

# Energy-Aware Clustering of Ubiquitous Devices for Collaborative Mobile Applications

Carmela Comito  
ICAR-CNR  
Rende (CS), Italy  
Email: ccomito@dimes.unical.it

Domenico Talia  
DIMES, University of Calabria  
Rende (CS), Italy  
Email: talia@dimes.unical.it

**Abstract**—The continuous advances in mobile computing technologies have paved the way to the spreading of new classes of collaborative mobile applications. *Clustering* the mobile nodes into local groups may promote and ease collaborations. In this paper we propose a clustering scheme formulated as a multi-criteria optimization problem based on the weighted-sum approach. The aim of the optimization is to maximize the network lifetime and the mobility similarity of the devices within a group. Depending on the specific application scenario, the weights associated to the cost functions can be properly tuned. Experimental results show that by using the proposed approach, the network lifetime is extended and the number of alive devices is significantly higher compared to alternative clustering strategies, while meeting application-level performance constraints as high mobility similarity within a cluster.

## I. INTRODUCTION

With the dissemination and increasing power of wireless networks and mobile devices, ad-hoc networking is gaining an always greater importance with the growing number of widespread applications. Typical applications include body-health monitoring, vehicle monitoring, emergency/rescue operations for disaster relief efforts, law enforcement, wireless security systems. In such scenarios, communications are often among teams which tend to coordinate their movements. Hence, the need arises for developing efficient environments to support real-life group mobility scenarios. A key aspect to be addressed to enable effective and reliable computing over mobile devices is ensuring energy efficiency, as mobile devices are battery-power operated and lack a constant source of power.

Efficient resource allocation, and energy management can be achieved through *clustering* of mobile nodes into local groups. In this paper we propose a clustering scheme where mobile devices are organized into local groups (clusters or mobile groups). Each cluster has a node referred to as cluster-head that acts as the local coordinator of the cluster. The focus of our approach is building and maintaining a cluster structure in a network of cooperating mobile devices. The problem of assigning devices to clusters has been formulated as a multi-criteria decision making problem based on the weighted-sum approach. We defined a multi-objective cost function where the aim of the optimization is to maximally extend the network lifetime and the mobility similarity of the devices within a group. Depending on the specific application scenario, the

weights associated to the two costs can be properly tuned. This way optimal decisions may be taken in the presence of trade-offs between the two objectives. A weighted metric that combines the effect of energy and mobility of nodes is also introduced to select suitable cluster-head nodes.

An extensive experimental evaluation has been performed to assess the performance of the proposed strategy in different network and application scenarios. The experimental evaluation shows that the proposed strategy is able to build a quite stable cluster structure in which devices within a group exhibit very similar mobility. Furthermore, it addresses energy saving more than related approaches. In fact, the network lifetime is extended and the number of alive devices is significantly higher compared to alternative strategies, while meeting application-level performance constraints.

The remainder of the paper is organized as follows. Section II outlines the adopted energy model. Section III describes the target application scenario. The energy-aware clustering protocol and related algorithm are presented in Section IV. Section V presents the experimental results. Finally, Section VI concludes the paper.

## II. ENERGY MODEL

Energy consumption of mobile devices depends on the computation (*EC*) and the transmission (*ET*) loads. To the scope of the validation of the clustering algorithm we focused only on the transmission energy as we assume that the computational load is the same for all the devices. This assumption is motivated by the fact that to evaluate the clustering scheme we only need to measure the energy depleted for transmission as nodes need to communicate in order to establish and maintain the clustering scheme. In mobile networks, nodes must always be ready to receive traffic from neighbors due to the absence of base station nodes. Therefore, every node overhears every packet transmission occurring in its transmission range consuming this way energy uselessly. This idle energy consumption is referred to as *overhearing*. Consequently, to model the energy consumed for transmission, the costs to send, receive and discard a packet must be included:

$$ET = E_{\text{send}} + E_{\text{receive}} + E_{\text{discard}} \quad (1)$$

We modeled the energy for transmission by exploiting a well-known and widely-adopted energy characterization model [1],

[2] has been validated experimentally. Based on this model, the cost for a node to send or receive a message is modelled as a linear function. In this function there is a fixed cost associated with channel acquisition and an incremental cost proportional to the size of the message. The fixed channel access costs, denoted as  $b_{\text{send}}$  and  $b_{\text{recv}}$ , and the incremental costs,  $m_{\text{send}}$  and  $m_{\text{recv}}$ , are the same for broadcast and point-to-point. For the ease of notation we denote them as just  $b$  and  $m$ , but clearly their values are specific to the mode of the network interface.

$$E_{\text{send}} = E_{\text{receive}} = E_{\text{discard}} = m * |MSG| + b \quad (2)$$

where  $|MSG|$  is the size of the message exchanged. In [2] have been described a series of experiments reporting detailed measurements of the per-packet energy consumption of a IEEE 802.11 wireless network interface operating in ad-hoc mode. Coefficients for the equations 2 have been determined by performing these measurements.

### III. REFERENCE APPLICATION SCENARIO

Main mobile applications range from disaster relief (fire, flood, earthquake) to law enforcement (crowd control, search and rescue), healthcare and digital battlefield communications. Some key characteristics of these systems are team collaboration of large number of mobile units, limited bandwidth, low latency access to distributed resources. The nodes in ad hoc networks move according to various patterns. However, group motions occur frequently. Communications are often among teams which tend to coordinate their movements (e.g., a firemen rescue team in a disaster recovery situation). This team relationship makes it possible to partition the network into several groups, each with its own mobility behavior.

