IoT_ProSe: Exploiting 3GPP Services for Task Allocation in the Internet of Things

Virginia Pilloni[a], Emad Abd-Elrahman[b,d], Makhlouf Hadji[c], Luigi Atzori[a], Hossam Afifi[b]

[a] *Department of Electrical and Electronic Engineering, University of Cagliari, Italy*
[b] *Institut Mines-Telecom (IMT), Telecom SudParis, Saclay, France*
[c] *Technological Research Institute, SystemX, Palaiseau, France*
[d] *Computer and Systems Department, National Telecommunication Institute (NTI), Cairo, Egypt*

Abstract

This work focuses on the problem of task allocation among mobile objects that aim to collaborate in the execution of IoT (Internet of Things) applications. We consider 3GPP Proximity Services (ProSe) that enable Device to Device (D2D) communications for direct interaction among IoT objects. We first define the framework where devices willing to cooperate: i) discover and join a cluster of objects to take part to the IoT_ProSe service; ii) compete with the other objects in the cluster when a request for a service is sent to the cluster from the IoT application server. We then focus on two important problems. The first problem is the decision about which node should win the competition, for which we propose a game-theory based approach to find a solution maximizing objects' utility functions. The second problem is the computation of the optimal size of devices' clusters within a cell by maximizing the signal quality across the cell. Experimental results provide insights on the strategy performance. By appropriately setting working parameters, it is possible to improve system reliability up to 21%, and system lifetime up to 68%.

Keywords: D2D, Internet of Things, Game Theory

1. Introduction

LTE (Long Term Evolution) based D2D (device-to-device) communication is considered as a promising solution for many short/long range communication services in 4G and 5G networks [1] [2]. This allows for the creation of D2D clusters [3], which can be defined as cellular communication groups that implement

Email addresses: virginia.pilloni@diee.unica.it (Virginia Pilloni), emad.abd_elrahman@telecom-sudparis.eu (Emad Abd-Elrahman), makhlouf.hadji@irt-systemx.fr (Makhlouf Hadji), l.atzori@diee.unica.it (Luigi Atzori), hossam.afifi@telecom-sudparis.eu (Hossam Afifi)

Proximity Services (ProSe) without (or with small) intervention of the central e-NodeB (eNB). Those groups can communicate internally using D2D-Direct mode, whereas externally they can do it by using D2D-Assisted by eNB.

For several IoT (Internet of Things) application scenarios, D2D communications represent a good opportunity. Objects in the IoT can collaborate in the provisioning of a cooperative activity locally, when served by the same eNB. Indeed, objects that are close to each other may be capable of performing similar tasks (e.g., sensing the temperature in a given location) and may need to collaborate to decide which of them should perform the task or may collaborate to provide a more accurate value acquired by more sensors. This collaboration can leverage the D2D communication scenario so that a local cluster is created among these objects in a dynamic way and they can decide which one should be assigned to a given task. We assume that each group has a cluster head (CH), which can be either auto-elected or elected by the other nodes in the group. Downstream communication can happen through the CH or directly from the eNB. Upstream communication and inter-cluster communication has to go through the CH.

Indeed, this is our scenario for task allocation in the IoT where different objects opportunistically can take part to task groups and coordinate with the other members for the provisioning of services to applications running on top of the IoT. The contributions of this paper are the following:

- We present the framework exploiting D2D communications for task allocation in the IoT, which we call IoT_ProSe. Accordingly, objects willing to collaborate in the implementation of IoT applications may create new clusters or join existing ones. Then, they collaborate with the other peers in the cluster to provide local services.
- We analyze the problem of task allocation within a cluster, where objects capable of performing the same tasks compete to get relevant remunerations (based on a utility function). Task allocation is assessed taking into account both node reliability and lifetime. To this, we propose a game-theory based approach where objects find a solution towards the maximization of their own utility.
- We evaluate the impact of the number and size of clusters on the communication interference in a cell and we provide a solution to optimize the setting.
- We perform extensive simulations which are focused on the evaluation of the energy consumption, provided reliability and distribution of tasks among all the available nodes.

The rest of the paper is organized as follows. In Section 2, we present the background in 3GPP ProSe services and past works in task allocation in the IoT. In Section 3, we present the proposed framework IoT_ProSe. In Section 4 we present the game theory based approach for task allocation, whereas in Section 5, we address the issue of optimizing the number and size of clusters of objects. Then in Section 6, we present the experimental results. Finally, conclusions are drawn.

2. Work Background

2.1. The 3GPP ProSe services

3GPP started the Proximity Services (ProSe) either for direct communication or public safety applications [1] [2]. As a result, LTE based D2D communication is considered as a promising solution for many short range communication services in 4G and 5G networks. IoT is one of the promising emerging technologies that can be integrated in the ProSe applications. Integrating IoT as a service in cellular architectures will add some exchanges for D2D discovery procedure proposed by 3GPP. Moreover, it will extend the benefits of universal LTE architecture facilities (i.e. reliability, quality and security) to IoT applications and platforms. In this work we focus on the ProSe direct communication mode [1], which has three techniques: one-to-one, one-to-many or ProSe UE-to-network relaying. We will detail in Section 3.3 this stage for the one-to-many use case.

According to D2D-4G Proximity Services' use cases classifications [1] [4], the commercial and social ProSe(s) are imposed as an important category. D2D clustering communications had been proposed in order to avoid interferences in cellular networks due to the services expected on D2D. It can also enhance the eNB throughput by distributing separate groups constructed based on the proposed 3GPP ProSe type. In our previous works [5] [6], we handled the D2D clustering in terms of the Intelligent Transportation System (ITS) ProSe. The proposed architecture is called Cellular Vehicular Network (CVN) and aims at extending the V2V networks using D2D aspects to enhance the covering range, accelerate data dissemination or data aggregation, and also improve the overall throughput for eNB in such emergency applications.

ProSe services have a good business model in the IoT. Task allocation optimization in IoT devices formed by D2D groups is an incentive solution for improving ProSe based IoT architecture. Notwithstanding the importance of coordinating IoT objects in the deployment of distributed applications and the potentialities of the ProSe services, this issue has not been addressed in the past.

2.2. Past works in task allocation in IoT

The problem of task allocation consists in finding the optimal task executor among several available providers. This problem has been extensively studied in the Wireless Sensor Network (WSN) field [7], where a big effort has been put into resource allocation to extend WSN lifetime. In [8], a centralized task allocation algorithm is proposed, which assumes that a single node has a complete updated view of the status of all the nodes in the wireless network. This study focuses on the reduction of the overall energy consumption into a heterogeneous WSN, with attention to nodes' residual energy. In [9] the same problem is analyzed taking into account also task execution time.

In the IoT, the problem is different as it happens that objects frequently join and leave the network as they are switched off/on by the owner depending on her personal needs; the objects' connectivity is also often not very reliable so that they disappear from the network and leave the community of IoT objects; additionally, they also move so that their geographical position varies and the task they are able to perform changes. This scenario entails the situation in which objects that in a given area may perform a task of interest for IoT applications may change over the time and there is the need for a dynamic coordination of objects in a given area to decide which one should perform the task of interest. Most of the existing studies on resource allocation for IoT are focused on IoT service provisioning, such as in [10] and [11]. In these studies, the aim is to find and allocate the resources that enable service execution. Only a few works in the literature try to find the optimal resource allocation associated with the lowest impact of the application assigned to the network. This is the case for instance of papers [12] [13], where a distributed optimization protocol, based on consensus algorithm [14], addresses the problem of resource allocation and management. Still the authors in this work assume that the objects are connected through short-range communication technologies. Additionally, no remuneration is considered to be given to the objects that finally provide the requested service. In this respect, we are proposing a new approach in the present paper: the use of the 3GPP ProSe service for the connectivity between objects and the presence of a remuneration for the sensors providing the service.

2.3. Clustering Methods Background

The state of the art in clustering focuses on different architectures for adhoc wireless sensor networks, which can be centralized, distributed and hybrid [15]. The centralized one represents the current status for 4G network devices controlled by the eNB as a central node for the whole cluster. While this architecture can suffer from the interference generated by many devices covered by the same cell, it can be extended to be a distributed clustering that selects some CHs to enable the communication from other devices to and from eNB. Those CHs are linked to eNB through direct communications (one hop). The hybrid architecture can exploit previous methodologies to control the usage of the upstream and downstream link traffic to and from eNB. The measures that are typically used to drive the cluster creation can be parametric (such as costs related to probability functions in terms of transmission power and interference/distances D2D & cellular communications) and non-parametric (based on heuristic, such as mobility patterns similarities and density distribution of D2D users).

In the current developments of emerging technologies like IoT and D2D, the massive number of IoT devices that can use D2D-4G communication technology will bring heavy burden to eNB for assisted communications. So, creating clusters of devices can ameliorate the overall performance for eNB and reduce the interference across the communication range. Moreover, the D2D can improve the overall spectrum efficiency across the covering range [16]. Different D2D clustering schemes have been studied in [17]. In this work, the authors supposed three alternatives for grouping the D2D devices; CH dependent, CH independent and threshold-based signal quality. The main idea behind those algorithms is to form the clusters that enable the devices to discover relevant peers in their coverage to construct the D2D connections.

The work in [18] introduced a clustering algorithm for enhancing the overall network survivability. Through this algorithm, the CH is selected with some redundancy backup CHs for failure recovery using probability models. Authors in [19] proposed a contextualized approach for D2D clustering to include social aspects besides separation distance as physical parameter. The work proposed in [20] used stochastic modeling tools to measure the performance of engaged D2D communication in proximity services. Moreover, they focused on the optimal number of D2D links that should be simultaneously activated in each cluster. Recently, the work proposed in [21] made use of a good opportunistic model to optimize the D2D cluster size in order to maximize the number of active cooperative D2D users. In the same context, we are proposing in this paper a mathematical approach for optimizing the D2D clusters' size in terms of separation distances and interference levels.

3. Proposed Framework

In this section, we firstly describe the reference scenario for the cooperation of objects taking part to the deployment of IoT applications and the proposed IoT_ProSe solution. We then present the relevant architecture and the group discovery process that is implemented by each object to take part in a cluster.

3.1. Reference Scenario

One of the greatest potentialities expected by IoT deployments is the opportunistic collaborative sensing of devices. It is the scenario where nodes allow their sensors to be remotely tasked on someone else behalf, collecting and reporting sensor data on a best-effort basis when the conditions permit. In this way, sensors capabilities can be exploited by other users other than by the object owner for the benefit of the collaborative communities.

