transpose operation). Figure 2-17 and Figure 2-18 provide the pseudo-code and the flowchart presentation of the core solution algorithm for cooperative node selection game, respectively.

START
GET matrixes **b**, **A**, and **c** from the node selection BLP problem
SOLVE the dual LP problem that is defined as follows:

$$\mathit{Minimize}\ w=\mathbf{b}^T\mathbf{y}\ \mathit{over\ the\ the\ field\ of\ non-negative\ real\ numbers}$$

$$\mathit{Subject\ to}:\mathbf{A}^T\mathbf{y}\geq\mathbf{c}^T$$

RETURN the solution $\mathbf{y}=[u_1,...,u_m,v_1,...,v_n]$
END

Figure 2-17. Pseudo-code for the core solution of the cooperative node selection game

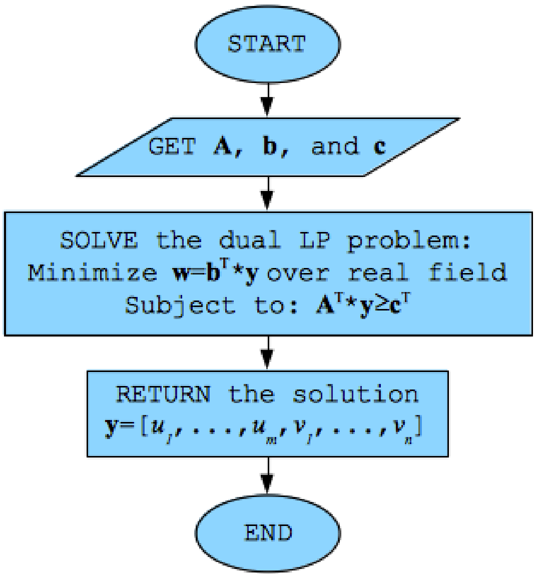


Figure 2-18. Flowchart for the core solution of the cooperative node selection game

Updating the Energy Credits

After finishing a cooperative short range communication session (CSRCS), source nodes should compensate the consumed energy by their corresponding relay nodes. Moreover, we should distribute the saved energy during a cooperative session among all cooperating players—either sources or relays. We assume that there is a central database in the network that records the share of each C2POWER MT from the saved energy.

To figure out the consumed energy by relay nodes as well as the total saved energy during a cooperative session, we need the total number of relayed bits during the session. We assume that the BS, as a trust reference for all cooperative parties, is responsible for counting the total number of relayed bits during any session. After each cooperative session, we update the energy credits of cooperating nodes as follows: to calculate the spent energy by the relay node, we get the cost of relaying one bit for each relay (which is given by Joule/bit) and multiply it by the total number of relayed bits during the session; to calculate the share of each cooperative node from the saved energy, we get the share of each source or relay node from the saved energy per bit and multiply those figures by the total number of relayed bits during the session.

When a cooperative session ends the whole saved energy is accumulated only in the source nodes (relay nodes only consume energy). Hence, to balance the energy credits, for each source node, we first decrease the saved energy from the source nodes' energy account; then, we add the share of each node (from the saved energy) to their account. In fact, we ask all source nodes to virtually return the saved energy back to the network; then, the network virtually distribute this energy among all cooperating nodes. Figure 2-19 and Figure 2-20 illustrate pseudo-code and flowchart presentation of this algorithm, respectively.

```
START
GET the core solution y = [u_1, ..., u_m, v_1, ..., v_n]
GET the total number of relayed bits during the CSRCS
FOR any relay node in the cooperative cluster DO
GET the ID of the corresponding source node
    CALCULATE the total consumed energy by the relay during the session
    ADD the calculated energy to the energy account of the relay
    REMOVE the calculated energy from the energy account of the corresponding source
ADD the share of the relay from the saved energy to its energy account
ENDFOR
FOR any source node in the cooperative cluster DO
ADD the share of the source from the saved energy to its energy account
ENDFOR
END
```

Figure 2-19. Pseudo-code for energy account updating algorithm

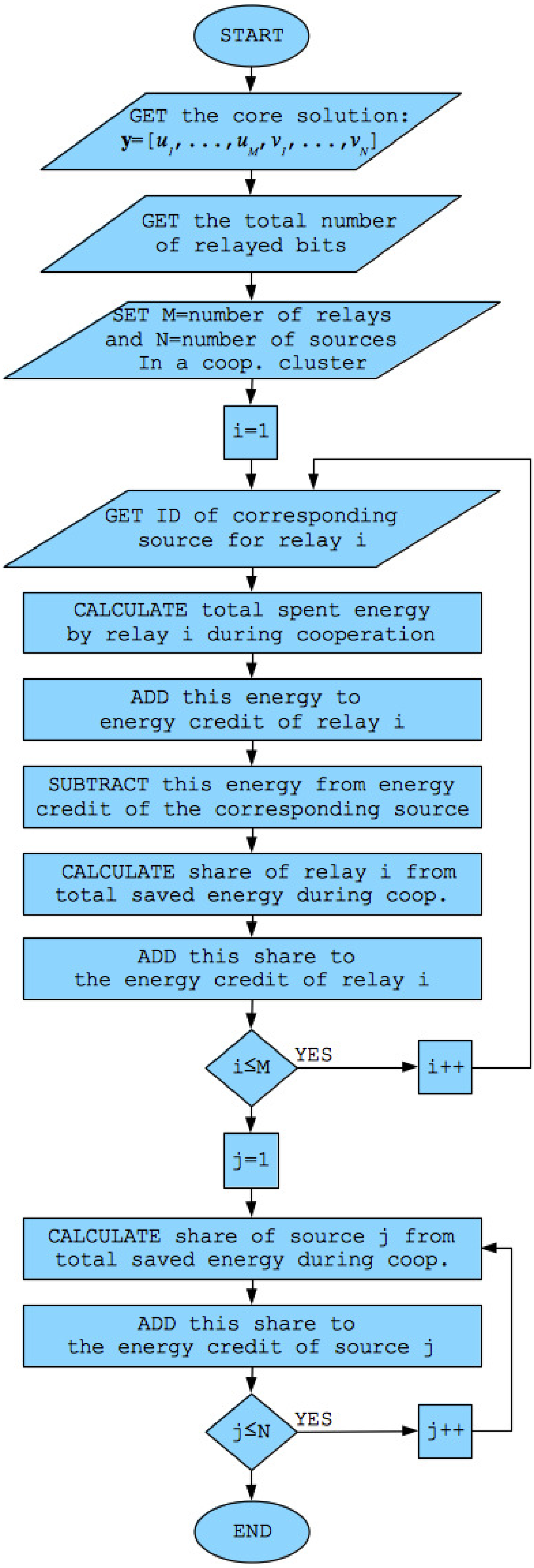


Figure 2-20. Flowchart of energy account updating algorithm

2.4 Energy efficient multi-radio cooperative relaying module

This section discusses a C2POWER energy efficient multi-radio relaying technique, designed with the goal to minimize the total energy consumption in wireless communication systems. The main idea behind the technique is derived from the fact that multi-radio relaying in wireless networks can greatly increase capacity and performance by exploiting the best available links and network interfaces to a destination node.

Assuming capability of multi-rate in wireless communication systems, if channel quality is sufficiently high, shorter transmission duration time is achieved by employing higher modulation schemes. However, multi-rate advantage might as well cause fairness problem. It is obvious that the low data rate stations use more channel time than the high data rate stations for the same volume of data. This affects the system in two ways: stations with low data rate experience poor service, in addition to reducing the bandwidth available to high data rate stations due to increased duration channel occupation time. Both reduced throughput and increased channel occupation time cause the growth of energy consumption in the network.

In order to minimize the use of long range devices, the C2POWER application can direct data frames via short range interfaces. The advantage of such a concept is multi-fold. First, a C2POWER node can save some energy by using short range low-power consuming radio interface instead of long range radio. Second, while relaying, it does not participate in long range radio channel sharing, causing higher probability of getting the access to the channel by other nodes including potential relay. Finally, data transmission may be achieved faster, if two-hop relay transmission provides higher average data rate.

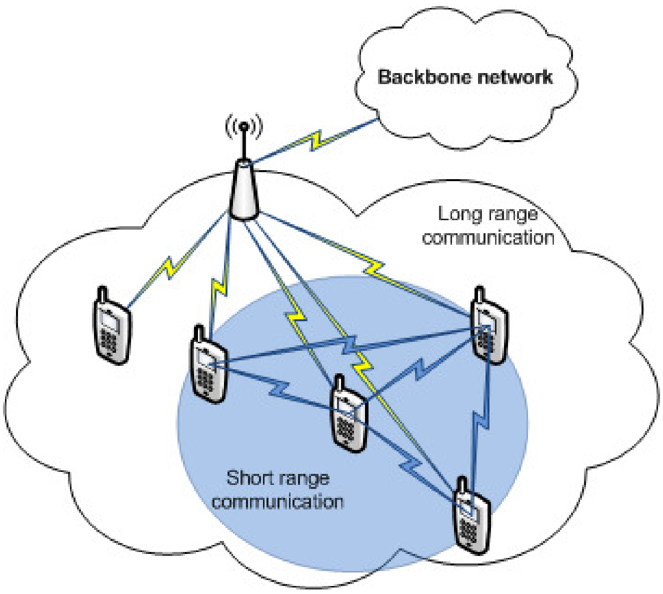


Figure 2-21. C2POWER with multi-radio relaying

Figure 2-21 illustrates a scenario for C2POWER multi-radio relaying. Nodes create an infrastructure based network using long range radio interface (e.g. WiMAX or WiFi). Some of them employ additional radio interface and form an ad-hoc network by the use of short range communication system (e.g. WiMedia or again WiFi in ad hoc mode). By taking the advantage of C2POWER application, nodes serve as multi-radio relay and use other nodes as relays with the aim to minimize the energy consumption required for data transmission.

For the sake of simplicity, we assume all nodes, willing to cooperate in a cluster, broadcast periodically a beacon frame. By sending periodical C2POWER beacons, nodes acknowledge their self-enforcement willingness to cooperate. It is especially desirable in case of shared wireless medium. If low data rate stations occupy the channel for longer duration time, the energy consumed by all nodes in the neighbourhood increases. Thus, providing relaying by high data rate station decreases total transmission time in the system and leads to faster access to the channel by relay node. The structure of <C2PowerBeacon> for multi-radio relaying, which is sent as a payload of C2POWER management frame (Table 2-2), is presented in Figure 2-22.

```
 0             7 8            15 16           23 24           31
+--------------+--------------+--------------+--------------+
|                 C2POWER Unique Identifier                 |
+--------------+--------------+--------------+--------------+
|               Long Range Interface IP Address             |
+--------------+--------------+--------------+--------------+
|               Long Range Interface Network ID             |
+--------------+--------------+--------------+--------------+
|         Net ID cont.        |     SNR      |  Data rate   |
+--------------+--------------+--------------+--------------+
```

Figure 2-22. C2POWER Beacon Structure

C2POWER Unique Identifier is a unique 4-byte C2POWER application address (or ID). The field is computed based on the last 4-bytes of the devices MAC address of the short-range interface. Next 4 bytes contain an IP address of long range interface. The following 6 bytes indicate ID of the long range network, which the device belongs to. In the case of 802.11 used as a long range network, *Long Range Interface Network ID* comprises MAC address of an access point (BSSID). 8-bit field *SNR* carries information about average signal-to-noise ratio on long range device and the final 1 byte informs the neighbourhood about currently available data rates to the attached access point.

Every C2POWER node detects beacon frames and creates a neighbour list containing specific information which is required to decide about potential multi-radio relaying.

Figure 2-23 presents a structure of neighbour list which is maintained by C2POWER nodes.

ID	(IP address)	SNR	BSSID	(IP addr)	SNR	rate	time
2	3.0.0.2	34.560	0	2.0.0.2	31.952	Mode18Mb	02.3
3	3.0.0.3	34.389	0	2.0.0.3	19.809	Mode02Mb	00.8
4	3.0.0.4	32.674	0	2.0.0.4	30.554	Mode11Mb	13.2
5	3.0.0.5	33.117	---	-------	------	-------	09.9
6	3.0.0.6	30.228	0	2.0.0.6	39.732	Mode54Mb	11.6
7	3.0.0.7	39.822	0	2.0.0.7	29.098	Mode06Mb	04.2

Figure 2-23. Neighbour List example

The first column contains *C2Power Unique Identifier*, and the second represents the IP addresses of neighbour short range devices. *SNR* in the third column indicates the direct channel state between neighbours on short range interface. Next column of the Neighbour List consists of information about long range radio interface. Column *BSSID* refers to network ID, next the IP address of the interface is stored, *SNR* and *rate* are related to latest signal-to-noise and available data rates on long range communication. Finally, time stamp indicates elapsed time since the last beacon has been received.

All decisions regarding multi-radio relaying implemented by C2POWER application are based on entries in neighbour list. A general scheme of the technique is shown in Figure 2-24.

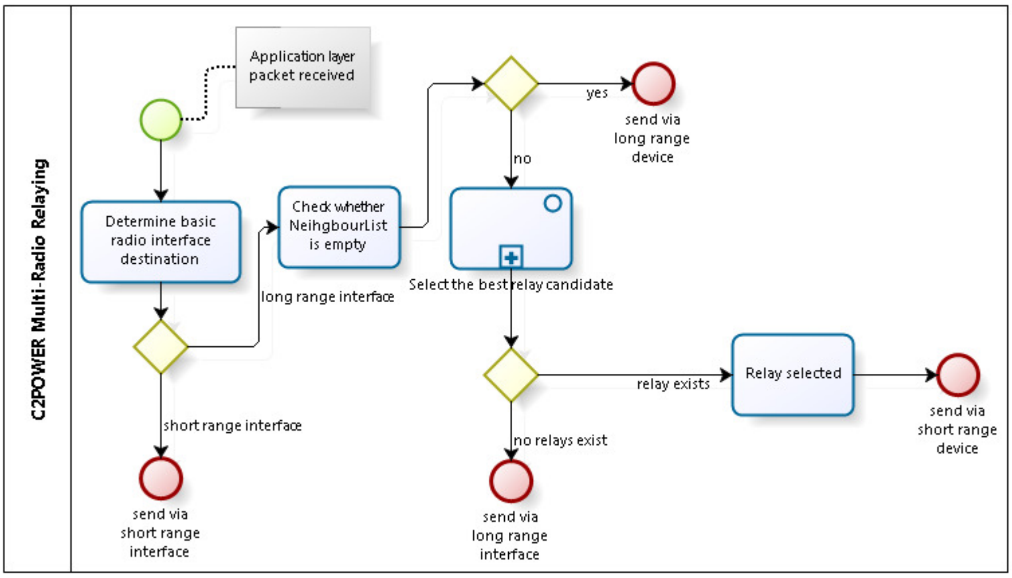


Figure 2-24. Multi-Radio Relaying Algorithm within the C2POWER network device

In order to determine if the packet is to be relayed, a C2POWER node decides whether the data packet is being basically directed to long range interface. In case any application realizes its communication via short range device, no relaying is needed. If the packet should be transmitted to an access point using long range radio interface, the algorithm checks the neighbour list to find some candidates for relaying.

Regarding a relay selection procedure, the C2POWER node compares entries of its neighbour list and tries to find a neighbour that can provide the highest rate of long range transmission (*Rr*) to the access point. The condition (*1/Rd>(1/Rr+1/Rs)*) is defined, where *Rd* and *Rs* are the data rates of long range interface from the node to the access point and short range interface data rate between the node and the neighbour, respectively. If the condition (*1/Rd>(1/Rr+1/Rs)*) is satisfied, the neighbour becomes a relay candidate. If more than one neighbour meet the constraint, SNR on long range channel is taken into consideration and finally, if such comparisons do not provide a relay, the candidate with the latest beacon update is selected to be the relay node. In next step, the packet traverses either short range device, if a relay exists, or long range devices as direct transmission if no neighbour satisfies relaying conditions.

2.5 Energy Aware Routing Technique

The energy-aware routing technique (EAR) is designed with the objective of finding optimal energy conserving routes based on aggregated network information. Our approach is based on an on-demand routing strategy where the route discovery is initiated when a node in the network has a packet to send to a destination.

The routing problem is mainly to identify the most cost effective path to the nearest available gateway unless the destination is located locally within the multi-hop environment. Generally in on-demand routing protocols, the source node generates and broadcasts a route discovery packet (RDP) to search for a path from source to destination. The destination node, upon receiving the RDP packets, unicasts a route reply packet (RRP) to the source in order to set up a selected path. In our approach, we consider periodic single-hop neighbourhood information exchange for every node in the network. This is a valid consideration given the cooperative approach considered in C2POWER. Secondly, unlike the on-demand approach where the destination node chooses the first arriving route request packet with minimum hop count, in our approach, the destination node considers all the RDP packets received within a pre-defined time window to estimate the most energy efficient path without compromising the end-to-end path delay to ensure adequate QoS for the flows.

During the route discovery process, each intermediate node piggybacks the energy related measures (such as the remaining battery level, energy cost per bit, etc.,) as well as its identity on the RDP message and forwards the packets (re-broadcasts). The destination node receives multiple RDP packets, but chooses the best route with respect to the optimum route path selection strategy chosen by the destination. The destination node upon receiving the first RDP packet will issue a time out period within which it will listen to and receives all the RDP packets with the respective identifier. The route discovery process and the path information aggregation approaches are shown in Figure 2-25 and Figure 2-26 respectively.

The RDP packets are given a particular hop-limit to introduce a restricted flooding of route discovery messages in the network. The hop-limit value is determined based on the overall network size and density. Another aspect for controlling the flooding of route discovery packets is the packet re-broadcast limit implemented to control the re-broadcast of the RDP packets since the destination is already implementing a timeout period for reception of the RDP packets and it will be logical for the intermediate nodes to control the re-broadcast of RDP packets for the same flow request. The re-broadcast limit will also depend on the network size and density and the hop-limit value. Such mechanisms inherently balance the traffic surge introduced by the route discovery broadcast mechanism and also indirectly balancing the energy consumption at the nodes for the control packets. As the network is comprised of nodes equipped with multiple interfaces and the routing is over multiple interfaces, the RDP packets are broadcasted for all the available links and access technologies (see Figure 2-25). This will provide a global view for the destination in making route choices and specifically the weight for the energy cost per bit will also be taken into account by delineating the path between the nodes independently based on the access technologies. Once the route path is computed, the destination node generates the RRP messages which will unicast the message to the source node, which eventually starts the packet flow.

```
Procedure at the source node(s)
unicast(DAT A : srdId, dstId, data)
if routeDb.dstId = □ {S starts route discovery} then
    unicast(DATA)
    send(DATA) {S sends a data message to D}
else
    buffer(DATA) {S puts the message on the buffer}
    for all Interface ∈Node do
        broadcast(RDP ) {S creates a query message}
        send(RDP )
    end for
end if
```

Figure 2-25. Pseudo-code describing the Route Discovery Process

Another aspect which is considered in our routing algorithm is the willingness of the nodes to participate in the routing path and to forward packets on behalf of others. In this regard, each node will determine whether to accept and forward RDP message depending on its remaining battery capacity. When it is higher than a threshold value, the RDP is forwarded; otherwise in the case of the battery capacity dropping between a secondary threshold, the RDP will be forwarded only on the minimum power interface, and lastly upon exceeding the secondary threshold, the packet is dropped. This will ensure that the destination is receiving the RDP packets only through the best available links and through nodes with a minimum available battery capacity. Eventually, in the event of the entire network nodes below a particular threshold, to ensure that the data forwarding process is still active, the revised RDP (in case the source does not receive a route to the destination for a period of time) will be sent with a higher sequence flag to ensure that the nodes use a revised threshold for the battery capacity.

Procedure at intermediate node(s)

```
receive(RDP ) {N receives a query message}
if RDP.hopNb = hopLimit then
  delete RDP {RDP achieves the hopLimit}
  return {Terminate}
end if
if ∃i∈rdpDb :rdpDb(i).rdpId = RDP.rdpId and rdpDb(i).srcId = RDP.srcId and rdpDb(i).reBr> rebrLimit then
  delete RDP {RDP achieved the rebroadcast limit}
  return {Terminate}
else
  if ndId = RDP.dstId then
    rdpDb ← rdpDb ∪ (RDP.rdpId, RDP.srcId) {N adds new rdpDb entry}
    rdpDb← rdpDb.reBr+ 1N increases the number of re-broadcast message in 1
    RDP.hopNb ← RDP.hopNb + 1 {N increases the hopNb in 1 hop}
    for all Interface∈Node do
      RDP.uniqueId ← RDP.uniqueId[n] ∪uniqueId[n + 1] {N appends its own uniqueId to the RDP}
      RDP.lifeT ← RDP.lifeT [n] ∪lifeT [n + 1] {N appends its own lifeTime to the RDP}
      RDP.costEnrLink ← RDP.costEnrLink[n] ∪costEnrLink[n + 1] {N appends its own link energy cost to RDP}
      send(RDP ) {N re-broadcast the RDP}
    end for
  else
    tempBuffer(RDP ) {N is the destination (D), D puts the message on the temporal buffer}
    delay(Timerseconds)
    calculate(OptPathCost) {route selection strategy}
    pathRoute(k) ← uniqueId_1 ∪uniqueId_2... ∪uniqueId_n
    unicast(RRP ) {S creates a new RRP}
    send(RRP )
    routeDb ← routeDb ∪ (U ) {U=(RDP.path, RDP.srcId, RDP.OptPathCost)}
  end if
end if
```

Figure 2-26. Pseudo-code describing the Path Information Computation

```
Update of routeDb at the node(s)
receive(RRP ) {N receives a RRP}
if ndId ≠RRP.srcId then
  send(RRP)
else
routeDb ← routeDb ∪ (Z) {Z = (RRP.path, RRP.dstId, RRP.OptPathCost)}
  delete RDP { S drops the message}
  If buffer≠∅ {S has messages on the buffer} then
    unicast(DATA)
    send(DATA)
  end if
end if
```

Figure 2-27. Pseudo-code describing the Route Reply Process

2.5.1 Optimum Route-Path Selection Strategies

The strategy to select the optimum route path is one of the crucial aspects of energy aware routing. In this section we provide the strategy to choose the route path during the route discovery process to minimize the total energy link cost and to maximize the network lifetime. The optimization is performed by the destination node which will have all the necessary information to perform the task (as obtained from the route discovery process). The lifetime of the network is maximized by considering the residual lifetime of every node participating in route discovery. Maximizing the network lifetime also implicitly means that load balancing is achieved amongst the participating nodes distributing the traffic flow amongst them. Here, we define the lifetime of the network as the time taken for the first node to drain-out its power.

Now we explain the route path selection strategy. In a network of wireless nodes, let $K \in \mathcal{N}$ be the total number of possible route-paths from $S \rightarrow D$ for a given environmental condition at a given time, which could vary in practice with time depending on node mobility and changing channel conditions. We consider a time period where the network environment shows no significant changes during the route discover process, and as mentioned before any changes in the network are reported periodically for route maintenance after the discovery process. Let $N(k) \in \mathcal{N}$ be the total number of nodes per route-path k, where $k = 1, 2, \ldots K$. Furthermore, the $n(k)^{th}$ link and the $n(k)^{th}$ node in the k^{th} route-path has the information metric $\Lambda(n, k)$, described by the pair

$$\Lambda(n, k) = \{E(n, k, T(n, k))\} \tag{2.2}$$

where, $n(k) = 1, 2, \ldots N(k)$, the element $E(n,k) \in \mathfrak{R}^{+}$ is the energy cost for the n^{th} link in the k^{th} path, and $T(n,k) \in \mathfrak{R}^{+}$ is the estimated residual lifetime (remaining lifetime) of the n^{th} node in the k^{th} path if the k^{th} route is selected by the destination D as the selected route path. Note that $T(1,k)$ for all k refers to the residual lifetime of the source node itself, the knowledge of the residual lifetime of the source node is important if the destination node is required to choose the strategy to minimize the link energy cost for the first hop to increase the lifetime of the source node itself. It is important to note here that simultaneous optimization of the total energy link cost and the network lifetime may not be possible always since in many cases one needs to be sacrificed instead of the other, that is to be traded off. Hence we try to find a good trade-off between the two cost parameters based on some internal policies. If α is the trade-off factor then we can define a utility function given by

$$\mathrm{U}_1(k) = \alpha \sum_{n=1}^{N(k)} E(n,k) + \frac{1-\alpha}{\min\{T(n,k);\ \forall n \neq 1\}} \tag{2.3}$$

where, $0 \leq \alpha \leq 1,\ \forall\ \alpha \in \Re^{+}$. Then the corresponding optimum path for a given α is obtained by minimizing the utility function given by,

$$\hat{k} = \arg\min_{k}\{\mathrm{U}_1(k)\} \tag{2.4}$$

The above optimization is basically a trade-off between selecting the minimum energy link cost route path and the maximum network lifetime as we discussed. It is important to note here that due to the differences in the orders of magnitude of the values for the first and second terms on the r.h.s of equation (2.3) it is a challenging task to select α to have a quantitative trade-off between them. In order have an absolute quantitative trade-off we could normalize the two such that they have the same orders of magnitude approximately, if not it is only possible to have relative trade-off between the total energy link cost and the network life-time. At the destination, the total link energy cost can be normalized with the total energy cost for all the participating links given by,

$$\Gamma = \sum_{n=1}^{N(k)} \sum_{k=1}^{K} E(n,k) \tag{2.5}$$

and the second term of $U_1(k)$ may be normalized with the inverse of the network lifetime, where the network lifetime is defined as the residual lifetime of the node corresponding to the lowest residual lifetime in the network, given by,

$$\Upsilon = \max\{T(n,k)^{-1}; \forall n, k\} \tag{2.6}$$

The utility function based on the normalized values is then given by,

$$\mathrm{U}_1(k) = \alpha \sum_{n=1}^{N(k)} E(n,k)\Gamma^{-1} + \frac{(1-\alpha)\Upsilon^{-1}}{\min\{T(n,k);\ \forall n \neq 1\}} \tag{2.7}$$

Based on (2.7), now we can say that for a given value of α, total energy link cost is traded off with the network lifetime with a quantitative measure of $\alpha\%$. Note that Γ and Υ will change over time but such changes however would not affect the optimization process which is only performed for a given time only.

Another route path selection strategy is to basically choose the minimum energy link cost path for the source node given by min{$E(1, k)$} for the first hop in order to help the source node that has a low residual lifetime, and then perform the optimization by defining a utility function similar to $U_1(k)$ for the rest of the links for all possible paths. Such a situation suits the scenario for a source node in need of help with a low battery power level. The modified utility function in this case would then be,

$$\mathrm{U}_2(k) = \alpha \sum_{n=2}^{N(k)} E(n,k) + \frac{1-\alpha}{\min\{T(n,k);\ \forall n \neq 1\}} \tag{2.8}$$

and the corresponding route path selection strategy would be,

$$\hat{k} = \arg\max_{k}\{\mathrm{U}_2^{-1}(k)\mathbf{I}(E(1,k) = \min\{E(1,k)\})\} \tag{2.9}$$

Where **I**(x) is a Boolean function that gives a 1 if the argument x is true and 0 otherwise. This particular strategy could also be imposed from the source node's side by simply selecting its minimum energy cost path to send the route request message, which can be embedded in the routing algorithm. However, performing this at the destination node, that is the destination node deciding whether the second strategy could be adopted or not, would have a network level benefit. For example, if the source node's low cost energy path has a node with much lesser residual lifetime than the source node's residual lifetime (i.e. if $T(1, k) >> T(n, k)$ for some n) then the destination node may choose to avoid this path in order to maximize the lifetime of the network which cannot be done if the source node already selects the link corresponding to the minimum energy cost during the first hop. Therefore, it is beneficial for the network to perform the route-path selection at the destinations node regardless of adopting the strategies corresponding to the utilities $U_1(k)$ or $U_2(k)$.

2.6 Relay-based Energy Efficient Routing Technique

In this section, we present the Relay-based Energy Efficient Routing approach (REER). The crucial aspect of the routing strategy is that every MT informs its neighbours about its minimum achievable energy-per-bit to reach an access gateway. The approach is to improve energy efficiency based on the minimization of energy-per-bit for transmission. The routing decisions are taken, at every MT, on the fly considering all the possible choices to reach the access gateway.
The proposed method is composed by two different stages running in parallel:

1) *Context Information Dissemination* Stage: responsible to disseminate context information and maintain it in neighbour tables of the MTs.
2) *Short-Range Routing Stage*: responsible to decide how to route the packets in order to reach the BS.

In the following, we explain the operation of the Context Information Dissemination and Short-Range Routing modules. Finally a numerical example is given in order to exemplify the functioning of the proposed technique.

Context Information Dissemination Stage

The scope of the context information dissemination is to maintain the neighbour table for each MT participating in the cooperation. The information at each MT is stored like in Table 2-4; for each neighbour it is maintained the IP address for the short-range communication and the energy per bit necessary to reach the gateway (BS) through this MT regardless the number of hops needed.

Table 2-4. Neighbour table information (example)

Entry	IP-Address SR	Eb
1	IP_Addr(1)	Eb(1)
2	IP_Addr(2)	Eb(2)
3	IP_Addr(3)	Eb(3)

Every MT that is willing to cooperate sends periodically a beacon containing its IP address for short-range communications and its best E_b to reach the gateway. While the IP address is related to the neighbouring MT, the best energy-per-bit could reflect energy employed for a more than two hops route and is calculated by the procedure described below.
In an infrastructure-based wireless network with more MTs having short and long range interfaces, let be $E_b^{SR}(n,k)$ the energy per bit to transmit via short-range from MT n to MT k; $E_b^{LR}(n,k)$ the energy per bit to transmit from MT n to the BS employing the long-range interface; and $E_b(n)$ the lowest energy per bit

found to reach the BS from MT *n* (energy is calculated as power consumed to maintain the interface in TX state divided by the achievable data-rate).
The value $E_b(n)$ is calculated in two steps. Firstly we choose the best neighbour MT using (2.10)

$$E_b(n) = \min_k \{E_b^{SR}(n,k) + E_b(k)\} \ ; \forall k \in NeighborTable \tag{2.10}$$

Secondly the long-range constraint is checked as in (2.11) in order to find which value of energy per bit to tell via beaconing.

$$\begin{cases} E_b(n) > E_b^{LR}(n) \rightarrow TellE_b^{LR}(n) \\ E_b(n) < E_b^{LR}(n) \rightarrow TellE_b(n) \end{cases} \tag{2.11}$$

If a neighbour is found to be convenient its related E_b is communicated via beaconing, otherwise it is communicated the E_b related to the Long-Range interface.
In order to maintain the neighbour table, a parameter is set in order to define the period for the beaconing, and another parameter is set in order to define the time out for the table entries. Since broadcasted packets can be lost, every beacon updates the timestamp of the related neighbour entry while the neighbour table timeout defines how many beacons can be lost before to erase a neighbour.