The focus of our approach is on building and maintaining a cluster structure in a network of cooperating mobile devices. Nodes may serve different roles, such as *cluster-*

within a group interact through ad-hoc connections (e.g., wi-fi, bluetooth). Interactions among mobile groups (*cluster-to-cluster connections*) take place through ad-hoc connections among the cluster-heads of the groups.

### IV. THE CLUSTERING APPROACH

In this section we describe the design principle of the proposed clustering approach together with the clustering algorithm.

#### A. Design Principle and Related Work

In literature there are several clustering approaches for MANETs [12]. Different approaches focus on different goals and performance metrics.

Most of the clustering schemes proposed in literature are proactive in the sense that a constant refresh rate of cluster-related information is required, introducing, this way, significant background control overhead even if there is no data traffic in the network. In most of the real-world applications such as covert military operations, rescue operations and emergency responses, this periodic control traffic can preclude the success of the action to be taken. (e.g., exposure to enemy interception, power consumption, etc).

To overcome the drawbacks of proactive clustering protocols, reactive on-demand or passive clustering approaches have been proposed [3], [4], [5]. In general, these approaches do not use any explicit control messages to maintain clusters. Instead, they piggyback the control information on the out-going data packets so reducing the control overhead. However, most of the proposed passive approaches has the main limitation of not being able of facing dynamic scenarios or sudden emergency (e.g., nodes join/leave, battery draining) not taking into consideration in the clustering process any node-related parameter like energy, mobility, etc.

According to the above considerations, we propose a hybrid clustering solution that integrates the advantages of both low-maintenance and combined-metrics-based clustering. Among proactive approaches, the most related to our clustering scheme (that we will use in the comparison evaluation in Section V) are mobility-based and combined metrics-based approaches. Mobility-aware clustering [6], [9] takes the mobility behavior of mobile nodes into consideration. By grouping mobile nodes with similar speed into the same cluster, the cluster structure can be correspondingly stabilized regardless of nodes mobility. However, due to change in the network topology it may produce high cluster-head re-elections. Differently to such approaches, we consider not only mobility but also energy metric so as to more properly configure clusters and extend network lifetime. Combined metrics based clustering (WCA) [8] considers multiple metrics, such as node degree, cluster size, mobility speed, and battery energy, in cluster configuration, especially in cluster-head selections. Compared to WCA, we use different parameters for cluster-heads selection. Instead of the battery power we use the residual life because it considers not merely the remaining energy of the device but also its energy load. Another key

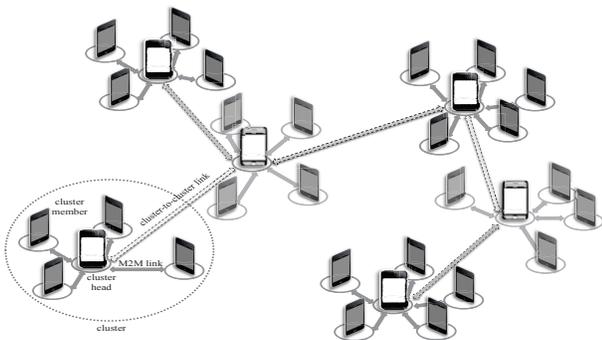


Fig. 1. The clustering architecture.

*head* and *cluster member*. A cluster-head normally acts as the local coordinator for its cluster, performing intra-cluster transmission arrangement, data forwarding, and so on. In our architecture the cluster-head acts also as a gateway for inter-cluster communications. Finally, a cluster member is an ordinary node without any inter-cluster links. We refer to the system architecture depicted in Figure 1. Mobile nodes

difference is that we cluster devices according to their mobility behaviour, instead WCA does not take into account such feature. Moreover, conversely to combined-metrics approach, we do not perform the clustering procedure periodically; procedures like node joining and cluster-head re-election are event-driven. In fact, the parameters (e.g., remaining energy, node mobility) needed for cluster construction and maintenance, are transmitted through periodic messages exchanged by neighbors in MANETs for topology management.

This work extends a previous one [7] where a preliminary version of the scheme was presented. Here we modified that approach to fit collaborative scenarios with group mobility. To the purpose, the clustering algorithm with its optimization function have been substantially modified to account mobility similarity for cluster formation. Moreover, in [7] the scheme was experimentally validated in a static network where devices were not moving. Here, instead, mobility is key in the evaluation and different mobility models have been implemented to test the approach in various real-world scenarios.

### B. Cluster-head Selection

Choosing cluster-heads is key in the clustering approach. The number of cluster-heads depends on many factors like the number of nodes in network, their physical location, the transmission range and the energy level. To select the cluster-head nodes we introduce a combined weighted metric that takes into account the following parameters:

- *Residual life (RL)*. Let be  $RE_i(t)$  the residual energy available at node  $i$  at time  $t$ , and  $P_i(t)$  the instantaneous power; the residual life of node  $i$  at time  $t$ ,  $RL_i(t)$ , is defined as follows:

$$RL_i(t) = RE_i(t)/P_i(t) \quad (3)$$

A node with higher residual life corresponds to a more powerful node that is a good candidate to be elected as a cluster-head.

- *Neighbors (NB)*. The size of each local group also depends on the number of nodes within the cluster-head range. The larger the number of neighbors of a node, the greater the likelihood of being elected cluster-head.
- *Velocity (V)*. A node with lower speed has a higher chance of being a cluster-head. According to [8], we estimate node mobility behaviour by taking the average movement speed till current time  $t$ :

$$V_i(t) = 1/t \sum_{j=1}^t \sqrt{(x_j - x_{j-1})^2 + (y_j - y_{j-1})^2} \quad (4)$$

where the values  $(x_j, y_j)$  and  $(x_{j-1}, y_{j-1})$  are the coordinates of node  $i$  at time  $j$  and  $j - 1$ .