In the depicted scenario, it frequently happens that some nodes perform the same sensing operation, such as the measurement of the traffic on the same street, the measurement of the humidity and/or the temperature in a room, the detection of moving objects/persons in a given environment, the monitoring of the luminosity in a public square. Accordingly, groups of nodes are identified, namely, task clusters, where each member performs similar and replaceable tasks. To understand the meaning of task cluster, suppose, for example, that the network is performing a temperature sensing in a specific area: only those nodes that are equipped with a temperature sensor and that are deployed within that area are included in the task cluster related to this task.

We assume that the IoT applications are running somewhere in the cloud in an Application Server (AS) and that they frequently need information and services related to the physical world that can be provided by the IoT objects, as sketched in Fig. 1. In order not to burden central ASs with the duty of tracking and recording all the available nodes in each task cluster, we foreseen the self-creation of this node clusters with the presence of a cluster head (the already mentioned CH) that locally coordinates the activities within the group and manage the communication with the IoT platform(s) in the cloud to connect with the applications requiring specific tasks/operations.

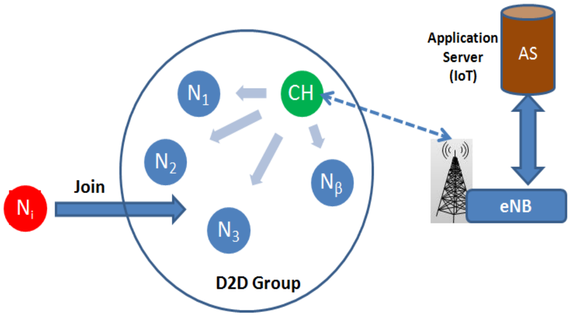


Figure 1: Reference scenario with the application server and cluster of objects.

3.2. IoT_ProSe Solution

To address the needs of the previously depicted reference scenario, we propose the exploitation of the 3GPP ProSe Service without the need to deploy a dedicated IoT platform infrastructure. Accordingly, objects willing to cooperate in the provisioning of IoT services take part in the creation of ProSe clusters, bringing to the IoT_ProSe solution.

In the current developments of emerging applications, the massive number of IoT devices with cellar communication technologies will bring a heavy burden to eNBs. So, creating clusters of devices can improve the overall performance for eNBs and reduce the interference across the whole communication range. This benefit is in addition to the reduction in the workload at the AS to keep track of each single available object, as described in the previous subsection.

Whenever the IoT AS requests some data, this request is forwarded to the relevant CH(s), which then triggers the distributed task allocation algorithm within the members of the cluster to decide which nodes will address this request. Once the allocation task procedures within each cluster are terminated, then the CH forwards the resulting bid to the AS, that then decides from which cluster to buy the service.

Fig. 2 depicts our Cellular IoT (CIoT) architecture, which integrates the ProSe service of IoT with the Evolved Packet Core (EPC) plane components (i.e., MME, HSS, S-GW and P-GW). IoT services and applications can benefit from this D2D facility. Our architecture relies on the D2D as detailed hereafter,

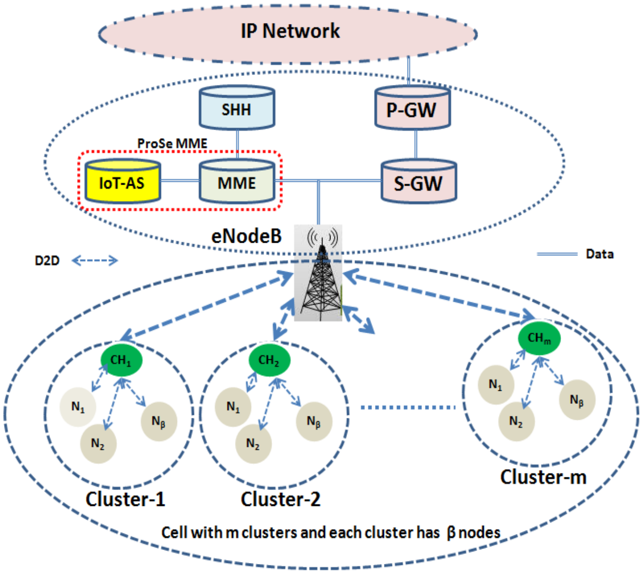


Figure 2: Cellular IoT_ProSe Architecture

to announce the service (bidding process), to manage subscribers (bidders) and to collect the periodic local information and transfer it to the querying server.

3.3. Group discovery process in IoT_ProSe

In the ProSe discovery procedure in [1], a ProSe-enabled UE (User Equipment) can be in the Announcing mode, the Monitoring mode or both. In the Announcing mode, presence information including a ProSe_App_Code is broadcast by the device to all nodes in its coverage range to announce a specific ProSe service. In the Monitoring mode, a device monitors information from a specific ProSe service using a ProSe_mask: upon the reception of a ProSe_App_Code announced by an Announcing ProSe UE, the Monitoring ProSe UE performs a matching report procedure in which it checks if the received code matches with the mask that it is monitoring. In the CIoT architecture, we assume that initially all IoT ProSe UEs are configured in the Monitoring mode with a global IoT ProSe_mask. This ProSe_mask is common to all IoT services and allows all CIoT Monitoring nodes to communicate with all CIoT Announcing nodes independently of the used ProSe application.

Cellular communications are based on the allocation of physical resource blocks (PRBs) corresponding to groups of wireless symbols (including OFDM). This applies to normal communication and on the complete D2D process (discovery and communication). The 3GPP Discovery Model [1] involves one UE announcing "as I am here". The other nodes will be in monitoring mode to join this CH. The sequence diagram in Fig. 3 explains the discovery phase for all nodes and the resources assigned for the CHs (Announcing mode nodes). Step 0:

this is the initialization phase for 4G network asking authentication and authorization based on the smart card-USIM information. Step 1: the UE configuration for IoT_ProSe application launches the specific service. Step 2: the UE CH sends the ProSe_Discovery_Service_Request specific to our scenario, asking for service authorization and resource allocation using the required App_ID and its IMSI information. Step 3: the integrated IoT_Prose with MME will check the ProSe_App_ID against the registered list in its database. Step 4: if access control is ok, the ProSe_Authorization_Procedure will start between a server called MME and the ProSe Function in the UE according to the 3GPP standard [1]. Step 5: after the authorization, the MME will generate the ProSe_App_Code for this proximity service applying required filtration and service masking. Step 6: the ProSe_Discovery_Service_Response answers with App_code and Timers. Also, it will define the IoT service type defined by the IoT_ProSe entity. Step 7: the UE sets the configuration parameters defined by the previous message acting as D2D Monitoring or Announcing modes. Step 8: the Radio Resource Controller assigns the PRB for discovery process by the eNB. Step 9: the discovery process starts. Step 10: CHs are in the announcing mode and all other nodes will be in monitoring mode. Who will be acting as CH is defined by the eNB as discussed in Section 5.

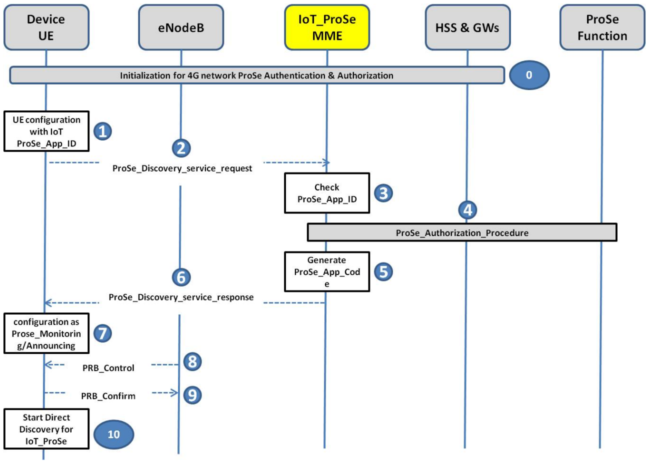

Device UE
eNodeB
IoT_ProSe MME
HSS & GWs
ProSe Function
Initialization for 4G network ProSe Authentication & Authorization
0
UE configuration with IoT ProSe_App_ID
1
2
ProSe_Discovery_service_request
Check ProSe_App_ID
3
4
ProSe_Authorization_Procedure
Generate ProSe_App_Code
5
6
ProSe_Discovery_service_response
configuration as Prose_Monitoring/Announcing
7
PRB_Control
8
PRB_Confirm
9
Start Direct Discovery for IoT_ProSe
10

Figure 3: Discovery phase for Cellular IoT_ProSe Architecture

After the allocation of ProSe discovery radio resources PRBs, ProSe Monitoring devices that enter the coverage area of an eNB will receive information of all the available ProSe CHs in their communication range. They receive the PRBs that are used for direct communication and the IoT bidding process can start normally. The CH forwards the local information to the querying server.

In the next section, Section 4, we describe how the task assignment is executed in a single cluster. In the following section, Section 5, we then analyze the problem of selecting the CH and the identification of the most optimal size of the clusters.

4. Task Assignment

Once the task clusters are formed, each one handles the assignment of the tasks to each node within, whenever a relevant request arrives at the CH from the AS. Let us then assume that the number of nodes in the cluster is N (excluded the CH). The task assignment procedure is aimed at deciding in which order the objects in the group will be inquired by the CH to perform the sensing and send the relevant output whenever requested by the AS. The result of this procedure is then a vector V with the indexes of the nodes in order of priority:

$$V = \{v_1, ..., v_N\} \tag{1}$$

where x_{v_1} is the node with the highest priority and x_{v_N} is the one with lowest priority.

The task assignment algorithm usually starts when the CH advertises an offer for task k with a remuneration r_k (expressed in coins per second) and a frequency q_k (frequency of performing the required sensing task). Following the advertisement phase, each node x_i interested in the task sends a bid $b_{k,i}$ (which should be lower than r_k). This bid is then used by the CH to generate vector V, which is then broadcast to all the nodes in the cluster. It results by ordering the nodes according to the bid and to the reliability of the node. Specifically, the following index is computed

$$Y_{k,i} = b_{k,i} - w_k R_{k,i} \tag{2}$$

where $R_{k,i}$ is the reliability level assigned to node i to complete task k. The priority vector V is then created putting in increasing order the values $Y_{k,i}$, i.e. the node with the highest priority is the one with the lowest $Y_{k,i}$. Note that w_k is a weighting factor selected by the CH per task on the basis of the importance of the money with respect to the reliability of the provided service. When a sensing task has to be performed, the CH queries the nodes according to the priorities and then a number of cycles are waited till one node provides the response according to the priorities in V. Each node x_i provides its bid taking into account:

- The gain it can get from this bid;
- The cost in terms of energy consumption, which affects its lifetime.

