IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 12, 2019, accepted August 26, 2019, date of publication August 30, 2019, date of current version September 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2938619

CER-CH: Combining Election and Routing Amongst Cluster Heads in Heterogeneous WSNs

MATTEO MICHELETTI[1], **LEONARDO MOSTARDA**[1], **(Member, IEEE), AND ALFREDO NAVARRA**[2]
[1]Computer Science Department, University of Camerino, 62032 Camerino, Italy
[2]Mathematics and Computer Science Department, University of Perugia, 06123 Perugia, Italy

Corresponding author: Leonardo Mostarda (leonardo.mostarda@unicam.it)

This work was supported in part by the European Project Geospatial-Based Environment for Optimisation Systems Addressing Fire Emergencies (GEO-SAFE) under Contract H2020-691161, and in part by the Italian National Group for Scientific Computation (GNCS-INdAM).

ABSTRACT Heterogeneous Wireless Sensor Networks (WSNs) are essential to the IoT vision. WSNs implement a virtual layer that can gather data about the real world. WSNs are composed of wireless battery powered devices that can have heterogeneous features related to computational power, memory, and communication capabilities. Because devices are battery powered, gathering data in an energy efficient way is crucial for the lifespan of the network. Clustering is a reasonable solution. This organises the devices into sets (clusters). Each cluster has a cluster head (CH) that gathers data from the nodes belonging to its cluster and communicates with other CHs in order to report data to a centralised base station (BS). This is usually achieved via a CH routing tree that is rooted at the BS. Beside clustering, the rotation in the role of CHs amongst the nodes of the network is a standard means to better distribute energy consumption. In this paper we propose a novel approach (CER-CH) where the CH routing tree definition and the CH rotation are combined together. More precisely, starting from any clustering criteria, we propose a novel rotation heuristic combined with a novel top-down CH routing tree definition in order to balance the node energy consumption and generate more energy efficient CH routing trees. Our experiments show that our rotation strategy enhances on average the network lifespan of 20% when compared to the state of art protocols.

INDEX TERMS Heterogeneous wireless sensor networks, energy saving, routing.

I. INTRODUCTION

Internet-of-Things (IoT) is one of the modern technological revolutions that allows communication amongst a variety of different devices. Heterogeneous Wireless Sensor Networks (WSNs) is an essential part to implement the IoT vision. WSNs are often used within an IoT system to collect data and send data through a router as part of the infrastructure system. Main applications [1]–[4], include transportation, smart homes, smart supply chain, smart cities, connected cars, smart industry, and smart retails.

As opposed to homogeneous WSNs where all nodes are equal, heterogeneous ones are composed of nodes that can have different features such as computational power, memory, communication capabilities and battery power. When devices are battery powered, gathering data in an energy efficient way is a fundamental requirement. Clustering is one of the solutions proposed by the research community. Clustering organises the nodes into clusters. In each cluster, a cluster head (CH) is elected with the aim of collecting data from the member nodes of the cluster itself (intra-cluster communication) and to communicate with other CHs to report data to a base station (BS) (inter-cluster communication).

Rotation is a widely used technique that aims at reducing the number of cluster head elections and cluster formation phases thus reducing the amount of control messages. Rotation is used to balance energy in static clustering [5], [6] where the network is clustered once at the beginning or in combination with well known dynamic clustering strategies [7]–[10] in order to reduce the amount of re-clustering phases. Static clustering divides the network according to some virtual grids [11]–[13], virtual layers [5], [14], [15], concentric circles [5], [6]. In these cases rotation usually includes some predetermined scheduling that is calculated by using a node energy consumption model. This estimates the energy consumption in average conditions (e.g., average distance

The associate editor coordinating the review of this article and approving it for publication was Peter Langendorfer.

between a member and its CH) and simplified settings (e.g., virtual grids). This may lead to lifetime performance degradation when the average case is not representative. On the contrary, dynamic clustering have clusters that change over the WSN lifetime. Rotation is usually adaptive since considers the node residual energy [8], [9]; few approaches may add other node features such as the node rate and the node initial energy [7]. Although rotation based on residual energy is suitable for a wide range of WSN settings, it may be inefficient for heterogeneous networks [7]. After CH selection a CH routing tree is defined [16]–[18]. This allows the delivery of data from each node to the BS. Routing tree algorithms aim at ensuring connectivity and very few approaches consider the energy efficiency of the generated routing tree [16], [17]. These are limited at building energy efficient routing paths after the CH routing tree has been defined. A CHs selection that also considers the energy efficiency of the generated routing tree could lead to a more energy efficient rotation approach.

In this paper we propose a novel approach (CER-CH) which combines a novel rotation heuristic with a novel top-down CH routing tree definition in order to balance the node energy consumption and generate more energy efficient routing trees. Our rotation heuristic combines the node residual energy with a node consumption model. This estimates the node consumption energy by using node local information (i.e., transmission rate, hardware and initial node energy), cluster level information (i.e., energy for intra-cluster communication) and routing path information (i.e., energy for inter-cluster communication). While this information can suggest the best CH candidate and the best routing path, its combination with the node residual energy balance off situations where the model foreseen residual energy is lower than expected. Our approach actually defines the rotation and the routing amongst CHs, regardless the strategy chosen for the initial CHs election and cluster formation. Any clustering algorithm that produces a CH per cluster can be used. To this respect, our approach can be considered as a plug-in. In this paper we make use of REECHD [7] for CH election and cluster formation. As we are going to show, this choice is dictated by the performance of REECHD compared to other well-known approaches. We fairly compare REECHD enhanced with CER-CH with the state of art routing protocols. More precisely, we implement well-known approaches for routing information amongst CHs, we use various WSN settings and a widely accepted energy model. These settings are used to equally compare all protocols by considering the first node die lifetime measure. Our experiments show that CER-CH ensures a gain on average of 20% with respect to the state of art clustering protocols.

The rest of the paper is organised as follows: Section II describes the network and the energy models; Section III formalises two routing approaches which can be used to route data amongst CHs; Section IV recalls REECHD clustering protocol; Section V introduces our novel approach; Section VI describes the simulation settings and the collected results; Section VII discusses our simulation results; Section VIII reviews the state of art of routing and clustering for homogeneous and heterogeneous WSNs; finally, Section IX concludes the article and outlines future work.

II. NETWORK MODEL

We use a network operation model that has been adopted in various protocols such as LEACH [19], HEED [18], RUHEED [8], FMUC [14], and clearly in REECHD [7]. Although many different strategies can be adopted for clustering purposes (see, e.g. [20]–[22] and references there in), a clustering protocol usually includes the following phases: (i) cluster formation and CH election; (ii) CH routing tree definition; (iii) network operation phase; (iv) rotation (if any) in the role of CH within each cluster. Routing amongst CHs is defined in order to deliver data to the BS. Routing is usually achieved by defining a routing tree rooted at BS. During the network operation phase data gets delivered to the BS. A widely used data delivery model is based on the concept of TDMA. This is composed of the following two activities: (i) each member node of a cluster sends one variable size message to its CH; (ii) CHs route data to the BS via the defined routing tree. In other words, a TDMA starts from the collection of data from the member nodes and ends when all the data reach the BS. A round is composed of multiple TDMAs. After a round, CH election and cluster formation is repeated unless rotation is used. This replaces the CH election and cluster formation and allows each current CH to designate one of its members as new CH. Effectively, rotation does not modify any WSN cluster.

We define two types of WSN nodes referred to as homogeneous and heterogeneous nodes. All homogeneous nodes have the same amount of initial energy while heterogeneous nodes have a variable initial energy. This must fall within an interval. We define the heterogeneity level as the ratio between the number of the heterogeneous nodes and all WSN nodes. For instance, a heterogeneous level of 30% means that 30% of the WSN nodes are heterogeneous. Furthermore, all homogeneous nodes send data messages of the same size, whereas heterogeneous nodes can send data messages of different size.

A. RADIO MODEL

The adopted radio model utilises free space and multi path channel model [19]. Transceiver circuitry of a sensor node consumes $E_{elec} = 50nJ/bit$. Sensor node amplification energy E_a depends on the distance d between sender and receiver. When $d < d_0 = 75m$, E_a becomes $E_{fs} = 10pJ/bit/m^2$ (in this case a free space model is assumed) and when $d \geq d_0 = 75m$, E_a reduces to $E_{mf} = 0.0013pJ/bit/m^4$ (in this case a multipath model is assumed). Eq. 1 defines the transmission energy that is needed in order to send k bits at distance d while Eq. 2 defines the reception energy that is spent for receiving k bits. The exponent n is set to 2 for the

TABLE 1. Notation and definition.