Depending on the specific application, different subsets of the parameters above can be used in the metric to elect cluster-heads. Thus, to establish whether a node  $i$  can be elected a cluster-head, a *Cluster-Head Selection function (CHS)*,  $CHS_i(t)$ , is defined:

$$CHS_i(t) = \alpha RL_i(t) + \beta/V_i(t) + \gamma NB_i(t) \quad (5)$$

where  $\alpha$ ,  $\beta$  and  $\gamma$  are the weights corresponding to the above cited performance parameters.

The node within the transmission range having the maximum value of the *CHS* function will be selected as the cluster-head. The weighting factors are chosen such that  $\alpha + \beta + \gamma = 1$ . Therefore, the contribution of each parameter in the *CHS* function can be tuned by selecting suitable combinations of the weighting factors. For example, in a particular configuration where energy is critic, the weight associated to residual life can be set to a larger value.

### C. Clustering algorithm

Using the proposed weighted approach, in this section we describe our clustering scheme formulated as an heuristic-based decentralized algorithm. A mobile node  $i$  that wants to join the network of clustered devices will execute the *joining node* algorithm described in the following.

A node  $i$  that wants to enter the network has to establish whether it can join an already existing cluster or it has to create a new one. To this aim, node  $i$  broadcasts a *join request* message to advertise its presence and to check the presence within its transmission range of either other not member nodes or already established clusters. After having sent such a message the node waits for responses. One of the two events may occur:

- Node  $i$  does not receive any reply. In this case it will form a new group electing itself as the cluster-head of the group.
  - Node  $i$  receives one or more replies. Also in this case we have to distinguish two possible scenarios.
- Node  $i$  receives one or more responses from only nodes not belonging to already established clusters (that we referred to as not member nodes). In this case the node will form a new cluster. Once again we have to distinguish two different cases.

- 1) In case node  $i$  received just one response will form a new cluster with the replying node by immediately activating the cluster-head election procedure. The node with higher CHS will be elected as the cluster-head.
- 2) Conversely, in the case of multiple responses the cluster-head election phase is not activated as node  $i$  will be elected as cluster-head and its members will be all the repliers. Only in a next phase, if necessary, the cluster-head election phase will be activated. This choice is motivated as follows. In case of multiple responses it is quite complicated to elect the cluster-head by issuing the cluster-head election procedure as a set of messages has to be exchanged among all the pairs of involved nodes to check not only the values of the CHS functions but also the respective reachability of nodes (we remind here that a node can act as cluster-head only if reaches all the other node in the cluster). This last condition is obviously satisfied by node  $i$  that is the recipient of all these messages.

- Node  $i$  receives one or more responses from both not member nodes and already established clusters. In this case,

we formalize the problem of joining node as a multi-criteria optimization problem based on the weighted-sum approach. The aim of the optimization is to maximally extend the network lifetime and the mobility similarity of the devices within a group. Depending on the specific application scenario, the weights associated to the two costs can be properly tuned. This way optimal decisions may be taken in the presence of trade-offs between the two objectives. More precisely, we optimize the problem by iteratively trying to improve a candidate solution. A feasible assignment is optimal if the corresponding network lifetime and/or mobility similarity are maximized among all the feasible assignments as expressed by the following equation:

$$\text{Max} \sum_{j=1}^N (\alpha_j \text{RL}_{LG_j}(t) + \beta_j \text{MS}_{(i, LG_j)}(t)) \quad (6)$$

where  $\text{RL}_{LG_j}$  denotes the residual life of local group  $LG_j$ ,  $N$  is the number of groups in the network,  $\text{MS}_{(i, LG_j)}$  represents the mobility similarity among nodes  $i$  and local group  $LG_j$ . The parameters  $\alpha_j$  and  $\beta_j$  take into account the importance of the group  $i$  in the network; details about such parameters are given in the following. Node  $i$  will join the cluster associated with the maximum value of the objective function. The residual life of each local group is described by the following equation:

$$\text{RL}_{LG_j} = \sum_{k=1}^{N_{LG_j}} \alpha_k \text{RL}_k(t) \quad (7)$$

where  $N_{LG_j}$  is the number of nodes within the local group  $LG_j$ ,  $\text{RL}_k$  is the residual life of node  $k$  in the group and the parameter  $\alpha_k$  takes into account the importance of the node  $k$  in the local group.

The mobility similarity is calculated as follows.