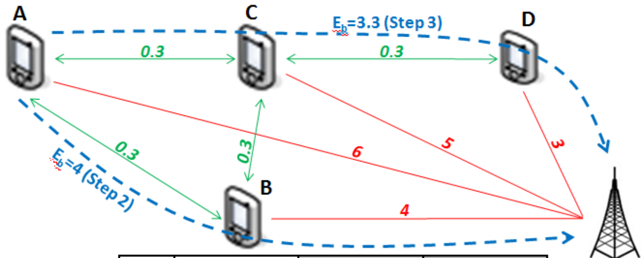
Short-Range Routing Stage

In the previous paragraphs we described how the context information is disseminated amongst cooperating MTs. Here we explain how this information is exploited for the routing decisions.
Suppose the MT *n* is receiving a data packet. A two-steps procedure is run in order to decide how to route the packet. Firstly, like in the *Context Information Dissemination* stage, we find $E_b(n)$ with (2.10). Secondly, checking the long range constraint (2.12), we decide whether to route the packet toward a neighbour MT or send it directly to the BS.

$$\begin{cases} E_b(n) > E_b^{LR}(n) \rightarrow Tell\ E_b^{LR}(n) \\ E_b(n) < E_b^{LR}(n) \rightarrow Tell\ E_b(n) \end{cases} \tag{2.12}$$

Since the above procedure is run upon every data packet reception, the packet can be relayed several times and the route is dynamically selected. For sake of clarification, in the following we present and example showing how the proposed technique works.

Numerical example



Node	Step 1		Step 2		Step 3	
	Next hop	E_b^{LR}	Next hop	E_b	Next hop	E_b
A	AP	6	*B*	*4*	*C*	*3.3*
B	AP	4	C	5	C	3.3
C	AP	5	*D*	*3*	*D*	*3*
D	AP	3	C	5	C	3.3

Figure 2-28. Numerical example for the proposed technique

In Figure 2-28 it is depicted a simple scenario in which green and red arrows represent short-range and long-range links respectively with the related costs of transmission representing the energy per bit. Moreover, in the table attached with the figure the best routing choice for each MT is shown in three different points in time:

> ***Step 1:*** At the beginning, every neighbour table is empty and every MT will use long-range interface for transmission;
> ***Step 2:*** Each MT sent its first beacon;
> ***Step 3:*** MT C sent its second beacon.

As we can see in *Step 2* MT A and C will route packets toward B and D respectively as they find a gain with respect to their long-range cost. In *Step 3* MT C notified its lowest cost ($E_b^{C \to D} + E_b^{D \to BS} = 0.3 + 3$) via beaconing reached by MTs A and B. After reaching MT C beacon, MT A will route its packets toward MT C instead of MT B, since he indirectly knows that through MT C he is able exploit connection of MT D, bringing its total cost at *3.6* [J/Mb] ($E_b^{B \to C} + E_b^{C \to D} + E_b^{D \to AP}$). Moreover, MT B is aware that he can transmit toward MT C with a cost of *3.6* [J/Mb] ($E_b^{B \to C} + E_b^{C \to D} + E_b^{D \to AP} = 0.3 + 0.3 + 3$). Note that MT D has MT C in its neighbour table; however, since MT C redirects traffic back to D, the cost is higher than the long-range TX. This shows that the loops in the routes are naturally avoided.

3. Simulator Implementation

In Section 2 we described and divided C2POWER application into functional elements. The description is intended to provide functionalities, requirements and algorithms that have to be implemented in the simulator. Herein, we evolve the description into the object model of the C2POWER application applied to ns-miracle. The ns-miracle simulator can be downloaded from the repository of the Department of Information Engineering (DEI) of Padova [7] while its documentation can be found online [8].
As explained in the previous section, the C2POWER layer is located between the IP routing layer and the IP assignment layer. Under the C2POWER layer, each interface has got its own IP layer, and the relative stack down to the PHY layer and the physical channel. Over the C2POWER layer there is one IP routing layer, common to all the interfaces, and on top of that the transport and application layers that can be used in ns-miracle.

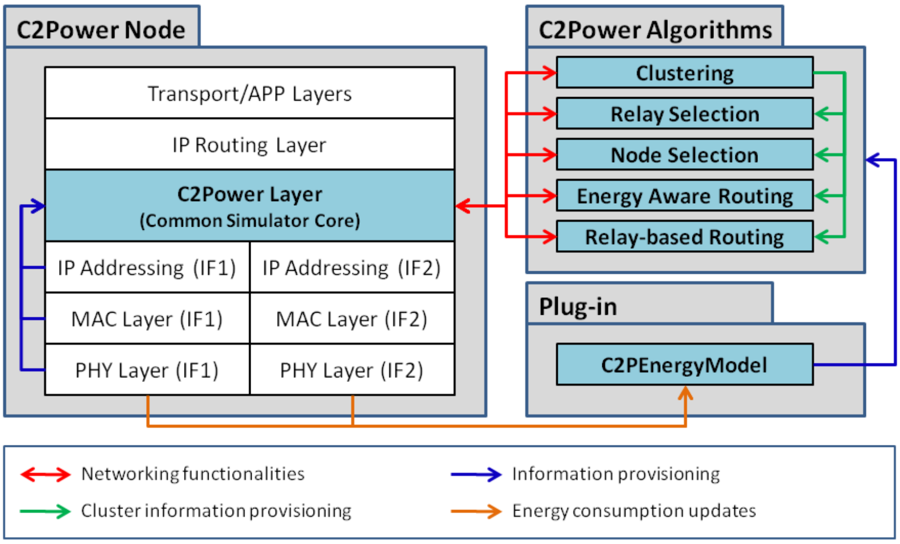


Figure 3-1. C2Power simulator structure

In Figure 3-1 the structure of a C2Power node is presented as implemented in the ns-miracle simulator. In the C2Power Node box there are the layers implementing ns-miracle Module class functionalities. Within the node the C2Power layer is able to retrieve necessary information from the layers of each stack below (i.e. data-rate, power consumption, etc…). We implemented the C2PowerEnergyModel as a plug-in in order to have a central place implementing a “battery” simulating the energy drain due to the consumption of different interfaces and have this information at disposal to the different algorithms.

The C2Power Layer module is the **core** of C2POWER with the following goals:

- connecting together the C2POWER elements,
- isolating them from the low-level details of the simulator, and
- providing the functionalities that are common between different approaches.

The first two goals are realized by C2Power Layer, which manages the interactions between the application layer and either IP layer(s) or MAC layer(s). The C2Power Layer —once upon receiving an ns2 packet— is able to decide whether to pass the packet to subsequent layer (either up or down), read C2Power control information or relay it. Apart from interaction with network simulator common framework, the C2Power Layer is responsible also for the integrity of C2Power application, as it is able to assign itself a unique ID of the C2Power application located in the particular network device and also to create and distribute <C2PowerBeacon> frame as according to C2Power cooperation protocol. The

C2Power Layer is the only part of the simulator that can contain invocations to ns-miracle methods. The ideal goal is to have only pure C++ code into the rest of the components, which is in line with idea of network simulator agnostic C2Power application (the code could be ported between different simulators, e.g. ns2, ns-miracle, ns3, omnet). Nevertheless, an effort in the direction of dividing C2Power elements into policies and mechanisms facilitates the maintenance and the readability of the code, ending up in a clear design of the software, and a code that is easier to develop and debug.

As a result of these motivations, we consider that all the calls to ns2/ns-miracle methods are invoked from the core. The other components invoke methods from the core, which in turn give access to the ns2 calls. A low-level detail that can or cannot be in the skeleton is the timer, and in particular the timers that will regulate the beacons.

The last goal of the C2POWER core is realized by the multiple support functions, which interact with the simulation environment to provide additional functionality required to support C2POWER cooperative networking and energy saving gain. The support functions are discussed in more details in the following subsection.

Eventually, C2Power Layer—as the skeleton of C2POWER application—contains methods that invoke algorithms developed within C2POWER, which are further discussed in following sections.

The algorithms implemented within the C2Power simulator are the following:

- Energy Efficient Cluster formation mechanism
- Energy Efficient Relaying protocol
- Energy Efficient Node Selection based on Game Theory
- Energy Aware Routing protocol
- Relay-based Energy Efficient Routing Technique

In the following of this section we will provide the details of the implementation of the C2Power core and the algorithms implemented.

3.1 C2Power Layer core

As explained before, the core of C2POWER simulator is the ***C2PowerLayer*** class that extends the ns-miracle *Module* abstract class and has the goals of *(I)* connecting together the C2POWER elements; *(II)* isolate them from the low-level details of the simulator; and *(III)* provide functionalities that are common between different approaches. In Figure 3-2 a class diagram for the C2Power core is shown and it is discussed below.

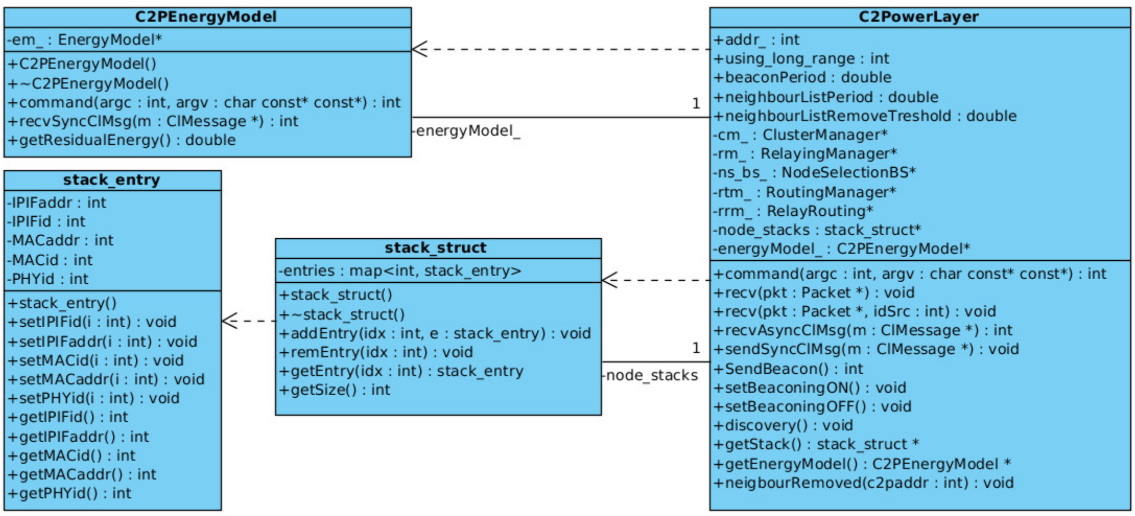


Figure 3-2. Class Diagram for the C2POWER Layer (Simulator Core)

The ***C2PowerLayer*** class extends the ns-miracle Module class by implementing necessary functions to handle packets (see 3.1.1), handle cross-layer messages (see 3.1.2) and set simulation parameters from TCL scripts (***C2PowerLayer::command(.)*** method). Moreover, it contains different pointers to the interfaces defined for the different C2POWER algorithms:

- ***cm_***: Pointer to the Cluster Manager
- ***rm_***: Pointer to the Relaying manager
- ***ns_mt_*** and ***ns_bs_***: Pointer to interfaces for node selection
- ***rtm_***: Pointer to the Routing Manager
- ***rrm_***: Pointer to the Relay-Based Routing Manager

On the other hands, each C2Power algorithm has a pointer to the ***C2PowerLayer*** class in order to have access at the common functionalities provided by the C2Power core.

As shown in Figure 3-2 an important functionality for all the C2POWER components is the ***C2PEnergyModel***. This class, implemented as an ns-miracle *plug-in* [8], is responsible for the maintenance of the battery state. Each C2POWER component has access to the Energy Model, through the ***C2PowerLayer*** module, in order to have access to the current battery status. Note that the responsible class for energy calculation is actually the PHY layer. The issue is solved with the following expedient:

- The ***C2PEnergyModel*** plug-in contains the battery model implemented with the ns-miracle ***EnergyModel*** Class.
- Every PHY layer, having its own levels of power consumption, is aware about the pointer to the ***EnergyModel*** Class in order to update energy values when necessary.

The way we disseminate the pointer for the ***EnergyModel*** Class is explained later in 3.1.2.

The structure **stack_struct** has been introduced for convenience and enables each component to be aware about the stack below the C2Power layer. This facilitates the sending of cross-layer messages addresses to a particular layer belonging to a particular stack. This class is initialized at the beginning of the simulation with the function *discovery()* which make use of cross-layer messages in order to retrieve necessary information from layers below.

The C2Power simulation core is also responsible to provide functionalities that are shared with all its components which are:

- the capability to switch ON and OFF the beaconing system,
- the capability to advise each component about the elimination of a node from neighbor list.

Switch ON and OFF the beaconing system

The ***C2PowerLayer*** is responsible for creating and distributing *<C2PowerBeacon>* frames according to C2POWER cooperation protocol. On the basis of the active components within a node, the beaconing system is not necessarily requested. For this reason, the C2POWER core allow each component to switch on and off the beaconing system using ***setBeaconongON()*** and ***setBeaconongOFF()*** methods.

Neighbor removal notification

Some algorithms making use of the neighbor table maintained by the clustering component may have the necessity to be aware about a leaving neighbor. For this reason a notification function is forecasted in case of neighbor removal.

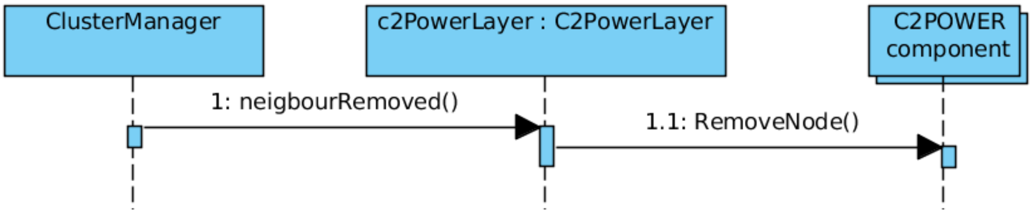


Figure 3-3. Sequence diagram for neighbor removal event

The responsible component for maintain the neighbor list is the ***ClusterManager*** class. Upon a neighbor removal event it will call ***neighbourRemoved()*** function in *C2PowerLayer* class which will in turn notify the event by calling the ***RemoveNode()*** method for all the C2POWER components implementing this method.

3.1.1 Packet management

C2POWER simulation core is responsible for creating and distributing *<C2PowerBeacon>* frame according to C2POWER cooperation protocol. Apart of this feature, the core is also responsible to handle data packets. Within the ns-miracle simulator we distinguished the following cases:

- Transmitting data packets (coming from higher layers)
- Received Packets –either data packets or C2POWER control packets– (coming from lower layers)
- Received Packets coming from a C2POWER component (control packets)

Ns-miracle makes use of *hdr_cmn* header in order to handle the direction of the packets through the stack by means of the attribute *direction_* which can be *UP* or *DOWN*.
In case of reception of a packet with direction *DOWN*, the possible cases are:

- The core is receiving a data packet from routing layer to be sent, or
- The core is receiving a C2POWER control packet from a component to be sent

Instead, in case of reception of a packet with direction *UP*, the possible cases are:

- The core is receiving a data packet from another node
- The core is receiving a C2POWER control packet from another C2POWER device

In both the last two cases the packet will be handled accordingly with the active policy which is related to the active C2POWER component.

Every C2POWER component may have its own control packets and it will be responsible to handle their fields. The C2POWER core handles different kind of control packets on the basis of the different C2POWER control packet type defined as one of the following:

- ***C2POWER_BEACON***: packet for the cluster manager component
- ***C2POWER_NODSEL_STATS***: packet for the node selection component
- ***C2POWER_NODSEL_ROUTE***: packet for the node selection component
- ***C2POWER_ROUTING***: packet type for the routing component.
- ***C2POWER_RR_BEACON***: packet type for the beaconing of relay-based routing.
- ***C2POWER_RR_DATA***: packet type for data packets handled by relay-based routing.

Figure 3-4 shows how the C2POWER core handles packets with “UP” direction. The *C2PowerLayer* class receives packets travelling through the stack, extracts the C2POWER header and, on the basis of the field *type* it decides how to handle the packet. In case of a C2POWER control packet, the correct method defined by each C2POWER component interface is called, otherwise the packet is sent up to the Routing layer above.

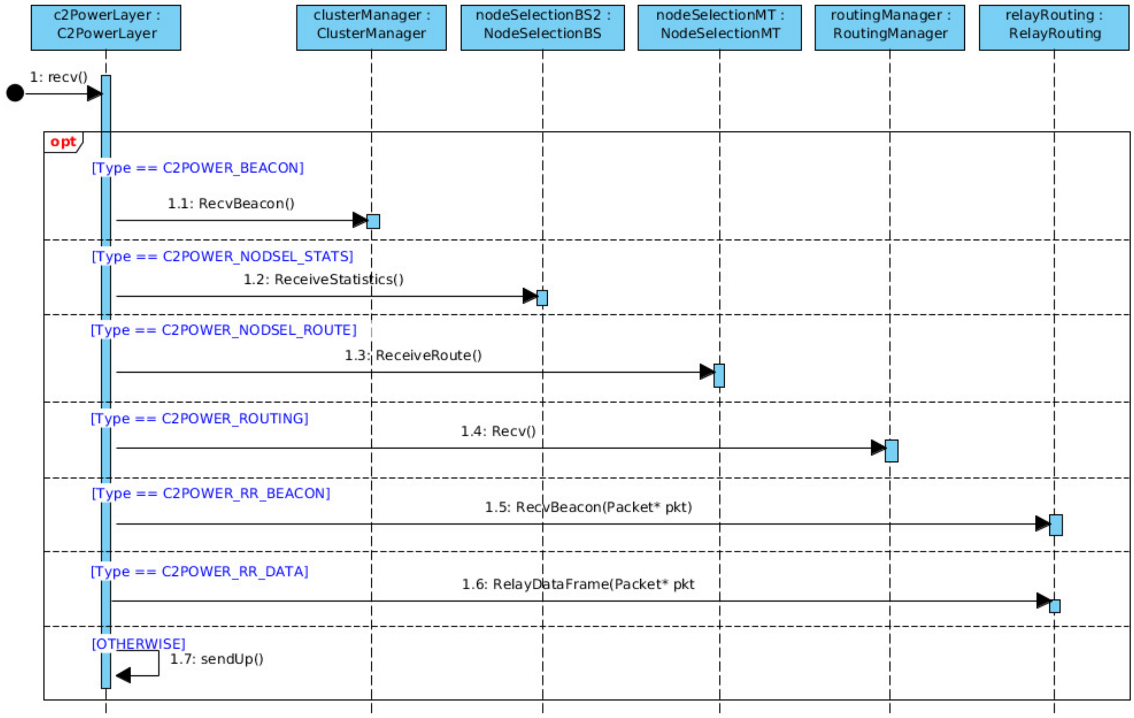


Figure 3-4. Handle received packets with C2Power header (*hdr_cmn.direction_=UP*)

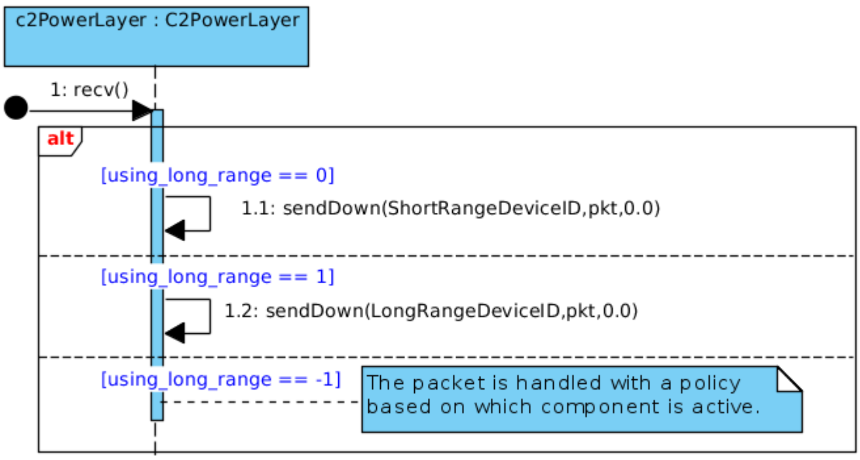


Figure 3-5.Handling of packets with direction "DOWN"

Figure 3-5 shows how the C2POWER core handles packets with "DOWN" direction. The policy is driven by *using_long_range* attribute that belongs to *C2PowerLayer* class.

In normal conditions, thus upon a reception of a data packet, the *using_long_range* parameter is kept always equal -1. In this case, the C2Power core is responsible to select the handling policy (on the basis of the active algorithms) and call the correct method to relay the data packet or send it directly to the BS.

On the other hands, if a packet is sent from a C2POWER component, the *using_long_range* parameter is set accordingly to its intention to send the packet with the C2POWER cooperative link or with the long-range interface.

3.1.2 Cross layer messages

In order to allow the operations of C2POWER layer and related algorithms, necessary information has to be retrieved from interfaces below (i.e. data-rate, IP address, etc…). In this sub-section we explain cross-layer messages employed in order to gain this objective.
In ns-miracle there is two kind of cross layer messages: synchronous and asynchronous. For a detailed explanation about synchronous and asynchronous cross-layer messages behavior we refer to [8], while here we want to explain the purposes of each cross layer message.
In Table 3-1 we list messages used by the C2POWER layer and the purpose of each of them is explained below. In ns-miracle every cross layer message is defined by a class implementing the abstract class *ClMessage* and distinguished by the parameter *ClMessage_t(unsigned int)* defining the kind of cl-message. Every layer handles cl-messages within *recvSyncClMsg(ClMessage* m)* or *recvAsyncClMsg(ClMessage* m*) methods.

Table 3-1. Cross Layer messages defined for C2POWER simulator.

Cross layer message kind	Class implementing the cross layer message	Library defining the message
CLMSG_C2P_DISC	ClMsgC2pDisc	ns-miracle (c2power_clmsg.h)
C2P_80211_SNR	C2P_Notify_Crl_Message	dei80211mr (mrcl-mac80211.h)
CLMSG_C2P_GETTXPOWER	ClMsgC2pPtx	ns-miracle (c2power_clmsg.h)
CLMSG_C2P_GETDATARATE	ClMsgC2pDataRate	ns-miracle (c2power_clmsg.h)
CLMSG_C2P_GETNRGPOINTER	ClMsgC2pGetNrgPointer	ns-miracle (c2power_clmsg.h)

CLMSG_C2P_DISC

This cl-message is sent from the *C2PowerLayer* class in order to discover information about the stacks below the C2POWER layer. This message is a UNICAST synchronous cl-message and it is used in order to retrieve IP interface address and its module ID, MAC address and its module ID, and finally the ID of PHY layer.
Every IP, MAC and PHY layers handles this message in order to update fields of this cl-message which will return values to the *C2PowerLayer*. This information is used whenever the C2Power Layer has to retrieve parameters of a particular layer belonging to a particular stack. The needed information (e.g. retrieve power consumption, retrieve data-rate, etc…) can be requested directly from the C2PowerLayer or by an algorithm. In the latter case, the C2PowerLayer will act as an intermediary between the C2Power algorithms and the ns-miracle protocol stack.

C2P_80211_SNR

Cross layer belonging to the 802.11 implementation.
Every time the 802.11 MAC layer receives a packet from the AP this cross layer message is sent in order to disseminate info regarding:

- SNR

- SINR
- BSS_ID_
- dataRate_

Every time this cl-message is sent, *C2PowerLayer* handles it within the implemented method *recvAsyncClMsg(ClMessage* m)* and tracks these values.

CLMSG_C2P_GETTXPOWER

This cl-message is sent within the C2POWER layer by each component that needs to know the power consumption in transmission for a particular interface.
With the current implementation, this is used as a synchronous cross-layer message of kind UNICAST and addressed to a PHY of a particular interface using the stack info contained in *stack_struct* explained earlier. The PHY layer handles this message within the *recvSyncClMsg(ClMessage* m)* methodand update the TX power value.

CLMSG_C2P_GETDATARATE

This cl-message is sent within the C2POWER layer by each component that needs to know the data-rate in transmission of a particular interface.
With the current implementation, this is used as a synchronous cross-layer message of kind UNICAST and addressed to a MAC of a particular interface. The MAC layer handles this message within the *recvSyncClMsg(ClMessage* m)* method and updates the data-rate value.

CLMSG_C2P_GETNRGPOINTER

As explained earlier, we need to disseminate the pointer for the *EnergyModel* Class contained in the *C2PEnergyModel* plug-in, to every PHY of the stacks below the C2POWER layer.
This cl-message is sent from every PHY layer at the initialization of the simulation. The *C2PEnergyModel* plug-in handles this synchronous cl-message providing to the PHY layers the pointer of *EnergyModel* Class contained in the *C2PEnergyModel* plug-in.

3.1.3 Timers

C2POWER layer uses various timers in order to schedule events for different C2POWER components. In particular, the current implementation contains the following timers:

- **C2PowerTimer** that is used in order to schedule packets and refresh periodically the neighbor lint entries,
- **RouteTableTimer** that is used in order to refresh the route table for the Energy Aware Routing Module
- **RDPTableTimer** that is used in order to refresh the RDP table for the Energy Aware Routing Module
- **RDPTimeoutTimer** that is used by the Energy Aware Routing Module in order to define the time window for the acceptance of RDP packets at the destination node.

Each timer and its relation with the related component are explained in the next sections dedicated to specific components.

3.2 Clustering Module

Exchange beacons between the nodes via the short-range interface, used indirectly by most other support classes, which use the information collected by this class. The idea is that the core knows about a *ClusterManager_Intrf* implementation and it access its methods to inform it that a node arrived or went away, and to provide it with the beacons, that have information about cluster management. Moreover, the cluster manager will be in charge of generating beacons. On the other hand, the code in the ClusterManager implementation will never "send" a frame directly, but instead it invokes the *SendBeacon()* method on the skeleton, that encodes and sends the frame directly, and maybe provide also caching and aggregation services.

The *RecvBeacon(Packet* pkt)* method is invoked by the core when it receives a control frame regarding cluster management. The *AddNode(.)* method adds nodes from a cluster upon the reception of beacon packets. *BeaconingON()*and *BeaconingOFF()* methods are used in order to switch on and off the beaconing service.
GetNeighbourList() method is invoked by the C2POWER layer in order to allow the C2POWER layer to be aware about the presence of neighbors and in order to obtain the best node in case of relaying.
The *ResreshNeighbourList()* method is responsible for the refresh out-dated entries and in case calls the *neigbourRemoved(.)* in the C2POWER layer to inform about a leaving neighbor.

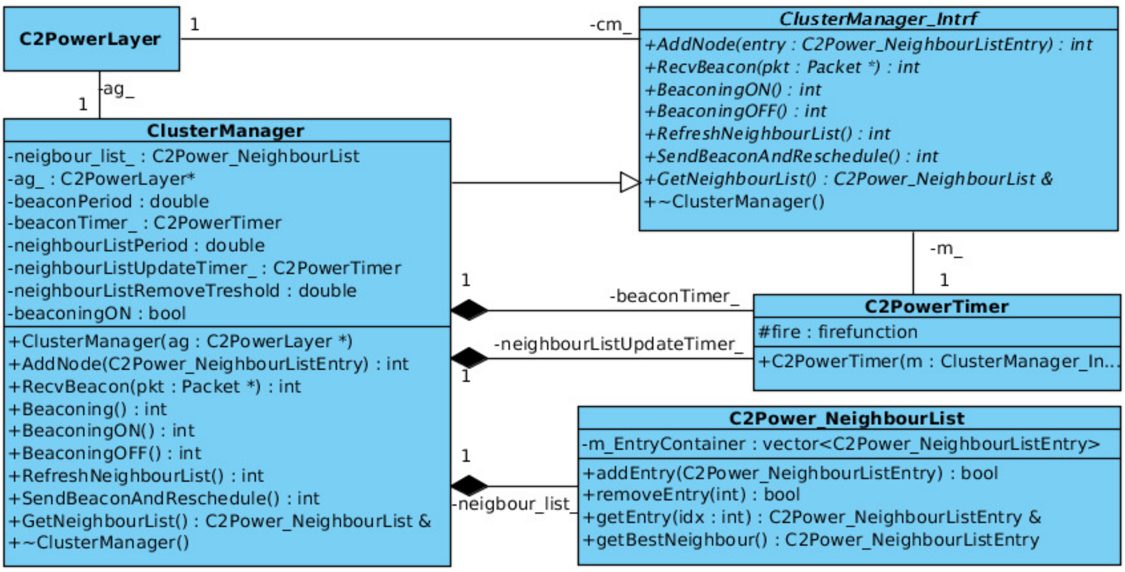


Figure 3-6. Implementation of Clustering Module

Table 3-2. Parameters for the C2POWER Cluster component

Parameter	Type	Description
beaconPeriod	double	Define the time interval between beacon packets
neighborListPeriod	double	Period use in order to parse neighbor list for out-dated entries
neighborListRemoveTreshold	double	Define the validity period for the neighbor list entries
beaconingON	Bool	Define whether to send beacons or not

3.3 Node Selection Module

Node selection Manager is present in each C2POWER-enabled node, and it coordinates with a process running on the network side in the Cooperation Mediator, to set up a two-hop relay between mobile terminals and base stations. Some of the nodes act as sources (communicating indirectly through one relay), while some act as relays (communicating directly with a base station, and forwarding data to/from one source). On the other hand, some nodes decide that cooperation is not beneficial for them and communicate directly with the base station without any relaying. This class uses *ClusterManager* to access the data about the quality of the connections among MTs themselves and the quality of connections between MTs and BS on the long range technology.

NodeSelectionBS_Intrf and *NodeSelectionMT_Intrf* are interfaces implemented by *NodeSelectionBS* and *NodeSelectionMT* classes, which uses the technique reported in [5] to decide which nodes should be relays, which should be sources, and which nodes should not cooperate.

A C2POWER application uses context information to perform informed decision about cooperation between MTs to achieve energy saving. The goal is achieved by using mathematical techniques from game theory to maximize energy saving with fair distribution of the gains. The node nelection accesses context information about the channel conditions to other mobile terminals and base stations, which are collected through beacon exchange described above in the *ClusterManager*.