The object utility can be expressed as follows:

$$H(b_{k,i}) = \Psi(b_{k,i})(b_{k,i} + \alpha u^e_{k,i}) \tag{3}$$

where $\Psi(.)$ is the probability to win the competition, $u^e_{k,i}$ is the cost associated with the use of the battery and α is a (negative) parameter that weights the energy cost with respect to the gain. Let us consider that each object generates a bid independently from the task k and to associate the variable b to the generic bid of any object in the community of objects in the cluster whatever the task k and that this variable is distributed according to pdf $f(.)$ and cdf $F(.)$, then

$$\Psi(b_{k,i}) = (1 - F(b_{k,i}))^{N-1} \tag{4}$$

We may assume that $f(.)$ is a distribution which is estimated dynamically by the cluster nodes and whose average μ and variance σ values are computed by the objects themselves from past observations. This assumption will be discussed during the experiments. As to $u^e_{k,i}$, it depends on the battery energy E_i at the moment of receiving this advertisement and on the frequency q_{i-k} at the which the object is executing other sensing functions. Let us assume that for the same sensing task any object consumes the same amount of energy e_k; then the lifetimes without and with the new task are the following:

$$L_{i-k} = \frac{E_i}{e_k q_{i-k}} \tag{5}$$

$$L_{i+k} = \frac{E_i}{e_k(q_{i-k} + q_k)} \tag{6}$$

The cost $u^e_{k,i}$ can be expressed in terms of ratio between the reduced lifetime (in case the competition is won) and the current lifetime:

$$u^e_{k,i} = \frac{L_{i+k}}{L_{i-k}} = \frac{q_{i-k}}{q_{i-k} + q_k} \tag{7}$$

$u^e_{k,i}$ expresses the amount of battery life (in time units) the object is devoting to task k for any time unit of its life.

4.1. Node reliability

The reliability of a mobile node i for a given task k depends on two components: data accuracy that it can provide for the required task $\mathbb{A}_{i,k}$, and node's trustworthiness, which is its reputation $\mathbb{T}_i$

$$R_{k,i} = \gamma \cdot \mathbb{A}_{k,i} + (1 - \gamma) \cdot \mathbb{T}_i \tag{8}$$

where γ is a weight that can be adaptively modified based on the required relation between data accuracy and node reputation. Data accuracy only depends on the capability of node i to fulfill task k's requirements, regardless of its past behavior. On the other hand, node's trustworthiness does not depend on the task demanded by the CH, but it only depends on the number of times that the node has been able to provide reliable data before.

Data accuracy for node i and task k relies upon task's requirements. The task requirements that mainly influence the provided data accuracy are: node location with respect to the required location (which is influenced by node speed); timeliness of provided data with respect to the required timeliness. Thus, motivated in part by [22], data accuracy can be estimated as

$$\mathbb{A}_{k,i} = \exp\left(-\Delta\left(p_i, p_k^{req}\right)\right) + \exp\left(-\Delta\left(t_i, t_k^{req}\right)\right) \tag{9}$$

where $\Delta(\cdot)$ is the distance between two positions or two time instants, p_i is node i's position, p_k^{req} is the position required for data resulting from task k, t_i is the time instant when node i is in position p_i, and t_k^{req} is the time instant at which data is required for task k. Since we are referring to a mobile node, its position will change after its reply to the CH. Therefore, its location is estimated based on the speed at which it is moving, and by the direction it is going towards.

As far as node's trustworthiness is concerned, it does not depend on the task to be assigned, but it only depends on the maliciousness of the node. To evaluate it, the CH observes the quality of the service obtained from the nodes each time they are requested to. Specifically, the following factors are considered: the node computational complexity C_i, how often the node provided a response M_i, how close the accuracy was to the expected one $B_{k,i}$, and the number of past observed transactions N_i. The trustworthiness level is then computed as follows

$$\mathbb{T}_i = \frac{C_i}{1 + log(N_i + 1)} + \frac{log(N_i + 1)}{1 + log(N_i + 1)} \left(\frac{M_i + \sum_k B_{k,i}}{N_i} \right) \tag{10}$$

where all the parameters are in the range 0–1, except N_i. Note that when no past observations have been collected, the trust is equal to the node complexity, which is a static characteristic of the object and does not vary over the time. The rationale is that we expect a smart object to have more capabilities to cheat with respect to a "dummy" object, leading to riskier transactions. Accordingly, the lower the complexity the higher C_i. As the number of transactions increases, the other two elements gain more importance. $\frac{M_i}{N_i}$ measures directly the frequency of responses received whenever i has been queried, so that it is equal to 1 if a response has always been given. $B_{i,k}$ measures the difference between the expected and real accuracy observed in the past and is computed as follows

$$B_{k,i} = \mathbb{A}_{k,i}^{obs} - \mathbb{A}_{k,i}^{est} \tag{11}$$

where $\mathbb{A}_{k,i}^{obs}$ is the observed accuracy and $\mathbb{A}_{k,i}^{est}$ is the estimated one according to (9). Note that other solutions can be adopted in case the proposed system can inter-operate with other IoT platforms and systems. For instance, the trust management algorithms proposed in [23] could be used by the proposed system, so that the CH won't need to compute the trust of each object but could just query the existing external services.

4.2. A Game-Theory Solution

This section describes and formulates the problem of objects' task assignment as a mathematical model using game theory. Let N be the total number of objects sharing the tasks provided by node CH. Each object i can propose a bid noted by $b_{k,i}$ for a task k. Furthermore, a reliability level $R_{k,i}$ is associated to node i to complete task k, so that the priority index $Y_{k,i}$ can be computed according to (2).

The objective of object i consists of maximizing its own utility function given by (3), and at the same time each object j such that $j \neq i$ is also maximizing its utility function. The problem to consider in this section consists to find the rational amount $b_{k,i}$ for each object i involved in the same game, leading to maximize its utility function. To win the game, each object i should propose a bid $b_{k,i}$ such that its priority index $Y_{k,i}$ is the smallest one.

The central or main goal of this section is to find the rational value of the bid to be proposed to win the task when maximizing a utility function. This is a non-cooperative game, where each object is interested in optimizing its utility function, and this can be modeled as a Nash game [24].

By solving the Nash game, we obtain the Nash Equilibrium Point (NEP) in which each object has no incentive to deviate unilaterally from the NEP (see definition of the NEP in [24]).

Proposition 4.1. *There exists a Nash Equilibrium Point $b^*_{k,i}$ for the described game. It is given by the equation:*

$$F(b^*_{k,i}) = 1 + C_0 \left(b^*_{k,i} + \alpha \frac{q_{i-k}}{q_{i-k} + q_k} \right)^{-1/(N-1)} \tag{12}$$

Proof. Recall that $(\Psi(b_{k,i}))$ is the probability to win the task by the object/player i when proposing a bid $b_{k,i}$. Let $Y_{k,i}$ be a random variable following a given probability density function, and for each player j, we also associate a random variable noted by $Y_{k,j}$ following the same density function as i. Then, for a player i, winning the task k consists to propose the value of $b_{k,i}$ that corresponds to the smallest value of $Y_{k,i}$. In other words, to win the task k, player i should propose the following action/bid:

$$b_{k,i} = w_k R_{k,i} + \min\{Y_{k,1}, Y_{k,2}, \dots, Y_{k,N}\} \tag{13}$$

Thus, to characterize the value of $\Psi(b_{k,i})$ for an object i, we have:

$$\Psi(b_{k,i}) = P(Y_{k,j} > Y_{k,i}, \forall j \neq i) \tag{14}$$

Assuming the distributions are identically independent, we find:

$$\begin{aligned} \Psi(b_{k,i}) =& P(b_{k,1} - w_k R_{k,1} > b_{k,i} - w_k R_{k,i}) \times \dots \\ & \times P(b_{k,N} - w_k R_{k,N} > b_{k,i} - w_k R_{k,i}) P(b_{k,N} > b_{k,i}) \end{aligned} \tag{15}$$

$$\begin{aligned} \Psi(b_{k,i}) =& \left(1 - P\left(b_{k,1} - w_k R_{k,1} \leq b_{k,i} - w_k R_{k,i}\right)\right) \times \dots \\ & \times \left(1 - P\left(b_{k,N} - w_k R_{k,N} \leq b_{k,i} - w_k R_{k,i}\right)\right) \end{aligned} \tag{16}$$

This leads to obtain:

$$\Psi(b_{k,i}) = (1 - F(b_{k,i} - w_k R_{k,i}))^{N-1} \tag{17}$$

where $F(b_{k,i} - w_k R_{k,i}) = P(Y_{k,j} \leq Y_{k,i})$ is the cumulative distribution function (i.e. cdf) of the random variable $Y_{k,i}$.

Using the result of (17), one can give an explicit formula of (3) as follows:

$$U(b_{k,i}) = \left[(1 - F(b_{k,i} - w_k R_{k,i}))^{N-1} \right] \left[b_{k,i} + \alpha \frac{q_{i-k}}{q_{i-k} + q_k} \right] \tag{18}$$

To find the Nash Equilibrium Point of this game, we investigate solutions of the derivative of the utility function given by (18). This leads to finding the rational bid allowing to win the task k.

$$\frac{\partial H(b_{k,i})}{\partial b_{k,i}} = 0 \tag{19}$$

This is equivalent to write:

$$\begin{aligned} &\left[(1-F(b_{k,i}-w_kR_{k,i}))^{N-2}\right](1-N)F'(b_{k,i}-w_kR_{k,i})\times \\ &\times\left(b_{k,i}+\alpha\frac{q_{i-k}}{q_{i-k}+q_k}\right)+\left[(1-F(b_{k,i}-w_kR_{k,i}))^{N-1}\right]=0 \end{aligned} \tag{20}$$

By simplifying (20), we obtain:

$$\begin{aligned} &(1-F(b_{k,i}-w_kR_{k,i}))^{(N-2)}\times[(1-N)F'(b_{k,i}-w_kR_{k,i}) \\ &\left(b_{k,i}+\alpha\frac{q_{i-k}}{q_{i-k}+q_k}\right)+(1-F(b_{k,i}-w_kR_{k,i}))\Big]=0 \end{aligned} \tag{21}$$

This means that we will consider two cases:

1. $F(b_{k,i}-w_kR_{k,i})=1$ or,
2. $(N-1)F'(b_{k,i}-w_kR_{k,i})\left(b_{k,i}+\alpha\frac{q_{i-k}}{q_{i-k}+q_k}\right)+F(b_{k,i}-w_kR_{k,i})=1$

In the following, we solve the second case, which is more general than the first one. In other words, the first case is included in the second case. Thus we solve a differential equation of first order, and the result is given by:

$$F(b_{k,i}-w_kR_{k,i})=1+C_0\left(b_{k,i}+\alpha\frac{q_{i-k}}{q_{i-k}+q_k}\right)^{-1/(N-1)} \tag{22}$$

where $C_0 \in R$. If $C_0 = 0$, then we find the result of case 1.