We characterize the group mobility similarity in terms of (1) the average velocity of the group, according to equation 4, and (2) the average angle of movement of the nodes in the group. We define the average angle of movement of a node  $i$  till current time  $t$  as follows :

$$\theta_i(t) = \frac{1}{t} \sum_{j=1}^t [\arctan \frac{y_j - y_{j-1}}{x_j - x_{j-1}}] \quad (8)$$

Thus, the average angle of movement of a local group  $LG_j$  till time  $t$  is as follows:

$$\theta_{LG_j}(t) = \text{avg}(\theta_i(t)), i \in [1, N_{LG_j}] \quad (9)$$

According to equation 4, the average velocity of the group  $LG_j$  till time  $t$  is expressed as follows:

$$V_{LG_j}(t) = \text{avg}(V_i(t)), i \in [1, N_{LG_j}] \quad (10)$$

The mobility similarity among the joining node  $i$  and a generic local group  $LG_j$  in the network is determined by taking into account both the average velocity of the group as well as the average angle of the movements of the nodes

in the group:

$$\text{MS}_{(i, LG_j)}(t) = \frac{1}{|V_i(t) - V_{LG_j}(t)| + |\theta_i(t) - \theta_{LG_j}(t)|} \\ \text{if } (|V_i(t) - V_{LG_j}(t)| + |\theta_i(t) - \theta_{LG_j}(t)|) > 1 \text{ else } \text{MS}_{(i, LG_j)}(t) = 1 \quad (11)$$

The parameters  $\alpha_j$  and  $\beta_j$  take into account the relevance of a node/group within the group/network from energy and mobility perspectives, respectively. In fact, a group with specific expertise could have more relevance than others in specific application scenarios. For example, in an emergency rescue case a team of medical staff could require more energy; thus, the value of the corresponding  $\alpha_j$  parameter could be set higher.

The algorithm followed by a node to join the network and, thus, to select its cluster-head is described in Figure 2. We use a state diagram formalism to model the behavior of a generic node, which includes the behavior of nodes performing join requests (NOT\_CLUSTER\_MEMBER nodes), as well as the behavior of nodes responding to join requests (CLUSTER\_HEAD nodes). The state diagram is represented using a pseudo-code that describes states, macro-states (containers for other states), and events that produce transitions among states. Due to lack of space we show only the key procedures of the algorithm.

The joining node algorithm is also executed to let nodes (re-)affiliate to clusters during the evolution of the clustering. Nodes may need to be re-affiliated because the signal strength received from the cluster-head decreases under a given threshold, and the node cannot stay connected to that cluster-head. In such a case, a re-affiliation is needed: the node can join another cluster following the joining node algorithm described above. Moreover, the nodes in the network representing a single-member cluster periodically invoke the joining node algorithm to assess whether they can join another cluster.

### Complexity.

**Lemma 1.** The clustering algorithm has a worst-case processing time complexity of  $O(n)$  per joining node, where  $n$  is the number of nodes in the network.

**Proof.** The worst case happens when the joining node receives multiple join responses. In this event it is necessary to distinguish two different cases.

- 1) The joining node receives responses from only not member nodes. In this case, it will elect itself as the cluster-head and as such it needs to calculate the residual life of the just formed cluster. So, it will take a processing time of  $O(m)$  where  $m$  is the number of nodes replying, with  $m \ll n$  and  $n$  the number of nodes in the network. In the worst case, when all the nodes in the network are still not members as they do not have started yet the cluster formation protocol, the joining node takes a processing time of at most  $n$  to compute the cost, thus the complexity is  $O(n)$ .
- 2) The node receives multiple responses from cluster-heads. In that case it will join the cluster allowing to maximize the cost function in equation 6. In particular, each replying



function (equation 6) and it will send just a message to the cluster that it will join. Thus, the total number of exchanged messages is  $1 + NB_i + 1$ . Generally,  $NB_i$  is strictly lower than  $n$  unless all the nodes in the network are cluster-heads or as many single-node clusters. Therefore, in the worst case the message exchange complexity is  $O(n)$ .

**Lemma 3.** The joining node algorithm terminates in a constant number of iterations.

**Proof.** From proof. of Lemma 2, we can also note that the algorithm requires at most a number of rounds equals to 3.

## V. PERFORMANCE EVALUATION

In this section we present a simulation study of our clustering scheme.

### A. Simulation Setting

A custom discrete-event simulator has been implemented to perform the experimental evaluation. As a first step, the simulator builds a network composed of 100 mobile devices distributed over an area of  $250,000 m^2$ , and let them grouping into clusters based on the proposed clustering algorithm.

Unless otherwise specified, the simulation setting is as follows. Each node is powered by a battery. An initial energy capacity ranging from 3,000 J to 17,982 J is assigned to each device, following a normal distribution. The transmission range is set to 150 meters. Each device is equipped with a network interface 802.11 b/g, with a bandwidth of 11 Mbps. The data rate is 2Mb/s and the traffic is modeled according to a Pareto distribution. Pareto traffic model is an ON/OFF traffic: it generates traffic during ON period (burst time). Average ON and OFF (idle time) times are 1.5s and 0.5s respectively with a shape parameter of 2.5. Initially, each mobile node is assigned a unique node ID, a x-y position as determined by the group behaviour, a mobility speed. At every time unit, the nodes are moved according to the random Group Mobility model (unless otherwise specified) with velocity distributed according to the group politics. This behavior is repeated for the duration of the simulation. Several runs of each simulation scenario are conducted to obtain statistically confident averages. The final result is the average of 50 simulation results. While our algorithm neither relies on, nor makes use of, any specific routing algorithm, we assume for the sake of evaluation our algorithm, that all communication to and from a node goes via its cluster-head.

For the *CHS* function, we used the following weights:  $\alpha = 0.55$ ,  $\beta = 0.35$  and  $\gamma = 0.1$ . The high value for  $\alpha$  accounts the importance of residual life in cluster-head selection. The lower values of  $\beta$  and  $\gamma$  are because velocity and neighbors number are less important in the target group mobility scenarios.