Notation	Meaning
CR	a WSN cluster; $\{n_1, \dots, n_k\}$ are elements in CR
CR_{CH}	the WSN cluster that contains the cluster head CH
M_{CH}	the member nodes of the cluster defined by CH; $M_{CH} = CR_{CH} \setminus \{CH\}$ and $\{m_1, \dots, m_h\}$ are elements in M_{CH}
$T(n)$	the bits transmitted by a node n in a TDMA
$R(n)$	the bits received by a node n in a TDMA
E_{r_n}	the residual energy of a node n
E_{max_n}	the maximum energy of the node n (it is equal to a fully charged battery)
R_0	the transmission radius a CH uses to alert nodes of its presence. This defines the size of a cluster.
CH_f	the cluster head father of the cluster head CH in the routing tree
CDN_{CH}	the set of CH children of the cluster head CH in the routing tree
$\Re_{BS}$	the routing radius of the BS
$\Re_{CH}$	the routing radius of a CH

free space model and 4 for the multipath one.

$$E_{Tx}(k, d) = k(E_{elec} + E_a d^n) \tag{1}$$

$$E_{Rx}(k) = k(E_{elec}) \tag{2}$$

B. AGGREGATION

In this section we introduce the aggregation rate (AR) which is a number between 0 and 1 to calculate the amount of intra-traffic that is forwarded by a CH. In the rest of the paper we use the notation of Table 1. This describes the basic notation we use for clusters, nodes and routing tree.

We denote by $T(CR)$ the summation of the bits each node inside the cluster CR sends in a TDMA. This is refereed to as the total cluster rate and can be defined as follows:

$$T(CR) = \sum_{i=1}^{|CR|} T(n_i) \tag{3}$$

where $|CR|$ is the cardinality of the set CR, $T(n_i)$ the number of bits sent by the node n_i in a TDMA (in the following $T(n_i)$ will be referred to as node transmission rate).

The forwarded intra-traffic of a cluster head CH can be defined as follows:

$$T_{intra}(CH) = max(T(CR_{CH}) \times (1 - AR), M_{min}) \tag{4}$$

where AR is the aggregation rate and M_{min} is a constant that denotes the minimum amount of intra-traffic (i.e., bits) CH must forward in a TDMA. When $AR = 0$, CH packs all messages received by the members (during a TDMA) and forwards them to the next hop. In this case no aggregation takes place. When $AR = 1$, then CH aggregates all messages received by the members in a TDMA by producing a message of minimum size M_{min}. This minimum size can be set to the minimum rate of a node. In our model a CH can aggregate intra-traffic communications but not inter-traffic ones (i.e., a CH never aggregates the data received from other CHs).

III. ROUTING DATA AMONGST CLUSTER HEADS

In this section we present two strategies for building the routing tree amongst cluster heads.

Algorithm 1 Peer to Peer Routing

```
 1: procedure BS_p2p
 2:     $\Re_{BS} \leftarrow$ INITIAL VALUE
 3:     $BS_{nbr} \leftarrow \{CH : d_{BS,CH} < \Re_{BS}\}$
 4: end procedure
 5: procedure CH_p2p
 6:     $\Re_{CH} \leftarrow$ INITIAL VALUE
 7:     $step \leftarrow \Re_{CH} \times 0.2$
 8:     repeat
 9:         $CH_{nbr} \leftarrow \{CH_s : d_{CH_s,CH} < \Re_{CH}\}$
10:         $CH_f \leftarrow pick(CH_{nbr})$
11:         $\Re_{CH} \leftarrow \Re_{CH} + step$
12:     until $CH_f ==$ null $\wedge\ \Re_{CH} < \Re_{CH_{max}}$
13: end procedure
```

A. PEER TO PEER ROUTING ALGORITHM

Algorithm 1 shows the *BS_p2p* and *CH_p2p* procedures that are run by the BS and all CHs, respectively. These are used to build the routing tree in a peer to peer way. The *BS_p2p* is used by the *BS* to discover surrounding CHs. To this ending the routing radius $\Re_{BS}$ (line 2 of Algorithm 1) is initialised and the set BS_{nbr} is built (line 3 of Algorithm 1). This contains all CHs whose distance from the BS is less than $\Re_{BS}$. These CHs (if any) will always set the *BS* as father.

A CH uses procedure *CH_p2p* in order to find its father (this is denoted by CH_f) in the routing tree. To this ending a CH sets an initial value of the routing radius R_{CH} and a step value. The routing radius R_{CH} is used to discover surrounding *CHs* (if any). The step value is used to increase R_{CH} when CH finds no surrounding cluster heads. As we are going to see in the following, $\Re_{CH}$ needs to be carefully set in order to provide connectivity, generate an energy efficient routing tree and minimise the energy consumption during the routing tree definition. Procedure *CH_p2p* defines the set CH_{nbr} at line 9. This contains all cluster heads whose distance from CH is less than $\Re_{CH}$. A pick function selects a father from the set CH_{nbr}. When no father is found the routing radius is increased by a step quantity and the CH_{nbr} is defined again. Procedure *CH_p2p* terminates when the maximum transmission radius $\Re_{CH_{max}}$ is reached or a father is found. The definition of a suitable pick function is essential in order to ensure connectivity (i.e., all CHs can reach the BS) and to generate an energy efficient routing tree. In this respect we propose a $pick(CH_{nbr})$ function (running at the WSN node CH) that ensures the following three conditions:

$$(i)\ CH \neq CH_f$$

$$(ii)\ d_{BS,CH_f} < d_{BS,CH}$$

$$(iii)\ \forall CH_i \in father\_set,\ d_{BS,CH_f} \leq d_{BS,CH_i}$$

The condition (i) specifies that a cluster head cannot be selected as being its own father. The condition (ii) ensures connectivity since a node CH can always find a father closer to the BS or can directly communicate with the *BS*. The condition (iii) is used to break the tie when different cluster head fathers verify the condition (ii), i.e., more than one father that is closer to the BS is found. In this case our simulations have shown that choosing the father closer to the BS is the most energy efficient choice. This choice implies that a CH always connects to the BS if this is directly reachable.

It is worth mentioning that our pick function ensures connectivity (i.e., conditions (i)-(iii)) when the WSN is dense enough or a node can always expand its transmission radius to reach a suitable father.

B. TOP-DOWN ROUTING ALGORITHM

Algorithm 2 shows a top-down routing strategy. The BS runs the *BS_topdown* procedure that sets the routing radius $\Re_{BS}$ to an initial value and the layer L equal to zero. The BS broadcasts a [*Join_msg*,L+1] message which contains the layer L equal to one (line 4); all nodes reachable from the BS are considered layer one nodes.

Algorithm 2 Top-Down Routing

```
 1: procedure BS_TOPDOWN
 2:     $\Re_{BS} \leftarrow$ INITIAL VALUE
 3:     $L \leftarrow 0$
 4:     SEND($broadcast$, [$Join\_msg$, $L+1$])
 5: end procedure
 6: procedure CH_topdown
 7:     $\Re_{CH} \leftarrow$ INITIAL VALUE
 8:     $my\_level \leftarrow \infty$
 9:     $father\_set \leftarrow \emptyset$
10:     repeat
11:         RECEIVE($CH_s$, [$Join\_msg$, $L$])
12:         if $L < my\_level$ then
13:             $my\_level \leftarrow L$
14:             $father\_set \leftarrow \{CH_s\}$
15:             SEND($broadcast$, [$Join\_msg$, $L+1$])
16:         end if
17:         if L==my_level then
18:             $father\_set \leftarrow father\_set \bigcup \{CH_s\}$
19:         end if
20:     until time_out_not_expired
21:     if father_set!= $\emptyset$ then
22:         $CH_f \leftarrow pick(father\_set)$
23:     else
24:         CH_p2p( )
25:     end if
26: end procedure
```

A cluster head runs the *CH_topdown* function. This sets the routing radius $\Re_{CH}$, the *father_set* to empty and the layer *my_level* to infinity. The *father_set* contains all nodes the CH can select as father while L is the length of the shortest path to reach the BS. A CH processes a [*Join_msg*, L] message sent by a cluster head CH_s according to the following strategy:

- When the CH receives a level L from CH_s that is smaller than the CH level *my_level*, the *father_set* is updated with CH_s, *my_level* is updated to L and the message (*Join_msg*, L_{i+1}) is forwarded. A shorter path to the BS has been found.
- When L is equal to the CH level *my_level*. The cluster head CH_s is added to the *father_set*.