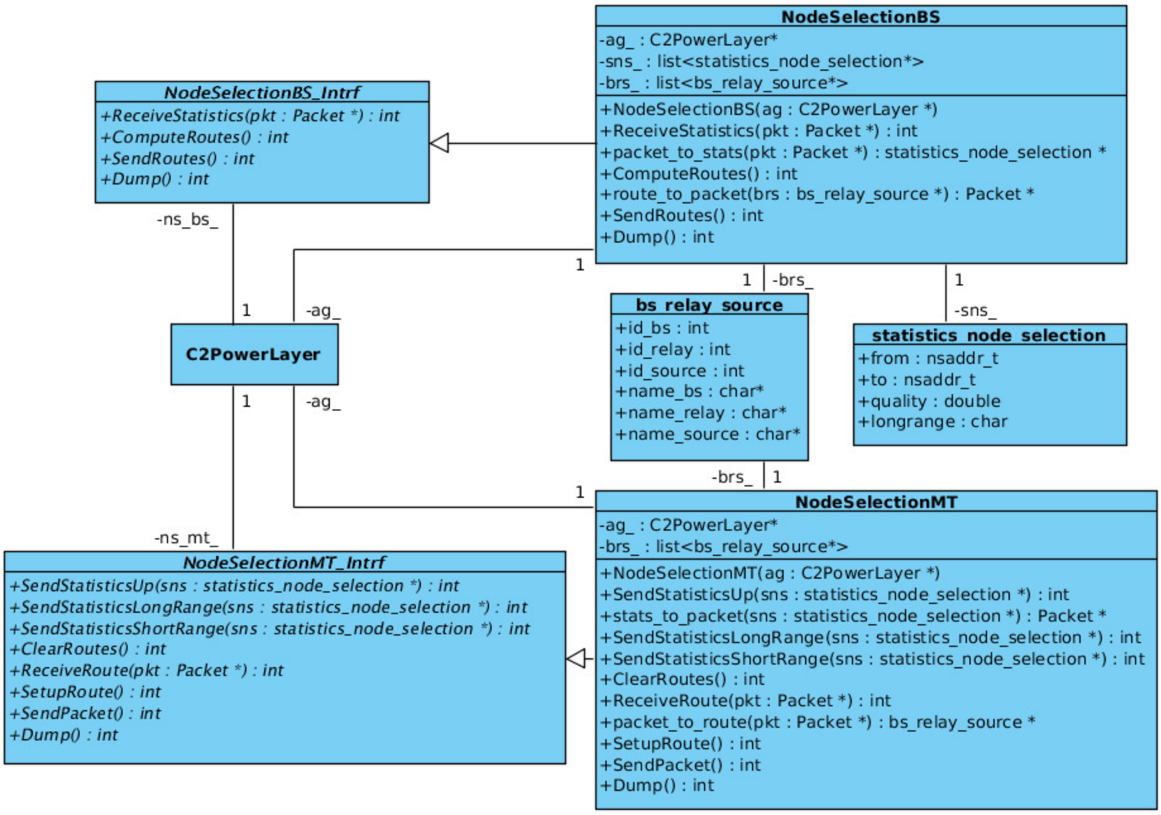


Figure 3-7. Implementation of Node Selection Module

The managing of the node selection process is started as soon as the class is instantiated. It provides a set of functions to send the collected information to the Cooperation Mediator, directly by long-range RAT, or by being relayed by other mobile terminals. These functions are called by the application layer or executed iteratively by a timer. Moreover, the *NodeSelectionMT* on the MT receives the selected cooperation policy from the Cooperation Mediator on the network side and sets up its role in the topology according to the performed computation.

On the base station side, the *NodeSelectionBS* receives information about the conditions of the short-range and long-range channels of the topology. Two methods, *ComputeRoutes()* and *SendRoutes()*, are called by the application layer or repetitively through a timer, to respectively use the collected information to compute an optimal node selection and send the computed routes to MTs.

3.4 Relay Selection Module

RelayingManager is responsible for deciding whether the relaying is available and if it is justified to be performed. It uses *ClusterManager* to obtain access to *NeighbourList* object. RelayingManager class is a general interface implemented by MultiRadioRelayingManager.

In case C2POWER application receives a packet from upper layers, *MultiRadioRelayingManager* is obliged to determine if it should be sent directly via long range device or by using an intermediate relay via short range interface. Relay candidates are gathered in *NeighbourList*. The decisive algorithm takes into account data-rates on direct long-range interface path and rates on short-range device to relay and long-range device from relay to base station. If total transmission time might be minimized by using relayed transmission, the relaying is performed. In order to select best relay candidate from the neighbor list, the algorithm checks data rates, if some neighbors can provide the same data rate, SNR is compared, and finally time stamp in order to use the most up-to-date neighbor.

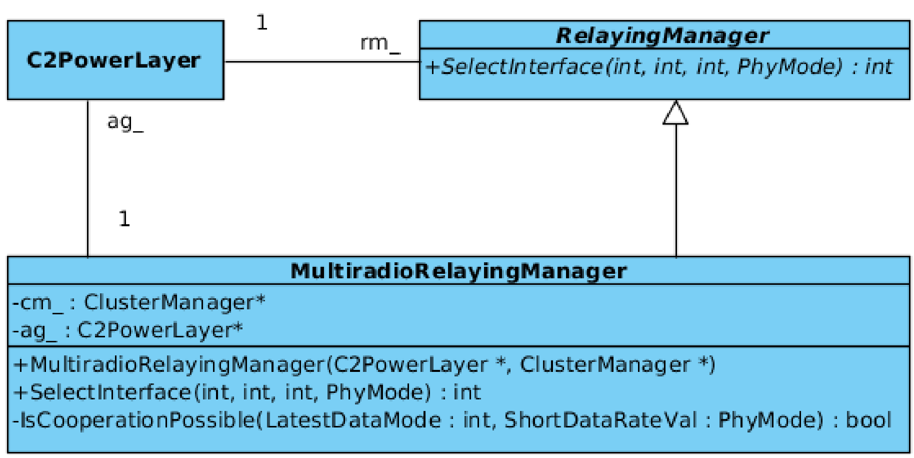


Figure 3-8.Implementation of Relay Selection Module

MultiRadioRelayingManager provides a main function *SelectInterface(.)*. The function returns the ID of the radio device, which has been selected in order to send a packet using either direct or relayed transmission. The *IsCooperationPossible()* private method, called from the *SelectInterface(.)* method, is the actual place where the policy described above takes place.

3.5 Energy Aware Routing Module

In Figure 3-9 the class diagram for the routing component is shown. The *C2PowerLayer* class is aware about the methods defined by the interface *RoutingManager_Intrf* which is implemented by the *RoutingManager* class which is also associated with the *C2PowerLayer* in order to have access to its functionalities described earlier in this section.

The *RoutingManager* class is associated with the *RDPTable* class, with the instance *rdpTable*, in order to handle route discovery packets.

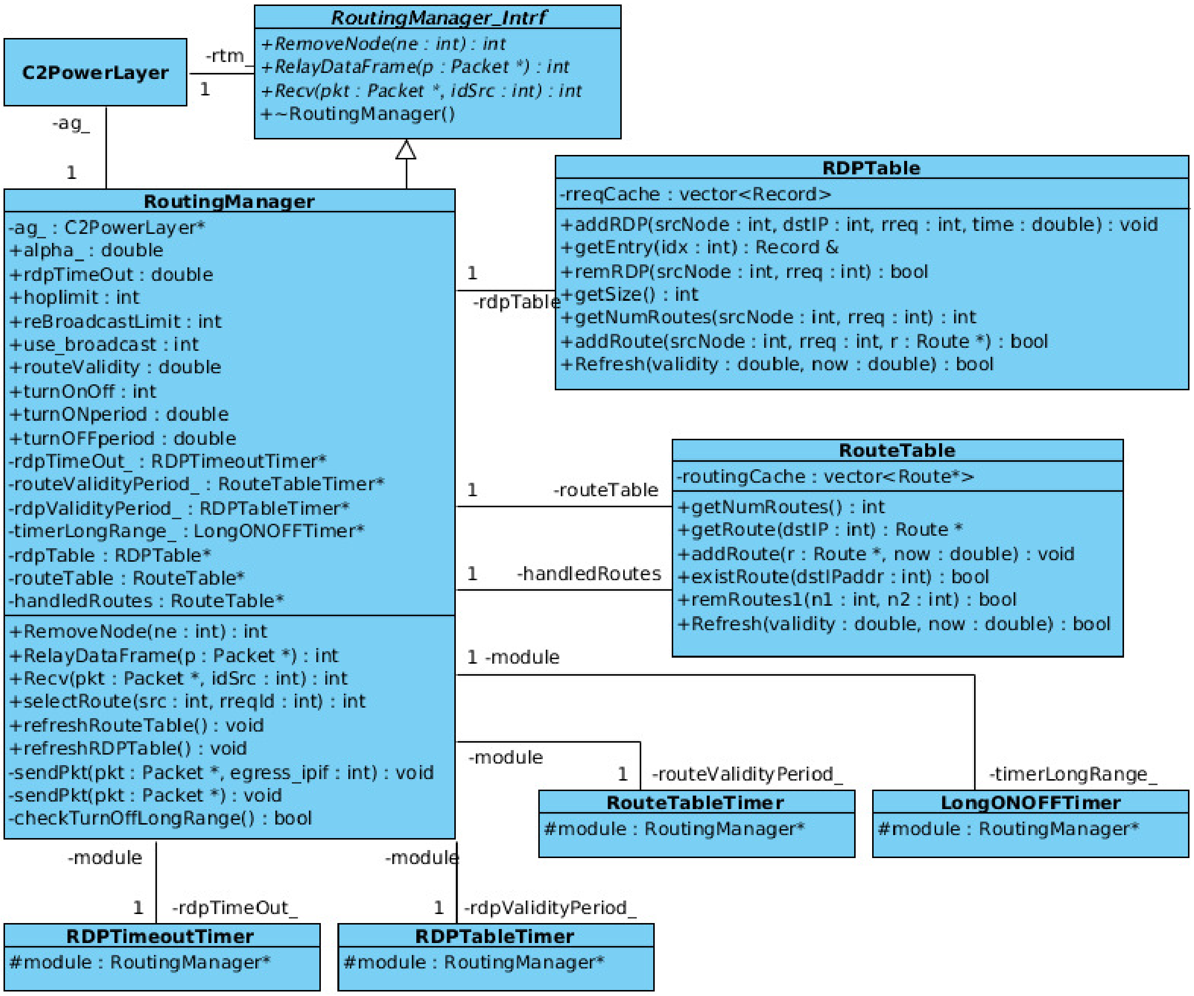


Figure 3-9. Implementation of Energy Aware Routing Module

Two associations with the class *RouteTable* exists; although it is possible to handle all routes with one table we preferred to handle routes that belong to the source node within *routeTable* instance, and routes handled by a relaying node within *handledRoutes* instance.

Figure 3-9 highlights also timers used within the routing component. *RDPTableTimer* and *RouteTableTimer* are used in order to keep *RDPTable* and *RouteTable* updated erasing outdated entries.

RDPTimeoutTimer is used in order to define a time window in which RDP packets are accepted at the destination node. The *RDPTimeoutTimer* is started when a new RDP timer is received at the destination. When this timer expires *selectRoute(.)* method is called and a route reply packet is sent toward the destination.

As said before, *RoutingManager* class implements methods defined in *RoutingManager_Intrf* interface. Those methods and their role within the C2POWER core are explained below.

- ***RemoveNode(ne : int);***

This method is called from the C2POWER layer upon a notification of a leaving neighbor from the *ClusterManager*. The implementation of this method performs a check on the consistency of the stored routes. If a route contained in *handledRoutes* uses the removed node as the next hop, a route error message is triggered towards the source of the route. On the other hands, if the *routeTable* contains a route having the removed node as the first hop, the route is removed.

- ***RelayDataFrame(p : Packet*);***

This method is called by the *C2PowerLayer* when it wants to relay a data packet using the C2POWER routing component.

- ***Recv(pkt : Packet*, idSrc : int);***

This method is called by the *C2PowerLayer* when it receives a packet that belongs to the C2POWER routing component.

The *C2PowerLayer* is also responsible to set parameters for the routing component through the method *command()*. These parameters are listed in the table below.

Table 3-3. Parameters for the C2POWER routing component

Parameter	Type	Description
*alpha_*	double	Define the α value to use during the route selection process
rdpTimeOut	double	Define how long is the time window for *RDPTimeoutTimer*
hoplimit	int	Put a threshold on the length of a route
reBroadcastLimit	Int	Put a threshold on the number of times that a node can rebroadcast a RDP with the same RDP_ID.
use_broadcast	int	By default RRP and REP packets are sent as UNICAST packets. We optionally added the feature to send them as broadcast packets.
routeValidity	double	Define how long a route is valid

3.6 Relay-based Routing Module

In Figure 3-10, the class diagram for the relay-based routing is shown. C2Power layer is aware about the interface *RelayRouting_Intrf* which is implemented by the *RelayRouting* class. A new beaconing system was implemented to meet the beaconing system for the relay routing described in section 2.6. For this reason *RelayRouting* module has its own functions and parameters to handle beacon packets:

- *SendBeacon()* and *RecvBeacon(Packet*)* to send or handle a beacon packet
- *StartBeaconing()* and *StopBeconing()* to allow a node participating in the cooperation.

Moreover, the class *C2Power_nbl_relrout* has been introduced in order to handle neighbors. It is responsible to keep update the neighbor entries and to select the best node to relay data packets.

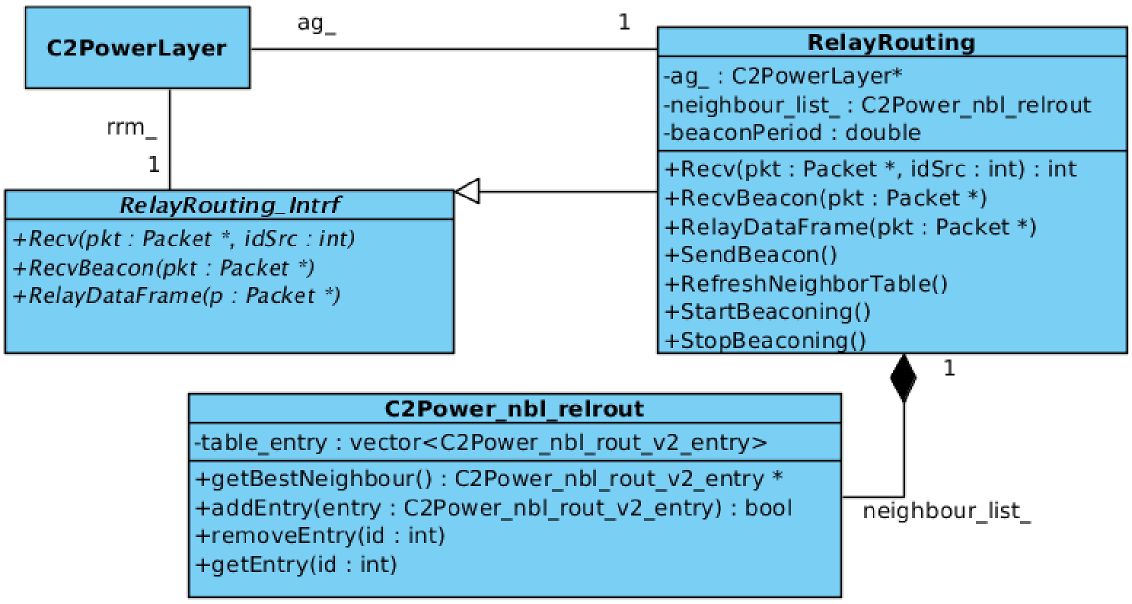


Figure 3-10. Implementation of Relay-based Routing Module

The function *RelayDataPackets(Packet*)* is a function dedicated to the handling of data packets to be relayed. As explained in section 3.1.1, it is called by the *C2PowerLayer* class when a data packets is received with the field *type=C2POWER_RR_DATA*. Alternatively, it can be called upon a reception of a data packet coming from the application layer. In both the cases, the *getBestNeighbour()* method of the class *C2Power_nbl_relrout* is called in order to find the most suitable relay following the strategy described in section 2.6.

4. Simulation scenarios

The objective of this deliverable is to describe the simulation tool developed within WP5, and present simulation results for the C2POWER proposed cooperative algorithms. More precisely, this part of the project is focused on the energy saving exploiting cooperative short-range communication. For this topic the project is focused on two different kinds of scenario:

- ***Scenario 1:*** Power saving strategies using short range cooperative clusters in homogeneous networks;
- ***Scenario 3:*** Power saving strategies using short range cooperation among heterogeneous nodes

In this section we want to introduce the main aspects taken into account in order to perform simulations for both of these scenarios.

4.1 Scenario 1

In this scenario, mobile terminals take advantage of the good channel conditions and low energy requirements of the short range communications. The detailed description of the C2POWER scenarios was already presented in the previous deliverable D2.2 [11]. In this section, we elaborate on the details of C2POWER scenario 1 and extend that to describe the simulation scenario. We first describe the benchmark scenario, i.e. scenario with no C2POWER capabilities. The benchmark scenario will be used as the reference baseline in order to compare the performance of the proposed solutions. Following the benchmark description, we describe C2POWER cooperative scenario (C2POWER scenario 1). At the end of the section, we describe the simulation environment. The performance of the proposed algorithms is analyzed in different environments, to evaluate the impact of different factors on the performance of proposed algorithms.

In all considered scenarios of C2POWER, heterogeneous environment is considered. Heterogeneous environment indicates the availability of more than one wireless technology for mobile devices to use; hence all MTs are equipped with multiple interfaces. In WP5, the cooperation scenario, at least two different interfaces exist on all mobile terminals. One of the two interfaces is a long range (LR) technology and the other interface is a short range (SR) technology. In C2POWER, LR technology refers to infrastructure based network, where MTs have to communicate directly with a BS or an AP. Within the project, we use SR communications to refer to the technology used for cooperative communications between MTs. Using SR interfaces, MTs are able to communicate between them directly without going through a wired BS or AP. Within C2POWER project, we consider different technologies for LR and SR. Moreover, certain technologies can be used for both LR and SR, as long as the technologies support peer to peer communications and infrastructure based communications. While some technologies like WiMAX, UMTS and LTE are considered LR technologies, others are considered SR technologies such as WiMedia and Bluetooth. On the other hand, WiFi is used as SR or LR in C2POWER, since WiFi has the capabilities of providing infrastructure-based communications through an AP, as well as peer to peer communications through its ad hoc mode. Our simulations consider WiMAX or WiFi as LR technologies and WiFi as the SR technology.

4.1.1 Benchmark Scenario

This scenario represents the benchmark. C2POWER performance is evaluated against the performance of this scenario. In this scenario, MTs do not use cooperation between themselves; hence MTs are usually connected to the LR BS or AP. Figure 4-1 represents a snapshot of the benchmark scenario. In the figure, multiple MTs are available in the vicinity of each other, but they do not cooperate due to the absence of C2POWER capabilities. All MTs use the LR interface to communicate directly with the BS (AP). MTs can be equipped with SR interfaces, as well. Current smart phones require users to be aware of available technologies and decide which interfaces should be ON at any specific time; hence SR interfaces can be in any modes. Based on that, the simulations results consider different modes of the SR interfaces, to show the effect of having a second interface on the mobile terminal. Moreover, this is used to study the effect of the context awareness in C2POWER. Context awareness provides information about available technologies and neighboring nodes in the vicinity, which potentially allows MTs to turn OFF (switch to SLEEP mode) interfaces with no use at certain point of time. This should result in significant energy saving, since these interfaces are not wasting energy scanning.

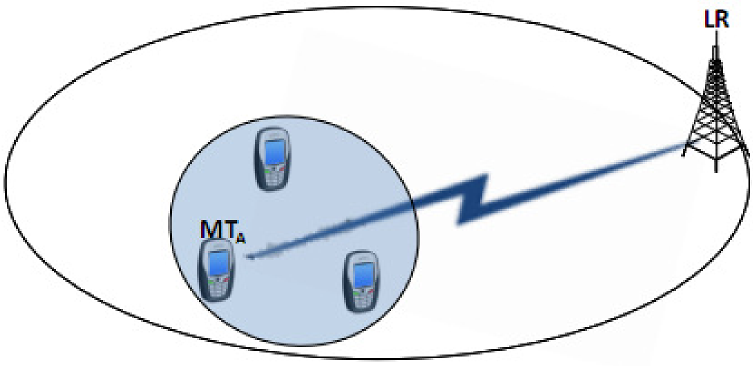


Figure 4-1. Benchmark scenario: No cooperation. All MTs communicate directly with the Base Station.

4.1.2 C2Power Cooperative Scenario

In C2POWER Scenario 1, MTs take advantage of the energy efficiency and the good channel conditions of the short range communications. MTs exploit short range communication capabilities to reach the LR BS consuming less energy. In Scenario 1, nearby MTs form a cooperative cluster. MTs with data to transmit, sends their data through other cluster members using short range communications. An instance of short range communication between two MTs is shown in Figure 4-2.

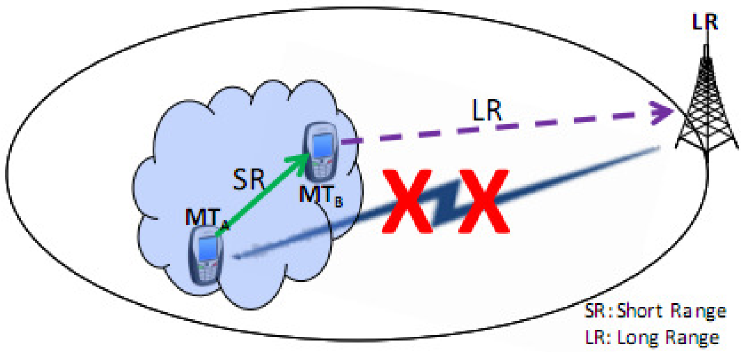


Figure 4-2. C2Power Scenario1: Energy saving using cooperative short range cooperation.

In the figure, MT$_A$ has data to send on the uplink toward the BS. The channel between MT$_A$ and LR BS is experiencing bad conditions. This can be due to the fact that MT$_A$ is too far from LR BS or that MT$_A$ is behind a building or some metal blocks, which renders the channel bad. Due to the bad conditions of the direct channel to LR BS, MT_A searches for more energy efficient connection to upload its data to the network. MT_A may as well be just running on low battery and the user of MT_A is willing to spend some money (credits) to save its battery power. At the same time, MT_B can be willing to sacrifice some of its energy to gain some credits. Moreover, MT_B can have high battery level or can be connected to a power supply, which means MT_B is not so concerned with energy consumption.

Using the context information available through C2POWER, MT_A is able to find out about MT_B in its vicinity. MT_A and MT_B then perform cluster formation, and after negotiation decide that MT_B relays data from MT_A toward LR BS.

It is worth elaborating that the connection from MT_A to MT_B is using SR communication, while the connection from MT_B to LR BS is using LR communication. The case shown in Figure 4-2 is the simplest case with two MTs cooperating. More generalized cases are considered in the simulations, with multiple nodes with data to transmit (sources) and multiple relays.

4.1.3 Simulation environment

The performance of the proposed algorithms is analyzed in different environments, to evaluate the impact of different factors on the performance; hence multiple simulation runs will be conducted with different conditions. In this section, we discuss different simulation parameters, which define the simulation environment for scenario 1.

Since we are investigating cooperation in the short-range, in the simulations all the MTs are assumed to reside inside the coverage area of a long range BS/AP. On the other hand, the simulation area can be set wider than the transmission range of the cooperative technology. In this situation, more than one-hop cooperation can be observed employing routing algorithms. All MTs have a LR interface, as well as a SR interface; hence MTs are capable of communicating directly with the wired BS/AP or communicating within each other through short range communications.

Moreover, the benchmark and the C2POWER scenario will be simulated considering the same LR conditions and the same kind of traffic for all the simulations.

Interfaces and technologies

In all simulation of Scenario 1, two interfaces are considered. Each MT is equipped with one LR interface and one SR interface. The LR interface will be either WiMAX or WiFi interface, while the SR interface will be a WiFi or a WiMedia interface. Energy consumption is considered only on these two wireless interfaces. It is assumed that the considered MTs have only two interfaces or other interfaces are turned OFF. This results in fair comparison between the benchmark scenario and C2POWER enabled scenario.

Number of nodes and Base Stations

C2POWER is mainly concerned with the energy savings due to the use of cooperative communications compared to the use of direct LR communications toward BSs or APs. In our simulation of C2POWER cooperative scenario, we consider only one BS/AP. All MTs reside within the coverage of one LR BS/AP.

Multiple nodes have to exist for cooperation to be performed. A viable question is how many nodes should be included in a cooperative cluster. Simulations should be carried out to find what a good size of a cooperative cluster is; hence different number of nodes will be considered in simulations. The results of simulation should determine whether more or less nodes are more beneficial. The number of nodes varies from the simple case of two nodes (One source node and one relay) and increases till cooperation is not useful due to expensive signaling.

Power consumption values

Since we are investigating the energy saving achievable using cooperation, an important thing to define is the power consumption values to be used in the simulations. In ns-miracle, every interface counts the time spent in transmission, reception, idle, and sleep states. Considering the time spent in each state and the associated power consumption, the consumed energy is calculated.

Table 4-1. Power consumption values for each state [W] (Values are taken from [12] and [13])

Technology	TX	RX	IDLE
WiFi (SR cooperation)	0.890	0.890	0.256
WiFi (LR infrastructure)	1.900	1.340	0.110
WiMAX	2.409	1.485	0.660

Table 4-1 shows the power consumption values, used for all simulations, for each interface. Please note that the SLEEP state is missing, in fact, this is because the actual implementation of physical interfaces within the NS-MIRACLE framework does not consider the SLEEP state.
Given that the C2POWER scenario has two active interfaces (SR and LR), while the benchmark scenario employs only one interface (LR), we can forecast that for the C2POWER scenario the absolute value of the energy consumed in the system will be probably increased with respect to the benchmark scenario. Thus, the aim of the simulations presented later in this deliverable is to show that we can achieve a gain in terms of *energy efficiency* of the system. In other words, we want to observe whether the goodput gain achieved by the cooperation can improve the energy efficiency in terms of Joules per Mb. Because of the above considerations, the obtained energy efficiency gain can be considered as a lower bound of the real gain obtainable with energy efficient MACs.

Multi-rate capabilities and channel condition

Both SR and LR technologies are assumed to have multi-rate capabilities, which is a common property of all technologies considered in simulations. Usually, different technologies achieve different data rates, by using different modulation schemes. Whenever the channel conditions are good, better modulation schemes are used which allow higher data rates to be achieved. In our simulations, we assume to simulate the channel condition assigning data-rates (or modulation scheme in case of WiMAX) randomly on the basis of some distribution.

In Table 4-2 we summarize the data-rates values employed in the ns-miracle implementation for WiFi and WiMAX interfaces. For the WiFi interface we can select directly the data-rate values to be employed for a deployed interface. On the other hands, within the WiMAX implementation we can select the modulation used for the OFDM system.

Table 4-2. Values of data rates for used interfaces as implemented in ns-miracle

WiFi
DataRate
6 Mbps
9 Mbps
12 Mbps
18 Mbps
24 Mbps
36 Mbps
48 Mbps
54 Mbps

WiMAX	
Modulation - (Code rate)	**Data Rate**
BPSK - (1/2)	3840000 bps
QPSK - (1/2)	8029090 bps
QPSK - (3/4)	18327272 bps
16QAM - (1/2)	16407272 bps
16QAM - (3/4)	37832727 bps
64QAM - (2/3)	44218181 bps
64QAM - (3/4)	56029090 bps

Traffic models

In C2POWER project, we are concerned with energy savings through cooperative communications or smart handovers. In cooperative communications, excessive delays can occur due to the multi-hop nature. Some applications may not be suitable for cooperative communications. Based on that, in our simulations, we focus on data transmissions instead of voice calls.

Thus the traffic model employed in the simulations is a CBR traffic simulating, for example, a real-time multimedia transmission. The main parameter characterizing CBR stream is the bit-rate of the data flow. The CBR traffic uses UDP as the transport protocol, since it is the obvious choice to perform best-effort data transmission, which is suitable for multi-media transmission which does not tolerate re-transmissions. The UDP is configured to use constant packet size.

User mobility

For cooperation to be successful, MTs need to be stationary or have low mobility. Stationary MTs will be considered in the initial simulation runs, which should be a good study of the efficiency of the proposed algorithms. Stationary environments are common in cases like users using their mobile devices at coffee shops, restaurants and airports.
In the more advanced simulation runs, mobility models will be considered in order to test the effect of low mobility on the performance of the algorithms.

4.2 Scenario 3

In the near future, networks will consist of numerous wireless devices and mobility plays the key role in this scenario. Moreover, the use of heterogeneous technologies is an important goal in order to achieve network coverage and even devices cooperation. Obviously each technology has different intrinsic constraints such as frequency bandwidth (i.e. data rate) and radio coverage. Those restrictions can put a limit in terms of both mobility freedom (speed of users) and services.