### B. Evaluation Metrics

As discussed before, there is a great variety of clustering algorithms. Choosing the right criteria to compare those algorithms is key. An important criteria concerns the stability of the clustering structure. A good clustering algorithm should maintain its cluster structure as stable as possible while the

topology changes, minimizing the number of node transitions from one cluster to another one. Moreover, the algorithm should be able to group together devices exhibiting similar mobility features so as to fit target collaborative scenarios. Another important criteria is energy-efficiency. Energy depletion may turn off mobile devices, thus, causing network partitions. Accordingly, the performance evaluation has been conducted along two main directions aiming at analyzing the topology and the energy behaviour of the proposed clustering scheme. We focus in particular on the following evaluation metrics:

- *Number of clusters.* This value defines the average number of logical partitions formed in the network with the mobile nodes.
- *Number of re-affiliations and number of cluster-head changes.* The first metric refers to the disassociation of a cluster member from its cluster-head and associating itself to another cluster without affecting the corresponding cluster-head. The second metric represents the number of times that cluster-heads give up this role (it is actually the number of cluster-head re-elections). A higher value for both metrics means higher control traffic overhead since all active routes to the node need to be updated. Whereas, a lower number implies a better cluster stability.
- *Average Group Mobility Similarity.* This metric measures the ability to cluster together devices with group mobility behaviour.
- *Network residual life and number of alive devices.* With those metrics we aim at evaluate the energy-efficiency of the clustering algorithms.

### C. Mobility Model

Mobility models are designed to describe the movement patterns of mobile users, and how their location, velocity and acceleration change over time. Mobility models are key to clustering performance, thus, it is essential that they properly emulate the movement patterns of targeted real life applications. Mobility models are divided into two main categories: entity mobility model and group mobility model. Entity mobility model specifies individual node movement. Group mobility model describes group movement as well as individual node movement inside groups. In this work, we consider two mobility models representative of each of the above mentioned categories, respectively. These models capture a wide range of mobility patterns for ad-hoc applications. The models are briefly described in the following .

Random Waypoint (RW) [11] is a model in which nodes move independently to a randomly chosen destination with a randomly selected velocity varying uniformly between 0 to a maximum value per time unit. RW includes pause times between changes in direction and/or speed. At the beginning of simulation, every node is independently choosing its mobility status: move or pause. If node is pausing at time zero, after a normal distributed random time period, it will change to moving phase. If node is moving, it will keep moving until it reached the destination. After that, it will pause for a normal

distributed random time period and go to the moving phase, again.

Most of the mobility models in literature describe independent behavior [12]. However, relationships among mobile nodes is key when they move with the same purpose, as it is the case for scenarios like disaster recovery or military deployment where several mobile nodes most likely move with a common objective. Some key characteristics of these systems are team collaboration of large number of mobile nodes and the need for supporting coordinated use to shared distributed resources. Among the group mobility models, we refer to the Reference Point Group Mobility (RPGM) [12], [13] which models nodes moving in group. Here, each group has a logical center (group leader) that determines the group's motion behaviour. Initially, each member of the group is uniformly distributed in the neighbourhood of the leader. Subsequently, at each instant, every node has a speed and direction that is derived by randomly deviating from that of the group leader. Each node has a reference point within a certain range from the group center which is moved together with the movement of the group center.

#### D. Experimental Results

We organized the experiments in two categories that evaluate the performance of the clustering scheme with respect to the mobility and transmission range parameters, respectively.

##### *Mobility*

In this set of experiments we first evaluated how the mobility model impacts on the algorithm performance. Then, based on the results of this evaluation we compared our algorithm with representatives related approaches.

Representative cases have been modeled through the RPGM model following the approach of [13]. These models have been then compared with the Random Waypoint mobility model in order to evaluate our clustering scheme.

The first model is a geographical partition model. The entire area is divided into several adjacent regions, with a different group in each region. This model can be used to represent large scale disaster recovery, where different paramedic, police, firemen teams work in separated neighborhoods. According to [13] we refer to this model as In-Place Mobility Model (IPM).

The second model describes an overlapped operation. Different groups carry out different tasks over the same area. For example, in a disaster recovery area, the rescue team, the medical assistant team and the psychologist team will be randomly spread out over the area. Each group has a unique motion pattern, speed, scope. We refer to this model as Overlap Mobility Model (OM).

The third model is a convention scenario. It models the interaction between exhibitors and attendees. A group of attendees roams from room to room. They may stop in one room for a while and then move on to another room. Or, they may pass through one room quickly. This is called the

Convention Model (CM). This model could also be used to represent the roaming behavior of drivers on a road network.

Speed is a key parameter to evaluate the mobility behaviour of our clustering scheme. The results reported in the performance graphs are based on average group speed and on mean motion displacement of nodes around their reference points. The radio transmission range is 150 meters. The data rate is 2Mb/s. We use 10 groups in the in-place model. The simulation area is divided into 10 regions. Each group moves around in one region. We also use 10 groups in the overlap model, but each group scatters over the entire area. Five of the 10 groups move in a circular pattern in different direction. Three groups move linearly, back and forth. The last two groups are almost static. In the convention model, we have 7 exhibitor groups moving slowly in each of the four partitions as in the in-place model. We also have 3 viewers group, which roam around the entire area. For the random waypoint model each node begins by pausing in one location for about 90 seconds and then chooses a uniformly distributed random speed and destination and moves to that destination in a straight line with that selected speed.

We set the weights of the multi-costs function of equation 6 as follows. All the devices within a group have  $\alpha = 1$ , meaning that in the target scenario all the devices have the same relevance. Also,  $\alpha = 0.6$  and  $\beta = 0.4$  for all local groups. This choice is because we focus on group mobility scenarios and mobility similarity is at a great extend already assured within nodes of a group. In fact, in this setting the average group mobility similarity is clearly very high for all the group mobility models (ranging from 0.9 to 0.75), whereas for the random model it decreases with devices' speed (ranging from 0.76 to 0.33 ).