The father can be picked with different strategies (these are implemented by the pick function of Algorithm 2, line 22). A CH performs the *CH_p2p* procedure of Algorithm 1 when it does not receive any (*Join_msg*, L) message (see Algorithm 2, line 24). This procedure is used by uncovered CHs to expand their routing radius and find a father. More precisely, our top-down routing has an hybrid approach: it builds the routing tree starting from the BS down to the CHs but uses a peer-to-peer approach for uncovered nodes.

When nodes receive and forward the *Join_msg* they are considered covered while uncovered nodes run the p2p routing. This requires the implementation of a neighbour discovery protocol which also needs acknowledge messages. As it will be shown by the experiments, the p2p approach would then consume more energy.

C. GRID BASED ROUTING

Quite often the WSN area is organised in virtual layers [14], [15] or in a virtual grid [11]–[13]. In these cases each cluster has nodes that belong to the same layer or grid. Such approaches are used in order to balance the inter-traffic communication and ease the routing. Figure 6 shows a WSN clustering where virtual layers are defined as opposed to Figure 5 where no virtual layers are defined.

IV. RECALLING REECHD

In this paper we make use of the Rotating Energy Efficient Clustering for Heterogeneous Devices (REECHD) clustering protocol in order to form the initial clustering. Any clustering algorithm that generates clusters with exactly one cluster head can be easily extended with our approach. More precisely, after an initial set of clusters are created our strategy can be used in order to rotate the CH role and build the CH routing tree, at the same time. We chose REECHD since it is an efficient clustering protocol that introduces a novel technique for the CH election which, not only considers the node's residual energy, but also its transmission rate. Eq. 5 shows the node probability of becoming CH in REECHD. Parameter K is simply needed to confine CH_{prob} between 0 and 1 (so that it represents a probability). A standard setting is $K = 2$. Value C_{prob} is a predefined initial probability (e.g., 5%) that sets the initial percentage of cluster heads amongst all WSN nodes. E_{rCH} is the residual energy of the node, E_{maxCH} is the maximum energy of the node (it is equal to a fully charged battery), $T(CH)$ is the transmission rate of the node, T_{max} is the highest transmission rate of the WSN (it corresponds to the rate of the node which has the highest transmission rate in the WSN). CH_{prob} value of a node is not allowed to fall below a certain threshold P_{min} (e.g., 10^{-4}), that is selected to be inversely proportional to E_{max}.

$$CH_{prob} = max\left(\frac{C_{prob}}{K}\left(\frac{E_{rCH}}{E_{maxCH}} + \frac{T(CH)}{T_{max}}\right), P_{min}\right) \quad (5)$$

REECHD is composed of the following phases: (i) cluster head election; (i) cluster formation; (iii) CH routing tree definition; (iv) network operation; and (v) rotation. Each node becomes CH according to the probability that is defined in Eq. 5. In the cluster formation phase each node attempts to join the least cost CH in order to form clusters. In the CH routing tree definition phase the routing amongst CH is defined. REECHD uses a top-down strategy to form the routing tree. Finally, the network operation phase can start. During this phase five TDMAs (i.e., one round) are performed. In a TDMA each CH gathers data from its member nodes and cooperate with other CHs in order to deliver them to the BS. After one round the rotation phase is triggered in order to elect new CHs. The current CH designates the next CH directly by using Eq. 5 as a weight function. More precisely the current CH calculates the quantity CH_{prob} of each member node and chooses the one with the highest CH_{prob} as the next CH. After each rotation, the CH routing tree is re-defined from scratch and the network operation phase can start again.

V. CER-CH: COMBINING CH ROTATION AND CH ROUTING TREE FORMATION

Most of the clustering protocols are composed of the following phases: cluster head election (*CH_Elect*), cluster formation (*Clustering*), routing tree definition (*RoutingTree*), network operational phase (*Networking*), and rotation amongst CHs (*Rotation*), if any. These activities can be performed cyclically until the WSN depletes its energy. In particular, the routing tree definition always follows the cluster head election. Figure 1 outlines the novelty of CER-CH and its basic building blocks. An initial cluster head election and cluster formation must be performed. As already pointed out, any algorithm for CH election and cluster formation can be used as long as the algorithm produces one CH per cluster. After cluster formation, a routing tree definition phase is performed such as the peer to peer routing of Algorithm 1. The clustering and the routing tree are taken as input to the CER-CH rotation and routing definition components of Figure 1. This performs an initial network operational run (e.g., five TDMAs in our case) and then combine the routing tree definition with the CH rotation phase (i.e., CH election). This combination allows the CH election and the routing tree definition to influence each other in order to elect CHs that produce an energy efficient routing tree. Rotation combined with routing tree definition plus network operational phases are performed in alternation until the first node dies. Although we change the CH at each rotation and we update the routing three amongst CHs, each CH will route information through the same cluster. In other words, if we consider the routing tree from the cluster prospective it never changes. Note that, the initial clustering greatly affects the energy performance of our approach but, as we are going to discuss in the performance evaluation section, in the average case we have a gain of 20% with respect to the state of art clustering.

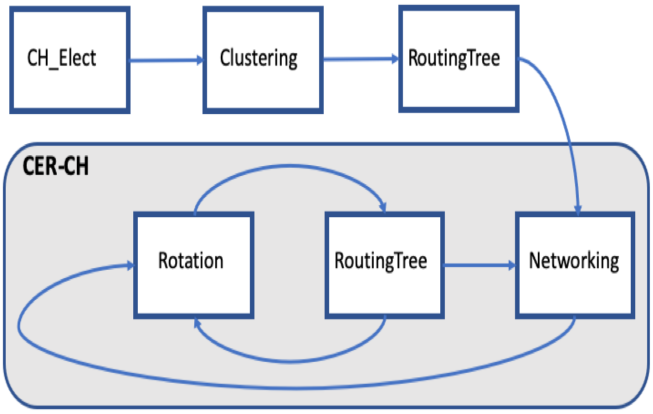

FIGURE 1. CER-CH main blocks.

When the CER-CH rotation and routing tree definition takes place the BS starts the procedure $BS\_CER-CH\_rotation$ of Algorithm 3. This multicasts a rotate message $[rotate, CH_f]$ to all nodes in the CDN_{BS} set. This contains all CH children of BS in the routing tree.

A cluster head runs the procedure $CH\_CER-CH\_rotation$ when it receives the $[rotate, CH_f]$ message from the old father CH_{old} (line 8 of Algorithm 3). This message contains the new father CH_f that is selected by the previous father CH_{old}. The current cluster head CH designates as the new CH the node n that maximises a function *weight*. This can take as an input a node n and various inputs such as the features of the father CH_f and the cluster node set CR_{CH}. In the next section we describe the implementation of a weight function. The new cluster head is saved in the $next\_CH$ variable and the old CH sends a message to every cluster member to indicate $next\_CH$ as the new CH. Finally, the $next\_CH$ is communicated to all children of CH in the routing tree.

We emphasise that the *weight* function is used by a CH to build the routing tree and elect a CH at the same time. In contrast, a CH uses the *pick* function, which supports the implementation of the traditional approaches (see Algorithm 1 and Algorithm 2 for details), for selecting an already existing cluster head as father. Effectively, *pick* only builds the routing tree.

A. CER-CH WEIGHT FUNCTION

In this section we describe the CER-CH weight function that is used in order to implement the CH role rotation. This function requires some definitions and notation that we introduce in the following.

We use $R_{intra}(CH)$ to denote the total amount of bits received by the cluster head CH from its member nodes in a TDMA (i.e., the intra-traffic data). $R_{intra}(CH)$ can be

Algorithm 3 CER-CH Routing

```
procedure BS_CER-CH_ROTATION
    $CH_f \leftarrow BS$
    SEND($CDN_{BS}$, [$rotate$, $CH_f$])
end procedure
procedure CH_CER-CH_ROTATION
    $max \leftarrow 0$
    $next\_ch \leftarrow 0$
    RECEIVE($CH_{old}$, [$rotate$, $CH_f$])
    for each node n inside cluster $CR_{CH}$ do
        if $max < weight(n, CR_{CH}, CH_f)$ then
            $next\_CH \leftarrow n$
            $max \leftarrow weight(n, CR_{CH}, CH_f)$
        end if
    end for
    $CH_f \leftarrow next\_CH$
    SEND($CR_{CH}$, $members$, $CH_f$)
    SEND($CDN_{CH}$, [$rotate$, $CH_f$])
end procedure
```

calculated as follows:

$$R_{intra}(CH) = \sum_{i=1}^{|M_{CH}|} (T_{m_i}) \tag{6}$$

where T_{m_i} is the transmission rate of the member node m_i in M_{CH}.