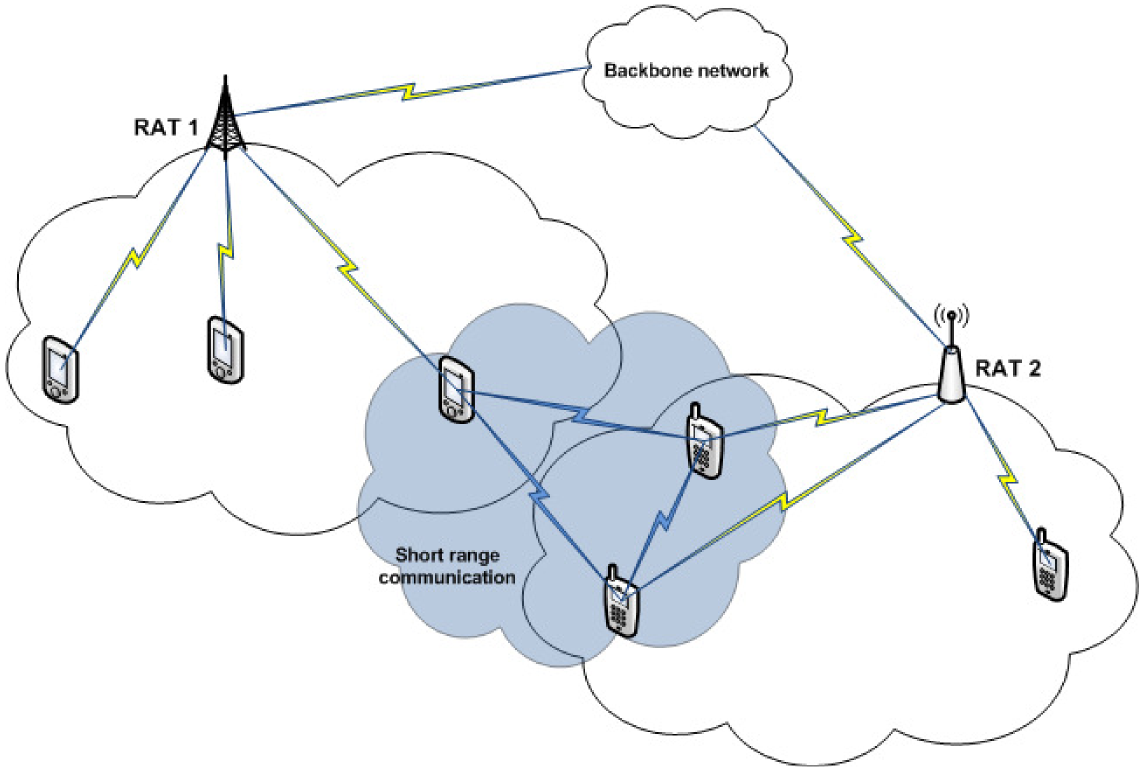


Figure 4-3. Illustration of C2Power Scenario 3

Figure 4-3 presents an example illustration of C2Power Scenario 3. It comprises heterogeneous networks with 2 radio access technologies that are additionally connected to a backbone network; and possible short range wireless communication technology to employ C2Power cooperation between nodes.
Next we describe benchmark and cooperative scenarios, where different cases with different wireless technologies are taken into account.

4.2.1 Benchmark Scenarios

Benchmark scenario nC2P I

In Figure 4-4 a case **nC2P I** is presented. It refers to the simplest benchmark situation that comprises nodes only with single radio interfaces. Therefore either WiMAX or WiFi users within their own networks generate traffic separately and there is no possibility to perform vertical handovers between the networks.

Benchmark scenario nC2P II

Figure 4-5 presents a case **nC2P II.** The case includes nodes that are distributed among two wireless networks. Additionally all nodes employ two radio interfaces, WiMax and WiFi or WiFi and WiFi. However, the nodes do not implement any cooperative techniques, hence by having second radio interface (WiFi) extra energy is spent on maintaining the second interface in, at least, idle state.

Benchmark scenario nC2P III

In benchmark case (Figure 4-6) **nC2P III** a similar situation to nC2P II is taken into consideration. Namely all nodes have two radio interfaces albeit they do not cooperate between themselves. Either WiMAX and WiMedia nodes or WiFi and WiMedia nodes are sources of traffic in the scenario. In such a case extra energy expenses are devoted to maintaining WiMedia interface. Since WiMedia radio interface based on UWB technologies is less energy consuming than WiFi technologies, nC2P III is expected to be more efficient than nC2P II case.

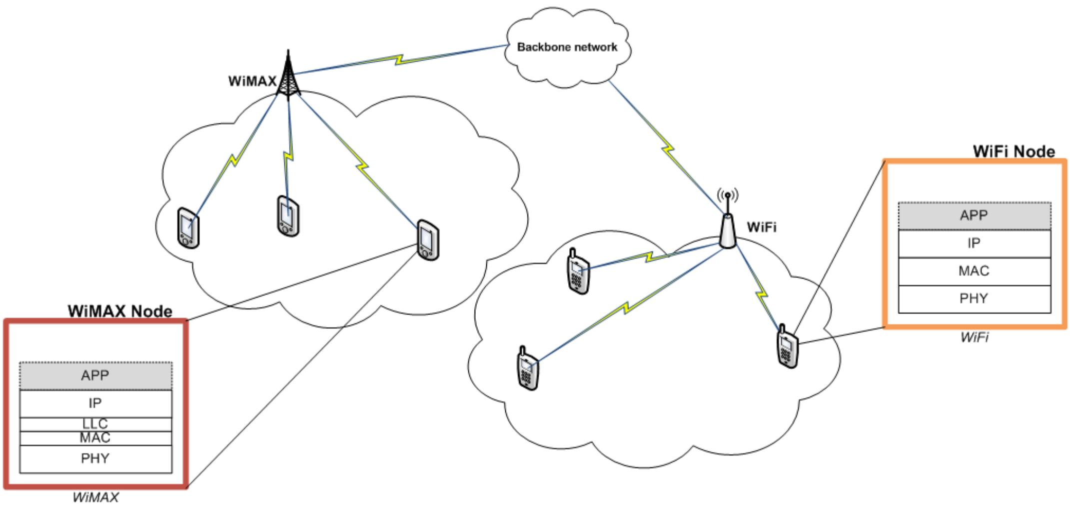


Figure 4-4. Benchmark scenario nC2P I

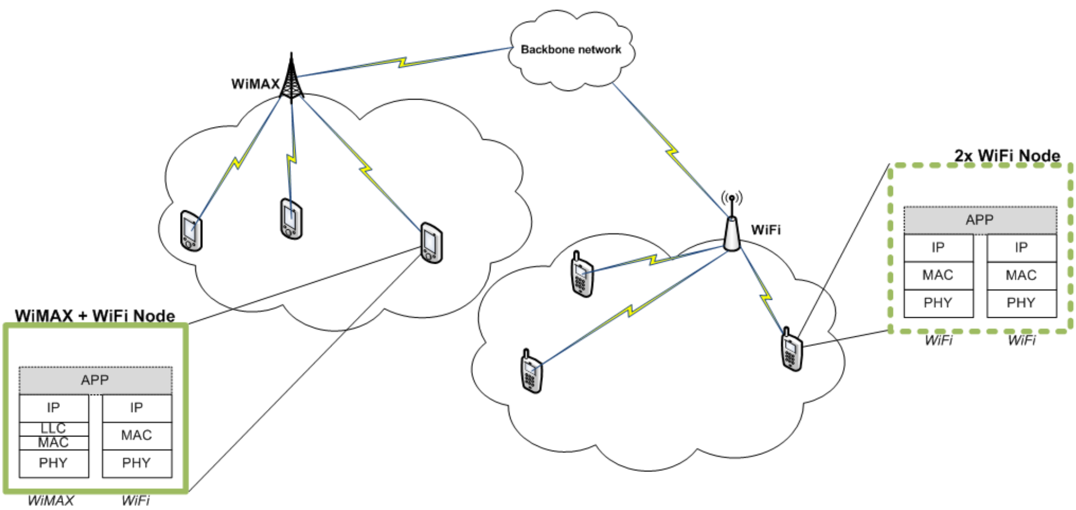


Figure 4-5. Benchmark scenario nC2P II

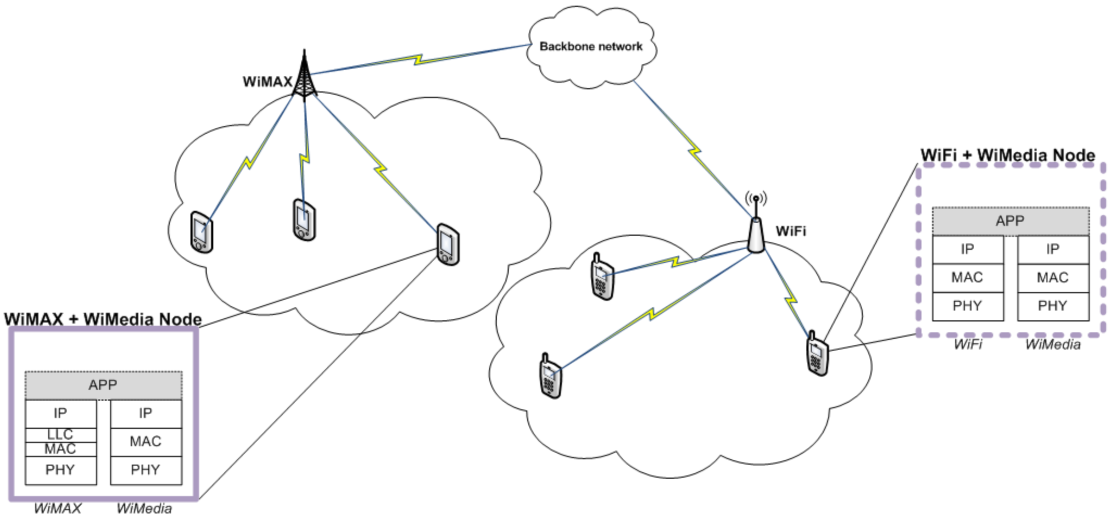


Figure 4-6. Benchmark scenario nC2P III

4.2.2 C2Power Cooperative Scenarios

Cooperative scenario C2P I

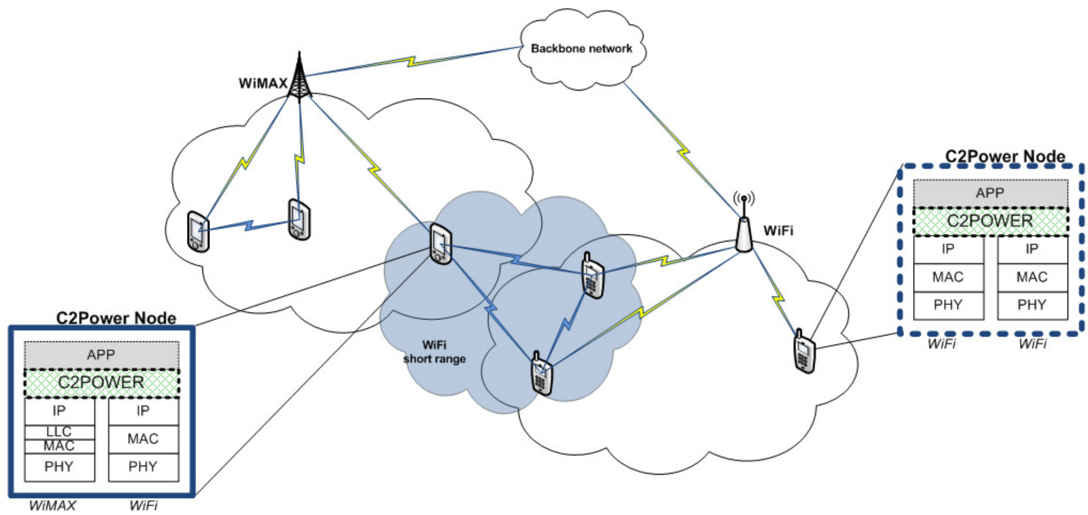


Figure 4-7 C2Power cooperative scenario C2P I.

Cooperative scenario C2P II

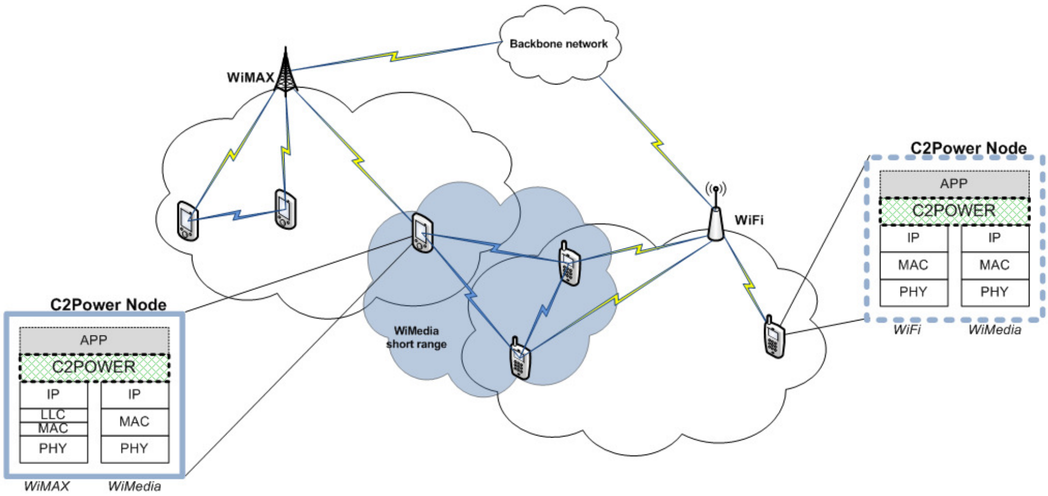


Figure 4-8 C2Power cooperative scenario C2P I.

4.2.3 Simulation environment

Due to the random nature of user behavior and network parameters, the performance of the proposed algorithms is analyzed in different environments. In order to observe the influence of different factors and system's elements on the general network performance, multiple simulation runs will be carried out with different starting conditions. This subsection discusses a general simulation environment setup, where different simulation parameters are defined and described.

We consider a system comprising two heterogeneous wireless networks representing LR communications technologies. Without loss of generality, in the simulations all the MTs are assumed to reside inside the coverage of a long range BS/AP. That means, every MT is equipped with at least one radio interface connected to a dedicated RAT. Moreover, in some cases MTs also have SR interface that could be used for C2Power cooperation techniques. Therefore, C2Power capable MTs have an opportunity to communicate directly with the BS/AP or communicating via relayed transmission, where short range radio interface is employed. All benchmarks non-cooperative cases will be simulated considering the same LR conditions and the same kind of traffic for all the simulations; moreover, SR interface if used, will remain permanently in idle mode in order to show potential energy gains compared to unused MT's resources.

Interfaces and technologies

In all simulations of Scenario 3, at most two interfaces per node are considered. Each MT is equipped with one LR interface and in some cases it uses additional SR interface. The LR interface will be either WiMAX or WiFi interface, while the SR interface will be a WiFi or a WiMedia interface. Energy consumption is considered only on these two wireless interfaces. Depending on defined cases, it is assumed that the considered MTs have only LR interface or LR and SR interfaces that are turned ON and spends their energy at least in idle mode. Simulation results will provide fair comparison between the benchmark cases and cooperative-based C2Power techniques.

Number of nodes and Base Stations

C2Power is mainly concerned with the energy savings due to the use of cooperative communications compared to the use of direct LR communications toward BSs or APs. In our simulation of C2Power cooperative scenario 3, we consider heterogeneous RATs, where two BSs/APs are employed. All MTs reside within the coverage of one of two LR BSs/APs and are randomly distributed among the two wireless networks.

In C2Power cases, multiple nodes have to exist in order to guarantee a possibility for potential cooperation. A general issue addresses the question that how many nodes should be embraced in a cooperative cluster. Simulations should be carried out to find what a good size of a cooperative cluster is; hence different number of nodes will be considered in simulations. The results of simulation should determine whether more or less nodes are more beneficial. The number of nodes should vary starting from the simple case of two nodes (One source node and one relay) and increasing till cooperation is not useful due to expensive signaling. However, since cooperative C2Power techniques apply short-range technologies, we will consider no more than 20 nodes.

Power consumption values

Since we are investigating the energy saving achievable with the cooperation, an important thing to define is the power consumption values to be used in the simulations. In ns-miracle, every interface counts the time spent in transmission, reception, idle, and sleep states. Considering the time spent in each state and the associated power consumption, the energy consumed in calculated.

Table 4-3. Power consumption values for each state in Watts (values are taken from [12], [13] and Deliverable D2.2).

Technology	TX	RX	IDLE
WiFi	0.890	0.890	0.256
WiMAX	2.409	1.485	0.660
WiMedia	0.250	0.250	0.0003

Table 4-3 shows the power consumption values of different interfaces, used in all simulation runs. Please note that the we do not consider SLEEP state due to the fact that, the actual implementation of physical interfaces within the NS-MIRACLE framework does not consider the SLEEP state.

By having given that the C2Power scenario employs two active interfaces (SR and LR), while the benchmark scenario employs either only one interface (LR) or two interfaces (LR and SR), where SR remains IDLE, we can forecast that for the C2Power scenario the absolute value of the energy consumed in the system will be probably increased with respect to the benchmark scenario. Thus, the aim of the simulations presented later in this deliverable is to show that we can achieve a gain in terms of *energy efficiency* of the system. In other words, we want to observe whether the goodput gain achieved by the cooperation can improve the energy efficiency in terms of Joules per Mb. Because of the above considerations, the obtained energy efficiency gain can be considered as a lower bound of the real gain obtainable with energy efficient techniques.

Multi-rate capabilities and channel condition

Both SR and LR technologies are assumed to have multi-rate capabilities, which is a common property of all technologies considered in simulations (WiMax, WiFi and WiMedia). Usually, different technologies

achieve different data rates, by using different modulation schemes. Whenever the channel conditions are good, better modulation schemes are used which allow higher data rates to be achieved. In our simulations, we assume to simulate the channel condition assigning data-rates (or modulation scheme in case of WiMAX) randomly on the basis of some distribution.
Table 4-2 in previous subsection summarizes the data-rates values employed in the ns-miracle implementation for WiFi and WiMAX interfaces. For the WiFi interface we can select directly the data-rate values to be employed for a deployed interface. On the other hands, within the WiMAX implementation we can select the modulation used for the OFDM system. For the sake of simplicity, we assume that, WiMedia radio interface is based on CSMA/CA techniques and provides the same data rate values as in case of WiFi.

Traffic models

In C2Power project, we are concerned with energy savings through cooperative communications. In Scenario 3 cooperative communications we consider relaying techniques that use heterogeneous networks and might provide multi-radio relaying between the heterogeneous networks. Moreover some applications may not be suitable for cooperative communications and in our simulations devoting to Scenario 3 we focus on data transmissions instead of voice calls.

Thus the traffic model employed in the simulations is a CBR traffic simulating, for example, a real-time multimedia transmission. The main parameter characterizing CBR stream is the bit-rate of the data flow and a data portion size. The CBR traffic usually uses UDP as the transport protocol, since it is the obvious choice to perform best-effort data transmission, which is suitable for multi-media transmission which does not tolerate re-transmissions. The UDP is configured to use constant packet size.

User mobility

For cooperation to be successful, MTs need to be quasi-stationary or have low mobility. Stationary MTs will be considered in the initial simulation runs, which should be a good study of the efficiency of the proposed C2Power multi-radio relaying technique. Stationary environments are common in cases like users using their mobile devices at coffee shops, restaurants and airports, universities, etc.

5. Simulation results and discussion

In this section, we present and discuss simulation results of the common simulator considering various algorithms implemented. In the first part of the section, C2POWER scenario one is investigated which involves short range cooperation under homogeneous networks, meaning having only one infrastructure based network covering the whole considered area. In the second part of the section, results for C2POWER scenario 3 are presented. In this scenario, we study achievable C2POWER energy saving gains within an area covered by two different infrastructure based network covering the considered area.

5.1 Scenario 1

5.1.1 Evaluation of energy efficient node selection mechanisms

Figure 5-1 to Figure 5-4 show the energy saving gain (percentage) obtained from our proposed cooperative game theory solution for the node selection algorithm. The gains are due to employing correlative strategy using a single intermediate relay. All the results represent average values over 10 replicas with random seeds. As for the simulation scenario, nodes are randomly deployed. There is only one AP in the centre of the cell and the MTs are deployed uniformly in a rectangular cell area.

Figure 5-1 and Figure 5-2 illustrate the simulation results for simulation areas of 20mx20m and 60mx60m, respectively. Both of these results are for CBR traffic with a rate of 20 packets per second with packet size 1024 bytes and 100s traffic duration. Two types of traffics are considered: Traffic1 in which the simulation time (100s) equally divided among the mobile nodes (as time slots); in these time slots, participating nodes transmit one by one—only one node transmits at each time slot. On the other hand, in Traffic2, node 1 transmits from slot 1 on, node 2 transmits from slot 2 on, etc. That is, at the last time slot, all the nodes are transmitting.

As seen from these results, there is an optimal density of nodes for which the cooperation gain is maximum. For instance, in both Figure 5-1 and Figure 5-2, having 10 nodes, results in maximum energy saving gain. On the other hand, either having low density of nodes or having very high density of nodes result in poor energy saving gains. This is because, for the low density of nodes, the probability of finding a good relay is rare, while for a dense area, the collisions from contending nodes to capture the medium essentially deteriorate the cooperation gain.

Another observation from Figure 5-1 and Figure 5-2 is that for lower number of nodes the energy saving gain of traffic 1 exceeds the gains of the traffic 2, while as the number of nodes grows, the energy saving gain obtained from traffic 2 outperforms the gain from traffic 1.

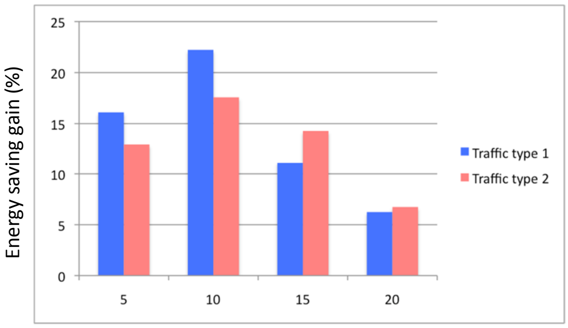


Figure 5-1. Energy saving gain (percentage) obtained by efficient node selection algorithm vs. number of nodes (area=20x20m, rate=20pps)

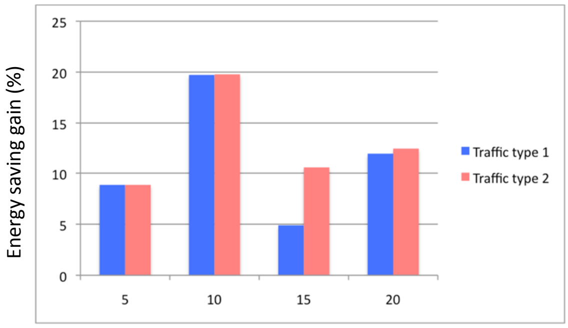


Figure 5-2. Energy saving gain (percentage) obtained by efficient node selection algorithm vs. number of nodes (area=60x60m, rate=20pps)

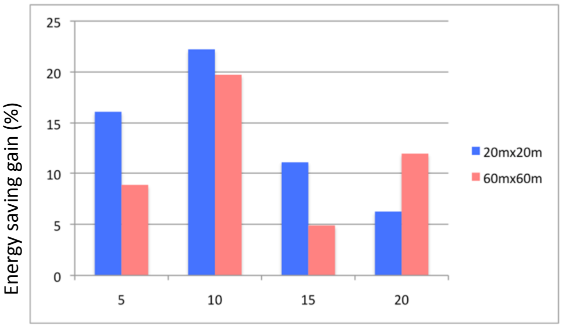


Figure 5-3. Energy saving gain (percentage) obtained by efficient node selection algorithm vs. number of nodes (Traffic type=1, rate=20pps)

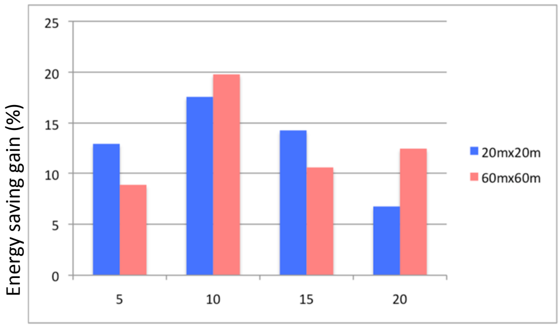


Figure 5-4. Energy saving gain (percentage) obtained by efficient node selection algorithm vs. number of nodes (Traffic type=2, rate=20pps)

Figure 5-3 and Figure 5-4 illustrate the simulation results for traffic type 1 and traffic type 2, respectively. Comparing the results of Figure 5-3 and Figure 5-4 reveals that despite of different traffic types, the trend of energy saving charts are almost similar.

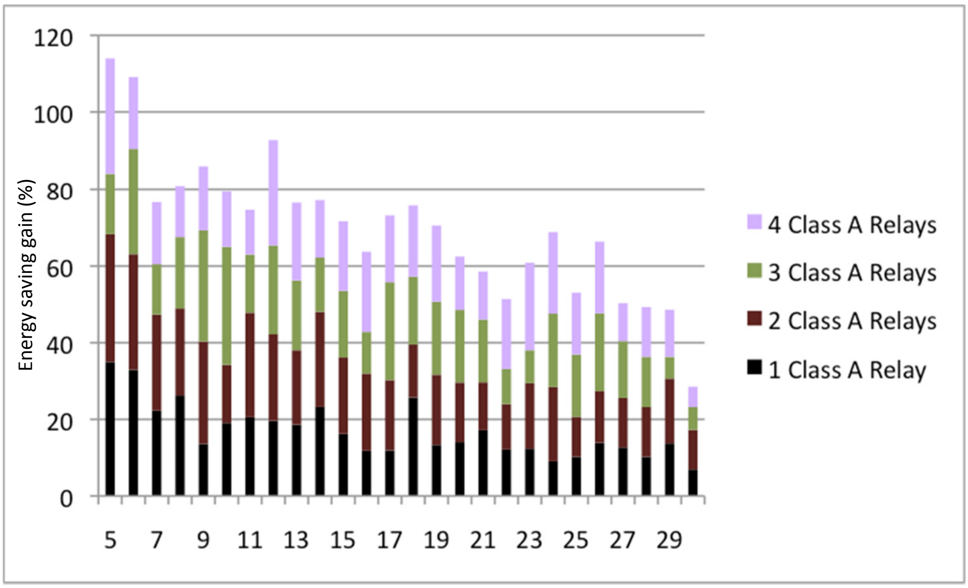


Figure 5-5. Energy saving gain (percentage) obtained by efficient node selection algorithm vs. number of nodes for different number of Class A relay nodes

Figure 5-5 provides energy saving gain as we increase the number of nodes in a simulation area of 60mx60m from 5 nodes to 30 nodes—adding one node at each step. We consider CBR traffic of type 1 and rate of 100 packets per second; the packet size and the flow duration are unchanged (1024 bytes and 100s). The figure presents the results for varying number of Class A relay nodes that are defined as those MTs having good LR channel qualities able to provide WiMAX connectivity with the rate of 54Mbps (c.f., Table 5-6). This figure demonstrates that:

- the average energy saving gain decreases as the number of nodes increases.
- the energy saving gain is more or less the same for different number of Class A relay nodes.

Table 5-1 summarizes the average energy saving gains presented by Figure 5-5 for different number of Class A relays. It validates that:

- we obtain no gain by deploying more than one MT with good LR channel quality.
- For a network with different number of MTs, on average we obtain around 17% energy saving gain.

Table 5-1. Average energy saving gain over varying number of nodes (from 5 to 30:increasing one node at each step) for different number of Class A relays—for each experiment with a given number of nodes and Class A relays, we repeat the simulation for 10 replicas with random seeds.

Number of Class A relays	Average energy saving gain (%)
1	17.06841236
2	18.8620471
3	17.05432724
4	16.98299333

In conclusion, we can summarize our comments for the cooperative game theory node selection algorithm as follows:

- To keep the computation burden of the LP problem reasonable, it is necessary to have 20 to 30 nodes in a cluster.
- For any cooperative cluster, there is an optimum density of nodes for which the energy saving gain is maximum: having lower density of nodes leads to lack of appropriate relay while having higher density of nodes introduces inefficient collisions.
- The maximum achievable gain is almost 50%, while the minimum value is around 3%.
- On average, we obtain 17% energy saving gain with the proposed node selection scheme.
- Traffic types 1 and 2 result in almost similar energy gain trends.
- For a specific cell area, the energy saving gain decreases when we increase the number of nodes.
- For the considered traffic, there is no difference between obtained energy saving gains from having a single Class A relay or more than one Class A relays.

5.1.2 Evaluation of energy efficient multi-radio cooperative relaying

Table 5-2 lists all the simulation parameters used in the simulation of the energy efficient multi-radio cooperative relaying.

Table 5-2. Summary for parameters used for simulations of Scenario 1 with multi-radio cooperative relaying

		Value	Notes
Parameters for the scenario			
Mobility of nodes		---	Static scenario
Number of nodes		2-20	
Simulation time		600	[s]
Topology		random	Infrastructure mode, WiMAX BS and multiple users are randomly distributed among static area.
LR power consumption WiMAX	Tx mode	2409	[mW] see: *Evaluating the Energy Efficiency of TCP Transmission over a WiMAX Network*, 2010 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5560108
	Rx mode	1485	
	Idle	660	
LSR power consumption WiFi	Tx mode	890	[mW] see*: CoolSpots: Reducing the Power Consumption of Wireless Mobile Devices with Multiple Radio Interfaces*, 2006, http://static.usenix.org/events/mobisys06/full_papers/p220-pering.pdf (NIC Linksys WCF12)
	Rx mode	890	
	Idle	256	
SR power consumption WiMedia	Tx mode	250	[mW] see: Deliverable D2.2
	Rx mode	250	
	Idle	0.30	
Long-range data rates		3.8-56	[Mbps] see: Table 4-2. Values of data rates for used interfaces as implemented in ns-miracle
Short-range data rates		36-54	[Mbps]
Parameters of traffic model			
Traffic Type		CBR	Every node generates traffic in the same way
Number of Traffic Sources		2-20	Every node in the system generates traffic
Data packets size		1024	[B], application layer data size
Sending rate		3000	Packets per second
Important parameters of the relaying algorithm			
Beacon period		5	[s] Beacon is broadcasted every 5 seconds
Timeout		15	[s] Nodes removes entries in their neighbour lists in case no beacon has been received within 10 seconds since the last update.

Regarding different cases, C2P represents C2Power-enabled nodes and nC2P is for non-C2Power or non-cooperative benchmark cases. In order to observe an impact of applying short-range wireless interface that might be used for cooperation, two variants are distinguished: WiFi and WiMedia. All C2Power cases embrace nodes equipped with two interfaces where C2P I and C2P II use WiFi and WiMedia technologies, respectively. In non-cooperative networks three possibilities are defined: nC2P I with nodes implementing only long-range WiMax interface; and cases where nodes has second interface, that is WiFi (nC2P II) or WiMedia (nC2P III).

Figure 5-6 illustrates energy efficiency in J/Mb for networks comprising 2-20 WiMAX users connected to the same base station.