Figure 3 (a) shows the number of re-affiliations per minute with respect to increasing devices' speed. When mobility increases, all the models show an increase in the affiliation rate as cluster members change clusters more frequently. As expected, the random mobility model has a higher affiliation rate than the group mobility models. In particular, the convention model shows the smallest affiliation rate, since the 6 exhibit groups move slowly. The in-place model and overlap model have different motion patterns, but they have similar affiliation rates. Figure 3 (b) shows that the random mobility model has a higher cluster-head change rate than the group mobility models. The cluster-head change rate of the overlap model is much higher than in the in-place model and the convention model. According to the overlap model, many groups (6 groups in simulation) have activities in the same field. Thus, the intermixing of the 6 groups generates more opportunities for cluster re-election. Contrarily, the in-place model only allows each group to move within its own geographical area, with fewer cluster change opportunities.

Figure 4 (a) shows that the network residual life decreases very slowly with nodes' speed for all the mobility models. However, the decrease is more pronounced for the random model. Accordingly, in Figure 4 (b) one can see that the number of alive devices slightly decreases with speed for all

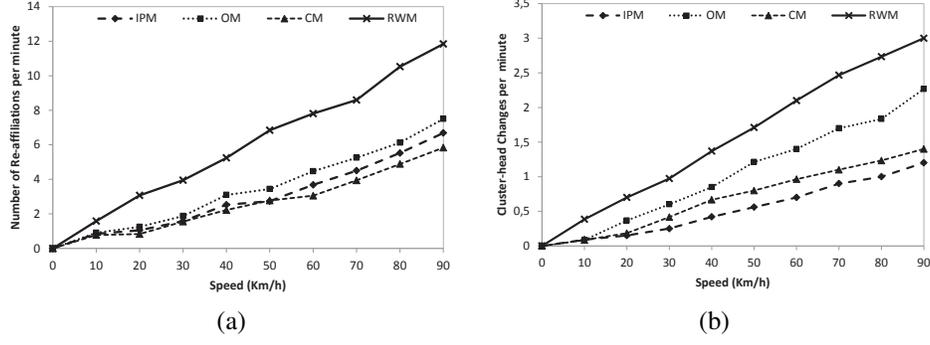


Fig. 3. Number of re-affiliations per minute (a) and Cluster-head changes per minute (b) w.r.t. devices' speed.

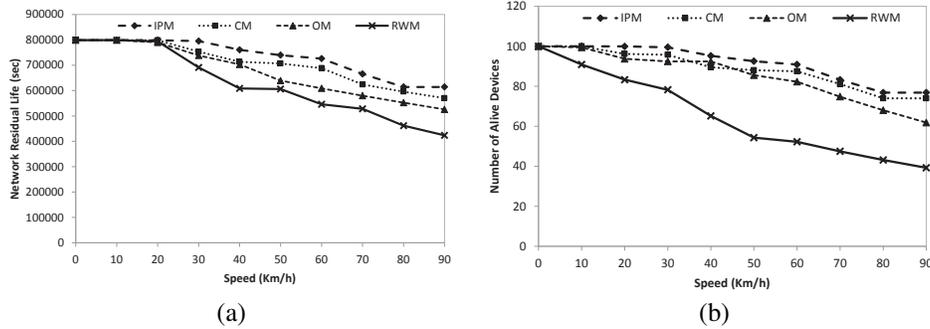


Fig. 4. Network residual life (a) and Number of alive devices (b) w.r.t. devices' speed.

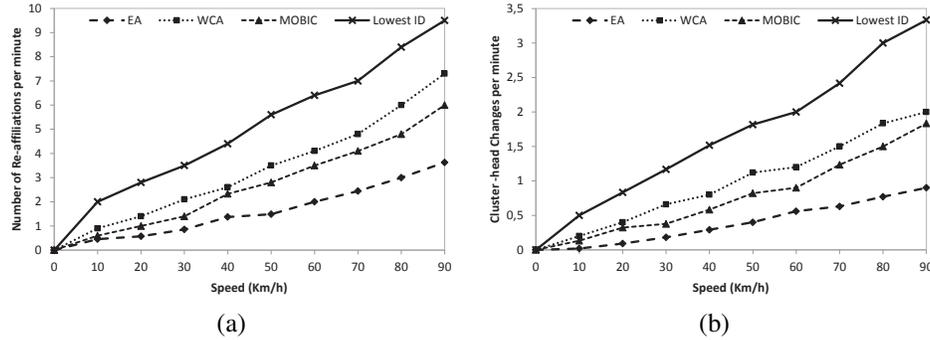


Fig. 5. Number of re-affiliations per minute (a) and Cluster-head changes per minute (b) for the different algorithms w.r.t. devices' speed.