We use $R_{inter}(CH)$ to denote the total amount of bits received by CH from its children CH nodes in a TDMA (i.e., the inter-traffic received data). $R_{inter}(CH)$ can be calculated as follows:

$$R_{inter}(CH) = \sum_{i=1}^{|subtree_{CH}|} (T_{intra}(CH_i)) \tag{7}$$

where $T_{intra}(CH_i)$ is the intra-traffic that CH_i forwards to its father (see Eq. 4 for details), a $subtree_{CH}$ contains all cluster heads that are in the sub-tree of the routing tree rooted at CH. Effectively, $subtree_{CH}$ contains all cluster heads that forward inter-traffic to CH. We recall that inter-traffic is not aggregated.

We can now calculate the energy $E_{intra}(CH)$ a CH spends to receive messages from its member nodes. $E_{intra}(CH)$ can be defined by the following equation:

$$E_{intra}(CH) = E_{Rx}(R_{intra}(CH)) \tag{8}$$

where $R_{intra}(CH)$ is the intra-traffic that has been defined in Eq. 6. The energy CH spends to forward data to its father (i.e., the inter-traffic transmission rate) is denoted with $E_{inter}(CH, CH_f)$ and can be calculated as follows:

$$\begin{aligned} E_{inter}(CH, CH_f) = {} & E_{Tx}(T_{intra}(CH), d_{CH,CH_f}) \\ & + E_{Rx}(R_{inter}(CH)) \\ & + E_{Tx}(R_{inter}(CH), d_{CH,CH_f}) \end{aligned} \tag{9}$$

where d_{CH,CH_f} is the distance between the CH and its father CH_f, the first adding is the energy CH spends for forwarding its intra-traffic to the its father, the second adding is the energy CH spends for receiving data from its children nodes while the last adding is the energy CH spends to send the data that are received from the children nodes to the father node.

The energy a node spends playing the CH role in a TDMA can be written as:

$$E_{TDMA}(CH, CH_f) = E_{inter}(CH, CH_f) + E_{intra}(CH) \tag{10}$$

The selected cluster head should reduce the E_{TDMA} energy consumption.

As we are going to discuss in the next section, a good estimate for the intra communication energy consumption E_{intra} can be provided while it is extremely difficult to estimate the inter-traffic energy communication E_{inter}. This estimate requires to know the amount of traffic forwarded by the children nodes. This can be unknown since routing tree definition can occur after the cluster head election. Our approach overcomes this problem by combining the routing tree definition and the CH election together.

$$CER - CH(CH, CH_f) = \frac{E_{rCH}}{E_{TDMA}(CH, CH_f)} \tag{11}$$

Eq. 11 describes the weight function that is used by CER-CH. This estimates the number of times a node can play the cluster head role. This is obtained by dividing the residual energy of the candidate node CH by the energy that CH spends in a TDMA.

B. WEIGHT FUNCTION ESTIMATE

In this section we show how CER-CH routing and rotation ease the estimate of the weight function CCE. This requires to estimate $E_{intra}(CH)$ and $E_{inter}(CH, CH_f)$ for the CH candidate node.

The intra-traffic energy $E_{intra}(CH)$ can be easily estimated by using Eq. 8. This only requires to know the transmission rate of the clustering nodes.

The inter-traffic communication $E_{inter}(CH, CH_f)$ is composed of the following addition:

1) $E_{Tx}(T_{intra}(CH), d_{CH,CH_f})$;
2) $E_{Rx}(R_{inter}(CH))$;
3) $E_{Tx}(R_{inter}(CH), d_{CH,CH_f})$

The energy (1) that is spent for forwarding the intra-traffic data T_{intra} to the father CH_f can be easily estimated by using Eq. 4. This requires to know the father of CH. Our rotation approach ensures the knowledge of the father since the cluster head election proceeds top-down. The energies (2) and (3) are difficult to estimate since the children nodes of CH have not been elected thus the received inter-traffic $R_{inter}(CH)$ is unknown. In our novel rotation Algorithm 3 we can prove the received inter-traffic $R_{inter}(CH)$ is constant throughout the rotation phase. More precisely, our rotation can change the CHs but clusters do not change. At each rotation the new CH always forwards data via the same cluster. Thus $R_{inter}(CH)$ (i.e., $T_{intra}(CH)$) is constant, no matter the CH which is designated during the rotation phase.

VI. SIMULATION SETTINGS AND RESULTS

In this section we describe our simulation settings, the routing radius selection and the results of our novel routing strategy.

A. SIMULATION SETTINGS

We assume nodes are uniformly distributed in a two dimensional area. Heterogeneous nodes have different data transmission rates and different initial energy within a defined range, while homogeneous nodes have equal initial energy and data transmission rates. More precisely, our simulation assumes homogeneous nodes have an initial energy of 1 joule and send messages of 3000 bits per TDMA. A heterogeneous node has an initial energy that falls within the interval $[1, 4]$ joules and sends messages of a size that falls within the interval $[500, 3000]$ bits. This message size is drawn in the first setup phase and then is kept constant throughout the entire WSN simulation. We set the heterogeneity (i.e., the percentage of heterogeneous nodes) to 50% and 100%. We only consider an aggregation rate of 100% since for lower values we have experienced simulation results in favour to our approach. Nodes have the same processing and aggregation capability, have a unique IDs and can transmit at various power levels which depend on the distance of the receiver.

The BS has no energy constraints and is located outside the WSN area. The BS has more communication and processing capabilities with respect to normal sensor nodes. Each CH can aggregate the intra-traffic data in order to reduce the amount of bits that are forwarded to the BS. Inter-traffic is not aggregated that is a CH forwards (towards the BS) messages received from its CH children with no aggregation.

We use a simulator that we have developed and validated in [8], [9], [23]. This simulator uses the energy model that has been presented in Section II and is optimised for parallel simulations. When compared with general purpose simulators such as OMNeT++ and NS2, it is up to ten times faster.

Each simulation we report in the next figures is the average result of various runs. The number of runs ensures the confidence interval is within 5%.

B. SELECTION OF THE INITIAL ROUTING VALUES $\Re_{BS}$ and $\Re_{CH}$

The initial routing radius value $\Re_{CH}$ (in the following referred to as v_{CH}) should be carefully set in order to ensure connectivity with few control messages and generate an energy efficient routing tree. Small values of v_{CH} can cause the following two problems: (i) high energy consumption for finding surrounding *CHs*; and (ii) inefficient routing tree generation. More precisely, the cluster head *CH* can fail to find surrounding cluster heads when v_{CH} is too small. In this case *CH* keeps sending control messages with an increasing $\Re_{CH}$ until at least a surrounding father is found. This causes a waste of energy and the selection of fathers that are closer to *CH*. This choice can generate routing paths with lots of hops, i.e., routing tree with lots of short connection links. Although short links require less energy for sending data, a high number of links can increase the inter-traffic communication. Large values of v_{CH} allow the *CH* to find a surrounding father with the first broadcast but can generate routing paths with few hops and links with long distance. More precisely, large values of v_{CH} can generate routing tree with a small number of long connection links. Although this keeps the inter-traffic communication low, CHs can require a lot of energy in order to send data over long distances.

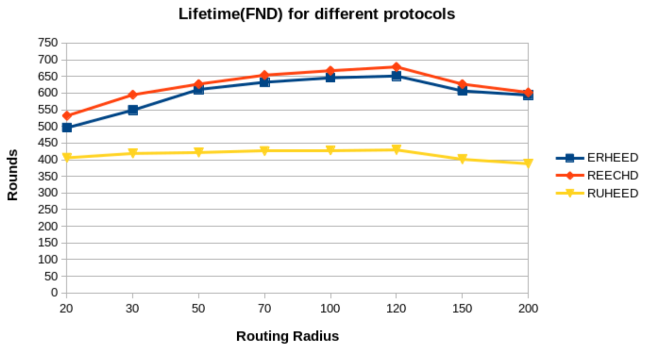

FIGURE 2. WSN 100 nodes, WSN grid 200x200, BS(275,50) Heterogeneity 100%, Aggregation 100%, Routing p2p, $R_0 = 70$.