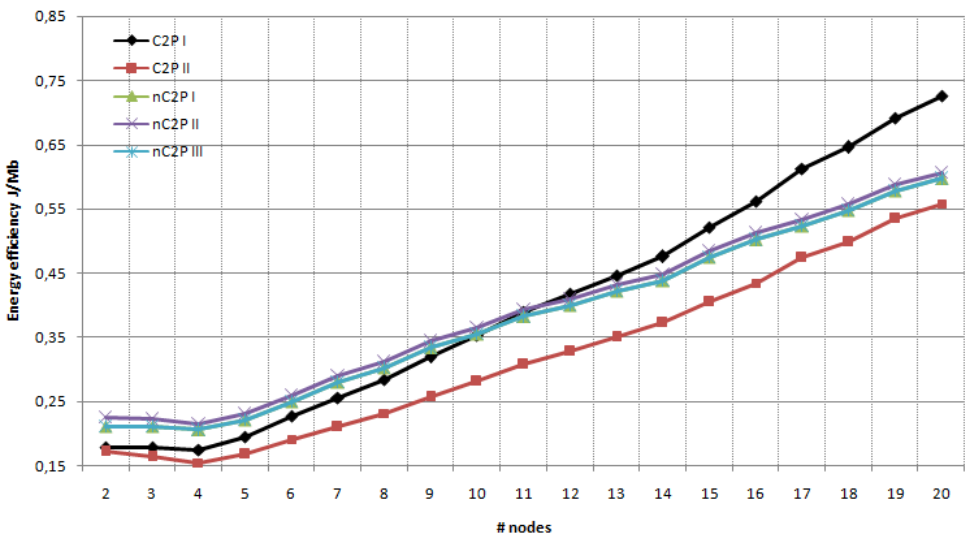


Figure 5-6. Energy efficiency in J/Mb

It might be observed that C2P I provides good energy efficiency for smaller networks up to 10 nodes. Starting from 11 nodes in the network the energy effectiveness falls below non-cooperative cases. Due to more nodes bring deployed in the system, on the one hand the probability of finding better helper node to perform cooperative relaying increases, but on the other, channel access is limited and energy expenditure for beaconing increases. Therefore it might be crucial to use completely low-power devices for short-range communication. As it is shown in Figure 5-6, C2P II, where WiMAX nodes use also WiMedia radio technology that enables C2Power techniques, provides the best energy efficiency in the meaning of J/Mb.

Figure 5-7 illustrates detailed comparison between C2P I (WiMAX-WiFI) case versus other non-cooperative setups. It is explicitly shown about 10 cooperating nodes provide relatively high efficiency in energy consumption. The gap might reach 17 % for smaller networks in comparison to nC2P I and even more than 25% if non-cooperative nC2P II cases with two-interfaced nodes but without installed C2Power techniques. Unfortunately, if more nodes exist in the system, some energy losses of using cooperation might appear. Thereby it might be valuable to apply some intelligent techniques that will help in making decisions about either cooperation or non-cooperative mode between nodes.

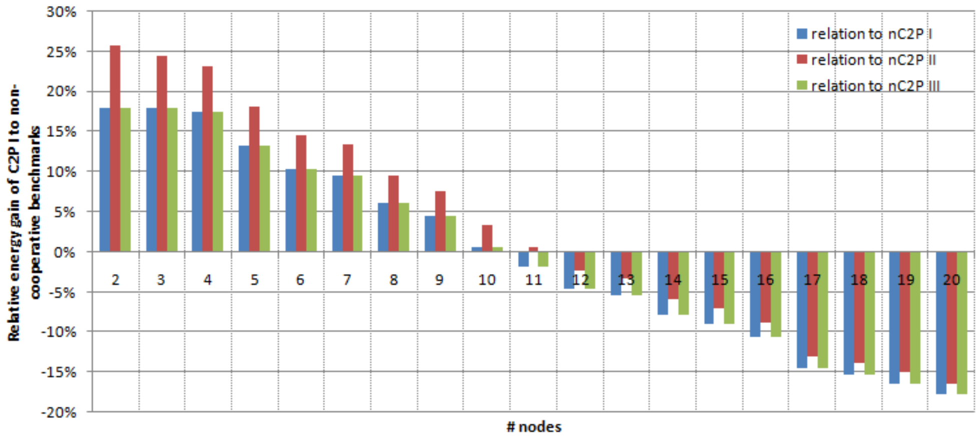


Figure 5-7. Energy gain [%] of C2P I to non-cooperative cases of Scenario 1

In Figure 5-8 energy gain of C2P II to all nC2P cases is presented. The most efficient setups comprise 4-7 nodes in neighborhood, however C2P II employing WiMedia as short-range cooperative interface provides energy gain for all network sizes with 2 to 20 nodes.

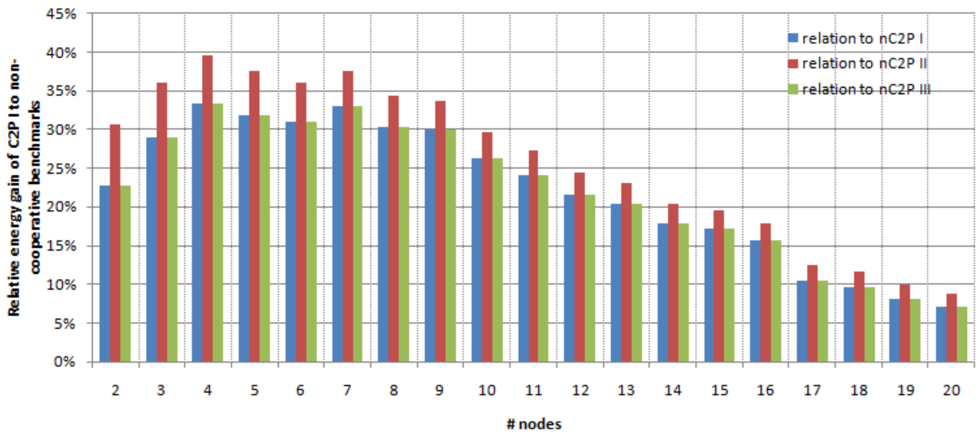


Figure 5-8. Energy gain [%] of C2P II to non-cooperative cases of Scenario 1.

Overall effectiveness of energy expenditures while applying cooperation strongly depends on a type of installed short-range wireless radio interface. Using WiFi (C2P I) provides explicit gain for smaller systems, while for larger number of nodes some losses are observed. On average, it reaches less than 1% in favor of enabling cooperative multi-radio relaying. However, WiMedia case (C2P II) guarantees energy gain for all instances separately and in average it equals to approximately 22-25%.

Figure 5-9 shows comparison between C2P I and C2P II. It can be noticed, C2P II consumes less energy than C2P I with about 4% (for networks with 2 nodes) to even 30% if 20 nodes exist in cooperation range.

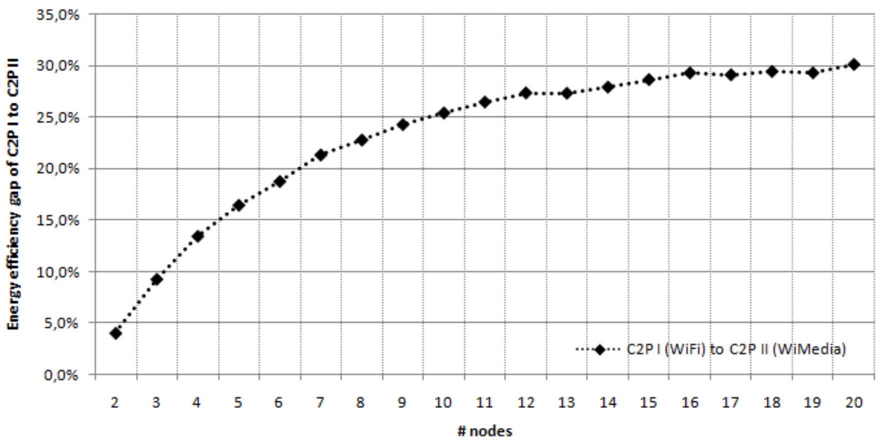


Figure 5-9. Relation of C2Power-enabled devices using WiMedia (C2P II) instead of WiFi (C2P I) as short-range interface

C2Power-enabled cooperative multi-radio relaying has a positive impact on energy efficiency seen at WiMAX base station. Herein WiMAX BS serves all nodes and its energy effectiveness is improved if cooperative techniques are applied. Figure 5-10 depicts energy expenditures at BS in C2P and nC2P cases. Cooperative C2P requires at least 12% less energy than nC2P and reaches even 30-35% for 4-10 cooperating nodes. Average C2P energy efficiency equals to 0.25 J/Mb, whereas nC2P consumes about 0.31 J/Mb that constitutes average energy gap between the cases of approximately 19%.

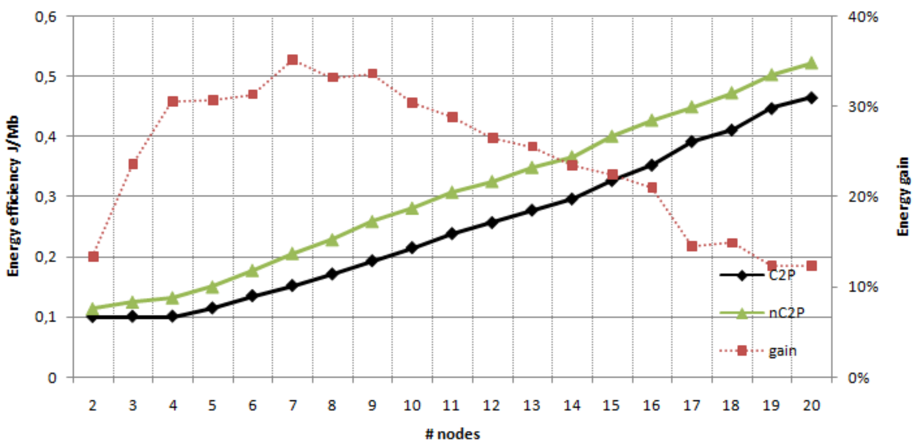


Figure 5-10. Energy efficiency in J/Mb and energy gain at WiMAX base station.

Energy effectiveness at WiMAX BS is related to better end-to-end throughput that is obtained by cooperation and multi-radio relaying. When base station serves more frames within the same time period, its energy is spent mainly for active modes rather than remaining idle. Figure 5-11 compares obtainable goodput for 2-20 nodes in Scenario 1. C2P that represents cooperative multi-radio relaying guarantees 24-35 Mb/s (averagely) whereas nC2P reaches only 20-29 Mb/s. Hence relative gap between the cases equals to 10-30% for different system's sizes, and average gap is approximately almost 22% in favor of C2P. It is worth noting that, improving goodput in the system yields better channel utilization and results in extremely better energy efficiency. Since some nodes might finish their transmission faster, other ones are granted faster channel access.

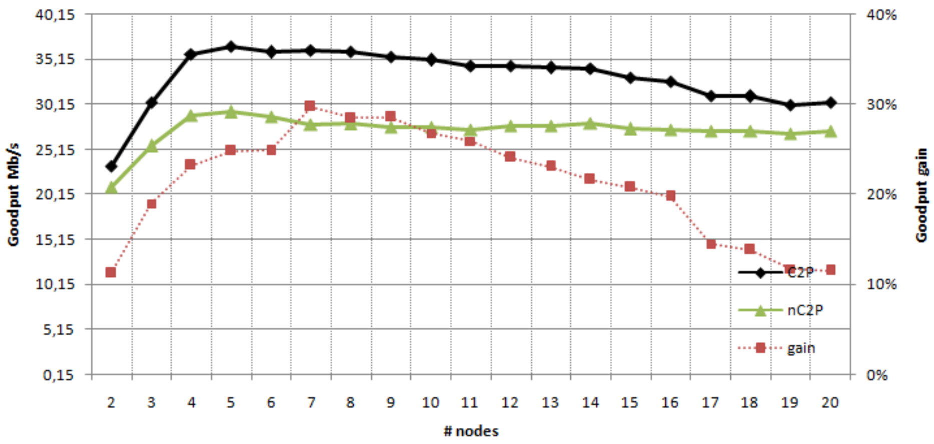


Figure 5-11. Average goodput in Mb/s.

Selected histograms of the simulation results of the energy efficient multi-radio relaying are presented in Appendix I.

5.1.3 Evaluation of routing techniques

In this section we evaluate the energy efficient routing mechanisms developed within the C2POWER project. In the first step, we simulate EAR within a scenario composed of WiFi-WiFi devices. In the second step, both routing algorithms (EAR and Relay-based routing) are simulated in a scenario composed of mobile devices with WiFi-WiMAX interfaces. The interest in a comparison between the two approaches comes from the fact that they belong to two different classes of algorithms. The EAR routing algorithm is an extension of a widely explored approach based on an on-demand routing strategy. In order to exploit energy saving, a route from the source to the destination needs to be created and maintained. On the other hands, the relay-based routing approach enables to exploit energy saving routes on the basis of on-the-fly decisions based upon the information shared between cooperating devices. Potentially, the relay-based routing, adding a little overhead to the beacons exchanged by cooperating devices, is able to exploit energy efficient routes without the need to spend resources in the route creation and maintenance. For this reason, it is more flexible and agile than a proper routing protocol.

5.1.3.1 Simulation Results with WiFi/WiFi devices

In this section the Energy Aware Routing (EAR) protocol is evaluated. In previous C2POWER deliverable D5.2 [6], EAR algorithm has been evaluated within an ad-hoc scenario. The goal of this section is twofold: first we present results of the algorithm ported within the common simulator; second, considering Scenario 1, we evaluate the achievable energy saving gain using C2Power devices with respect to the Benchmark scenario.

Simulation parameters were discussed in section 4.1.3 and are summarized for sake of clarity in Table 5-3. Using the routing algorithm a Node is able to relay its data using routes through the short-range interfaces. Within the chosen scenario, we can achieve some energy gain because of nodes having a bad channel to the AP, and thus having a low data-rate to the AP, can reach nodes with a good channel to the AP thus having a higher data-rate.

Nodes are randomly placed within the scenario area and we assumed that the entire scenario is covered by the AP. The transmission range for the short-range is 20m; however, in the scenario with the 20x20

meters area any node can reach any other node for the cooperation. Partitioned nodes (i.e. Nodes that don't have any neighbors for cooperation) transmit directly through the long-range interface.
Nodes in the Baseline scenario use only one interface, while C2Power devices have to handle two interfaces. In order to save energy spent in IDLE mode, the long-range is switched-off whenever is possible. This means that the long-range interface can be active during the discovery phase, or when a node handles a route in which it is the upload node.

Table 5-3. Summary for parameters used for simulations

	Value	Notes
Parameters for the scenario		
Mobility of nodes	---	Static scenario
Number of nodes	5-20	Randomly placed (5, 10, 15, 20)
Number of runs	10	In order to average results
Simulation time	100	[sec]
Long-range and Short-range power consumption	-	As described in section 4.1.3
Data-rates	-	Assigner randomly accordingly to Table 5-4 and Table 5-5
Transmission range for the short-range interface	20	[m]
Parameters of traffic model		
Traffic Type	CBR	- Traffic_Mode_1: one device transmitting at a time - Traffic_Mode_2: All the devices transmitting together
Sending rate	20/100	20, 40, 60, 80, 100 [Packets/sec]
Parameters of the routing algorithm		
Alpha [*]	1	We consider as best path the one using less energy per bit
Hop_limit [*]	3	Maximum number of hops for a short-range route
reBroadcastLimit [*]	2	Number of times that a node can forward a RDP
rdpTimeOut [*]	0.5	Time to wait at the AP for RDP (routes collection)
routeValidity [*]	100 [s]	Validity period for route entry

[*] - These parameters are presented in section 3.5.

Table 5-4. Deployment of long range data-rates assigned to each node to the AP

Data-Rate (Mbps)	% of Nodes
54	15
48	15
12	21
9	28
6	21

Table 5-5. Deployment of data-rates for short-range links

Data-Rate (Mbps)	% of Nodes
54	30
48	40
36	30

In Figure 5-12, simulation results for the 20x20 meters scenario are shown. Results represent the energy gain percentage, with respect to the Baseline scenario, considering the energy consumption in the whole network. Results are related to *Traffic_Mode_1* in which each node transmits without network congestion. Since the scope of C2POWER is to save energy using an intelligent packet delivery strategy, results are shown in terms of total energy consumed by nodes Figure 5-12 (a), and energy spent only in TX and RX states Figure 5-12 (b).

As we can see in Figure 5-12 (b), the routing algorithm is able to save energy spent in TX and RX states showing an almost constant gain of 50%. However, if we look at the overall energy gain, the maximum observed overall gain with this scenario is around 16%. Moreover, we can see that there are positive gains only above a certain number of nodes and above a certain amount of traffic. This behavior is due to the fact that, using C2POWER devices, we have to keep active two interfaces instead of one. This makes the energy spent in IDLE mode the predominant factor in the energy consumption (note that in the simulations both the interfaces have the same power consumption values). On the other hands, we can observe that with the increase of the traffic, the observed overall gain increases too. Since the gain in TX and RX is almost constant, this is because devices spend less time in IDLE mode if they send and receive a lot of packets.

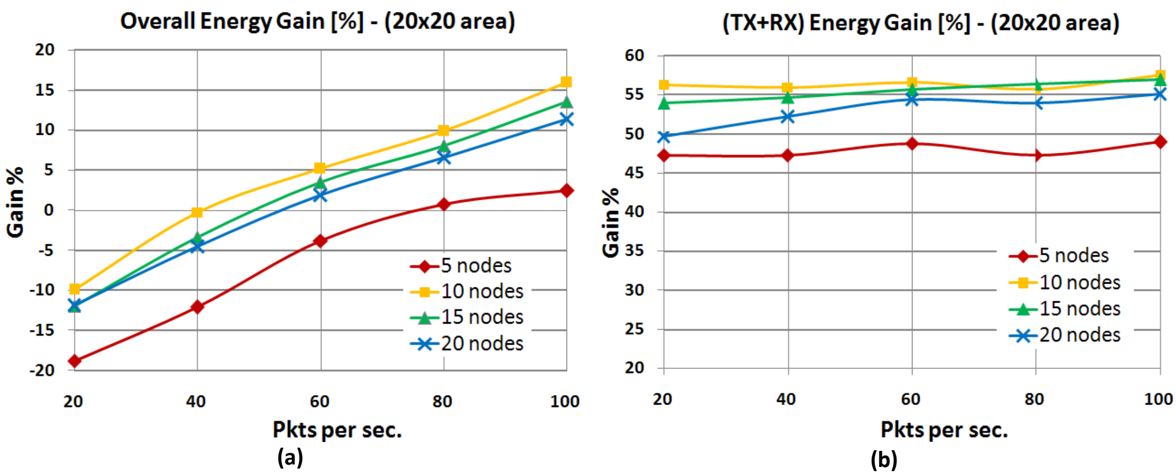


Figure 5-12. Total energy gain with *Traffic_mode_1* in a 20x20[m] area considering overall energy consumed (a) and energy consumed in TX and RX states (b)

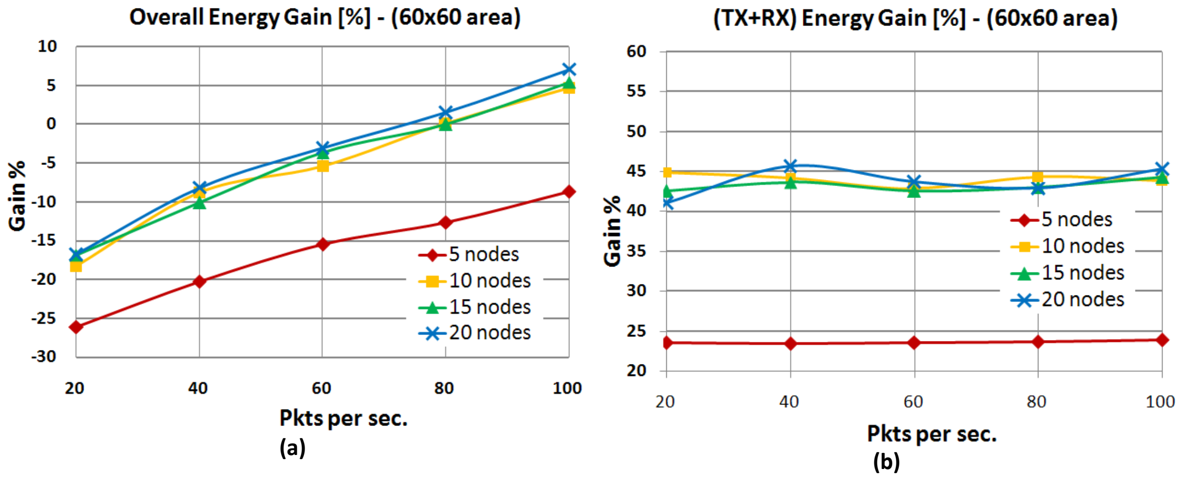


Figure 5-13. Total energy gain with *Traffic_mode_1* in a 60x60[m] area considering overall energy consumed (a) and energy consumed in TX and RX states (b)

In other words, using C2POWER devices, the energy spent in IDLE represent a loss in the energy gain (we have two active interfaces); while the energy spent in TX and RX represent always a gain.
This highlights the fact that the employment of an energy efficient interface for the C2POWER cooperation is of paramount importance.

In Figure 5-13, results in a 60x60 meters area, again under *Traffic_mode_1*, are shown. We can note that the energy gain obtained in TX and RX states is less with respect to the 20x20 meters area. This is because, within a bigger area, some nodes can be not reachable by other nodes with the short-range interface and thus cooperation cannot be achieved. Of course, the less is the number of nodes placed the higher is the number of nodes that have to upload directly packets to the AP. In fact, the energy gain in TX and RX for 5 nodes is around 24%, while the energy gain with 10 to 20 nodes is around 44%.
Considering the overall gain in Figure 5-13 (a) results are affected, like for the 20x20 meters area, by the losses represented by the energy spent in IDLE mode, bringing the maximum achievable energy gain around 7%. This highlights that with a bigger area the effect of the losses in IDLE mode is even worst and the employment of energy efficient IFs for cooperation is even more important.

As we can see from Figure 5-15, within a bigger area the routing is able to find 3 hops route. Of course, since a 2 hops route can be achieved by a relay strategy, 3 hops routes show cases in which the routing algorithm can really be useful for energy saving. On the other hands, Figure 5-14 shows the average number of nodes exploiting routes with two, three and zero hops. Nodes having zero hops are partitioned nodes, which have to upload directly packets to the AP without exploiting C2POWER functionalities. The low values of hops and the presence of partitioned nodes, indicates that the employed scenarios are not very suitable to test the routing algorithm, but on the other hands, bigger scenarios with higher number of nodes are outside the scope of C2POWER considering practical scenarios (i.e. coffee shops, halls, etc...). This suggests that a global policy in order to trigger the use of the routing only when really needed is advisable.

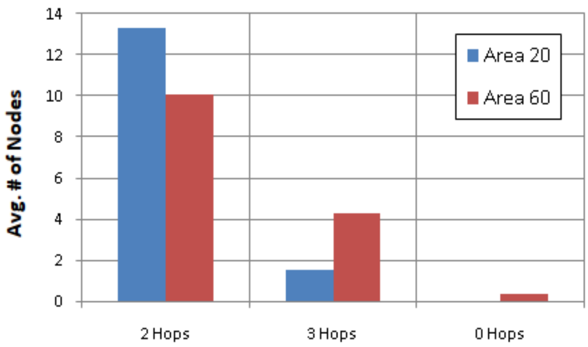


Figure 5-14. Number of nodes exploiting routes with a certain number of hops, averaged over 10 runs (*Traffic_Mode_1*). (Nodes in the network=15; traffic=60 Pkts/sec)

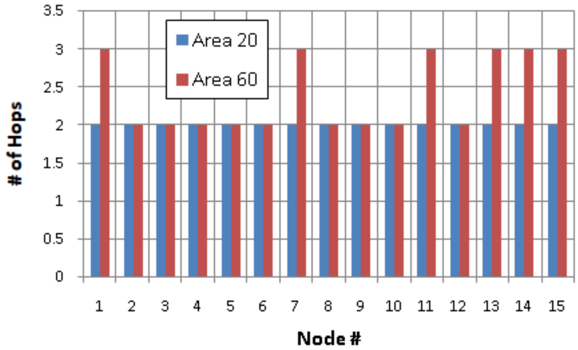


Figure 5-15. Number of Hops used for each node in the network for a single simulation run (*Traffic_mode_1*). (Nodes in the network=15; traffic=60 Pkts/sec)

Looking at the average number of nodes exploiting 3 hop routes, in Figure 5-14, we can observe that some nodes use 3 hops route also in case of 20x20m area. Note that in this particular scenario any node can reach any other node for cooperation (see previous subsection). This happens because of the loss of some broadcast RDPs which bring to incomplete information for the route choice at the AP. Of course, the bigger is the traffic, the bigger will be the effect of this behavior on the routing performance. This suggests that a route discovery procedure is not much advisable within the C2POWER context.

In Figure 5-16 and Figure 5-18, results in terms of energy spent for *Traffic_Mode_2* are shown in a simulation area of 20x20 meters. Moreover, in Figure 5-17 and Figure 5-19 the related energy gains are shown. As we can see, unlike with *Traffic_Mode_1*, considering the overall network gain is always positive. This is because all the nodes are transmitting together and the energy spent in IDLE mode is reduced. This brings the energy gain obtained in TX and RX states more relevant than before showing the highest gain obtainable as 29%.

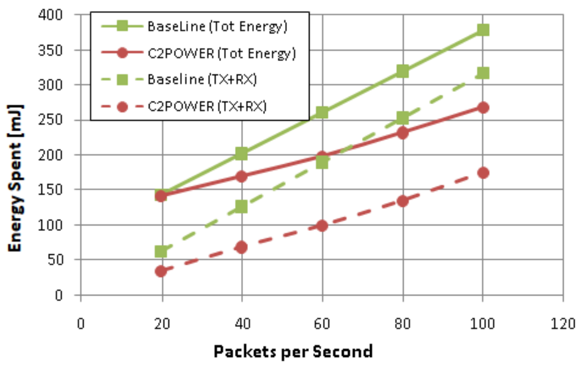


Figure 5-16. Energy spent VS Packets per second under Traffic_Mode_2. (No. Nodes=5; area=20x20[m])

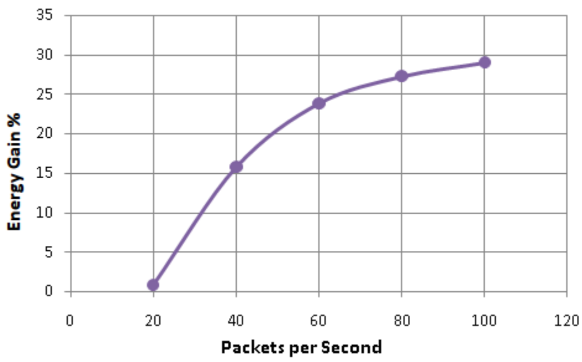


Figure 5-17. Overall Network Energy Gain VS Packets per second under Traffic_Mode_2. (No. Nodes=5; area=20x20[m])

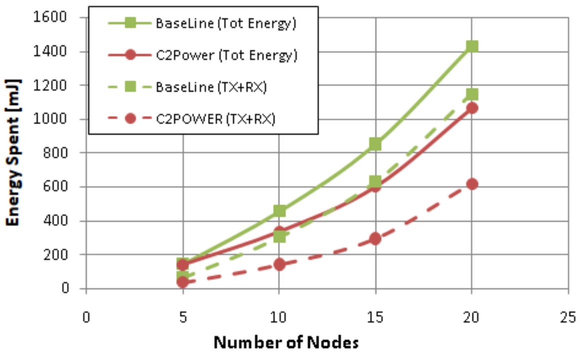


Figure 5-18. Energy spent VS Number of Nodes *Traffic_Mode_2*. (20 packets per second; area=20x20[m])

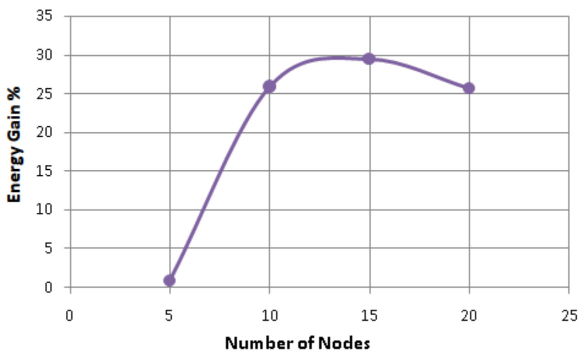


Figure 5-19. Overall Network Energy Gain VS Number of Nodes *Traffic_Mode_2*. (20 packets per second; area=20x20[m])

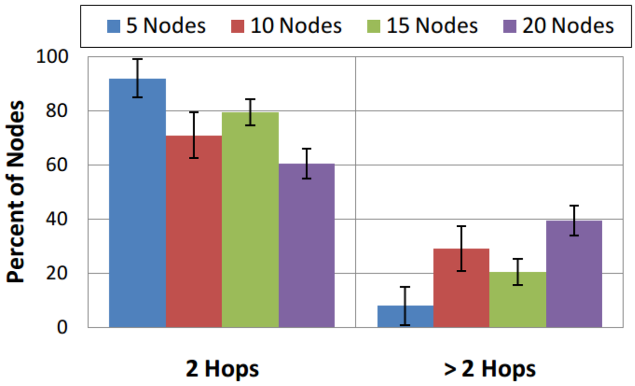


Figure 5-20. Percent of nodes exploiting routes with 2 Hops and routes with more than 2 Hops (area=20x20m; Traffic_Mode_2; 20 packets per second)

Looking at Figure 5-19, plotting the overall energy gain against the nodes in the network, we can see that the energy gain decreases for 20 nodes after reaching its maximum for 15 nodes. Moreover, if we look at Figure 5-20, we can see that the number of nodes exploiting routes with more than two hops increases. Note that in this scenario each node is reachable by every other node in the network, and nodes are homogeneous. This means that routes with more than 2 hops are not the best choice and thus the energy gain decreases. The optimality of a route is actually due to the fact that, under heavy traffic, some RDPs are discarded and the route is decided having only partial information. This suggests that under heavy traffic a relaying strategy is more suitable.

Figure 5-21 shows results in terms of latency in the received packets. As we can see average packet latency employing the C2POWER routing is higher. Also if it exploits higher data-rates with respect to the baseline scenario, the latency is affected by route discovery process during which packets need to be hold by the C2POWER layer. This behavior is also highlighted in Figure 5-22 showing the latency of received packets at the AP over time. Every peak reveals the packet latency when each node starts to transmit and a route discovery procedure is triggered.

Note that this latency period could be a problem in case of route errors for which a new discovery procedure is needed. For this reason, in case of high mobility in the network a relay strategy could be more suitable for QoS-sensible applications.