the group mobility models, whereas it decreases faster for the random model. Therefore, also from the energy perspective our clustering scheme performs better in case of a group mobility model confirming again that our clustering approach is specifically tailored for group mobility scenarios. Anyhow, with both mobility models we have low energy consumption with nodes' speed. This can be explained as follow. As our clustering approach is hybrid, in-between on-demand and proactive protocols, it exchanges clustering-related info only when necessary and, thus, the energy consumption tends to stay constant. Consequently, it is not so sensitive to speed and mobility compared to on-demand approaches. To confirm this, we compared our EA scheme to some of the main representative approaches in literature (see Section IV-A): the combined Weighted Clustering Algorithm (WCA) [8], a mobility-aware clustering as MOBIC [6],

and the Lowest-ID (LID) algorithm [10] (also known as identifier-based clustering) that chooses the node with the lowest ID as a cluster-head. In this comparison, we use a group mobility model, the in-place model described above, as it best represents real-world collaborative scenarios as the ones we want to model with our clustering scheme. In this case we have 20 groups. Figure 5 (a) shows the number of re-affiliations per minute for the different clustering schemes with respect devices' speed. The number of re-affiliations increases with speed for all mobility models, as expected. This is because nodes with higher speed quit rapidly their cluster to reach another one. We observe that, for very low mobility EA and MOBIC algorithms exhibit similar results. However, with the increasing of speed, EA exhibits lowest re-affiliation rates producing, thus, more stable clusters. The really bad behaviour of the Lowest-ID algorithm is because it

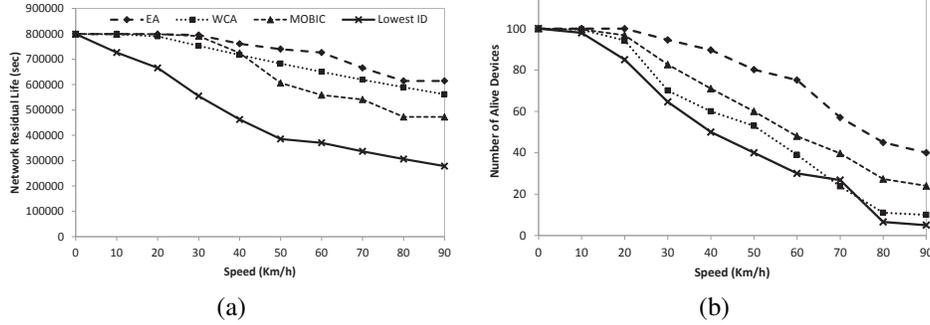


Fig. 6. Network residual life (a) and Number of alive devices (b) for the different algorithm w.r.t. devices' speed.

chooses the node with the lowest ID as a cluster-head leading to the battery drainage of such nodes. Thus, causing several node re-affiliations with increase in device's speed. Similar comments hold for Figure 5 (b) showing the cluster-head changes with speed. In fact, for very low speeds EA, WCA and MOBIC present again similar results. Then, with higher speed, EA considerably outperforms the other algorithms. Once again we can see the very bad behaviour of the Lowest-ID algorithm. Figure 6 (a) plots the network residual life for the four algorithms. For low mobility values once again EA, WCA and MOBIC present comparable results. However, for higher values of about 50 Km/h, MOBIC performance degrades while EA and WCA keep performing very similarly, even though EA shows the higher network residual life. Also from the energy point-of-view the Lowest-ID shows a bad behavior. Figure 6 (b) shows that EA maintains the higher number of alive devices. This outcome further confirms the good behaviour of our scheme also from the energy perspective.

#### Transmission Range

The transmission range of a mobile device is an important parameter for evaluating the performance of the clustering scheme. It particularly impacts on the topology of the clustering as evidenced by the following experiments. Assuming a uniform distribution of nodes in the network, we evaluated how the number of clusters and thus the number of cluster-heads varies with the transmission range. We compared again the performance of our clustering algorithm with those achieved by WCA, MOBIC and Lowest-ID algorithms, respectively. The simulation setting is as before with average devices' speed of 60 Km/h.

Figure 7 (a) shows that the average number of clusters decreases as the transmission range increases for all the different clustering algorithms, as expected. However EA and WCA exhibit very similar results, with the lowest number of clusters and thus, on average, the higher number of members per cluster. In Figure 7 (b) one can see that the average cluster mobility similarity grows with the transmission range, for all the algorithms. Again, EA outperforms related approaches exhibiting a great ability to cluster devices with group mobility behaviour. Figure 8 (a) shows that the re-affiliation rate

decreases with the transmission range. This is because cluster-heads cover a larger area and more nodes are within range of other nodes for longer period of time. Therefore, fewer clusters (which are large in size) are formed and the mobility of nodes does not cause them to frequently move in and out of range of each other, so the number of re-affiliations decreases. This behaviour is shown by all the algorithms. However, EA produces more stable clusters than those produced by the other algorithms. Same comments hold for Figure 8 (b) where the increasing of the transmission range results in lower cluster-head changes. This means a better connection among nodes and thus higher cluster stability as the heads take major role. Again, EA outperforms related approaches. Figure 9 (a) shows that the network residual life decreases with the transmission range. The decrease is quite low for EA and WCA as they produce more stable clusters (as shown on Figures 8 (a) and (b)) and, thus, the energy dissipated decreases. Moreover, from Figure 9 (b) we can see that our algorithm remains with the higher number of alive devices that slightly decreases with the transmission range. By summarizing, also taking into account the transmission range, the proposed clustering scheme behaves well in terms of both cluster topology and energy and mobility issues. In particular, it is able to build a quite stable cluster structure and addresses energy saving more than related approaches.

## VI. CONCLUSION

In this paper, we have presented a clustering scheme for mobile computing focusing on energy efficiency. To conservatively consume energy and prolong network lifetime we have introduced an energy-aware adaptive distributed clustering scheme based on a combined weighted metric. It dynamically reconfigures the clusters based on the energy requirements of applications to avoid the performance degradation. We focused on collaborative applications where mobile devices present group mobility. We evaluated the clustering scheme through simulation experiments in different network and application scenarios. Results showed that by using the proposed energy-aware approach, the network lifetime can be extended and the number of alive devices can be significantly higher compared to alternative strategies, while meeting application-level performance constraints.