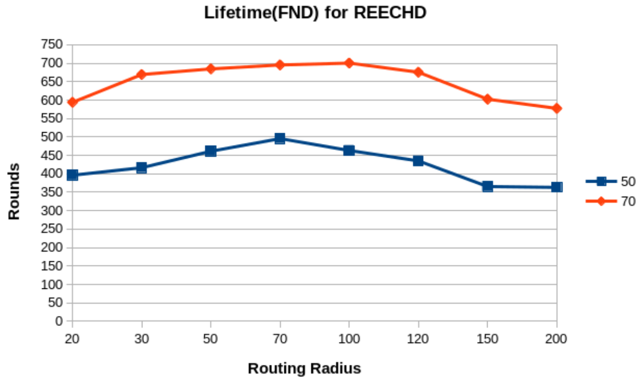

FIGURE 3. WSN 100 nodes, WSN grid 200x200, BS(275,50) Heterogeneity 100%, Aggregation 100%, Top-Down routing, $R_0 = 70$.

Figures 2 and 3 show on the X-axis the routing initial values and on the Y-axis the FND measure. Such results have been obtained by using the following simulation parameters:

- uniformly distributed deployment of 100 nodes over a WSN square area of 200 by 200 meters;
- BS located at position (275,50);
- both heterogeneity and aggregation rate at 100%;
- control parameter c for UHEED and RUHEED equal to 0.5 (see [8] for the definition of c);
- REECHD ITLR parameter equal to 100% (see [7] for the definition of ITLR)

The choice of considering an area of 200 by 200 meters rather than 100 by 100 (which is widely used in literature) allows us to better highlight the effect of the initial routing value. More precisely, as the network gets larger, the routing tree gets deeper and the effect of a badly chosen initial routing value becomes more visible. Figure 2 shows the FND for the p2p

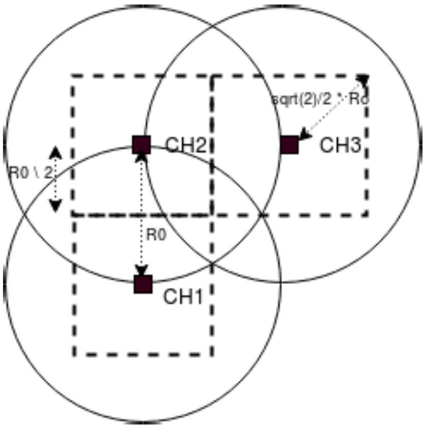

FIGURE 4. A network model where the clusters form a grid with cells of dimensionR_0 by R_0.

routing protocol where the competition radius is set to $R_0 = 70$. The FND measure is evaluated for the ER-HEED [9], REECHD [7] and RUHEED [8] protocols. REECHD uses the heuristic that is presented in [8] which is summarised by Eq. 5. Figure 3 shows the FND for the REECHD protocol for the top-down routing protocol where the competition radius is set to $R_0 = 70$ and $R_0 = 50$ (red and blue lines respectively). For each routing initial value we calculate the number of rounds it takes for the first node to die. Each value is obtained by averaging various simulation runs until the confidence interval is within 5%.

For all experiments and all routing protocols small values of v_{CH} produce a decrease in performance up to 20% when compared to the most energy efficient initial routing value. This same behaviour can be observed when the initial value of v_{CH} is large. We can also observe that the unequal protocol RUHEED is less affected by the initial value v_{CH}. This is a consequence of the RUHEED clustering strategy that has a dense amount of small clusters next to the BS. In this case small values of v_{CH} still allow to find a father with no extra messages.

We have performed an extensive set of experiments for different WSN settings that show $\Re_{CH} = 1.6 \times R_0$ to be the most energy efficient routing initial value. This initial value allows a CH to find a father with one broadcast message (for both p2p and the top-down routing) and to build an energy efficient routing tree. This can be easily proved for the equal variations of the HEED (e.g., ER-HEED and REECHD) which produce clusters of equal radius R_0. HEED prevents two nodes within the same transmission range from becoming CHs. Thus, the distance between two CHs must be greater than R_0. Figure 4 shows a network model where the clusters form a grid with cells R_0 by R_0. In this case a routing radius $\Re^{min}_{CH}$ equal to $\sqrt{2} \times R_0$ would allow a CH to discover all surrounding cluster heads. During the rotation phase a communication radius of $2\sqrt{2} \times R_0$ is the longest distance to be covered amongst two communicating CHs.

$$\Re^2_{CH} \times \pi \times \delta_{CH} \geq 2 \tag{12}$$

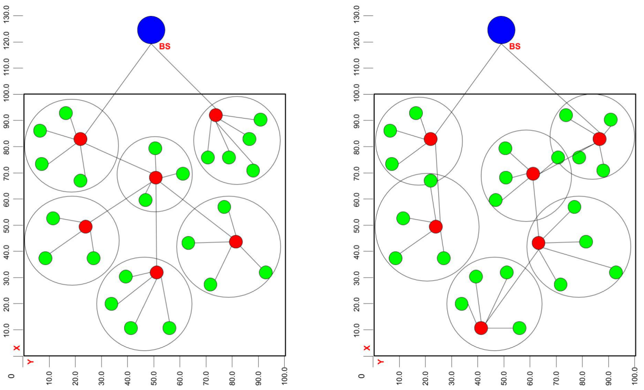

FIGURE 5. A WSN clustering without virtual layers. After the rotation phase, some CH can route information via a different cluster.

Eq. 12 can be also used to approximate the initial value of the routing radius $\Re_{CH}$ where δ_{CH} is the cluster head density. This equation ensures the initial routing radius always intercept a surrounding cluster head. This approach could be not viable since the density function δ_{CH} can be difficult to estimate. In fact, cluster heads are not always uniformly distributed since δ_{CH} could depend on the position. This is the case of unequal clustering protocols such as RUHEED [8] where regions that are closer to the BS have more clusters when compared to regions located farther away. The density δ_{CH} can be also difficult to estimate for protocols that produce equal-sized clusters since different protocols can produce different overlapping of clusters or the same cluster can have more cluster heads.

The radius $\Re_{BS}$ determines the percentage of cluster heads that are directly connected to the BS. We have performed an extensive set of simulations in order to verify that in our WSN setting the value of $\Re_{BS}$ does not significantly affect the lifetime. Suppose that the *BS* does not reach any surrounding CHs. Then CHs will be still able to reach the BS with respect to both routing Algorithms 1 and 2.

C. ROUTING PROTOCOL SIMULATION

We have performed the following two set of experiments: (i) simulations of the state of art protocols; (ii) simulations of our novel approach. Simulations in (i) are used to validate our simulator and to get the most energy efficient competitor of our approach. Simulations in (ii) are used to compare the energy efficiency of our approach with the most efficient competitor. All simulation results have been obtained by using the following parameters:

- a uniformly distributed deployment of 100 nodes over a WSN square area of 100 by 100 meters;
- a BS that is located at position (175,50);
- an heterogeneity and aggregation rate of 100% and 50%;
- a control parameter c for UHEED and RUHEED equal to 0.5 (see [8] for the definition of c);
- a REECHD ITLR parameter equal to 100% (see [7] for the definition of ITLR)
- each round is composed of 5 TDMAs

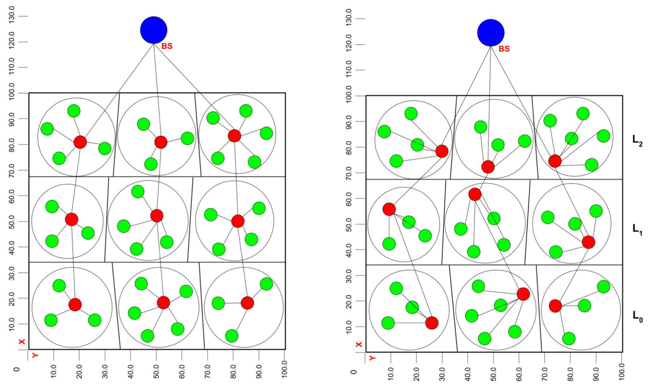

FIGURE 6. A WSN clustering with virtual layers. After each rotation phase, the routing tree between the clusters is unchanged.