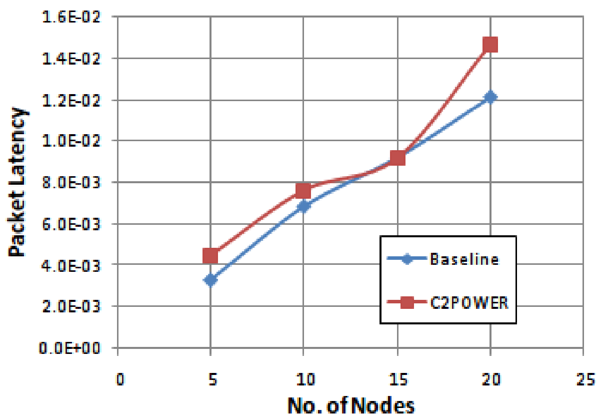


Figure 5-21. Average Packet Latency VS Number of Nodes for *Traffic_Mode_2*. (20 packets per second; area=20x20[m])

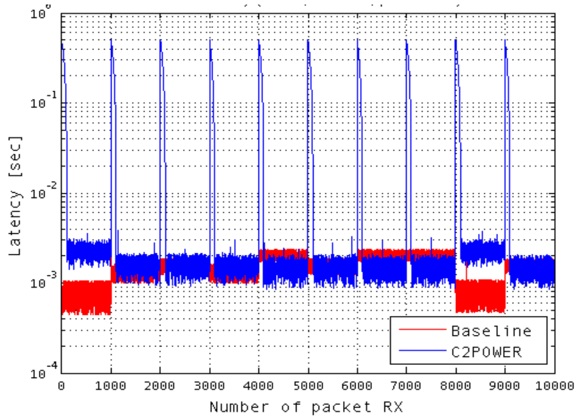


Figure 5-22. Latency of received packets over time for a single sim (*Traffic_Mode_1*)

In conclusion, we can argue that the C2POWER strategy adopted for the routing can be employed only if we have at disposal an energy-efficient interface for the short-range cooperation. In particular, the IDLE mode consumptions could be reason of energy waste instead of energy gain bringing the choice of the short-range interface of paramount importance.

The higher the traffic is, the higher gain could be achieved since the time spent in IDLE mode is reduced. On the other hand, the optimality of the choice of the route is affected by the traffic since some broadcasted RDP packet might be lost. Moreover, the possibly increased latency upon errors in routes might affect delay sensitive applications. Since the C2POWER scenario of interest is not so wide-range (i.e. coffee shops, halls, etc...), this suggests that another routing strategy could be forecasted exploiting additional information delivered with beacons instead of using a route discovery procedure based strategy.

Results also suggested that a global policy in order to trigger the use of the routing only when really needed is advisable mainly for two reasons. First, if routing is not needed, a relay strategy can be used and thus save some energy spent for increased overhead. Second, since the routing needs to perform route discovery to the AP and maintain active routes, it is more difficult to switch off the long-range interface when not really needed while, using relay strategies, the long-range could be switched off more frequently.

5.1.3.2 Simulation Results with WiFi/WiMAX devices

In the following we present the evaluation of the proposed routing techniques employing devices equipped with WiFi for short-range cooperation and WiMAX as long-range infrastructure network. We considered two simulation areas of *60* by *20* and *100* by *50* meters. For all the simulation we set the maximum transmission range for the cooperation at *20* meters.
In order to have reliable results we chose to avoid having isolated MTs for which the cooperation is not possible. For this reason, scenarios are created off-line for each simulation run with the following simple procedure: first we place MTs randomly within the simulation area and we create a graph with maximum edge length of 20 meters (maximum transmission range set in NS-2 for SR cooperation); second, we test over the created graph the any-to-any reachability running Dijkstra algorithm [14]. If the set of positions creates a connected graph, then the scenario is admitted; otherwise we try with another set of random positions.

The proposed technique exploits short-range cooperation to take advantage of MTs having better long-range connection. For this reason, we created scenarios defining two classes of MTs as shown in Table 5-6. For all the MTs data-rates of the short-range links (IEEE 802.11g) is always *54Mbps*, while for the long-range links (WiMAX) the data-rates are chosen varying the OFDM modulation. As first instance, for the cooperative scenario we can consider *ClassB* MTs as source MTs, while *ClassA* MTs as relaying MTs. The number of *ClassA* and *ClassB* MTs is varied in order to observe behaviours of the proposed technique varying the number source and relaying MTs.

Table 5-6. Data-rates used for different classes of MTs used in the simulations

	"ClassA" MT	**"ClassB" MT**
Short range Data-Rate (WiFi)	54 Mbps	54 Mbps
Long range Modulation (WiMAX)	64QAM (3/4)	QPSK (1/2)

In the following subsections we present results for the following cases: simulations varying the number of MTs in the network; simulations varying the amount of traffic sent from the sources; and finally simulations introducing mobility in the scenario.
For every set of parameters the simulation is repeated for *10* runs, and the simulation last *100* seconds. In order to generate traffic, we used CBR traffic generator and all the MTs within the scenario transmit data to the BS with same packet rate. The packet size is set to *1024B* for all the simulations. Finally we considered a beaconing period of *5* seconds for all the MTs for the clustering algorithm (EAR) and the context information dissemination (relay-based routing).

Simulations varying the number of MTs

For those simulations, we varied the total number of MTs, keeping the number of *ClassA* MTs constant (i.e. only number of *ClassB* nodes is varying). The CBR packet frequency is set to *3000 pkts/sec* and all MTs are transmitting to the BS. Results for EAR and the relay-based routing are shown, respectively, in Figure 5-23 and Figure 5.24.

As first instance we can observe that for a low number of nodes the gain strongly decreases in all the cases. This is somehow expected since if we have almost all the nodes with a good LR connection no improvements can be observed with respect to the benchmark case (i.e. the benefit of the routing do not compensate the energy overhead due to the employing of an additional interface for the short-range cooperation). This highlights that if we are in a scenario with a low diversity in the LR channel condition, the cooperation could be even counterproductive.

On the other hand, increasing the number of *ClassB* MTs, the observed gain, decreases after reaching a maximum. This suggests that there is an "optimal" number of source MTs that can be relayed given a number of MTs with good connection to the BS. This behaviour is due to the decrease of the ratio between the number of *ClassA* and *ClassB* MTs. In other words, too many source nodes need to be relayed by MTs with good LR connections and the WiMAX interface of the relaying MTs becomes the bottleneck of the system. Within the investigation ranges, this is not true employing the relay-based routing in the 60x20m area with 3 and 4 *ClassA* nodes. Having more MTs with good LR connection, in this case more MTs can be successfully relayed.

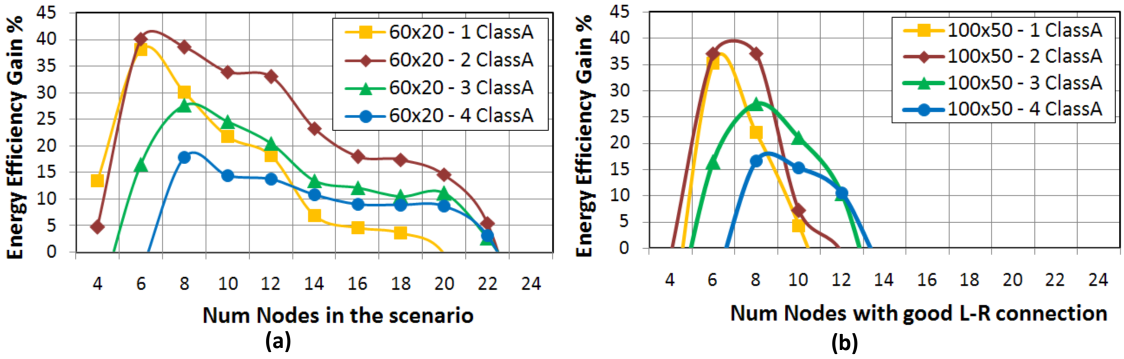


Figure 5-23. Results for EAR. Number of nodes vs. Energy eff. Gain %: 60x20 meters area (a) and 100x50 meters area (b); every curve refers to a constant number of "*ClassA*" MT, while the number of "*ClassB*" nodes varies

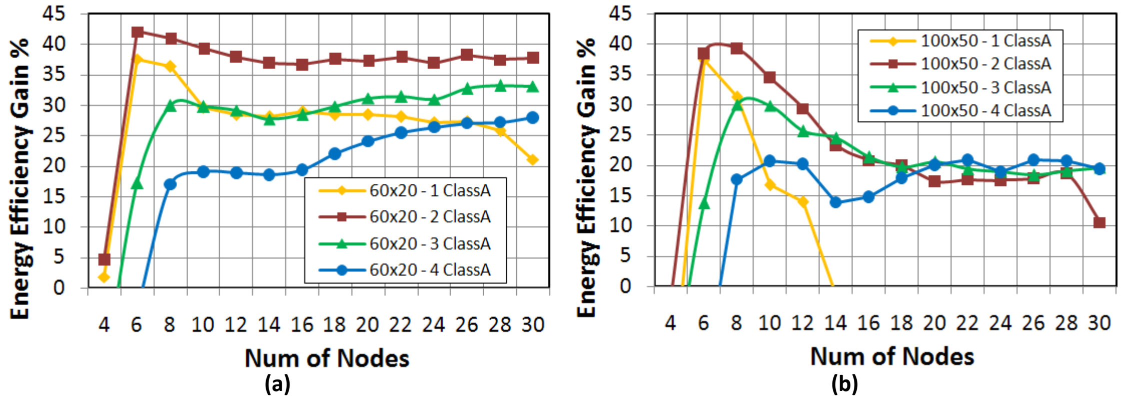


Figure 5-24. Results for relay-based routing. Number of nodes vs. Energy eff. Gain%: 60x20m area (a) & 100x50m area (b); every curve refers to a constant number of "*ClassA*" MT, while the number of "*ClassB*" MTs varies

For both the algorithms, results between the two simulation areas are almost similar for a low number of MTs. On the other hand, in the 100x50 area the gain decrease more rapidly with the increase of the *ClassB* nodes. Please note that in both areas the capacity limit for the LR network is the same (the same number of LR devices with same connection to the AP is deployed for the simulations). This tells us that the behaviour described above is primarily due to bottlenecks formed in the cooperation links since the lower connection degree in the peer to peer SR network. Therefore, stressing the cooperation increasing the number of *ClassB* MTs will have a higher impact on the achievable energy gain in case of wider areas. This is particularly true in our simulations, employing WiFi devices as SR technology, which have a low data-rate.

It is also interesting to note that the EAR is much more affected by the increase of MTs in the scenario. Increasing the number of nodes increases the probability of collisions. Consequently, for the EAR some routes could appear as broken requiring a new route discovery process and a transition period in which the cooperation is not exploited. On the other hands, the relay-based routing since its light approach is more agile and it is more resilient.

Simulations varying the traffic speed

For those simulations, we considered 10 MTs in the network; *2 ClassA* MTs and *8 ClassB* MTs. Results are shown for *60X20m* and *100X50*m areas. The CBR packet frequency is set ranging from *500* to *3000* *pkts/sec* and all MTs are transmitting to the BS. Results for EAR and the relay-based routing are shown, respectively, in Figure 5-25 and Figure 5-26.

We observed that, as expected, the total energy consumed by the C2POWER devices is always higher than the benchmark case, because of two active interfaces. Moreover, for the LR interface (WiMAX), the energy spent in IDLE mode is the primary source of energy drain (always more than 80% in all the cases). We also noted that the energy consumption in the LR interface is always comparable between the Benchmark and C2POWER case (deviations are inside a 7%).
Given this, the difference between the overall energy spent in the C2POWER and Benchmark could be considered as the energy overhead needed for the cooperation. This overhead is shown in Figure 5-25 (b) and Figure 5-26 (b) for EAR and relay-based routing respectively. As we can observe, for both the algorithms the energy overhead is almost constant over the source data-rate, since it is primarily due to the IDLE mode, and a bit higher in the small area because of increased energy spent in RX mode due to the higher node degree in the short-range network.

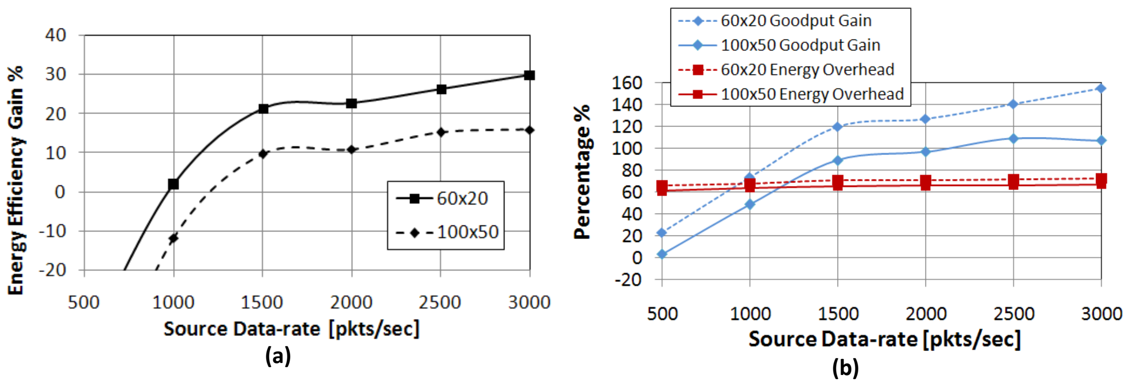


Figure 5-25. Results for EAR; effect of source data-rate over the Energy Efficiency Gain (a), goodput (b) and energy overhead for cooperation (b). (10 nodes: 2 *ClassA* nodes and 8 *ClassB* nodes)

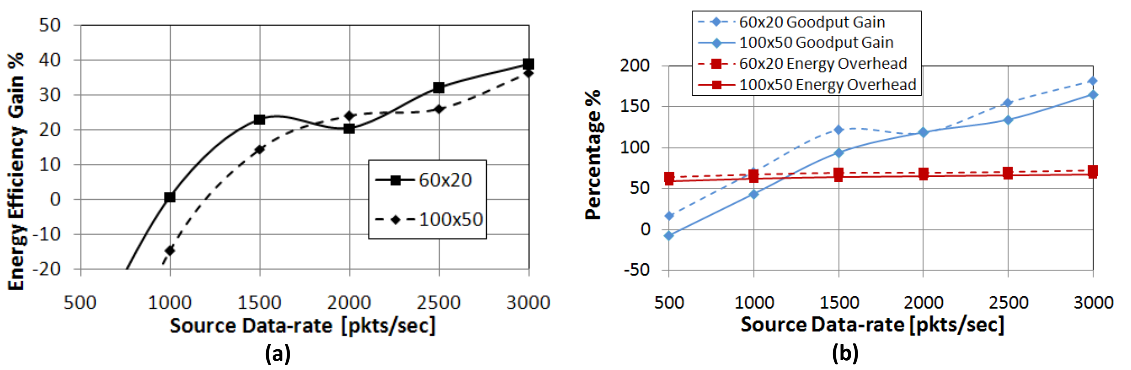


Figure 5-26. Results for relay-based routing; effect of source data-rate over the Energy Efficiency Gain (a), goodput (b) and energy overhead for cooperation (b). (10 nodes: 2 *ClassA* nodes and 8 *ClassB* nodes)

On the other hand, the overhead of the total energy consumed is compensated by the gain in terms of goodput. Both the proposed techniques, exploiting cooperation with MTs having good connection to the BS, are able to increase the network capacity and thus to reach a gain in terms of energy efficiency thanks to a higher goodput.

When the traffic is low (i.e. less than 1000 [pkts/sec]), we can observe that the gain becomes negative. In this case the gain in term of goodput is not able to compensate the energy overhead for the cooperation. On the other hands, increasing the data-rate of the sources, we are able to observe good gains up to 38% for the relay-based routing and 30% for the EAR. This behavior indicates that the C2POWER short-range cooperation system needs to be combined with energy efficiency mechanisms for the interfaces below the C2POWER layer in order to put the interfaces when not used in SLEEP mode.

Simulations with mobility

In this section, we discuss simulation results under mobility. Presented results consider 10 transmitting MTs, two of them are *ClassA* MTs and the other 8 are *ClassB* MTs. The source data is 3000 pkts/sec and all MTs are transmitting to the BS.

For the mobility model we used an implementation of the Gauss Markov mobility model. For the detail of the model we refer to [15] while in the following we give some assumptions made for the simulations. First of all MTs are always inside the simulation area and they bounce back when reaching the edges. Secondly, the value of the speed is intended as the average speed of MTs and, finally, the value of α expressing the randomness of the mobility is set to 0.5.

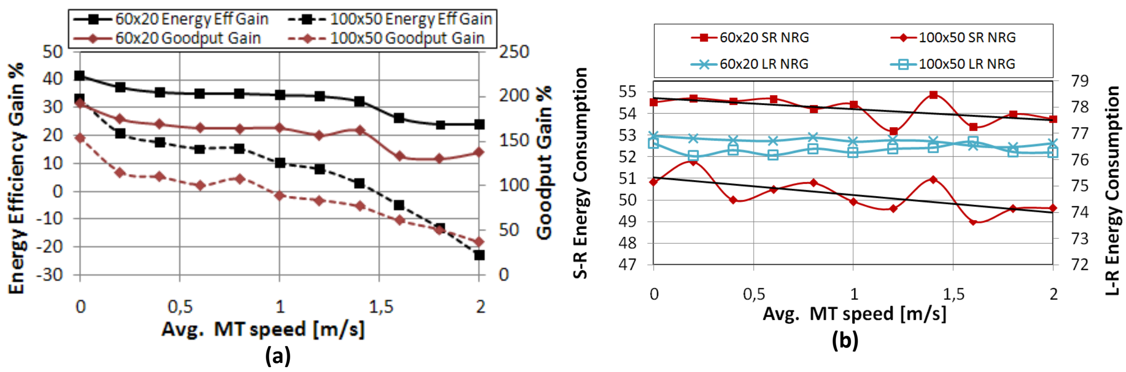


Figure 5-27. Results for EAR; effect of the mobility over the Energy Efficiency Gain (a), and the Energy Consumption for long-range and short-range in C2POWER scenario (3000 pkts/sec; 10 nodes: 2 *ClassA* nodes and 8 *ClassB* nodes)

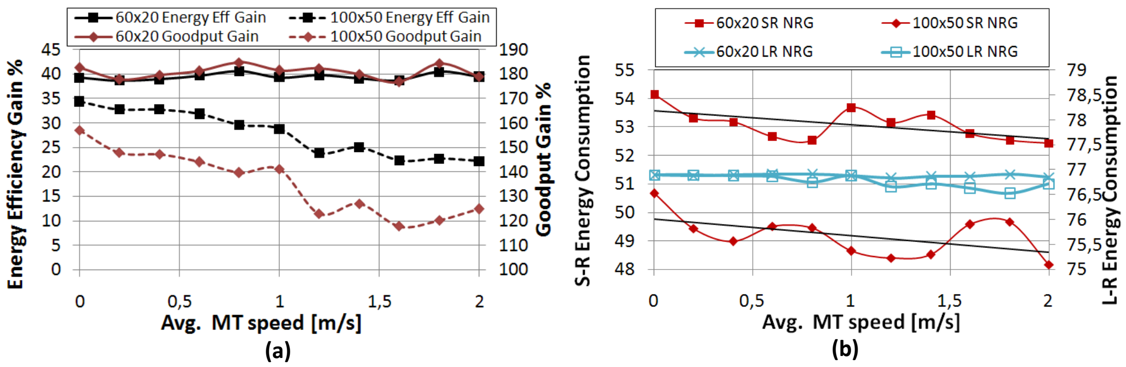


Figure 5-28. Results for relay-based routing; effect of the mobility over the Energy Efficiency Gain (a), and the Energy Consumption for long-range and short-range in C2POWER scenario (b). (3000 pkts/sec; 10 nodes: 2 ClassA nodes and 8 ClassB nodes)

For the relay-based routing (Figure 5-28), it is interesting to note that in the 60X20 meters area, the energy efficiency gain (except for some slight deviations) is almost constant at around 40%. In terms of energy consumption, we observed that for the C2POWER case while the long-range energy consumption is almost constant around 76.7 [J], the short-range energy consumption decreases from 54.1 J to 52.4 J and this is due to less energy used in RX state for short range. This means that adding mobility we have more sparse MTs with and a lower number of connections in average. However, this do not affects the goodput which is almost constant with slight deviations of 0.04 Mb/s, meaning that no cooperation fault occurs due to outdated neighbour entries. On the other hand, within the 100x50 area we can observe that the energy efficiency gain is affected by the mobility.
As previously noted, the LR energy consumption is almost constant, while the SR energy consumption decreases as in the 60x20 area. However, this time the goodput is affected by the mobility. In fact we observed a goodput gain of 157 and 124% for 0 and 2 m/s respectively, meaning that some cooperation fault occurs. The reason for this is that in a big area, it is more difficult to keep the neighbour table updated. Thus, having more hops between a source and a relay MT, more time is needed in order to have updated information about “neighbour of neighbours”.

For the EAR similar results can be observed in terms of energy consumption. However, we can note that, for the considered areas, the goodput gain and consequently the energy efficiency gain is much more affected by the mobility of MTs bringing to more cooperation faults. A cooperation fault can happens basically for two reasons. As first instance, the neighbor table could be out-dated and a MT recognized as a neighbor also if it has already left the reachability area. Another possibility, also if less frequent, a MT inside the reachability area could be deleted from the neighbor table, if a number of beacon packets is discarded because of subsequent collisions in the SR channel. In both the cases, the EAR needs to compute a new route and there is a period in which the source MT is not able to exploit the cooperation at all. With the relay-based routing we have that in the first case the cooperation will restart successfully immediately after the update of the neighbor table, while in the second case the cooperation is always exploited but sub-optimal decisions will be taken till the neighbor table update.

Results show that the lightness of the relaying-based approach is to be preferred in respect to an on-demand routing strategy since its higher resilience to mobility.

5.2 Scenario 3

Table 5-7. Summary for parameters used for simulations of Scenario 3 with multi-radio cooperative relaying.

		Value	Notes
Parameters for the scenario			
Mobility of nodes		---	Static scenario
Number of nodes		3-20	
Simulation time		600	[s]
Topology		random	Infrastructure mode, WiMAX BS and WiFi AP are employed and multiple users are randomly distributed among these networks
LR power consumption WiMAX	Tx mode	2409	[mW] see: *Evaluating the Energy Efficiency of TCP Transmission over a WiMAX Network*, 2010 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5560108
	Rx mode	1485	
	Idle	660	

LR/SR power consumption WiFi	Tx mode	890	[mW] see: *CoolSpots: Reducing the Power Consumption of Wireless Mobile Devices with Multiple Radio Interfaces*, 2006, http://static.usenix.org/events/mobisys06/full_papers/p220-pering.pdf (NIC Linksys WCF12)
	Rx mode	890	
	Idle	256	
SR power consumption WiMedia	Tx mode	250	[mW] see: Deliverable D2.2
	Rx mode	250	
	Idle	0.30	
Long-range data rates		3.8-56	[Mbps] see: Table 4-2. Values of data rates for used interfaces as implemented in ns-miracle
Short-range data rates		36-54	[Mbps]
Parameters of traffic model			
Traffic Type		CBR	Every node generates traffic in the same way
Number of Traffic Sources		3-20	Every node in the system generates traffic
Data packets size		1024	[B], application layer data size
Sending rate		3000	Packets per second
Important parameters of the relaying algorithm			
Beacon period		5	[s] Beacon is broadcasted every 5 seconds
Timeout		15	[s] Nodes removes entries in their neighbour lists in case no beacon has been received within 10 seconds since the last update.

Evaluation of energy efficient multi-radio cooperative relaying.

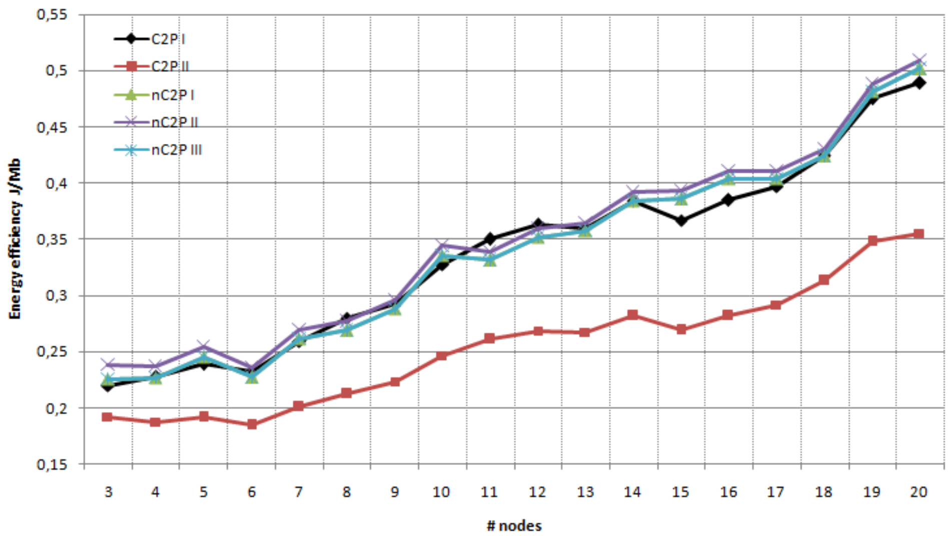


Figure 5-29. Energy efficiency in joules per transmitted Mb of data for network with 3-20 nodes.

Figure 5-29 presents an overall comparison of energy efficiency obtained for cooperative (C2P I and C2P II) and non-cooperative benchmark scenarios (nC2P I, nC2P II, nC2P III) for networks presented in Section 4. X-axis represents number of nodes, whereas Y-axis depicts energy efficiency expressed in joules per one realized mega bit of data in the whole heterogeneous system with 2 radio access technologies available. All illustrated results are average values among 50 independent tests performed for each case and number of nodes separately.

It might be easily noticed that, C2P I cannot reach explicit energy gain in the meaning of J/Mb in comparison to all non-cooperative cases. However, please note that C2P I employs WiFi technology for short-range cooperative interface. In case there are not a lot of nodes existing in the system, energy efficiency of such systems is higher, and by average, the cooperative network consumes about 0.220-0.238 J per transmitted Mb if 3-6 nodes actively communicate; However, non-cooperative networks use 0.226-0.255 J/Mb, which states for even 8% gain. Next when more nodes exist in the multi radio access technology system, overall energy consumption increases. Starting from 7 nodes for C2Power-enabled case C2P I energy consumption equals to 0.259 J/Mb and is getting higher to 0.424, 0.474 and 0.488 J/Mb for 18, 19 and 20 nodes, respectively. In comparison to non-cooperative cases C2P I guarantees average energy consumption in the range from 0.03 J/Mb better (gain, e.g., 15 nodes) to 0.02 J/Mb worse (loss, e.g., 11 nodes).

In order to compare relative energy gain of applied C2Power cooperative techniques in the form of multi-radio cooperative relaying, formula (5.2.1) is employed.

$$\boldsymbol{relative\ energy\ gain} = \left(\frac{\boldsymbol{nC2P}}{\boldsymbol{C2P}} - \mathbf{1}\right) \cdot \mathbf{100\%} \qquad (5.2.1)$$

Therefore, relative energy gain expresses percentage gap between non cooperative benchmark cases (nC2P) and cooperative cases of Scenario 3 (C2P). Note that positive gain refers to as real energy savings, whereas negative gain is equivalent to extra energy expenses, i.e., energy losses.

In Figure 5-30 relative energy gain of C2P I versus all non-cooperative cases is compared for 3-20 nodes distributed among two wireless networks, either WiMax or WiFi.

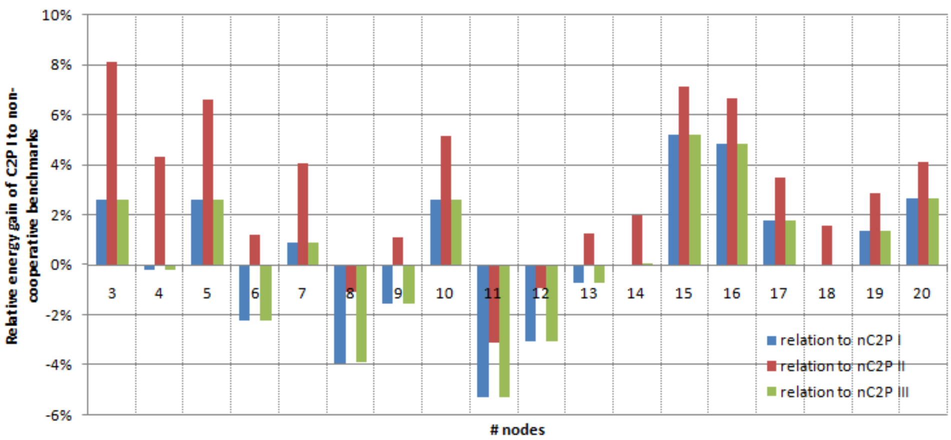


Figure 5-30. Relative energy gain of applying C2Power cooperative strategies C2P I in comparison to all non-cooperative benchmarks for 3-20 nodes.

For 9 instances (50% tests) C2P I provides better energy efficiency results (about 0.9-5.2%) in comparison to nC2P I, where non-cooperative system comprises single-interface nodes. On the other hand, C2P I case might also fall below non-cooperative benchmarks what is observed for 7 instances and potential losses reach at most 0.2-5.26%. Similar results are observed for relation of C2P I and nC2P III, where non-cooperative users take advantage of second wireless interface - WiMedia, albeit it only consumes extra portions of energy. The highest difference in energy gain relation is shown when

C2Power techniques are applied as C2P I case and non-cooperative users use two interfaces, where WiFi stays idle and practically unused (nC2P II). For such cases C2Power provides even 8% energy gain and exceeds non-cooperative cases in 15 systems configurations (83% situations), whereas potential losses (17% situations) reach only 0.9-3.1% in energy expenses. Finally, it is worth providing an overall values for gains, that equal 0.4%, 0.4% and 3% in favor of C2P I in relation to benchmark-defined nC2P I, nC2P III and nC2P II, respectively.

Next we present percentage relation between non-cooperative cases and C2PII that implements WiMedia as short-range technology. Figure 5-31 shows that C2P II outperforms all nC2P cases by at least 17%. In relation to nC2P I it reaches 17-43%, in relation to nC2P II 24-45% and about 17-43% in relation to nC2P III. By average among all network sizes, C2P II provides 32-35% better energy efficiency.