For sake of clarity, we give in the following a simple example, in which we suppose $Y_{k,i}$ has an exponential density function of parameter λ. We write $Y_{k,i}\sim Exp(\lambda)$. Then we have:

$$F(b_{k,i}-w_kR_{k,i})=1-e^{-\lambda(b_{k,i}-w_kR_{k,i})} \tag{23}$$

To find the NEP of this example, we use the result (20) as follows:

$$\begin{aligned} &(1-N)\frac{\partial\left(1-e^{-\lambda(b_{k,i}-w_kR_{k,i})}\right)}{\partial b_{k,i}}\left(b_{k,i}+\alpha\frac{q_{i-k}}{q_{i-k}+q_k}\right)+ \\ &+\left(1-\left(1-e^{-\lambda(b_{k,i}-w_kR_{k,i})}\right)\right)=0 \end{aligned} \tag{24}$$

Finally, to find the NEP of this example, we simply solve the following equation, (using mathematical tools (Matlab [25] for example)):

$$(1-N)\left(\lambda e^{-\lambda(b_{k,i}-w_k R_{k,i})}\right)\left(b_{k,i}+\alpha\frac{q_{i-k}}{q_{i-k}+q_k}\right)+ \\ +\left(e^{-\lambda(b_{k,i}-w_k R_{k,i})}\right)=0 \tag{25}$$

To find numerical values of the Nash Equilibrium Point of the considered game and for any cumulative distribution function, one can iteratively solve the equation (12) using mathematical tools (see [25]).

5. Selection of the CHs and optimal cluster size

We now focus on the problem of selecting the CHs and the optimal cluster size. To this, we assume that the CHs are the nodes who subscribed to act the role of (IoT_ProSe) executing tasks. They will hence be transferred to Announcing mode, while all the other nodes will remain under Monitoring mode to join the announcer for a specific IoT service. In the following, we discuss the selection of the CH and the importance of the cluster size, for which we propose an optimization algorithm.

5.1. D2D Cluster Head Election

Upon the validation of an IoT-ProSe device authorized for a specific ProSe_App_ID, the ProSe-MME executes a CH selection algorithm. The main purpose of the CH algorithm is to select the minimum number of ProSe-CHs in order to optimize and maximize the throughput usage in the Radio Access Network (RAN). After the election of ProSe-CHs by the CIoT core network and the allocation of ProSe discovery radio resources, ProSe Monitoring devices that enter the coverage area of an eNB will receive information of all the available ProSe-CHs in their communication range. The ProSe Gateway selection phase consists of the selection of the best ProSe-CH nodes, by the monitoring nodes. The CH acts as a gateway for uplink data transmission from the devices to the infrastructure network (i.e. to the IoT AS). In other words, a ProSe gateway selection consists of learning the best path to the ProSe-CH that depends on the architecture. We have two use cases architectures as follows:

Single-hop. Single-hop means all nodes have direct connections with the CH. In this topology, all nodes in the cluster have symmetric connections to the CIoT platform through one hop to the MME-ProSe_IoT through their CH. The service creation can be detailed as follows: (1) the CH node will start the group discovery either using 3GPP direct discovery or centralized discovery based on the operators vision for the proposed ProSe, (2) the CH sets the ProSe server with eNB, (3) if the discovery process is initiated by the CH, all the other nodes remain in monitor mode in order to join this Group (example: node N_i is listening to the discovery broadcast messages by CH so as to join this ProSe) as shown in Fig. 1.

Multi-hop. Here, some nodes have to relay their direct connections to other nodes before reaching a CH. In this case, there are no sufficient resources to relay the communications directly to the CHs but through intermediate nodes, as shown in Fig. 4. So, the connections for all nodes are asymmetric and the intermediate nodes can gain more score as their participation is weighted by other nodes. With the inverse direction of data flow from AS to Nodes, the CH motivated by the AS can assign tasks to do by the joined nodes based on some incentives and agreements between all participants, taking into account: (1) uniform lifetime, (2) the assigned task frequency, (3) energy optimization, (4) sub-groups formation in case of heterogeneous objects (like objects with [ZigBee, Bluetooth: BLE or WLAN-Direct] with 4G communications).

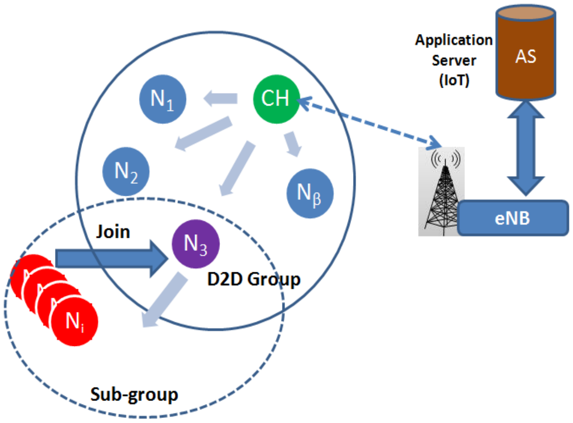


Figure 4: Multi-hop cluster for some nodes N_i to join a D2D CH through a relay node to the CH and then to eNB and IoT-AS.

5.2. D2D Cluster Size

The cluster size is considered as an important issue in the optimization. The larger the number of participants in the cluster, the more radio interference arises. So, optimizing the cluster size can improve the overall performance of the group. The Signal-to-Interference-plus Noise Ratio (SINR) information for all objects can be used as the main criteria for constructing small or big cluster sizes. Since the D2D clusters are isolated, performance is supposed to be good and protected from interference. The same could be done with eNB communication. The D2D clustering targets and issues can be summarized as follows.

1. Impacts from D2D clustering: (a) Reduce eNB signaling for D2D communications, (b) Reduce the D2D cluster size that will reduce the internal interference inside the cluster and the overall global interference across the cell, (c) For inter-cluster communications, it will be through eNB (i.e. best choice of each group CH that will reduce the transmitted power and D2D-Assisted interference across the cell).

2. Transmission power issues in D2D clustering: (a) Use Assisted-D2D (D2D-A) when the device power to eNB communication is lower than the required power for Direct-D2D (D2D-D) communication (i.e. for two D2D devices communicating through eNB when Power-cell is lower than Power-D2D) (b) Use D2D-D over D2D-A when Power-cell (P_C) is lower than Power-D2D (P_D).

3. Distance issues in D2D clustering: (a) When the distance (d_i) between two devices is large, then use D2D-A in order to reduce the interference, distance to eNB is (d_A). (b) When the distance is small (nodes are very close), then use D2D-D to reduce the interference, distance for D2D-D (d_D).

The next subsection proposes an optimization for the cluster size that takes into account the aforementioned impacts and issues.

5.3. D2D Cluster Size Self-Optimization

In D2D-LTE communication, eNB links for many D2D services can lead to much interference across the cell. So, in order to reduce this interference, D2D-clusters can improve the performance of D2D-ProSe services. This means that, without clustering, this eNB-D2D links will be interfered by the simultaneous D2D devices unless reducing the number of devices communicating to same receivers, which are the CHs. In this optimization, the SINR is the main criteria used for clusters formation. The power efficiency can be measured by the received SINR that is required to achieve reliable communication with a specified bandwidth efficiency in the presence of channel impairments [26]. The eNB is the responsible for this through the ProSe-IoT entity attached to the CIOT architecture. Fig. 5 depicts the system model for the SINR optimization over eNB cell self-organized with two type of SINR links (Devices to CH and CH to eNB) as follows.

- SINR general equation for connection i:

$$SINR_i = \frac{P_i}{I_{ri} + N} \tag{26}$$

  where (P_i) represents the power measured for the received signals relative to distance (d_i) (i.e. the average Received Energy for the signal), N indicates the background noise coefficient (i.e. the Noise Power), and I_{ri} represents the received interference on same connection.

- The SINR for one cluster consisting of β nodes, is the sum of the signals emitted in the cluster area as written as follows

$$\sum_{i=1}^{\beta} \frac{\frac{P_i}{d_{ii}^{\alpha}}}{N + \sum_{j \neq i} \frac{P_i}{d_{ij}^{\alpha}}} \tag{27}$$

  where α is the connection path loss exponent that describes the influence of the transmission medium.

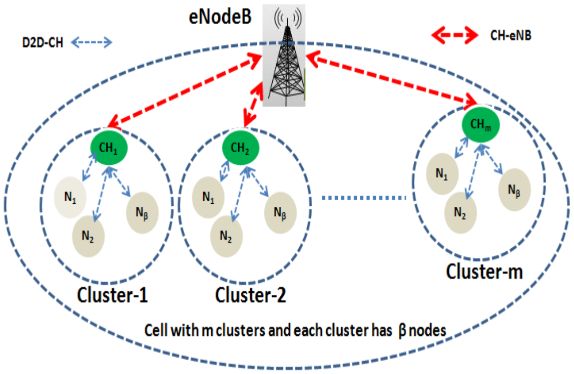


Figure 5: System model for D2D cluster size optimization based on two links interference as D2D-CH: β links for each cluster and CH-eNB: m links all over the cell.

- For m clusters (i.e. the total number of nodes covered by an eNB), the total SINR will be:

$$SINR_{tot} = \sum_{c=1}^{m} \sum_{i=1}^{\beta} \frac{\frac{P_i}{d_{ii}^{\alpha}}}{N + \sum_{j \neq i} \frac{P_i}{d_{ij}^{\alpha}}} \tag{28}$$

- So, after clustering, the objective is to maximize the total SINR ($SINR_{tot}$) by clusters formation (i.e. minimize the interference) which can be represented by the following optimization

$$\max SINR_{tot} = \sum_{c=1}^{m} \sum_{i=1}^{\beta} \frac{\frac{P_i}{d_{ii}^{\alpha}}}{N + S_i(m) - \frac{P_i}{d_{ii}^{\alpha}}} \tag{29}$$

Where

$$S_i(m) = \sum_{j=1}^{\beta} \frac{P_i}{d_{ij}^{\alpha}} = P_i \sum_{j=1}^{\frac{U}{m}} \frac{1}{d_{ij}^{\alpha}} \tag{30}$$

Where: U represents the number of all active $D2D$ devices (nodes) inside the cell, m represents the number of clusters while β represents the cluster size. So, the objective function can be more simplified by the following sequences

$$\max SINR_{tot} = Z = \sum_{c=1}^{m} \sum_{i=1}^{\frac{U}{m}} \frac{\frac{P_i}{d_{ii}^{\alpha}}}{N + \sum_{j=1, j \neq i} \frac{P_i}{d_{ij}^{\alpha}}} \tag{31}$$