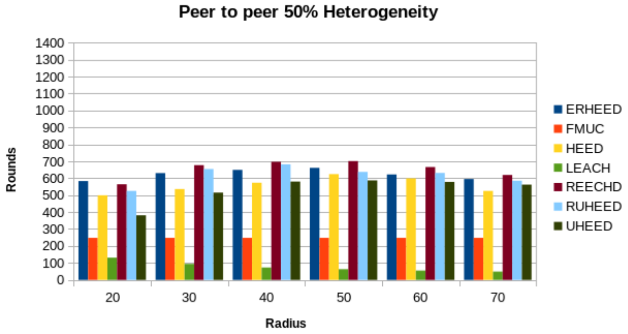

FIGURE 7. Simulation of several clustering protocols with Peer2Peer routing. Aggregation is 100%, Heterogeneity level is 50% and BS Position is at (175,50).

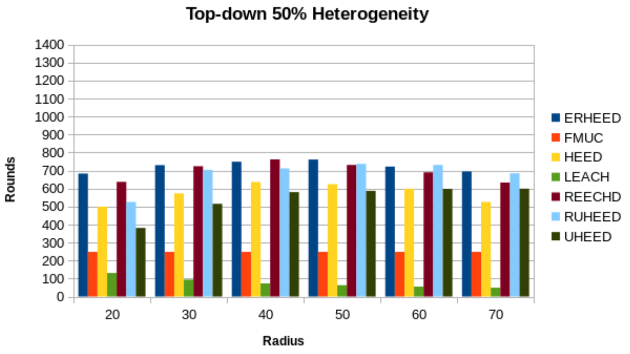

FIGURE 8. Simulation of several clustering protocols with Top-Down routing. Aggregation is 100%, Heterogeneity level is 50% and BS Position is at (175,50).

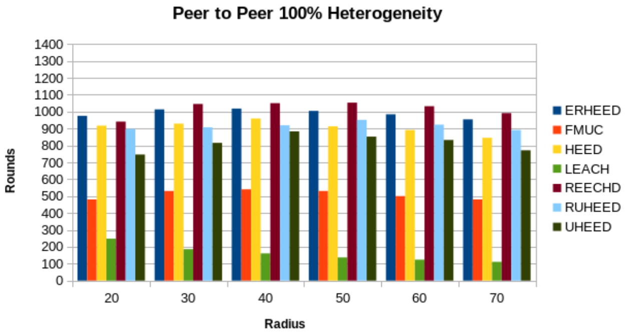

FIGURE 9. Simulation of several clustering protocols with Peer2Peer routing. Aggregation is 100%, Heterogeneity level is 100% and BS position is at (175,50).

We emphasise that the WSN square area of 100 by 100 meters is widely used in literature. This setting penalises our approach since the generated routing tree are usually deep at most three. Higher tree would better show the efficiency of our tree definition.

1) STATE OF ART PROTOCOL SIMULATION: THE EFFECTS OF P2P AND TOP-DOWN APPROACHES

Figures 7, 8, 9 and 10 show the simulation of the state of art protocols that are ER-HEED [9], HEED [18], REECHD [7], RUHEED [8] and UHEED [23], FMUC [14] and M-LEACH [24] (i.e., multi-hop LEACH). These figures have on the X-axis the competition radius R_0 which varies from 20 meters to 70 meters and on the Y-axis the FND measure. Figures 7, and 8 show the result for the p2p and top-down routing protocols of Algorithms 1 and 2 when the heterogeneity and aggregation levels are set to 50% and 100%, respectively. We can notice that REECHD and ERHEED have the best lifetime performance. More precisely, REECHD performance is sightly better than ERHEED and the best lifetime for both protocols is obtained at competition radius $R_0 = 40$. Both p2p and top-down routing protocols have a similar performance. Figures 9 and 10 show the result for the p2p and top-down routing protocols, respectively, when both the heterogeneity and aggregation levels are set to 100%. Again, we can notice that REECHD and ERHEED have the best lifetime performance. More precisely, REECHD performance is sightly better than ERHEED and the best lifetime for both protocols is obtained at competition radius $R_0 = 40$. We can conclude that REECHD is the most energy efficient protocol in our WSN settings (this is confirmed from the relevant research literature [7]). We can also notice that the top-down routing performs better than the p2p. Hence, in what follows we consider REECHD with its top-down implementation as competitor of our CER-CH where REECHD is used to set the initial set of clusters. In this way we show how our new approach can improve on the performance of a protocol which has been shown to be better than any other state of art protocol.

2) CER-CH: ENHANCING REECHD WITH TOP-DOWN ROUTING COMBINED WITH ROTATION

In this section we compare REECHD [7] with our novel rotation approach. The REECHD simulation makes use of the top-down routing and uses the weight function that is defined by the Eq. 5. CER-CH uses REECHD for the initial clustering, the novel top-down routing combined with rotation approach of Algorithm 3 and the Eq. 11 for rotation.

Figures 11 and 12 show our simulation results. The X-axis has the competition radius R_0 which varies from 20 meters to 70 meters while the Y-axis the FND measure. We can notice

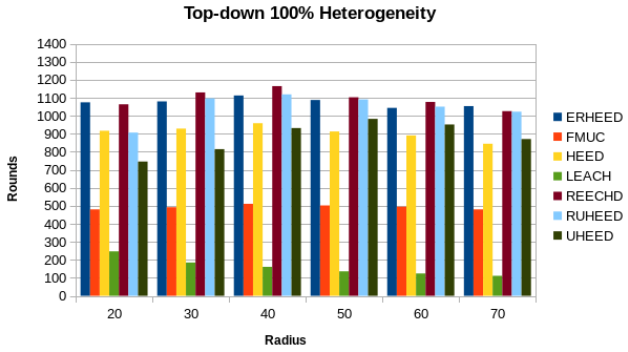

FIGURE 10. Simulation of several clustering protocols with top down routing. Aggregation is 100%, Heterogeneity level is 100% and BS Position is at (175,50).

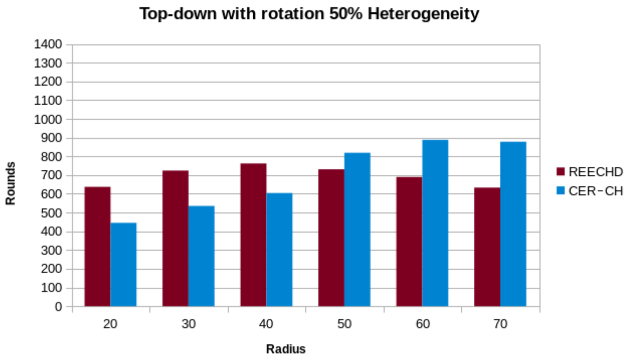

FIGURE 11. Simulation of several clustering protocols with top-down with rotation routing. Aggregation is 100%, Heterogeneity level is 50% and BS position is at (175,50).

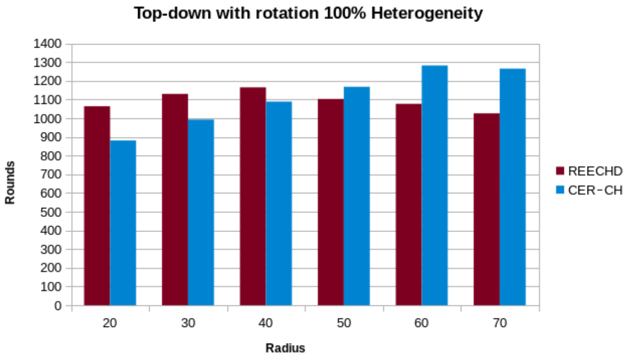

FIGURE 12. Simulation of several clustering protocols with top-down with rotation routing. Aggregation is 100%, Heterogeneity level is 100% and BS position is at (175,50).

that our CER-CH always outperforms the REECHD protocol. The lifetime performance has an average gain of 20% when CER-CH is compared with REECHD. We can also notice that as the heterogeneity level increases the lifetime of both REECHD and CER-CH increases. This is consequence of the higher energy that heterogeneous nodes have when compared to homogeneous ones.