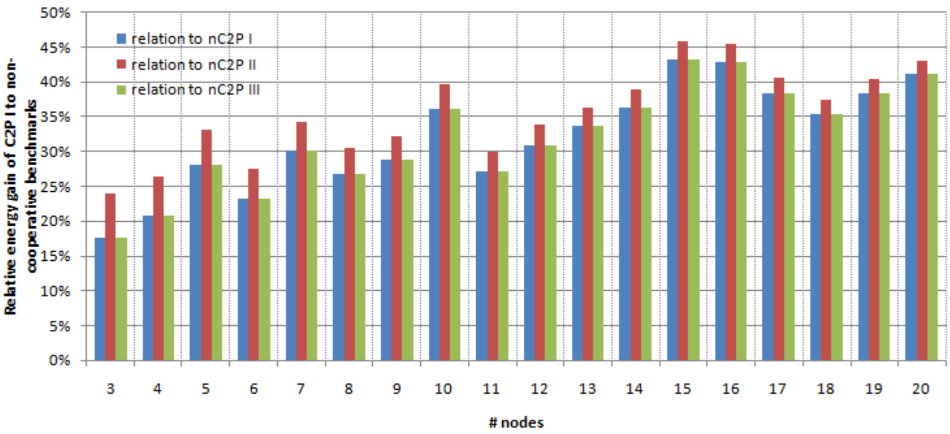


Figure 5-31. Relative energy gain of applying C2Power cooperative strategies C2P II in comparison to all non-cooperative benchmarks for 3-20 nodes.

Short range technology impact

In general, the most beneficial is to apply low-power consuming short range network interface cards. Presented cases use either WiFi technology or WiMedia wireless standard. The former consumes more energy while the latter requires explicitly less. Figure 5-32 illustrates relation between C2P I and C2P II cases and shows how much applying WiMedia as short-range communication technology outperforms using WiFi radio interface.

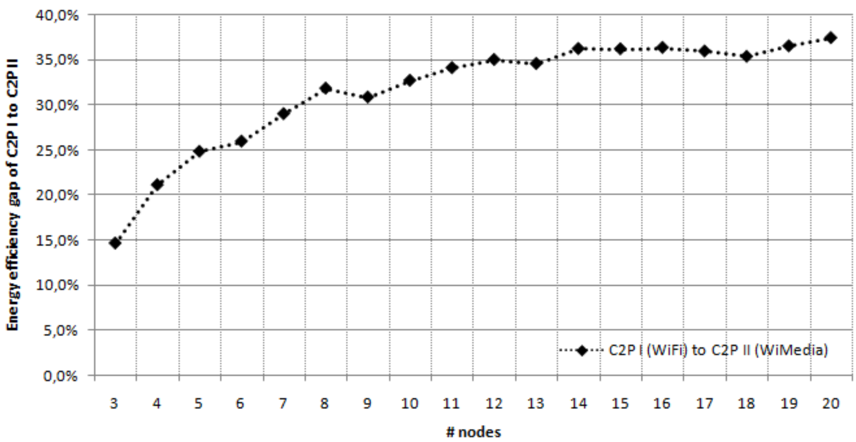


Figure 5-32. Energy efficiency gap of using WiMedia short range technology (C2P II) versus WiFi (C2P I).

Relative gain in favor of WiMedia grows in case more nodes are deployed in the Scenario 3 system. By achieving 15-37% gap, using WiMedia instead of WiFi as short-range technology applied in order to enable C2POWER mutli-radio cooperation cannot be ignored.

BS/AP Perspective

Next it is worth analyzing energy efficiency at WiMAX base station and WiFi access point. Figure 5-33 presents relation of C2P and nC2P cases at WiMAX BS. According to these results, applying C2Power techniques and employing multi-radio relaying to wireless devices helps in providing better effectiveness of energy spending. C2P case requires explicitly less value in Joules for serving every Mb of effective throughput and BS processes WiMAX-side traffic using averagely 0.1-0.2 J/Mb (depending on number of nodes), whereas non-cooperative cases consume 0.13-0.26 J/Mb which constitutes for approximately 6-60% gap.

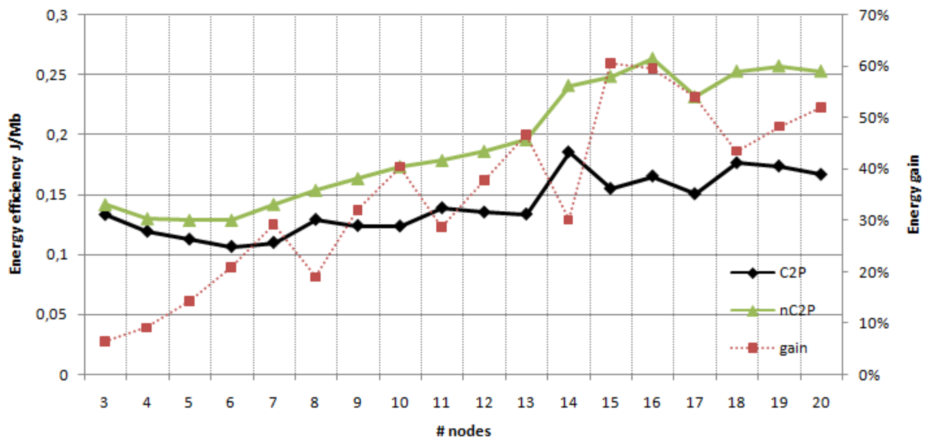


Figure 5-33. Energy effectiveness at WiMax BS

Moreover, from WiFi access point, as shown in Figure 5-34, enabling C2Power techniques guarantees better energy efficiency than using non-cooperative modes. Although energy gain is not as high as at WiMAX side, it still reaches 10-30% and slightly grows while network size is growing. 0.09-0.11 J of energy expenditures are required for serving each Mb of data in case C2Power-enabled devices are deployed. On the other hand, non-cooperative set up consumes about 0.10-015 J/Mb.

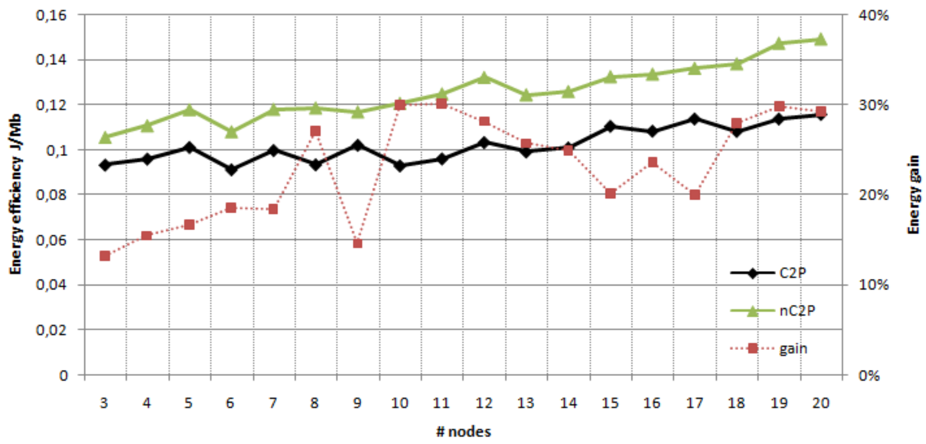


Figure 5-34. Energy effectiveness at WiFi AP

Goodput Comparison

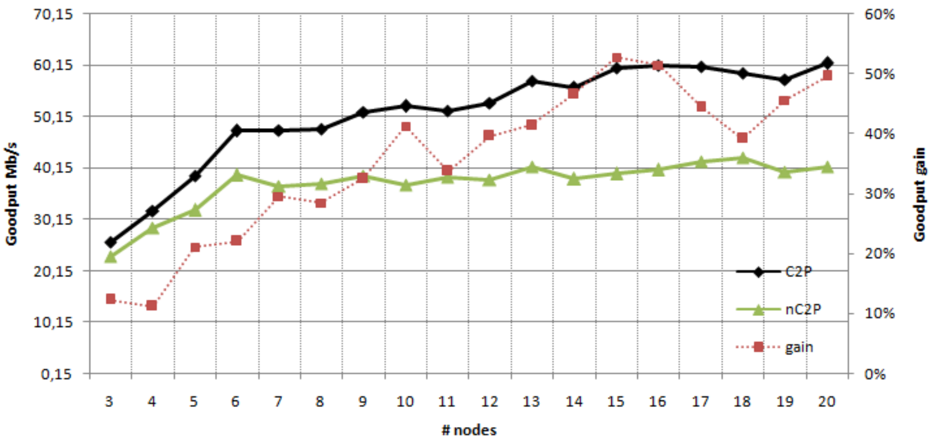


Figure 5-35. Goodput comparison in Mb/s and percentage relation for cooperative C2P cases and non-cooperative nC2P benchmarks.

Figure 5-35 illustrates goodput comparison in Mb/s obtained for cooperative and non-cooperative cases. In average, for all cases with 3-20 nodes, applying C2Power guarantees better effective end-to-end throughput. Generally the goodput gain (see Formula 5.2.2) starts from 10% for smaller networks (with 3-4 nodes); however it can reach even 40-50% where more nodes are available and possibility of efficient cooperation in the network increases.

Formula (5.2.2) defines relative goodput gain,

$$relative\ goodput\ gain = \left(\frac{C2P}{nC2P} - 1\right) \cdot 100\% \qquad (5.2.2)$$

Finally, improving goodput might have positive influence on general energy effectiveness. In case any node is helped by other it finishes its transmission faster. The faster the node's transmission finishes, the faster other node is granted the channel access.

6. Conclusion

This document represents the last outcome of WP5 in C2POWER project regarding the simulation tools to evaluate energy efficiency strategies in cooperative short-range networks. A common simulation framework has been implemented in order to demonstrate the algorithms developed. The main idea behind the C2POWER framework was to create a technology independent layer which creates a pre-IP routing in the short-range network. This layer hosts the developed algorithms separating them from the low level details of the simulator and radio interfaces below.

In order to test developed algorithms two kinds of scenarios have been considered. First C2POWER scenario 1 is investigated, which involves short range cooperation within a homogeneous LR network covering the whole considered area (short range cooperation among homogeneous MTs). Second C2POWER scenario 3 was simulated where an area covered by two different infrastructure based network have been considered (short range cooperation among heterogeneous MTs).

Results from different cooperative proposed algorithms vary. For instance, the node selection mechanism based on the cooperative game theory solution showed gains between 3 and 50% considering single MTs gains. This is not true for relaying and routing techniques which present some MTs with negative gains. As a conclusion the cooperative game theory solution results in a more fair solution since it has a more global view of optimization.

Simulations for the routing techniques, showed that increasing the size of the considered area for the scenario performance in terms of gain decreases. In bigger areas the SR connections used for the cooperation form a network with a lower connection degree. This could results in bottlenecks affecting the achievable goodput improvement and thus the achievable energy efficiency gain. Testing the routing techniques under mobility we observed that the relay-based routing is much more efficient than the EAR adapting itself in a more agile way to the effects of mobility.

In general, performed simulations show an energy efficiency gain up to 50% within the explored scenarios. Moreover, simulations pointed out some behaviour summarized in the following.

The achievable energy efficiency gain is strongly related to the number of nodes in the cluster. On the one hand, for low number of MTs it could be difficult to achieve good gains. This is because having less relaying choices, the probability to find a good relaying MT decreases. On the other hand, for a high number of MTs it is difficult to observe good gains because of increased collisions in the SR network. Moreover, results show that the overall effectiveness of energy expenditures while applying cooperation strongly depends on the type of installed short-range wireless radio interface. Simulation results also show that employing an energy efficient SR technique (as WiMedia) can have an improvement in the gain of up to 30% with respect to a WiFi interface. Another thing that could strongly affect the achievable gain is the LR data-rate distribution among MTs. The more the data-rate distribution is sparse, the higher will be the improvement margin in terms of goodput exploiting cooperation among MTs and, consequently, the energy efficiency gain. As a final conclusion, unless there is a variance in the quality of long range channels experienced by different MTs, no energy savings can be achieved (i.e. Some MTs have to be suffering bad channel conditions).

References

[1] C2POWER, "D5.1: Cooperative short-range strategies and protocols for power saving," 2010.

[2] "IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, 2007, pp. C1-1184.

[3] C2POWER, "D2.2: Scenarios, System architecture definition and performance metrics," 2010.

[4] G. Heidari, "WiMedia UWB: technology of choice for wireless USB and Bluetooth," Wiley Publishing, 2008

[5] F. B. Saghezchi, A. Nascimento, M. Albano, A. Radwan, and J. Rodriguez, "A Novel Relay SelectionGame in Cooperative Wireless Networks based on Combinatorial Optimization Problem," in Proc. 2nd International Workshop on Cognitive radio and Cooperative strategies for POWER saving (C2POWER 2011), VTC'11, May 2011.

[6] C2POWER, "D5.2: Modelling and Simulation Tools to Evaluate Energy Efficiency Strategies in Cooperative Short-Range Networks – Intermediate Version", 2011.

[7] NS-MIRACLE: Multi-InteRfAce Cross-Layer Extension library for the Network Simulator, http://telecom.dei.unipd.it/pages/read/58/, retrieved on 12/2012.

[8] NS-MIRACLE (Multi-InteRfAce Cross-Layer Extension library for the Network Simulator) Documentation, http://telecom.dei.unipd.it/ns/miracle/doxygen/, retrieved on 12/2012.

[9] S. Kurkowski, T. Camp, and W. Navidi, "Minimal Standards for Rigorous MANET Routing Protocol Evaluation," Technical Report MCS 06-02, Colorado School of Mines, 2006.

[10] Jean-Pierre Ebert, Stephan Aier, Gunnar Kofahl, Alexander Becker, Brian Burns and Adam Wolisz, "Measurement and Simulation of the Energy Consumption of a WLAN Interface", TKN Technical Report Series TKN-02-010, Telecommunication Networks Group, TechnischeUniversität Berlin, jun 2002.

[11] C2POWER, "D2.2: Scenarios, System architecture definition and performance metrics," 2010.

[12] T. Pering, Y.Agarwal, R. Gupta, and R. Want, "CoolSpots: reducing the power consumption of wireless mobile devices with multiple radio interfaces", In Proceedings of the 4th international conference on Mobile systems, applications and services (MobiSys '06). ACM, New York, NY, USA, 220-232.

[13] Shiao-Li Tsao; Shih-Yung Lee; , "Evaluating the Energy Efficiency of TCP Transmission over a WiMAX Network," Computer Communications and Networks (ICCCN), 2010 Proceedings of 19th International Conference on , vol., no., pp.1-6, 2-5 Aug. 2010.

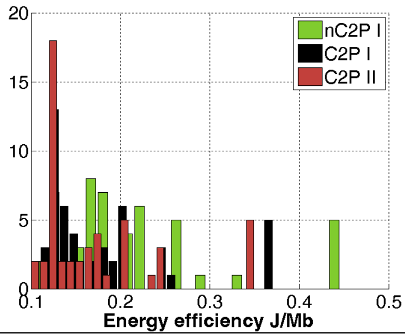
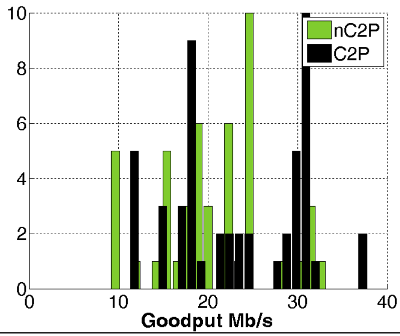
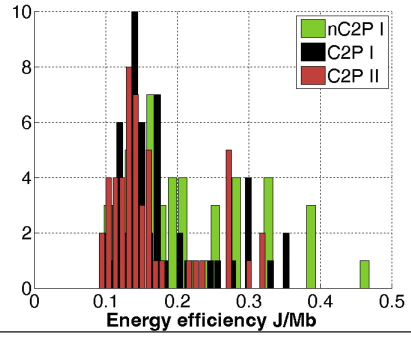
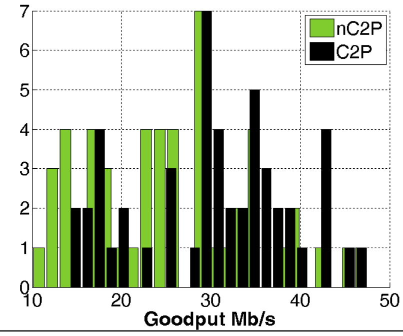
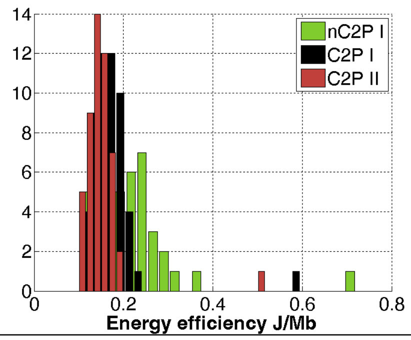
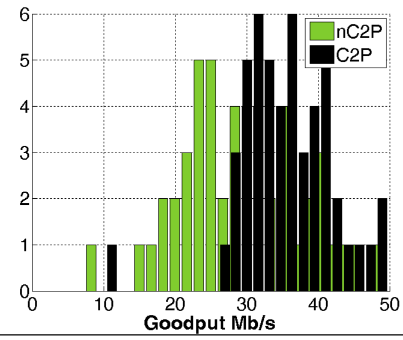
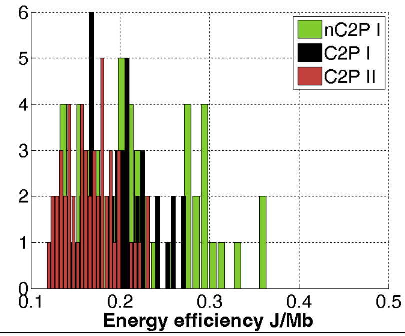
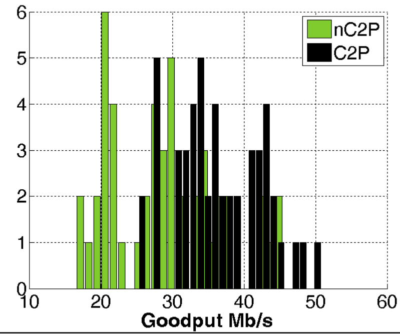
[14] Thomas H. Cormen; Charles E. Leiserson; Ronald L. Rivest; and Clifford Stein, Introduction to Algorithms, Second Edition. MIT Press and McGraw-Hill, 2001, ISBN 0-262-03293-7, Section 24.3: Dijkstra's algorithm, pp.595–601

[15] Ben Liang; Zygmunt J. Haas , Predictive Distance-Based Mobility Management for PCS Networks, INFOCOM 1999, 1377-1384.

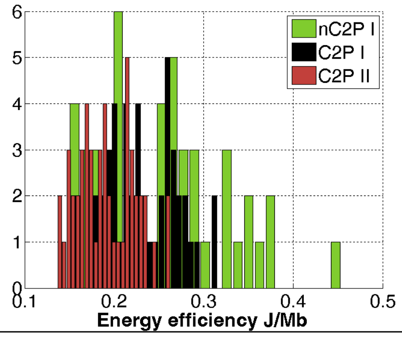
Appendix I

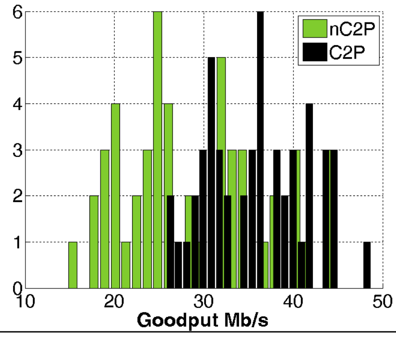
Selected histogram from the simulation results of the energy efficient multi-radio relaying algorithms are shown below.

Selected histograms

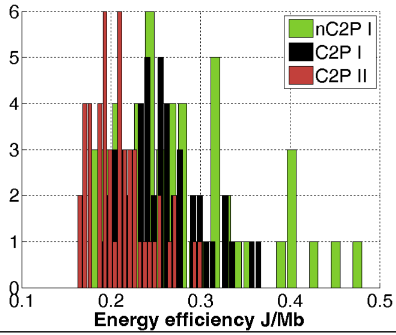
#nodes	Energy efficiency	Goodput
2		
3		
4		
5		

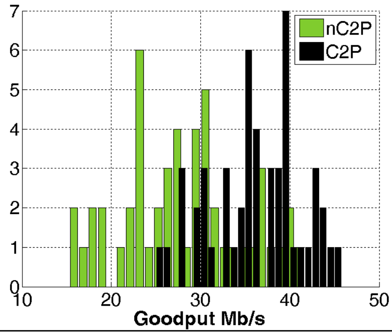
6


nC2P I
C2P I
C2P II
Energy efficiency J/Mb

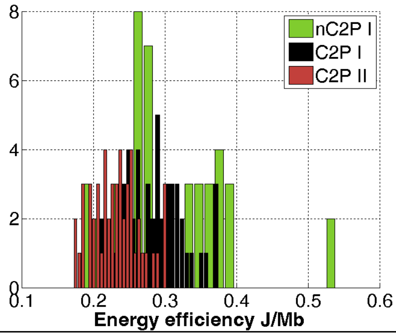

nC2P
C2P
Goodput Mb/s

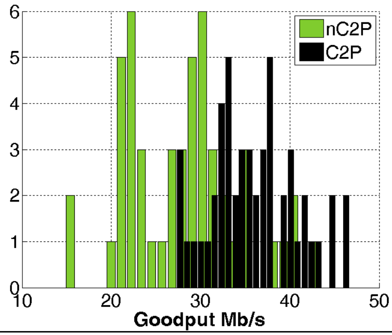
7


nC2P I
C2P I
C2P II
Energy efficiency J/Mb

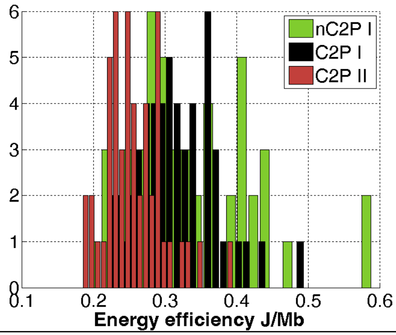

nC2P
C2P
Goodput Mb/s

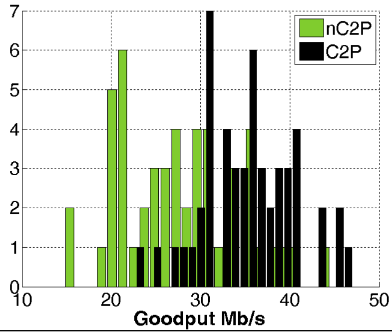
8


nC2P I
C2P I
C2P II
Energy efficiency J/Mb

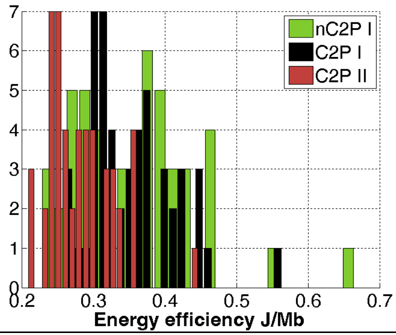

nC2P
C2P
Goodput Mb/s

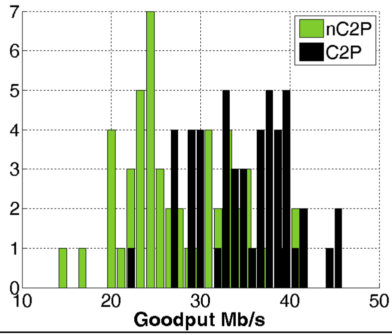
9

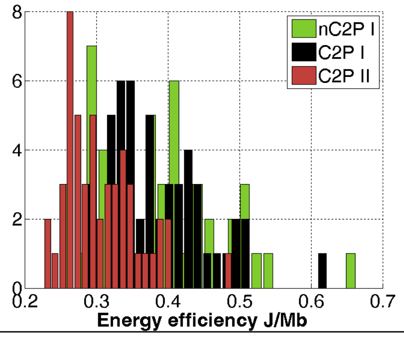
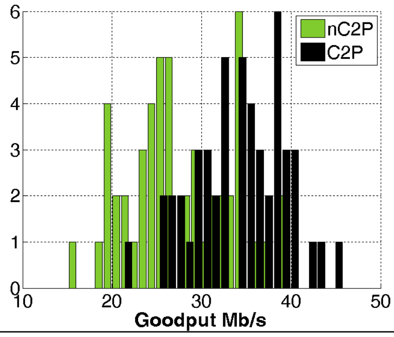
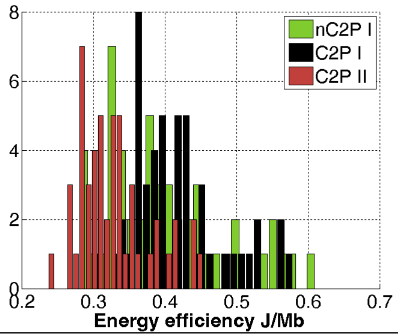
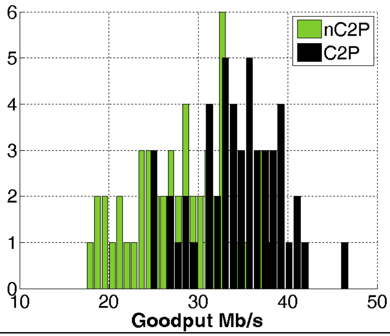
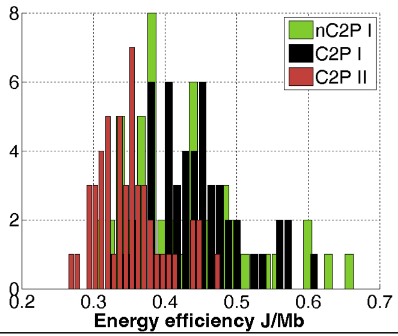
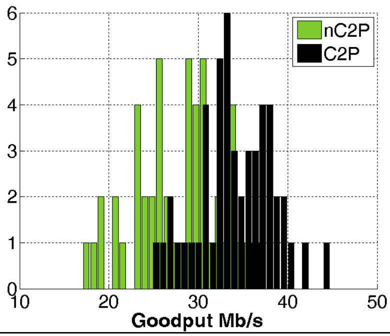
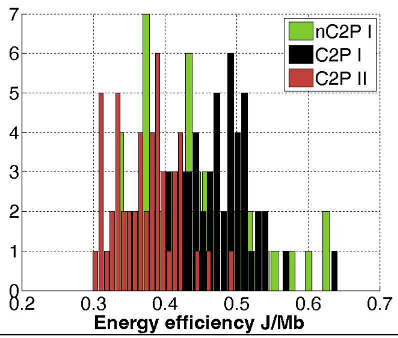
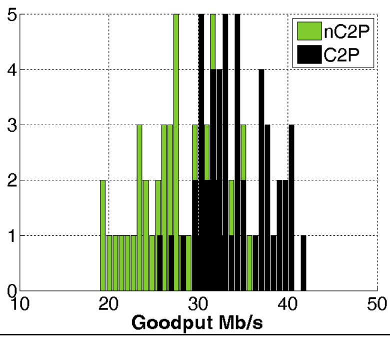
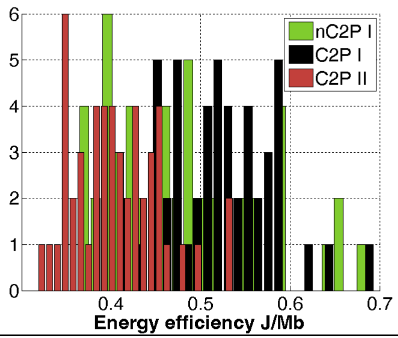
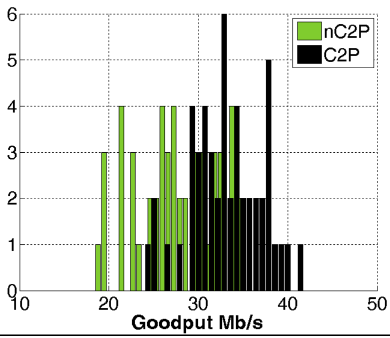

nC2P I
C2P I
C2P II
Energy efficiency J/Mb

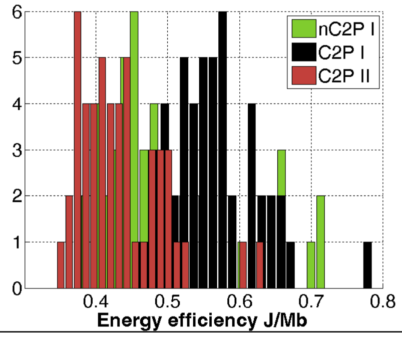
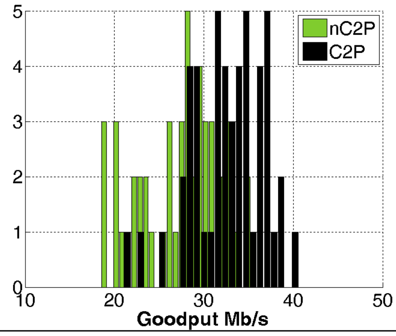
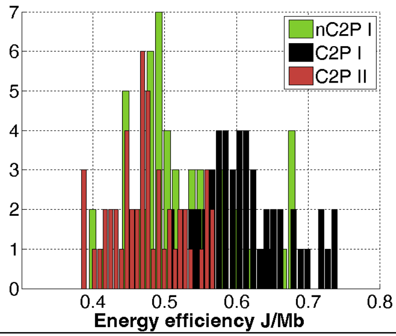
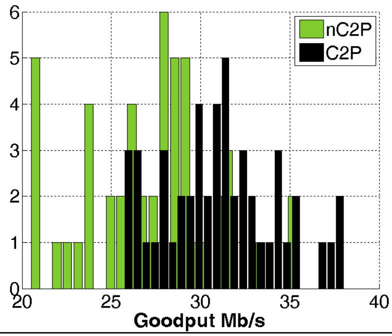
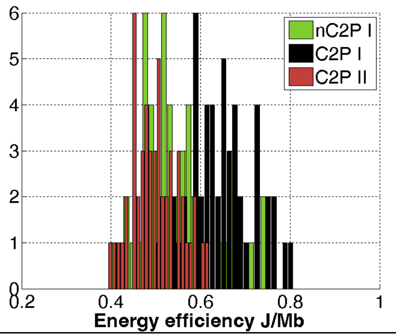
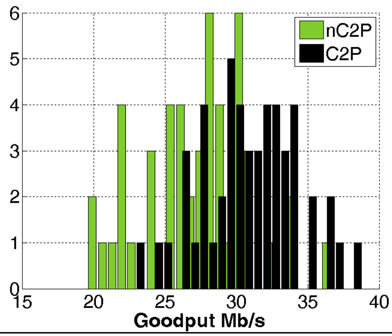
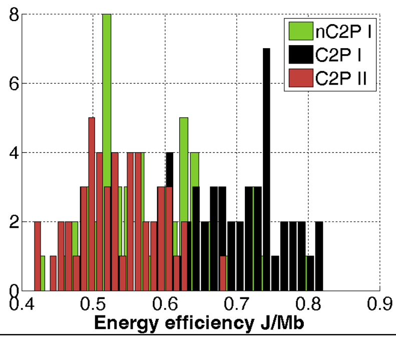
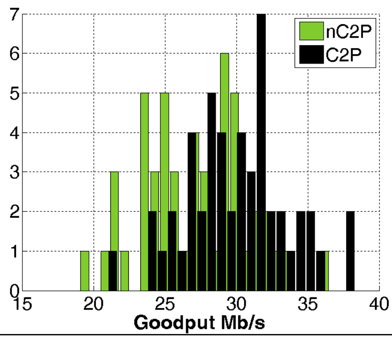
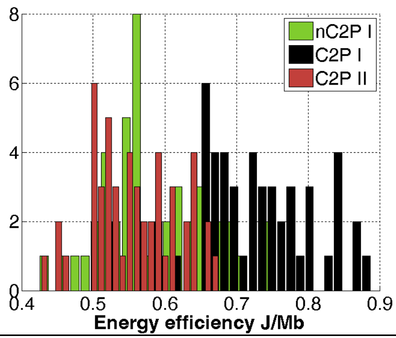
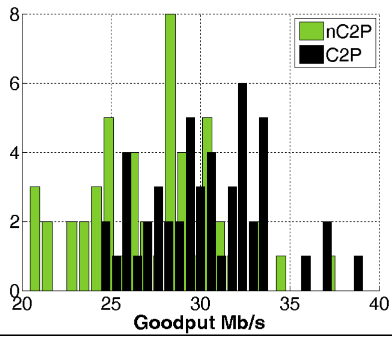

nC2P
C2P
Goodput Mb/s

10


nC2P I
C2P I
C2P II
Energy efficiency J/Mb


nC2P
C2P
Goodput Mb/s

11	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
12	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
13	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
14	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
15	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s

16	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
17	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
18	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
19	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
20	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s

Appendix II

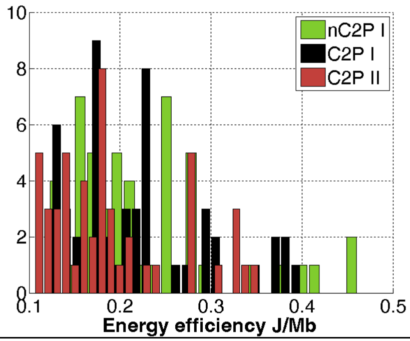
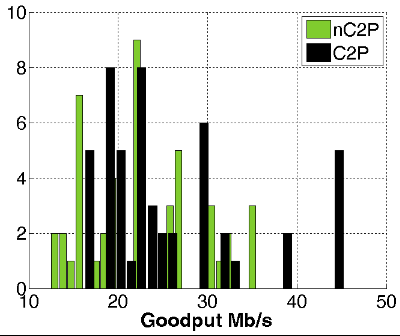
Detailed results and selected histograms of the simulation results for C2POWER scenarios 3 are shown below for reference.