Then:

$$= \sum_{c=1}^{m} \sum_{i=1}^{\frac{U}{m}} \frac{\frac{P_i}{d_{ii}^{\alpha}}}{N + S_i(m) - \frac{P_i}{d_{ii}^{\alpha}}} \tag{32}$$

$$= \sum_{i=1}^{\frac{U}{m}} \frac{mP_i}{(N + S_i(m))d_{ii}^{\alpha} - P_i} \tag{33}$$

$$= \frac{mP_1}{(N + S_1(m))d_{11}^{\alpha} - P_1} + \frac{mP_2}{(N + S_2(m))d_{22}^{\alpha} - P_2} +$$

$$+ \ldots + \frac{mP_{\frac{U}{m}}}{(N + S_{\frac{U}{m}}(m))d_{\frac{U}{m}\frac{U}{m}}^{\alpha} - P_{\frac{U}{m}}} \tag{34}$$

Now for optimizing Z:

$$\frac{\partial Z}{\partial m} = 0 \tag{35}$$

That can lead to:

$$\frac{P_1 \left[(N + S_1(m))\, d_{11}^{\alpha} - P_1\right] - mP_1[d_{11}^{\alpha} S_1'(m)]}{[(N + S_1(m))d_{11}^{\alpha} - P_1]^2} + \ldots +$$

$$\frac{P_{\frac{U}{m}} \left[\left(N + S_{\frac{U}{m}}(m)\right) d_{\frac{U}{m}}^{\alpha} - P_{\frac{U}{m}}\right] - mP_{\frac{U}{m}}[d_{\frac{U}{m}}^{\alpha} S_{\frac{U}{m}}'(m)]}{[(N + S_{\frac{U}{m}}(m))d_{\frac{U}{m}}^{\alpha} - P_{\frac{U}{m}}]^2} = 0 \tag{36}$$

$$\sum_{i=1}^{\frac{U}{m}} d_{ii}^{\alpha}[S_i(m) - mS_i'(m) + N] - P_i = 0 \tag{37}$$

Note that ($\forall i$, $\frac{1}{d_{ij}} = D_j$) then:

$$S_i(m) = P_i \sum_{j=1}^{\frac{U}{m}} D_j^{\alpha} = P_i \sum_{j=1}^{\frac{U}{m}} D^{\alpha} = P_i \frac{U}{m} D^{\alpha} \tag{38}$$

And:

$$S_i'(m) = \frac{-UP_iD^{\alpha}}{m^2} \tag{39}$$

By substituting $S_i(m)$ and $S_i'(m)$ in (37):

$$\sum_{i=1}^{\frac{U}{m}} d_{ii}^{\alpha} \left[\frac{UP_iD^{\alpha}}{m} - \frac{UP_iD^{\alpha}}{m} + N\right] - P_i = 0 \tag{40}$$

Solving (40) leads to the following result:

$$m = \frac{UN}{D^{\alpha} \sum_{i=1}^{\frac{U}{m}} P_i} \tag{41}$$

To find the optimal number of clusters m^*, we proceed as follows. We note that

$$T_{\frac{U}{m}} = \sum_{i=1}^{\frac{U}{m}} P_i \tag{42}$$

Then, it is clear that

$$T_{\frac{U}{m}+1} = \sum_{i=1}^{\frac{U}{m}+1} P_i \tag{43}$$

According to (42) and (43), we have:

$$\frac{T_{\frac{U}{m}+1}}{T_{\frac{U}{m}}} = \frac{\sum_{i=1}^{\frac{U}{m}+1} P_i}{\sum_{i=1}^{\frac{U}{m}} P_i} \tag{44}$$

Now, suppose that we ordered the powers P as follows $P_i \geq P_{i+1}$. Thus, we have:

$$\frac{T_{\frac{U}{m}+1}}{T_{\frac{U}{m}}} = \frac{\sum_{i=1}^{\frac{U}{m}+1} P_i}{\sum_{i=1}^{\frac{U}{m}} P_i} = \frac{\sum_{i=1}^{\frac{U}{m}} P_i + P_{\frac{U}{m}+1}}{\sum_{i=1}^{\frac{U}{m}} P_i} = 1 + \frac{P_{\frac{U}{m}+1}}{\sum_{i=1}^{\frac{U}{m}} P_i} \tag{45}$$

By considering the defined order, we let

$$\frac{P_{\frac{U}{m}+1}}{\sum_{i=1}^{\frac{U}{m}} P_i} = \epsilon \tag{46}$$

where $1 > \epsilon > 0$. We deduce that:

$$\frac{T_{\frac{U}{m}+1}}{T_{\frac{U}{m}}} = 1 + \epsilon \tag{47}$$

We generalize the formula (47), to finally obtain:

$$T_{\frac{U}{m}} = (1+\epsilon)^{\frac{U}{m}-1} T_1 = (1+\epsilon)^{\frac{U}{m}-1} P \tag{48}$$

where P can be the average power value, for instance.

Now, starting from (48), we replace this result in formula (41), and we find:

$$m = \frac{UN}{D^{\alpha}(1+\epsilon)^{\frac{U}{m}-1} P} \tag{49}$$

Using the following Taylor development:

$$(1+\epsilon)^{\frac{U}{m}-1} = 1 + \left(\frac{U}{m} - 1\right)\epsilon + o(\epsilon) \tag{50}$$

We replace the result of (50) in (49), we obtain

$$m = \frac{UN}{D^{\alpha} P \left[1 + \left(\frac{U}{m} - 1\right)\epsilon\right]} \tag{51}$$

By solving equation (51), we find the optimal cluster size (m^*) given by:

$$m^* = \frac{\frac{UN(1+\epsilon)}{D^{\alpha}P} - U\epsilon}{1-\epsilon} \tag{52}$$

6. Simulations

6.1. Performance of the Task Allocation Algorithm

The performance has been analyzed through simulations run within the Matlab environment [25]. The parameter values used in the simulations are listed in Table 1. The 640 scenarios that have been considered consist of 10 nodes placed randomly in an area of 10x10 m^2, supposing that they change their position randomly at each assignment of a new task. All these nodes represent the members of a single cluster and are those that start the competition to win the assignment of the task. For the analysis of the performance, it is not necessary to know exactly which one of these nodes acts as the CH. Each scenario differs with respect to the others in terms of initial nodes position and initial status of the battery energy of the nodes. It is important to note that no additional traffic is handled by the devices; accordingly, the performance results shown in this analysis are related only to the tasks assigned with the proposed algorithm. Node initial residual energy E_i was set to 2 J, while initial task frequency q_{i-k} was set to a value in the range $[0, 20]$ Hz. The γ weight parameter was set to 0.5. With reference to the trustworthiness (see (10)), node computational complexity C_i was set to a random value in the range $[0, 1]$. We observed the algorithm's behavior for 100 tasks, which were allocated one at a time. Their characteristic parameters q_k and e_k were initialized to random values in the range $[0, 10]$ Hz and mJ and kept constant during the whole simulation. We considered a normal distribution function, which μ and σ parameters were initialized to those of a standard normal distribution ($\mu = 0$ and $\sigma = 1$). At each step of the algorithm, they are set according to the mean and standard deviation of the winning bid values. The trustworthiness value is computed at each step of the algorithm for each node, as well. We define Δ as the lowest possible difference between the sum $((M_i)|_{k+1} + \sum_{k+1} B_{k+1,i})$ computed after the $(k+1)$-th task has been assigned and $((M_i)|_k + \sum_k B_{k,i})$ computed after the k-th task has been assigned

$$((M_i)|_{k+1} + \textstyle\sum_{k+1} B_{k+1,i}) - ((M_i)|_k + \sum_k B_{k,i}) \geq \Delta \tag{53}$$

The sum $((M_i)|k + B_{k,i})$ is set randomly according to Δ, at each step of the algorithm.

As an explanatory example, Fig. 6 shows how the values of $Y_{k,i}$ index, bid, reliability and lifetime change at each step of the algorithm, i.e. each time a new task needs to be allocated, for $w = 1$, $\alpha = -1$ and $\Delta = 100\%$. Note that the case $\Delta = 100\%$ corresponds to the case where, at each step, each node surely provides a response with an accuracy corresponding to the expected one. A different color is assigned to each of the 10 nodes. In Figs. 6(a), 6(b) and 6(c), the thicker line represents the corresponding value for the winning node. The thicker line in Fig. 6(d) represents the minimum lifetime, i.e. the system lifetime for the considered scenario. By definition, the node that wins the competition is the one that gives the bid corresponding to the lowest $Y_{k,i}$ index value. Since w and α have comparable values, it is possible to note a trade-off between reliability of the winning node and lifetime. In fact, Fig. 6(c) shows that it is

Table 1: Simulation parameters

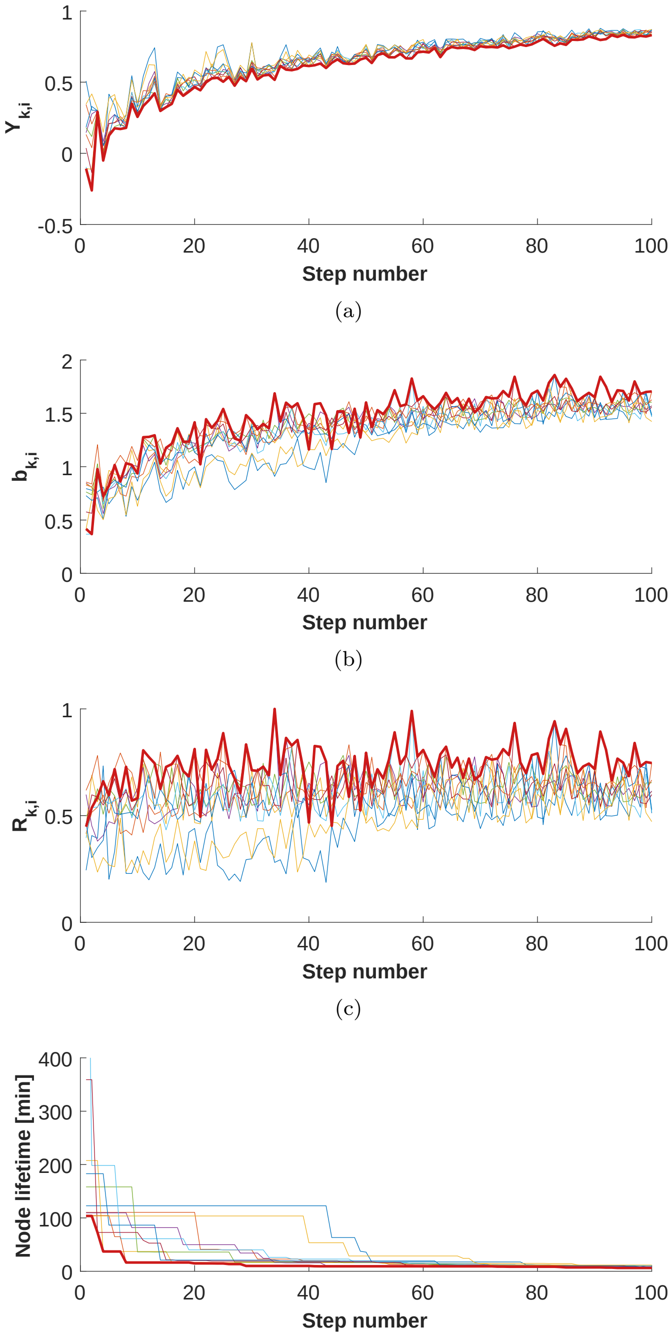
Parameter	**Value**	**Parameter**	**Value**
# scenarios	640	Init. μ	0
# nodes	10	Init. σ	1
Area	10x10 m^2	C_i	$[0, 1]$
# tasks	100	Δ	$\{100\%, 75\%, 50\%, 25\%\}$
E_i	2 J	γ	0.5
q_{i-k}	$[0, 20]$ Hz	α	$\{0, -1, -5, -10\}$
q_k	$[0, 10]$ Hz	w	$\{0, 1, 5, 10\}$
e_k	$[0, 10]$ mJ		