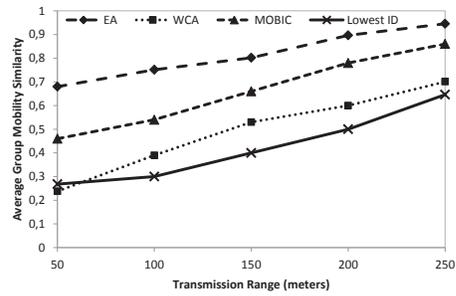
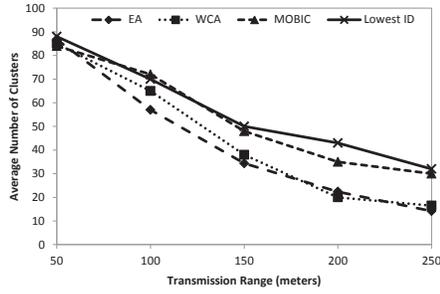


Fig. 7. Average number of clusters (a) and Average cluster mobility similarity (b) w.r.t. transmission range.

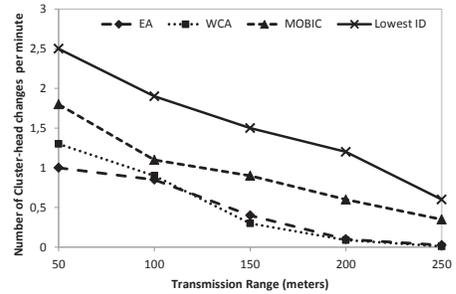
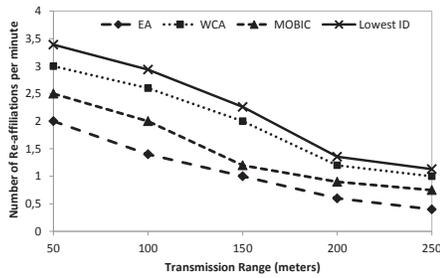


Fig. 8. Number of re-affiliations per minute (a) and Number of cluster-head changes per minute (b) w.r.t. transmission range.

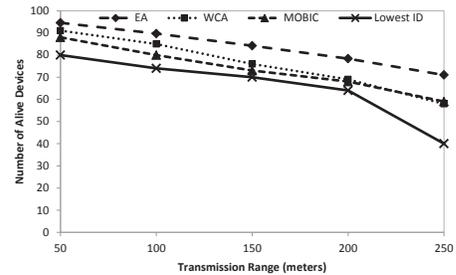
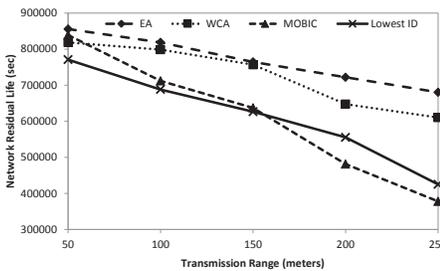


Fig. 9. Network residual life (a) and Number of alive devices (b) w.r.t. transmission range.

#### ACKNOWLEDGMENT

The work presented in this paper has been partially supported by the COST program Action IC1305, 'Network for Sustainable Ultrascale Computing (NESUS)'.

#### REFERENCES

- [1] L. M. Feeney. "An energy-consumption model for performance analysis of routing protocols for mobile ad hoc networks". *Mobile Networks and Applications Journal*, 6(3):239-250, (2001).
- [2] L. M. Feeney, M. Nilsson. "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment". *INFOCOM*, pp. 1548-1557, (2001).
- [3] T. J. Kwon and M. Gerla. "Efficient flooding with Passive Clustering (PC) in ad hoc networks". *Computer Communication Review*, 32(1): 44-56, 2002.
- [4] C. Cramer, O. Stanze, K. Weniger and M. Zitterbart. "Demand-Driven Clustering in MANETs". *ICWN*, pp. 81-87, 2004.
- [5] Foroozan, F., Datta, S. "A Low-Maintenance Energy-Aware Clustering Algorithm for Wireless Ad-hoc Networks". *WiMob*, pp. 457-462, 2006.
- [6] P. Basu, N. Khan, and T. D. C. Little. "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks". *ICDCSW*, pp. 413-18, 2001.
- [7] C. Comito, D. Talia, and P. Trunfio. "An Energy-Aware Clustering Scheme for Mobile Applications". *Scalcom*, pp. 15-22, (2011).
- [8] M. Chatterjee, S. Das, and D. Turgut. "Wca: A weighted clustering algorithm for mobile ad hoc networks". *Cluster Computing Journal*, 5(2):193-204, (2002).
- [9] A. B. McDonald and T. F. Znati. "A Mobility-based Framework for Adaptive Clustering in Wireless Ad Hoc Networks". *JSAC*, 17:1466-87, (1999).
- [10] M. Gerla and J. T.-C. Tsai, "Multicluster, Mobile Multimedia Radio Networks". *Wireless Networks* 1, pp. 255-265, 1995.
- [11] D. B. Johnson and D. A. Maltz. "Dynamic Source Routing in Ad-Hoc Wireless Networks". *Mobile Computing*, pp. 153-181, 1996.
- [12] T. Camp, J. Boleng, and V. Davies. "A survey of mobility models for ad hoc network research". *WCMC*, 2(5):483-502, 2002.
- [13] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks". *MSWiM*, pp. 53-60, 1999.