VII. DISCUSSION

Although CER-CH starts with the same set of clusters dictated by REECHD, it still has significant performance differences with respect to REECHD. The obtained improvements can be explained by considering their different rotation and routing strategies. REECHD CH rotation: (i) uses node rate and node residual energy; (ii) performs routing by selecting the closest father. Figure 5 shows an example of the REECHD routing strategy where (after the rotation) some CHs route information via different clusters. CER-CH rotation: (i) considers node rate, node residual energy and the energy cost for communicating with an already selected father (i.e., local routing path cost); (ii) CHs of a cluster *CR* always select a father from the same cluster (i.e., routing amongst clusters does not change). Figure 6 shows an example of CER-CH routing strategy where (after rotation) the routing tree between clusters is unchanged.

CER-CH rotation has some advantages over REECHD one. During CER-CH rotation only multicast messages flow from an old CH to its children while REECHED requires the top-down routing algorithm of Figure 2 to be performed from scratch. CER-CH rotation considers the cost for inter traffic communication while REECHD does not. This allows the selection of a CH that reduces the inter-traffic energy cost.

VIII. RELATED WORKS

Our related work mainly focuses on rotation techniques (i.e., model based and adaptive ones) and clustering protocols which aim at producing an energy efficient CH routing tree.

A. ROTATION

Rotation has been widely proposed in literature in order to balance the energy consumption and prolong the WSN lifetime. The authors in [5] propose an energy efficient protocol consisting of clustering, cluster head selection/rotation and data routing method to prolong the lifetime of sensor networks. The WSN area is organised in concentric circles. Nodes that are exactly midway between the two concentric circles have a higher probability of becoming CH in the first election. Clusters are formed only once during the lifetime of the sensor network, then rotation takes place. This results in substantial saving of energy. Rotation takes place after a fixed amount of rounds which is calculated by using a mathematical energy model. This considers the average energy consumption for a node playing the CH and member roles. The authors in [6] propose an Area-Partitioned Clustering where the network is statically divided in concentric circles. Rotation is used to provide a balanced consumption of energy. Three rotation strategies are proposed where rotation is based on a predetermined schedule or residual energy is used when the energy consumption is lower than expected. While a rotation based on a predetermined schedule may reduce overhead messages, energy consumption models usually estimate the energy consumption in average conditions (e.g., average distance between a member and its CH) and simplified settings (e.g., virtual grids). This may lead to lifetime performance degradation when the average case is not representative. This problem is solved by our adaptive rotation approach which incorporates node residual energy inside our energy consumption model.

Rotation is usually adaptive since considers the node residual energy [8]–[10]; few approaches may add other node features such as the node rate and the node initial energy [7]. In [9] the authors (including some of us) extend the HEED [18] protocol with rotation. They use HEED for the cluster formation thae rotation based on residual energy takes place. HEED cluster formation is performed every time a node depletes its energy completely. In [8] the authors (including some of us) extend the UHEED [23] protocol with rotation. They use UHEED for the clustering the WSN, then rotation based on residual energy takes place. UHEED clustering is performed every time a node dies. In [10] the authors use the CH current energy load to calculate a threshold for rotation. This reduces the premature death of cluster heads. A splitting policy divides the cluster when no cluster member can afford to be CH. Although rotation based on residual energy is suitable for a wide range of WSN settings, it may be inefficient for heterogeneous networks. The authors in [22] propose the Energy-Coverage Ratio Clustering Protocol (E-CRCP) for heterogeneous energy wireless sensor networks. They define an energy consumption model and an optimal number of clusters in order to minimise energy consumption. CHs that maximise the coverage are selected. CH nodes that consume a large amount of energy are replaced in the next iteration. Members join their nearest CH. Although this approach improves energy efficiency for various WSN simulation settings it does not consider node rate heterogeneity.

B. BUILDING AN ENERGY EFFICIENT CH ROUTING TREE

The problem of building a CH routing tree that ensures connectivity is widely studied. However, very few approaches consider the energy spent for the routing tree generation and the energy efficiency of the generated routing tree. In [25], the authors perform a field study on existing routing techniques applied to WSN, highlighting the performance issues of each technique. They classify the routing techniques into the following three categories: flat, hierarchical, and location-based. Moreover, depending on the protocol operation they include, further classifications as multipath-based, query-based, negotiation-based, QoS-based, and coherent-based routing have been studied. A similar but more up-to-date work has been presented in [26]. In addition, they explore in details possible optimisation of existing routing algorithms as well as the unsolved issues and research gaps. An even more recent work [27] focuses on the factors that affect the energy aware routing in WSNs. In particular, the authors suggest various approaches to make existing routing techniques energy aware. In a recent work [28], the authors propose a Tree-Based Energy-Balance Routing in which each node selects its father amongst its neighbours on the basis of the communication distance between sensor nodes and the BS, the nodes residual energy level, the energy required to transmit the data to the BS, and the number of associated child nodes. The authors believe that this leads to a uniform energy utilisation and offers a better energy balance mechanism with respect to other general purpose routing algorithms. In this paper, we focus on some existing clustering protocols to inspect how they cope with connectivity and energy issues and we propose a novel energy efficient routing approach which considers network topology and nodes heterogeneity.

HEED [18] and UHEED [23] build the routing tree between CHs in a peer to peer fashion. Each node sends a broadcast to search for neighbour CHs and connects to them. This approach only guarantees connectivity if CHs can increase their radius enough to reach the BS, and it doesn't allow to build the best possible routing tree. Other clustering protocols which implement rotation, like ER-HEED [9] and RUHEED [8], try to build a routing tree in a centralised way, starting from the BS. The BS sends a broadcast to find at least one CH, which will do the same to find neighbour CHs. This implies a lot of communication between the nodes. The only way to check for connectivity is that every CH sends a control message and waits for an ack from the BS (through the upper-level CHs). However, this synchronisation phase adds extra control messages. Finally, there are protocols like FMUC [14] which create clusters only between nodes at the same level. In this case, the only way to avoid the synchronisation phase is to choose the levels height accordingly to the nodes competition radius. Without this assumption, nodes would still need to find a way to understand if they are connected to the BS.

The authors in [21] survey the state of art clustering for Computational Intelligence (CI) and Machine Learning (ML). They compare the clusterings by using various parameters such as the data delivery rate; data aggregation; network lifetime; the scalability for an increasing number of nodes; centralised and distributed clustering; homogeneous and heterogeneous nodes (i.e., whether sensors have or not the same performance); the energy model and fault tolerance. Algorithms that are based on Swarm Intelligence (SI) seem the most energy efficient choice when artificial intelligence is considered. The authors in [20] provide a comparative analysis for sensor node deployment schemes and energy efficient clustering protocols. The state of art energy efficient cluster-based and grid-based techniques in WSN are evaluated by considering different parameters such as cluster formation metric, energy consumption, and lifetime. The authors also discuss the design issues and open research challenges. A comparative analysis is presented that helps in selecting the most appropriate technique for specific requirements. Although the surveys in [20] and [21] present a plethora of different clustering approaches, none of them seems to combine CH election and routing tree definition when heterogeneity in terms of rate and residual energy are considered.

IX. CONCLUSION AND FUTURE WORK

This paper proposes CER-CH that is a novel approach where the CH routing tree definition and the CH rotation are combined together. More precisely, a novel rotation heuristic is combined with a novel top-down CH routing tree definition in order to balance the node energy consumption and generate more energy efficient routing trees. Actually CER-CH can be considered as a plug-in defining the rotation and the routing amongst CHs, regardless the strategy chosen for the initial CHs election and cluster formation. Any clustering algorithm that produces a CH per cluster can be used.

In order to evaluate the performance of our approach we formalise well-known p2p and top-down routing algorithms that allow the definition of the CHs routing tree. We observed that the value of the routing radius a CH uses to find reachable fathers and the pick strategy to select a father (when different fathers are available) can heavily affect the WSN lifetime. We have set the best routing parameters and found the best competitor for our CER-CH that is REECHD. Finally, we have compared CER-CH and REECHD for various WSN settings and verified that our novel approach ensures an average gain of 20%.

As future work we plan to apply our rotation approach to other clustering algorithms, e.g. FMUC [14] or DEEC [29], in order to evaluate whether the obtained enhancement induced by CER-CH might be even more relevant with respect to the 20% obtained for REECHD. We are also planning to use our strategy for 5G device-to-device communications.

Finally, we plan to perform some experimental evaluation on real case studies. This can give interesting insights on the CER-CH energy performance.

REFERENCES

[1] C. Vannucchi, M. Diamanti, G. Mazzante, D. R. Cacciagrano, F. Corradini, R. Culmone, N. Gorogiannis, L. Mostarda, and F. Raimondi, “vIRONY: A tool for analysis and verification of ECA rules in intelligent environments,” in *Proc. Int. Conf. Intell. Environ. (IE)*, Aug. 2017, pp. 92–99.