not always the most reliable node that wins the competition, most likely because its lifetime is not high enough to bear a new task. This is confirmed by Fig. 6(d), where towards the end of the simulation nodes tend to converge to the same lifetime. Fluctuations of the $Y_{k,i}$ index and the bid from one step to the other are mainly due to the fact that many of the parameters that intervene in the algorithm are set randomly, and thus they can vary considerably for different tasks. However, it can be observed that values follow the same trends, as they are influenced by the value q_k of the required frequency for task k, which is equal for all the nodes. Also, note that these fluctuations tend to decrease as the number of considered tasks increases; this is due to the fact that μ, σ and the trustworthiness value for each node (which affects $R_{k,i}$) gradually converge to a value that reflects the overall behavior of the system.

Figs. 7, 8, 9 and 10 show average simulation results for different scenarios, for different values of the weight parameters α and w, and of the Δ parameter related to node reliability: $\alpha = \{0, -1, -5, -10\}$, $w = \{0, 1, 5, 10\}$ and $\Delta = \{100\%, 75\%, 50\%, 25\%\}$. Recall from Section 4 that α is a negative parameter that weighs the energy cost, i.e. the higher the absolute value of α, the higher the weight that energy cost has on the bid outcome; w is a positive parameter that weights the reliability, i.e. the higher the value of w, the higher the weight that reliability has on the bid outcome.

The average bid values of the nodes that won the competition, evaluated at each step, are shown in Fig. 7 for different values of w and $|\alpha|$, i.e. the absolute value of α. Note that, for each value of w, curves remain almost constant as long as $w \geq |\alpha|$. As soon as $w < |\alpha|$, curves get steeper and bid values increase rapidly. We can conclude that the average bid value is mostly affected by the highest between w and $|\alpha|$ values. Average bid values are not compared for different values of Δ because results were very similar, as it is possible to deduce from confidence intervals. This implies that there is no correlation between the reliability expected from a node, which depends on Δ, and the bid value of the winning node.

Fig. 8 shows how the average normalized reliability changes for different values of Δ, when the ratio $w/|\alpha|$ changes. By correctly setting w and $|\alpha|$ values, reliability can improve up to 21% with respect to the case where only lifetime



(a)

(b)

(c)

(d)

Figure 6: Example of the evolution of $Z_{k,i}$ index, $b_{k,i}$, $R_{k,i}$ and node lifetime at each step of the algorithm, for $w = 1$, $\alpha = -1$ and $\Delta = 100\%$

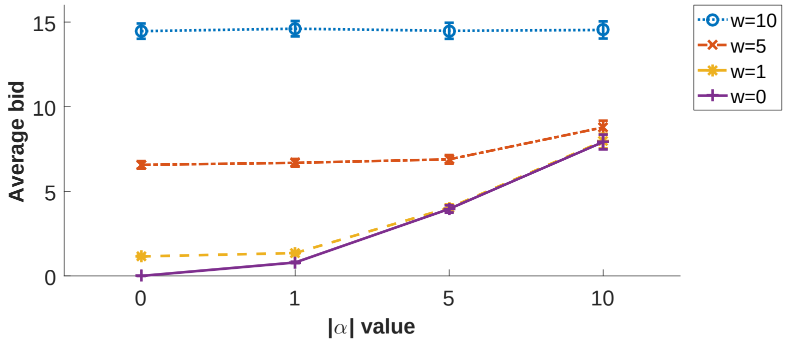


Figure 7: Average bid evaluated at each step of the algorithm, for different values of w and $|\alpha|$

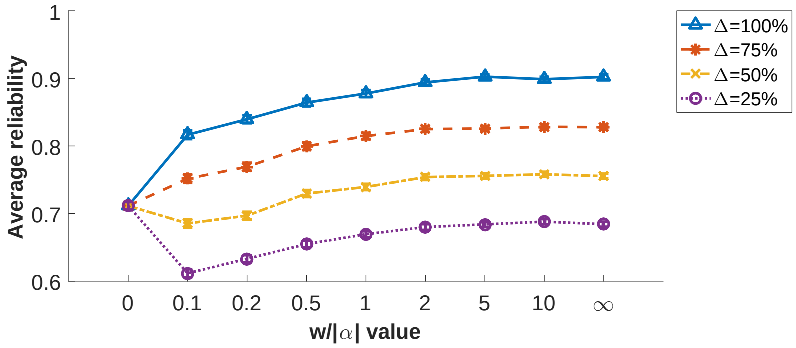


Figure 8: Average normalized reliability for different values of the ratio $w/|\alpha|$, compared for different values of Δ

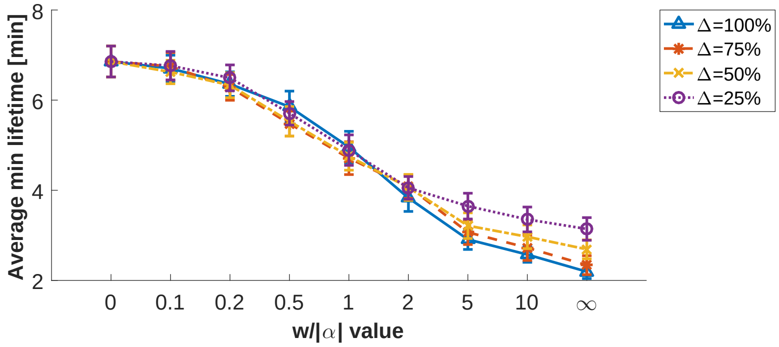


Figure 9: Average minimum lifetime of the system, evaluated at the end of the simulations, for different values of the ratio $w/|\alpha|$, compared for different values of Δ

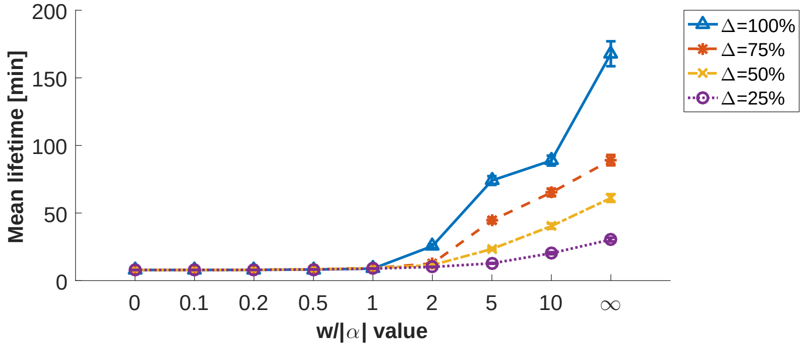


Figure 10: Average node lifetime evaluated at the end of the simulations, for different values of the ratio $w/|\alpha|$, compared for different values of Δ

is considered ($w = 0$). Since Δ is directly related to node's trustworthiness, i.e. to the probability that a node that won the competition actually gave a response and this response accuracy corresponds to the expected one, as a consequence it affects nodes' reliability. For this reason, the average reliability increases when Δ is higher, especially for higher values of w, i.e. when reliability weighs more on the algorithm. It is interesting to note that, for low values of Δ ($\Delta = \{50\%, 25\%\}$), the average reliability is sometimes higher when it is not considered in the competition. This is consistent with the fact that if, in a system, nodes' trustworthiness is low, even the node with the highest trustworthiness is not sufficiently reliable, and thus it may result in a lower reliability than choosing it randomly. In this case, if Δ is not too low (e.g. $\Delta = 50\%$), in order to have good reliability it can be either chosen to set a higher w, at the expenses of higher bids and lower lifetime values, or set it to 0. If Δ is lower than 50%, reliability is not ensured even for high values of w, so a better solution would be setting it to 0.

It can be observed that, according with the behavior that is expected by the algorithm, the minimum lifetime observed at the end of the simulations (see Figure 9) increases as the ratio $w/|\alpha|$ decreases. System's lifetime increases up to 68% with respect to the case where only reliability is considered in the utility optimization process ($\alpha = 0$). In fact, for higher values of w with respect to $|\alpha|$, the energy consumption component weighs less on the utility function, and therefore it is considered less important than the reliability component. This is particularly accentuated for higher values of Δ, for which the competition will reward more often the node with the highest trustworthiness, thus leading to a more rapid decrease of its lifetime with respect to other nodes. This is confirmed by Figure 10, which shows the average lifetime observed at the end of the simulations for all the nodes that were involved in the competitions. Note that the approach proposed in this paper aims to increase the system's lifetime by distributing the workload among all the nodes that participate in the competitions. Thus, the average lifetime is lower when the lifetime component weighs more (i.e. lower values of $w/|\alpha|$), because the workload is appropriately distributed among all the nodes. This is confirmed by the higher values of minimum lifetime reported. On the other hand, when the weight of the lifetime component decreases with respect to that of the reliability component, the most reliable nodes will be those that will be chosen more often to undertake most of the workload, and this results in lower minimum lifetime values and higher average lifetime values determined by the nodes that seldom win the competition. Reminding that trustworthiness values depend on the number of transactions already occurred, this effect is especially emphasized for higher values of Δ: since nodes are, on average, more reliable, the nodes that won the first competitions quickly increase their trustworthiness values with respect to the other nodes, and thus they are more likely to win the following competitions as well.

As a final performance evaluation, the proposed approach has been compared with other similar algorithms existing in the literature. Figure 11 shows a comparison of the total cost paid by the CH for an increasing number of 35 to 65 tasks. The total cost is calculated as the overall sum of the bids for different values of w and α, and is compared with the results obtained in [27] for the Greedy Algorithm (GA) and the Local Ratio Based Algorithm (LRBA), which is an auction-based algorithm for task allocation in crowdsensing. Compared to the other approaches, the proposed algorithm enables to choose whether to reward nodes more, or to spend the lowest amount possible, depending on the objective of the auction. As confirmed by the graph, by appropriately setting the w and α values, the total bid cost is consistently higher or lower than those obtained using GA or LRBA.