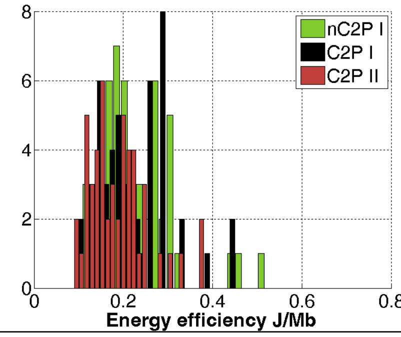
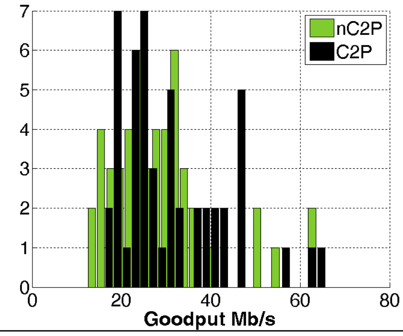
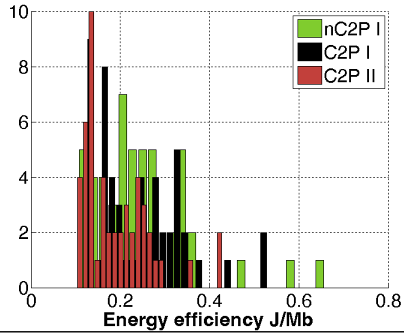
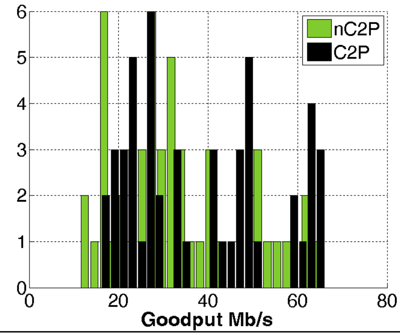
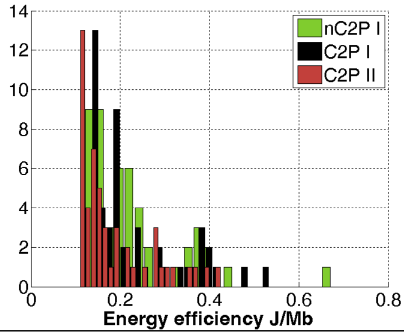
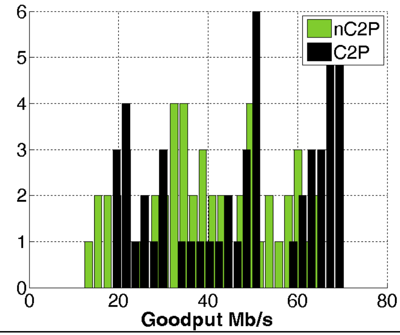
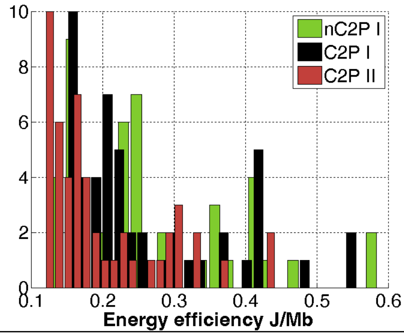
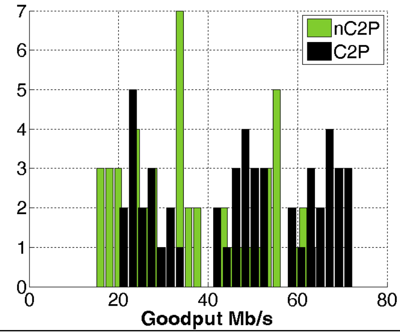
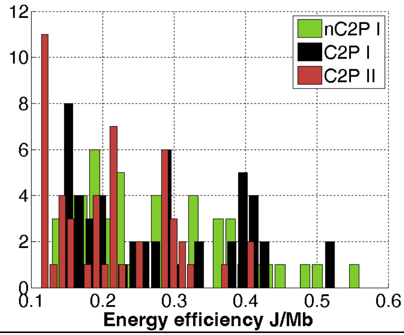
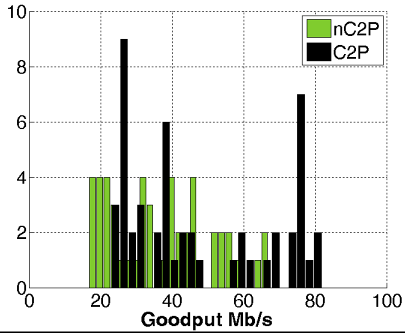
Detailed Results

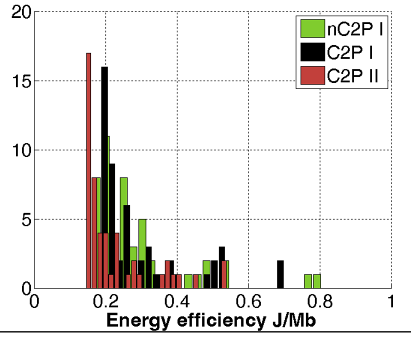
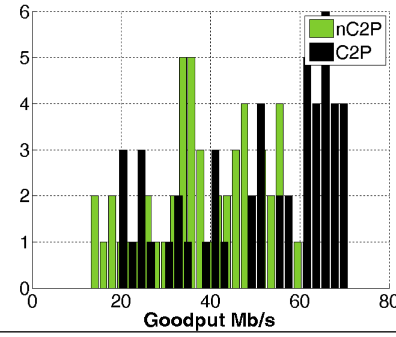
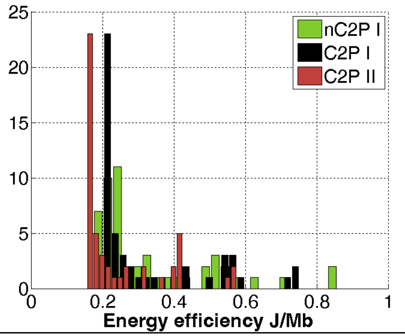
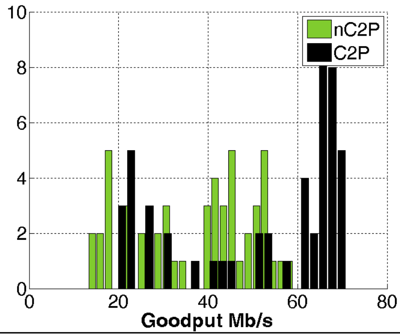
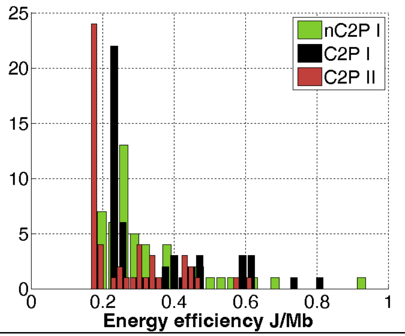
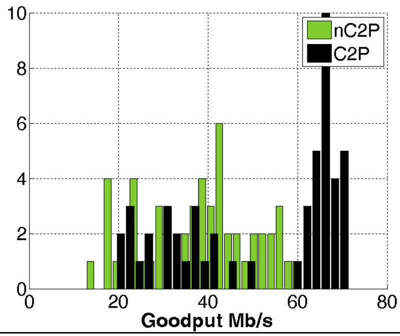
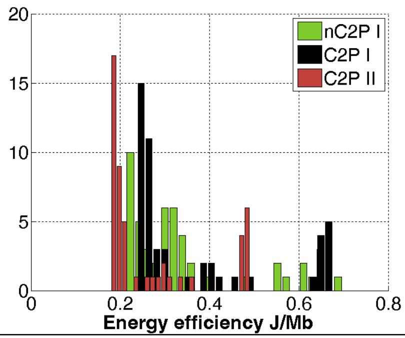
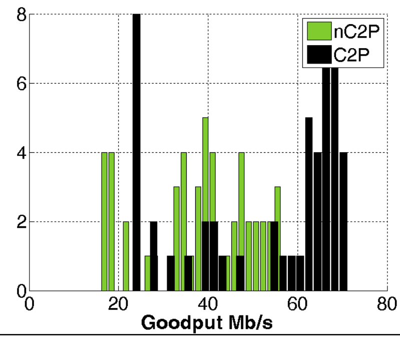
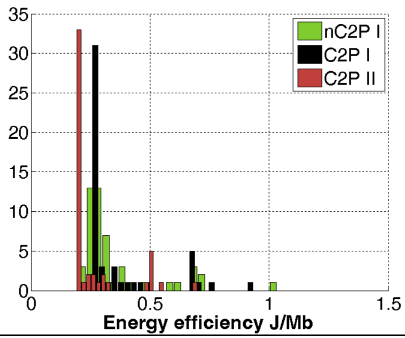
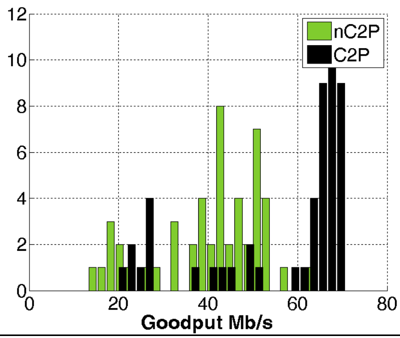
#nodes	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Energy consumption, C2Power cooperative cases (C2P) J/Mb																		
C2P I avg	0.220	0.227	0.239	0.232	0.259	0.280	0.293	0.327	0.350	0.363	0.360	0.384	0.367	0.385	0.397	0.424	0.475	0.489
C2P I med	0.208	0.217	0.210	0.188	0.217	0.275	0.233	0.234	0.249	0.269	0.274	0.299	0.312	0.328	0.342	0.356	0.383	0.394
C2P I min	0.125	0.098	0.125	0.136	0.150	0.145	0.188	0.203	0.221	0.238	0.256	0.274	0.287	0.304	0.236	0.241	0.350	0.360
C2P I max	0.398	0.451	0.529	0.534	0.556	0.526	0.700	0.751	0.820	0.676	0.935	1.196	0.844	0.843	1.119	1.300	1.238	1.301
C2P I std	0.076	0.078	0.098	0.103	0.109	0.107	0.132	0.162	0.156	0.159	0.170	0.176	0.134	0.140	0.154	0.199	0.209	0.239
C2P II avg	0.192	0.188	0.191	0.185	0.201	0.212	0.224	0.246	0.261	0.269	0.267	0.282	0.269	0.282	0.292	0.313	0.348	0.356
C2PII med	0.181	0.178	0.166	0.148	0.166	0.209	0.176	0.179	0.188	0.201	0.204	0.222	0.230	0.241	0.251	0.261	0.279	0.288
C2P II min	0.106	0.089	0.103	0.109	0.120	0.113	0.145	0.157	0.168	0.179	0.192	0.204	0.214	0.225	0.234	0.248	0.257	0.265
C2P II max	0.352	0.382	0.429	0.425	0.442	0.414	0.540	0.575	0.620	0.490	0.699	0.858	0.608	0.621	0.832	0.946	0.912	0.955
C2P II std	0.070	0.066	0.078	0.083	0.084	0.082	0.102	0.121	0.115	0.114	0.126	0.125	0.096	0.102	0.110	0.143	0.150	0.174
WiMax BS avg	0.133	0.119	0.112	0.106	0.110	0.129	0.124	0.123	0.139	0.135	0.133	0.185	0.155	0.165	0.150	0.176	0.174	0.167
WiMax BS med	0.112	0.107	0.103	0.089	0.100	0.122	0.110	0.110	0.132	0.137	0.128	0.148	0.144	0.160	0.151	0.166	0.164	0.157
WiMax BS min	0.059	0.046	0.050	0.046	0.057	0.060	0.059	0.059	0.059	0.059	0.062	0.074	0.064	0.062	0.059	0.064	0.059	0.059
WiMax BS max	0.234	0.339	0.444	0.339	0.234	0.319	0.343	0.237	0.343	0.357	0.342	1.774	0.412	0.412	0.237	0.546	0.344	0.343
WiMax BS std	0.055	0.053	0.063	0.060	0.044	0.055	0.061	0.047	0.064	0.048	0.051	0.236	0.067	0.070	0.045	0.091	0.061	0.065
WiFi AP avg	0.093	0.096	0.101	0.091	0.100	0.093	0.102	0.093	0.096	0.103	0.099	0.101	0.110	0.108	0.114	0.108	0.114	0.116
WiFi AP med	0.068	0.076	0.078	0.078	0.081	0.084	0.088	0.086	0.094	0.091	0.097	0.091	0.098	0.102	0.102	0.107	0.108	0.113
WiFi AP min	0.065	0.065	0.065	0.065	0.065	0.065	0.066	0.066	0.066	0.068	0.067	0.066	0.066	0.067	0.066	0.067	0.068	0.067
WiFi AP max	0.205	0.206	0.206	0.206	0.206	0.219	0.522	0.156	0.226	0.218	0.226	0.233	0.402	0.218	0.559	0.218	0.231	0.180
WiFi AP std	0.047	0.046	0.047	0.038	0.044	0.032	0.068	0.022	0.027	0.034	0.029	0.036	0.054	0.038	0.068	0.031	0.027	0.027
Energy consumption, non-cooperative benchmark cases (nC2P) J/Mb																		
nC2P I avg	0.226	0.227	0.245	0.227	0.261	0.269	0.288	0.336	0.331	0.352	0.357	0.384	0.386	0.403	0.404	0.424	0.481	0.502
nC2P I med	0.201	0.207	0.224	0.203	0.237	0.234	0.252	0.249	0.273	0.306	0.280	0.337	0.360	0.359	0.361	0.367	0.407	0.430
nC2P I min	0.122	0.091	0.106	0.120	0.129	0.128	0.162	0.174	0.184	0.213	0.198	0.238	0.232	0.264	0.251	0.256	0.292	0.300
nC2P I max	0.461	0.518	0.658	0.673	0.585	0.561	0.805	0.857	0.941	0.697	1.034	1.128	0.823	0.969	1.037	1.311	1.271	1.385
nC2P I std	0.084	0.086	0.110	0.109	0.114	0.108	0.141	0.173	0.150	0.145	0.174	0.162	0.127	0.156	0.151	0.189	0.209	0.249
nC2P II avg	0.238	0.237	0.255	0.235	0.270	0.277	0.296	0.344	0.339	0.359	0.365	0.392	0.393	0.410	0.410	0.431	0.489	0.509
nC2PII med	0.211	0.216	0.232	0.210	0.244	0.240	0.259	0.255	0.279	0.312	0.286	0.343	0.367	0.365	0.367	0.373	0.413	0.437
nC2P II min	0.130	0.096	0.110	0.124	0.133	0.132	0.166	0.179	0.188	0.217	0.202	0.243	0.236	0.269	0.255	0.260	0.296	0.305
nC2P II max	0.482	0.538	0.681	0.694	0.602	0.576	0.825	0.877	0.961	0.713	1.053	1.148	0.839	0.985	1.054	1.330	1.289	1.404
nC2P II std	0.087	0.090	0.114	0.113	0.117	0.111	0.145	0.177	0.153	0.149	0.178	0.166	0.130	0.159	0.154	0.192	0.212	0.252
nC2P II avg	0.226	0.227	0.245	0.227	0.261	0.269	0.288	0.336	0.331	0.352	0.357	0.384	0.386	0.403	0.404	0.424	0.481	0.502
nC2PII med	0.201	0.207	0.224	0.203	0.237	0.234	0.252	0.249	0.273	0.306	0.280	0.337	0.360	0.359	0.361	0.367	0.407	0.430

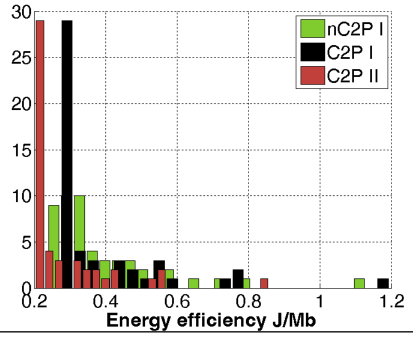
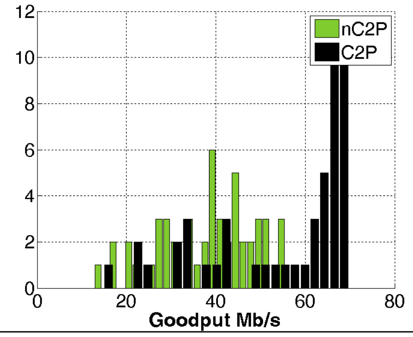
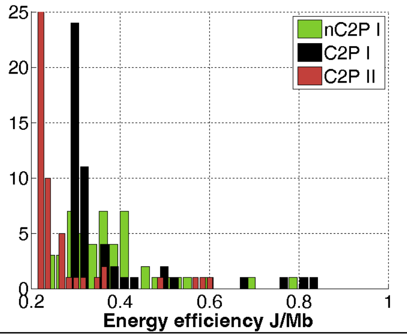
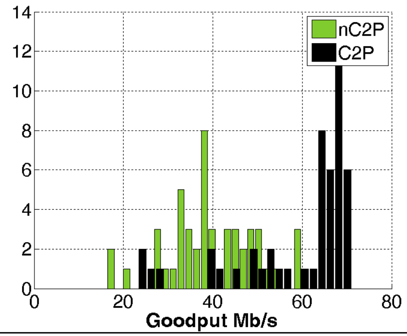
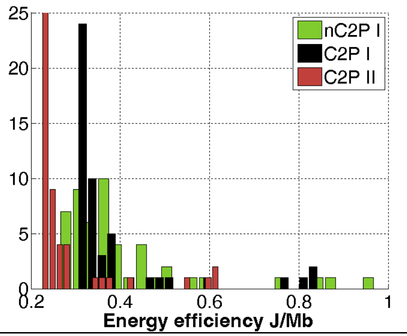
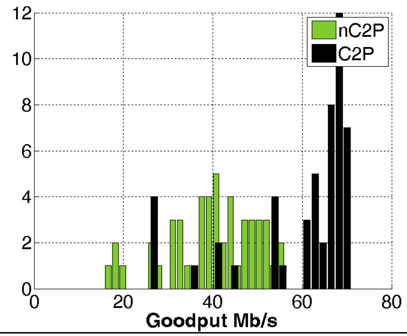
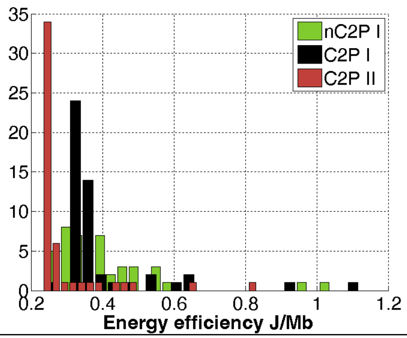
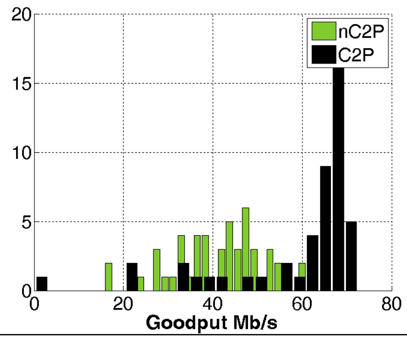
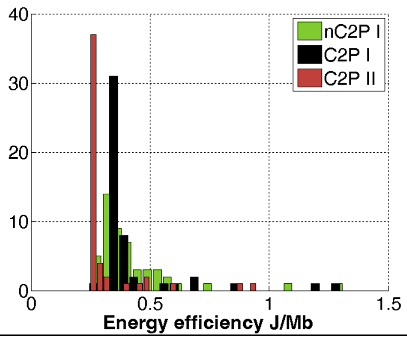
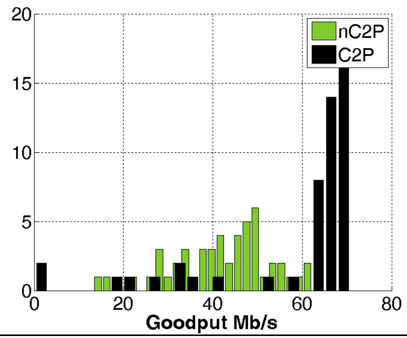
#nodes	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
nC2P II min	0.122	0.091	0.106	0.120	0.129	0.128	0.162	0.174	0.184	0.213	0.198	0.238	0.232	0.264	0.251	0.256	0.292	0.300
nC2P II max	0.461	0.518	0.658	0.673	0.585	0.561	0.805	0.857	0.941	0.697	1.034	1.128	0.823	0.969	1.037	1.311	1.271	1.385
nC2P II std	0.084	0.086	0.110	0.109	0.114	0.108	0.141	0.173	0.150	0.145	0.174	0.162	0.127	0.156	0.151	0.189	0.209	0.249
WiMax BS avg	0.142	0.130	0.129	0.129	0.142	0.154	0.163	0.173	0.179	0.186	0.196	0.241	0.249	0.264	0.232	0.253	0.258	0.253
WiMax BS med	0.130	0.120	0.128	0.108	0.136	0.139	0.152	0.145	0.148	0.169	0.184	0.216	0.258	0.266	0.226	0.240	0.242	0.240
WiMax BS min	0.082	0.046	0.048	0.046	0.064	0.076	0.061	0.072	0.065	0.075	0.063	0.089	0.077	0.077	0.075	0.077	0.075	0.074
WiMax BS max	0.234	0.339	0.444	0.431	0.289	0.455	0.461	0.528	0.461	0.526	0.447	0.612	0.578	0.563	0.607	0.560	0.516	0.567
WiMax BS std	0.051	0.058	0.061	0.072	0.056	0.070	0.075	0.084	0.088	0.085	0.088	0.118	0.105	0.110	0.103	0.115	0.106	0.111
WiFi AP avg	0.106	0.111	0.118	0.108	0.118	0.119	0.117	0.121	0.125	0.132	0.124	0.126	0.133	0.134	0.136	0.138	0.147	0.149
WiFi AP med	0.068	0.110	0.105	0.105	0.119	0.116	0.115	0.126	0.130	0.129	0.129	0.123	0.134	0.135	0.139	0.148	0.144	0.154
WiFi AP min	0.065	0.065	0.065	0.065	0.065	0.065	0.066	0.066	0.066	0.068	0.067	0.066	0.066	0.067	0.066	0.067	0.068	0.067
WiFi AP max	0.205	0.206	0.206	0.206	0.206	0.207	0.208	0.191	0.209	0.208	0.213	0.213	0.213	0.207	0.217	0.207	0.212	0.225
WiFi AP std	0.049	0.048	0.048	0.043	0.043	0.036	0.041	0.034	0.038	0.035	0.042	0.040	0.038	0.040	0.036	0.034	0.029	0.039
Goodput, C2Power cooperative cases (C2P) Mb/s																		
C2P avg	25.70	31.74	38.65	47.36	47.47	47.77	51.08	52.16	51.26	52.81	57.07	55.67	59.61	60.18	59.89	58.65	57.40	60.53
C2P med	22.60	27.70	34.33	49.48	48.63	39.47	55.90	62.07	62.89	62.47	66.09	63.78	65.46	65.88	66.48	66.23	65.16	66.90
C2P min	16.23	16.22	16.17	18.55	20.07	23.05	19.42	19.74	19.48	23.02	19.85	14.93	23.39	25.96	0.25	0.26	0.42	20.43
C2P max	45.31	65.69	66.17	70.43	72.38	81.93	70.88	70.84	71.51	71.19	70.74	69.80	71.06	70.99	72.42	70.72	72.40	72.77
C2P std	8.44	11.96	15.97	17.51	17.09	20.25	16.56	18.34	18.03	17.55	16.27	15.82	13.16	12.91	15.40	17.98	17.16	15.11
Goodput, non-cooperative benchmark cases (nC2P) Mb/s																		
nC2P avg	22.88	28.56	31.91	38.79	36.64	37.18	38.55	36.93	38.33	37.84	40.35	37.97	39.03	39.74	41.46	42.13	39.43	40.46
nC2P med	22.39	26.38	29.64	37.50	33.59	36.34	37.33	41.47	40.52	39.47	42.80	39.75	38.11	40.46	43.34	43.21	40.86	40.46
nC2P min	12.42	12.36	11.36	12.22	14.97	16.76	13.02	13.16	12.77	15.99	13.16	12.78	16.36	15.82	15.73	13.35	14.21	13.71
nC2P max	35.50	63.69	65.03	65.37	64.16	66.92	62.42	59.05	60.75	56.35	64.11	55.47	59.80	56.00	60.91	62.18	57.92	59.26
nC2P std	6.36	11.65	13.88	14.74	14.41	14.50	12.71	13.61	12.43	12.22	11.96	10.77	10.10	9.97	10.34	11.06	11.17	12.01

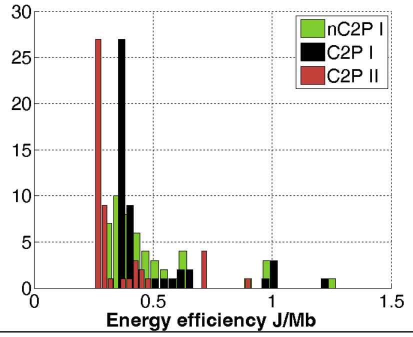
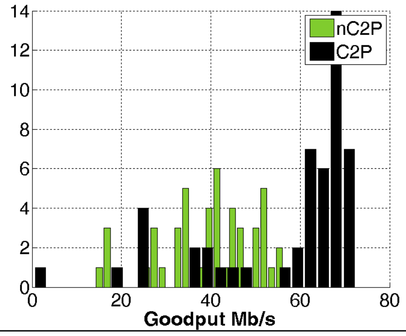
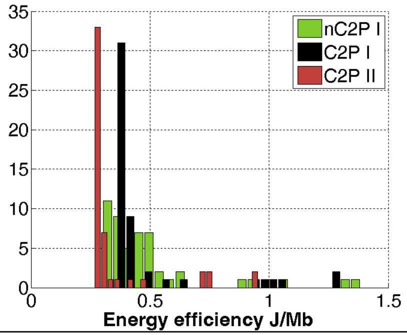
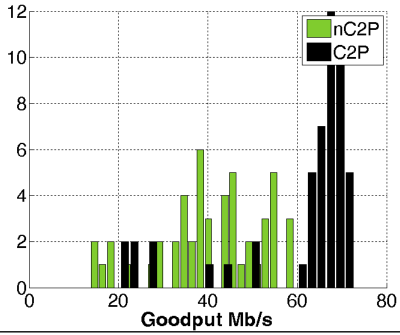
Selected histograms of Scenario 3 results.

#nodes	Energy efficiency	Goodput
3		

4	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
5	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
6	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
7	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s
8	 nC2P I C2P I C2P II Energy efficiency J/Mb	 nC2P C2P Goodput Mb/s

<table>
<tr><td>9</td><td>
</td><td>
</td></tr>
<tr><td>10</td><td>
</td><td>
</td></tr>
<tr><td>11</td><td>
</td><td>
</td></tr>
<tr><td>12</td><td>
</td><td>
</td></tr>
<tr><td>13</td><td>
</td><td>
</td></tr>
</table>

<table>
<tr><td>14</td><td>
</td><td>
</td></tr>
<tr><td>15</td><td>
</td><td>
</td></tr>
<tr><td>16</td><td>
</td><td>
</td></tr>
<tr><td>17</td><td>
</td><td>
</td></tr>
<tr><td>18</td><td>
</td><td>
</td></tr>
</table>

19	nC2P I C2P I C2P II Energy efficiency J/Mb	nC2P C2P Goodput Mb/s
20	nC2P I C2P I C2P II Energy efficiency J/Mb	nC2P C2P Goodput Mb/s