[2] F. Corradini, R. Culmone, L. Mostarda, L. Tesei, and F. Raimondi, “A constrained ECA language supporting formal verification of WSNS,” in *Proc. IEEE 29th Int. Conf. Adv. Inf. Netw. Appl. Workshops*, Mar. 2015, pp. 187–192.

[3] G. Russello, L. Mostarda, and N. Dulay, “ESCAPE: A component-based policy framework for sense and react applications,” in *Proc. Int. Symp. Compon.-Based Softw. Eng.* Berlin, Germany: Springer, 2008, pp. 212–229.

[4] L. Mostarda and A. Navarra, “Distributed intrusion detection systems for enhancing security in mobile wireless sensor networks,” *Int. J. Distrib. Sensor Netw.*, vol. 4, no. 2, pp. 83–109, 2008.

[5] A. Kumar, V. Kumar, and N. Chand, “Energy efficient clustering and cluster head rotation scheme for wireless sensor networks,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 3, no. 5, pp. 129–136, 2011.

[6] H.-W. Ferng and J.-S. Chuang, “Area-partitioned clustering and cluster head rotation for wireless sensor networks,” in *Proc. Int. Conf. Mach. Learn. (ICMLC)*, vol. 2, Jul. 2017, pp. 593–598.

[7] M. Micheletti, L. Mostarda, and A. Piermarteri, “Rotating energy efficient clustering for heterogeneous devices (REECHD),” in *Proc. 32nd IEEE Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, Krakow, Poland, May 2018, pp. 213–220.

[8] N. Aierken, R. Gagliardi, L. Mostarda, and Z. Ullah, “Ruheed-rotated unequal clustering algorithm for wireless sensor networks,” in *Proc. 29th IEEE Int. Conf. Adv. Inf. Netw. Appl. Workshops (AINA)*, Gwangiu, South Korea, Mar. 2015, pp. 170–174.

[9] Z. Ullah, L. Mostarda, R. Gagliardi, D. Cacciagrano, and F. Corradini, “A comparison of heed based clustering algorithms–introducing er-heed,” in *Proc. IEEE 30th Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, Mar. 2016, pp. 339–345.

[10] R. Pachlor and D. Shrimankar, “LAR-CH: A cluster-head rotation approach for sensor networks,” *IEEE Sensors J.*, vol. 18, no. 23, pp. 9821–9828, Dec. 2018.

[11] A. Navarra, C. M. Pinotti, and A. Formisano, “Distributed colorings for collision-free routing in sink-centric sensor networks,” *J. Discrete Algorithms*, vol. 14, pp. 232–247, Jul. 2012.

[12] A. W. Khan, A. H. Abdullah, M. A. Razzaque, and J. I. Bangash, “VGDRA: A virtual grid-based dynamic routes adjustment scheme for mobile sink-based wireless sensor networks,” *IEEE Sensors J.*, vol. 15, no. 1, pp. 526–534, Jan. 2015.

[13] A. Navarra, C. M. Pinotti, V. Ravelomanana, F. B. Sorbelli, and R. Ciotti, “Cooperative training for high density sensor and actor networks,” *IEEE J. Sel. Areas Commun.*, vol. 28, no. 5, pp. 753–763, Jun. 2010.

[14] T. Liu, J. Peng, J. Yang, G. Chen, and W. Xu, “Avoidance of energy hole problem based on feedback mechanism for heterogeneous sensor networks,” *Int. J. Distrib. Sensor Netw.*, vol. 13, no. 6, pp. 1–11, 2017.

[15] F. Barsi, A. A. Bertossi, C. Lavault, A. Navarra, S. Olariu, C. M. Pinotti, and V. Ravelomanana, “Efficient location training protocols for heterogeneous sensor and actor networks,” *IEEE Trans. Mobile Comput.*, vol. 10, no. 3, pp. 377–391, Mar. 2011.

[16] W. Liu and J. Yu, “Energy efficient clustering and routing scheme for wireless sensor networks,” in *Proc. IEEE Int. Conf. Intell. Comput. Intell. Syst.*, vol. 3, Nov. 2009, pp. 612–616.

[17] E. Babaee, S. Zareei, and R. Salleh, “Best path cluster-based routing protocol for wireless sensor networks,” in *Proc. UKSim 15th Int. Conf. Comput. Modeling Simulation*, Apr. 2013, pp. 663–667.

[18] O. Younis and S. Fahmy, “HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks,” *IEEE Trans. Mobile Comput.*, vol. 3, no. 4, pp. 366–379, Oct. 2004.

[19] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *Proc. 33rd Annu. Hawaii Int. Conf. Syst. Sci. (HICSS)*, Washington, DC, USA, vol. 8, Jan. 2000, p. 10.

[20] B. Jan, H. Farman, H. Javed, B. Montrucchio, M. Khan, and S. Ali, “Energy efficient hierarchical clustering approaches in wireless sensor networks: A survey,” *Wireless Commun. Mobile Comput.*, vol. 2017, Sep. 2017, Art. no. 6457942. doi: 10.1155/2017/6457942.

[21] D. W. Sambo, B. O. Yenke, A. Förster, and P. Dayang, “Optimized clustering algorithms for large wireless sensor networks: A review,” *Sensors*, vol. 19, no. 2, pp. 1–27, 2019. doi: 10.3390/s19020322.

[22] M. Zeng, X. Huang, B. Zheng, and X. Fan, “A heterogeneous energy wireless sensor network clustering protocol,” *Wireless Commun. Mobile Comput.*, vol. 2019, May 2019, Art. no. 7367281. doi: 10.1155/2019/7367281.

[23] E. Ever, R. Luchmun, L. Mostarda, A. Navarra, and P. Shah, “UHEED—An unequal clustering algorithm for wireless sensor networks,” in *Proc. SENSORNETS*, 2012, pp. 185–193.

[24] C. T. Sony, C. P. Sangeetha, and C. D. Suriyakala, “Multi-hop LEACH protocol with modified cluster head selection and TDMA schedule for wireless sensor networks,” in *Proc. Global Conf. Commun. Technol. (GCCT)*, Apr. 2015, pp. 539–543.

[25] J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: A survey,” *IEEE Wireless Commun.*, vol. 11, no. 6, pp. 6–28, Dec. 2004.

[26] A. Sarkar and T. S. Murugan, “Routing protocols for wireless sensor networks: What the literature says?” *Alexandria Eng. J.*, vol. 55, pp. 3173–3183, Dec. 2016.

[27] S. Jabbar, M. A. Habib, A. A. Minhas, M. Ahmad, R. Ashraf, S. Khalid, and K. Han, “Analysis of factors affecting energy aware routing in wireless sensor network,” *Wireless Commun. Mobile Comput.*, vol. 2018, Feb. 2018, Art. no. 9087269.

[28] V. K. Arora, V. Sharma, and M. Sachdeva, “A distributed, multi-hop, adaptive, tree-based energy-balanced routing approach,” *Int. J. Commun. Syst.*, vol. 32, no. 9, 2019, Art. no. e3949.

[29] Q. Li, Q. Zhu, and M. Wang, “Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks,” *Comput. Commun.*, vol. 29, no. 12, pp. 2230–2237, 2006.

MATTEO MICHELETTI is currently pursuing the Ph.D. degree with the Department of Computer Science, University of Camerino, Italy.

LEONARDO MOSTARDA was a Research Associate with the Computing Department, Distributed System and Policy Group, Imperial College London, in 2007. He was a Senior Lecturer with the Distributed Systems and Networking Department, Middlesex University, in 2010. He is currently an Associate Professor with the Department of Computer Science, University of Camerino, Italy. His main research interests include the areas of wireless sensor networks, middleware, and security.

ALFREDO NAVARRA received the Ph.D. degree in computer science from the Sapienza University of Rome, in 2004. Before joining the University of Perugia, Italy, in 2007, as an Assistant Professor, he has been with various international research institutes, such as the INRIA, Sophia Antipolis, France, Department of Computer Science, University of L'Aquila, Italy, and LaBRI, University of Bordeaux, France. Since 2015, he has been an Associate Professor with the Mathematics and Computer Science Department, University of Perugia. He has coauthored more than 140 publications in high-quality international journals, book chapters, and conference proceedings. His research interests include algorithms, computational complexity, distributed computing, and networking.

• • •