Finally, node lifetime has been compared in Figure 12 with other static and lifetime-extending mechanisms for IoT systems, namely:

- C: a static task allocation where all the tasks are assigned to the CH;
- CO: a centralized task allocation for energy consumption maximization proposed in [8];

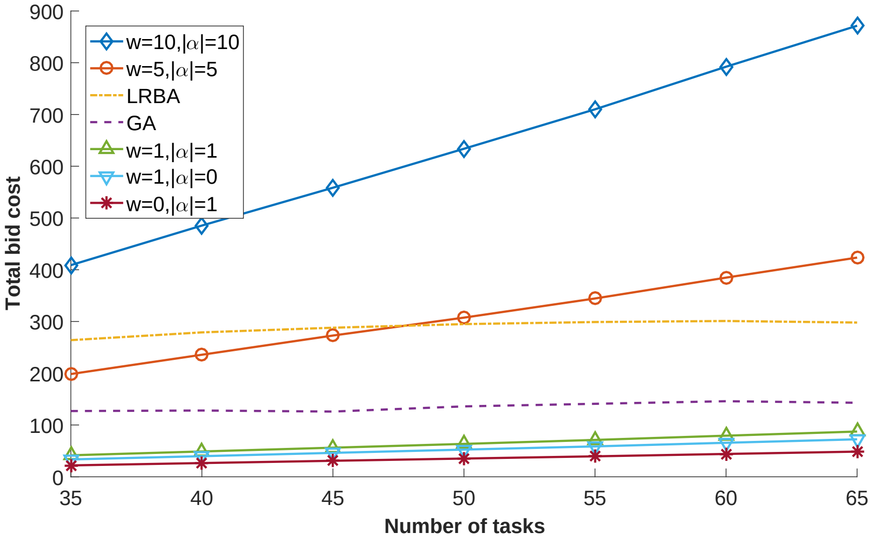


Figure 11: Total bid cost paid by the CH for different numbers of tasks, using the GA, the LRBA, and the proposed algorithm with different values of w and α

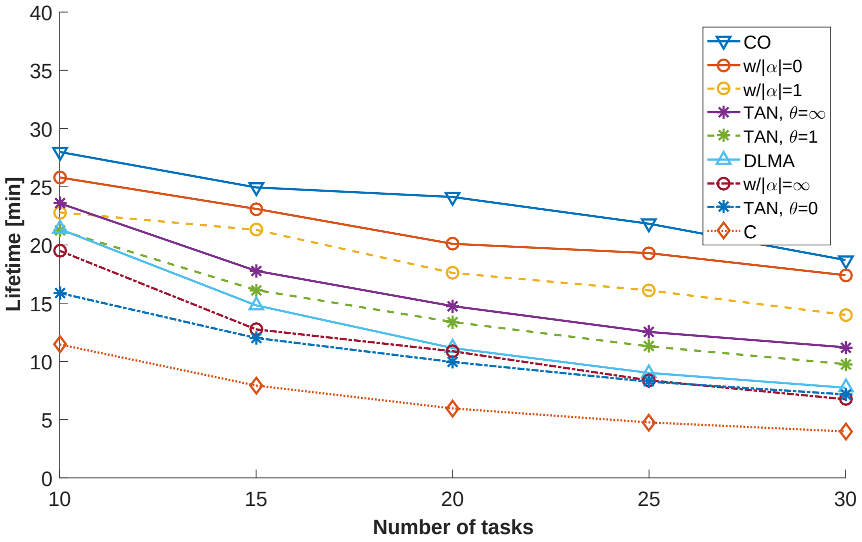


Figure 12: Lifetime results for a static (C), a centralized (CO), a distributed lifetime maximizing $(DLMA)$ and a distributed energy consumption and task execution time aware (TAN) task allocation algorithms, for an increasing number of tasks

- *DLMA*: a distributed task allocation for lifetime maximization proposed in [28];
- *TAN*: a distributed task allocation based on game theory, which finds a tradeoff between node energy consumption and task execution time [29]. This tradeoff is adjusted thanks to some weight parameters, which in Figure 12 are summarized by parameter θ. This parameter is proportional to the weight given to the lifetime-related component of the algorithm, and in inverse proportion with the one related to the task execution time, i.e. $1/\theta \propto w/|\alpha|$.

The graph shows that, when the reliability weight w is much lower than the lifetime weight α, the proposed algorithm outperforms all the other approaches, with the exception of the centralized optimal one. This is because in the centralized allocation case, the algorithm is aware of all the tasks to be assigned at the same time, and not in order of arrival as it is in the other cases. Results get lower only when the reliability component prevails on the lifetime component, i.e. when w is higher than α. Note that, among the compared approaches, TAN is the most similar to the proposed one. In fact, it is based on game theory, and it is a multiobjective algorithm with one component related to lifetime and one related to the task execution time. As explained above, $1/\theta \propto w/|\alpha|$, therefore results obtained for $\theta = 0$ are comparable to those obtained for $w/|\alpha| = \infty$, and results obtained for $\theta = \infty$ are comparable to those obtained for $w/|\alpha| = 0$.

6.2. Analysis of the Optimal Cluster Size

In this section, we analyze the optimal cluster size according to the presented optimization algorithm. The performance has been analyzed through numerical simulations run within the Matlab environment [25]. To do this we consider a single cell interference scenario: a single eNB with U active users, m D2D task clusters, and α equal to either 2 or 3. The D2D distance has been considered in the range between 10 and 50 meters and for average sensitivity for powerful mobile devices the power P_i has been taken in the range between $+20$ and -40 dBm. The numerical values used in our performance evaluation considered the proximity aspect in radio part as proposed in 3GPP specifications [30] and [31]. Moreover, for energy efficiency in LTE D2D aspect, we followed the work proposed in [32] for machine to machine communication. Finally, for the power value levels, we assumed the models presented in [33] for power thresholds in dBm. Table II summarizes the input parameters used in cluster size calculations and links them to their proposed references.

The proposed cluster size optimization algorithm in terms of optimizing m will also optimize the CHs selection as each cluster has one CH. Moreover, this algorithm can be integrated into the ProSe_MME EPC plane as described in Fig. 2 for CIOT architecture.

The results are shown in Figs. 13-15, which demonstrate that whatever the number of active users in the cell, there is no need for clustering in the case of α equal to 2, which means low noise levels. Especially for large scale networks, as

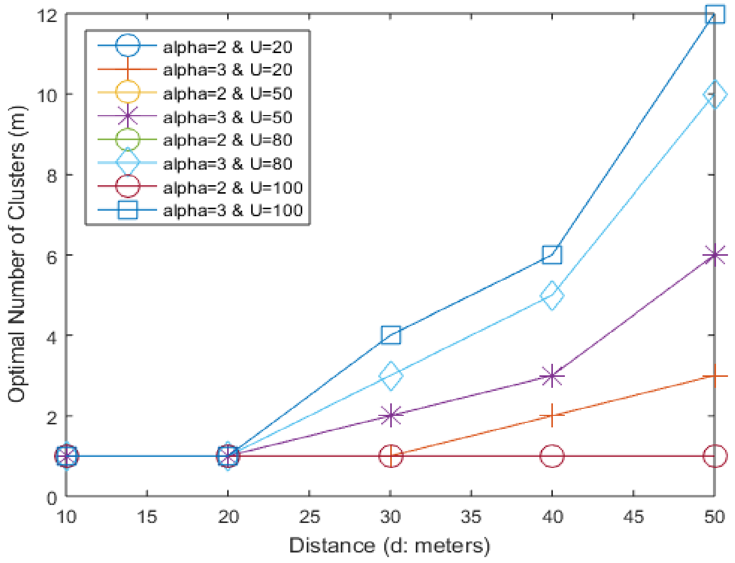


Figure 13: Small scale of D2D cluster size optimization based on two links interference as α equal 2 or 3 for different number of U inside the cell.

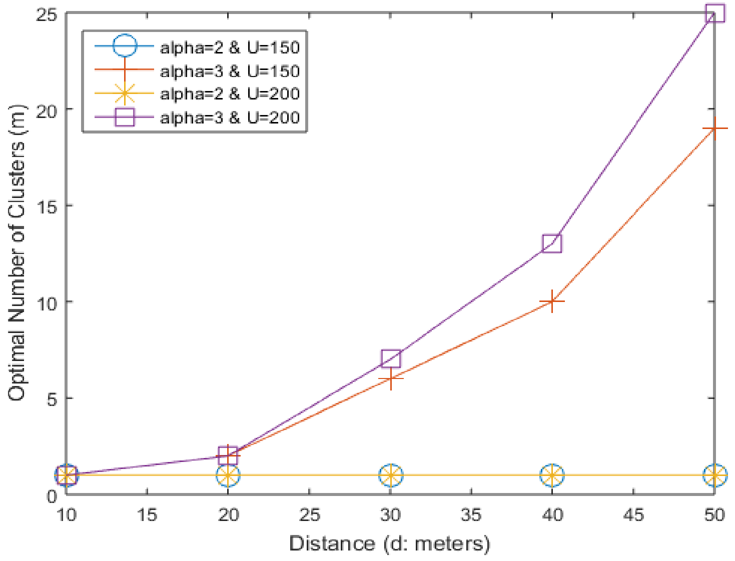


Figure 14: Medium scale of D2D cluster size optimization based on two links interference as α equal 2 or 3 for different number of U inside the cell.

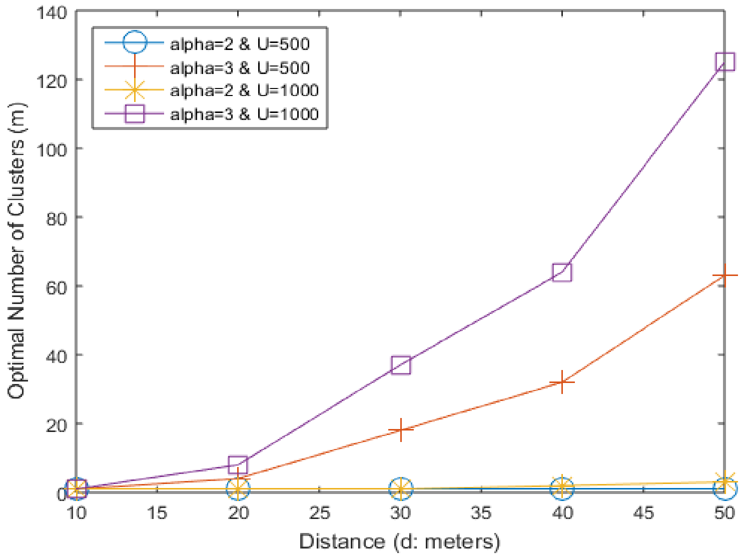


Figure 15: Large scale of D2D cluster size optimization based on two links interference as α equal 2 or 3 for different number of U inside the cell.

considered in Fig. 15, there is a chance for clustering for U equal to 1000 and the same value of α. This is expected as the number of users is large. But, in the case of α equal to 3, which means more interference added by increasing the path loss that will impose more power signaling, clustering is mandatory. As shown in the three figures, increasing the number of users brings to an increase in the number of the optimal cluster size m.

6.3. Comparative Analysis of the Optimal Cluster Size

This section provides a brief comparison study to clustering techniques well-known in the state of the art, on which our optimization for cluster size calcu-

Table 2: Cluster Size Simulation parameters

Parameters	**Values**	**References**
Number of active users (U)	20 to 1000	Follow [34]
D2D Separation distance (d)	From 10 to 50 m	Proposed
Number of eNB	One	3GPP [30] [31]
Cell radius	1 km	Proposed
Path-loss exponent range (α)	2 or 3	Follow [21] [32]
Power threshold range (P)	+20 to -40 dBm	Follow [32]
Noise (N)	0.01 MW	3GPP [31]
Epsilon range	From 0 to 1	Proposed

lation for one cell is based. The comparative study is mainly focused on the cluster size, the number of generated clusters and the interference aspects as these are important for our optimization approach. The following statements provide a comparison of our solution with respect to other five different clustering architectures when considering the mentioned features.

1. One-hop cluster: the number of generated clusters equals the number of CHs with less interference level through inter-clusters communications. Also, the network state points to full coverage by eNBs. The relevant work proposed in [21] analyzes this architecture through numerical results for D2D active users between 10 to 1000, while our approach achieves it through large scale network sizes and higher path loss as shown in Fig. 13
2. Centralized cluster: it selects one cluster controlled by an eNB with more interference due to large number of active D2D devices. Here, the same network condition for full coverage by eNB is considered. It is clear from our clustering approach results given in Figs. 13 and 14 for small scale size and lower path loss while this achievement occurred for [21] with a number of active D2D users lower than 10.
3. Other three architectures (multi-hop, distributed or hybrid): all subject to partial coverage of network devices by eNB. The interference level can be increased or decreased based on the number of generated clusters and the percentage of coverage devices. Neither our approach nor the work in [21] treated those use cases. We restricted our work to assisted-eNB D2D communication following 3GPP recommendations.

As a conclusion, the work in [21] built their simulation and numerical results based on caching content popularity to affect D2D cluster size optimization. So, if we neglect this parameter (i.e. put the content popularity equals zero as our approach is a mathematical generic approximation for optimal cluster size), then the optimal cluster size can be compared. Actually, we adopted the same number of active D2D users (order of tens till 1000) and obtained closer results for optimal cluster size. The advantage of our algorithm over the literature one is that we vary the path-loss and the average distance between users more, which mainly affects the interference levels and the optimal cluster size as it is shown in Figs. 13, 14 and 15.

7. Conclusions

In this paper, we proposed an IoT-D2D framework for task allocation. It manages the interactions among a group of D2D objects controlled by a CH. With respect to traditional D2D architectures, our approach aims to extend the communication range and improve the Quality of Service. A bidding process is presented. An IoT AS forwards its requests to the CH. It proposes an initial reward amount for the service. Cluster nodes are informed by the D2D process of the task request and start bidding. We also address the problem of the election of the CH and find the optimal size of the clusters.

The proposed game is evaluated and compared for different scenarios, with different trustworthiness levels of the nodes. Through the simulation results, it is possible to conclude that, in order to keep the average bid values low, w and $|\alpha|$ values, which respectively weigh reliability and energy cost on the utility function, should be set to low values. Furthermore, their value should be set taking into account that reliability can improve up to 21% when decreasing $|\alpha|$ up to 0, while the system lifetime can improve up to 68% when decreasing w up to 0. Nevertheless, it should be kept in mind that, for systems which nodes are, on average, not trustworthy, even increasing the ratio $w/|\alpha|$ may not lead to reliability results that are better than the case with $w = 0$.

Acknowledgements

This work was supported in part by Italian Ministry of University and Research (MIUR), within the Smart Cities framework (Project: CagliariPort2020, ID: SCN_00281).

References

[1] 3GPP TS 23.303, Architecture enhancements to support proximity services (prose) (2014).

[2] 3GPP TS 23.303 V13.4.0, Proximity-based services (ProSe); stage 2, release 13 (June 2016).

[3] B. Zhou, H. Hu, S.-Q. Huang, H.-H. Chen, Intracluster device-to-device relay algorithm with optimal resource utilization, IEEE Transactions on Vehicular Technology 62 (5) (2013) 2315–2326.

[4] T. T. Gunes, S. T. Kwong U, H. Afifi, Hybrid model for LTE network-assisted D2D communications, in: International Conference on Ad-Hoc Networks and Wireless, Springer, 2014, pp. 100–113.

[5] T. Toukabri, A. M. Said, E. Abd-Elrahman, H. Afifi, Cellular vehicular networks (CVN): Prose-based ITS in advanced 4G networks, in: 2014 IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems, IEEE, 2014, pp. 527–528.

[6] T. Toukabri, A. M. Said, E. Abd-Elrahman, H. Afifi, Distributed D2D architecture for ITS services in advanced 4G networks, in: Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd, IEEE, 2015, pp. 1–7.

[7] K. L. Mills, A brief survey of self-organization in wireless sensor networks, Wireless Communications and Mobile Computing 7 (7) (2007) 823–834.

[8] V. Pilloni, L. Atzori, Deployment of distributed applications in Wireless Sensor Networks, Sensors 11 (8) (2011) 7395–7419.

[9] J. Zhu, J. Li, H. Gao, Tasks allocation for real-time applications in heterogeneous sensor networks for energy minimization, in: Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007. SNPD 2007. Eighth ACIS International Conference on, Vol. 2, IEEE, 2007, pp. 20–25.

[10] D. Guinard, V. Trifa, F. Mattern, E. Wilde, From the Internet of Things to the web of things: Resource-oriented architecture and best practices, in: Architecting the Internet of Things, Springer, 2011, pp. 97–129.

[11] B. Silverajan, J. Harju, Developing network software and communications protocols towards the Internet of Things, in: Proceedings of the Fourth International ICST Conference on COMmunication System softWAre and middlewaRE, ACM, 2009, p. 9.

[12] G. Colistra, V. Pilloni, L. Atzori, Task allocation in group of nodes in the IoT: A consensus approach, in: 2014 IEEE International Conference on Communications (ICC), IEEE, 2014, pp. 3848–3853.

[13] V. Pilloni, L. Atzori, M. Mallus, Dynamic involvement of real world objects in the iot: A consensus-based cooperation approach, Sensors 17 (3) (2017) 484.

[14] R. Olfati-Saber, J. A. Fax, R. M. Murray, Consensus and cooperation in networked multi-agent systems, Proceedings of the IEEE 95 (1) (2007) 215–233.

[15] S. Vasudevan, D. Towsley, D. Goeckel, Neighbor discovery in wireless networks and the coupon collectors problem, in: ACM Mobicom '09, Beijing, China, 2009.

[16] G. Fodor, E. Dahlman, G. Mildh, S. Parkvall, N. Reider, G. Miklos, Z. Turanyi, Design aspects of network assisted device-to-device communications, IEEE Communications Magazine 50 (3) (2012) 170–177.

[17] Q. Lu, Q. Miao, G. Fodor, N. Brahmi, Clustering schemes for d2d communications under partial/no network coverage, in: 2014 IEEE 79th Vehicular Technology Conference (VTC Spring), Seoul, 2014, pp. 1–5.

[18] D. Jia, H. Zu, S. Zou, P. Hu, Dynamic cluster head selection method for wireless sensor network, IEEE.

[19] L. Wang, G. Araniti, C. Cao, W. Wang, Y. Liu, Device-to-device users clustering based on physical and social characteristics, International Journal of Distributed Sensor Networks 11 (8).

[20] M. Afshang, H. S. Dhillon, P. H. J. Chong, Coverage and area spectral efficiency of clustered device-to-device networks, in: Global Communications Conference (GLOBECOM), 2015 IEEE, San Diego (CA), 2015, pp. 1–6.

[21] B. Chen, C. Yang, G. Wang, Cooperative device-to-device communications with caching, in: Vehicular Technology Conference (VTC Spring), 2016 IEEE 83rd, Nanjing, 2016, pp. 1–5.

[22] Y. Wang, Y. Zhang, J. Liu, R. Bhandari, Coverage, connectivity, and deployment in wireless sensor networks, in: Recent Development in Wireless Sensor and Ad-hoc Networks, Springer, 2015, pp. 25–44.

[23] M. Nitti, R. Girau, L. Atzori, Trustworthiness management in the Social Internet of Things, IEEE Transactions on knowledge and data engineering 26 (5) (2014) 1253–1266.

[24] D. Fudenberg, J. Tirole, Game theory (1991).

[25] MATLAB (October 2015).
URL `http://fr.mathworks.com/products/matlab/`

[26] G. L. Stüber, Principles of mobile communication, Springer Science & Business Media, 2011.

[27] S. He, D.-H. Shin, J. Zhang, J. Chen, Near-optimal allocation algorithms for location-dependent tasks in crowdsensing, IEEE Transactions on Vehicular Technology.

[28] V. Pilloni, M. Franceschelli, L. Atzori, A. Giua, Deployment of applications in wireless sensor networks: A gossip-based lifetime maximization approach, IEEE Transactions on Control Systems Technology 24 (5) (2016) 1828–1836.

[29] V. Pilloni, P. Navaratnam, S. Vural, L. Atzori, R. Tafazolli, Cooperative task assignment for distributed deployment of applications in wsns, in: Communications (ICC), 2013 IEEE International Conference on, IEEE, 2013, pp. 2229–2234.

[30] X. Lin, J. G. Andrews, A. Ghosh, R. Ratasuk, An overview of 3GPP device-to-device proximity services, IEEE Communications Magazine 52 (4) (2014) 40–48.

[31] 3GPP TR 36.843 V12.0.1, Study on LTE device to device proximity services; radio aspects, release 12 (March 2014).

[32] K. Wang, J. Alonso-Zarate, M. Dohler, Energy-efficiency of LTE for small data machine-to-machine communications, in: 2013 IEEE International Conference on Communications (ICC), IEEE, 2013, pp. 4120–4124.

[33] L. Song, D. Niyato, Z. Han, E. Hossain, Wireless device-to-device communications and networks, Cambridge University Press, 2015.

[34] 3GPP, TR 22.891 feasibility study on new services and markets technology enablers (2016).