

Original Research Paper

# Energy-Efficient Ant Colony-Based $k$ -Hop Clustering and Transmission Range Assignment Protocol for Connectivity Construction in Dense Wireless Sensor Networks

<sup>1</sup>Gokou Hervé Fabrice Diédié, <sup>2</sup>Boko Aka and <sup>3</sup>Michel Babri

<sup>1,2</sup>Université Nangui Abrogoua, 02 BP 801, Abidjan, Côte d'Ivoire

<sup>2</sup>Institut National Polytechnique Houphouët Boigny, BP 1093 Yamoussoukro, Côte d'Ivoire

## Article history

Received: 20-12-2017

Revised: 16-02-2018

Accepted: 19-03-2018

## Corresponding Author:

Gokou Hervé Fabrice Diédié  
Université Nangui Abrogoua,  
02 BP 801, Abidjan, Côte  
d'Ivoire  
Email: herve.diedie@inphb.edu.ci

**Abstract:** Connectivity construction is the main phase of a communication-oriented topology control process. It consists of improving the current network physical topology while preserving important properties such as connectivity and symmetry. In this paper, we address the problem of combining two of the techniques commonly used for this purpose in networks composed of a large number of energy constrained wireless sensor-nodes namely, *clustering* and power control. We propose an ant colony-based asynchronous and localized protocol that helps to significantly reduce energy losses by simultaneously eliminating redundant and poor quality links, always keeping the Cluster Head-to-member distance up to  $k$ -hops ( $k \geq 1$ ) and minimizing signalization. Simulation results show that our protocol outperforms some state-of-the-art solutions in terms of Quality of the Topology (QoT) and network lifetime prolongation.

**Keywords:** *Clustering*, Range Assignment, Connectivity Control, Stigmergy, WSN

## Introduction

Wireless Sensor Networks (WSNs) are often randomly deployed to collect information in remote or hostile areas. This type of network is characterized by its large number of nodes and redundant links. The resulting interferences contribute to shortening both nodes and network lifetimes. Connectivity control is a technique that is used to restructure, reorganize and maintain such a chaotic communication topology (Li *et al.*, 2013; Labrador and Wightman, 2010).

When constructing nodes' connectivity a strategy called *clustering* is commonly used to cope with scalability. It consists in partitioning the network into groups of nodes that are geographically close. Each group or cluster is placed under the control of a dedicated node called Cluster Head (CH). The latter is selected according to a combination of criteria. This strategy helps to save nodes' energy by reducing communication distance (in terms of number of hops) and packet size. Regrettably, *clustering* is a logical topology-oriented technique. In order to further reduce energy consumption, it is mandatory to also try to improve the network's physical topology. To this end, a common method consists in providing each cluster

member the ability to minimize its transmission range through another well-known connectivity construction scheme referred to as *power control*.

*Clustering* and power control are formally related to two NP-hard problems respectively referred to as the minimum connected dominating set (Garey *et al.*, 1976) and the minimum range assignment problem (Clementi *et al.*, 1999). As a consequence, for efficiency purpose they are commonly solved using only approximate solutions such as heuristics. A substantial number of protocols that combine *clustering* and *power control* exist in the literature; however, these solutions fail to provide a strategy that simultaneously minimizes CHs reelections, keeps the CH-to-member distance to at most  $k$ -hops ( $k \geq 1$ ) and eliminates redundant and poor quality links while preserving symmetry. Such a shortcoming is detrimental to node and network lifetime maximization.

In this paper, we propose an ant colony-based asynchronous localized strategy which helps to create in each cluster a CH-rooted spanning tree only composed of edges that provide the best trade-off between length and quality. This scheme also helps to minimize signalization and the intra-cluster topology stretch factor i.e., to always keep the CH-to-member distance up to  $k$ -hops ( $k \geq 1$ ). All these features have the effect of increasing energy efficiency and extending network lifetime.

The rest of the paper is organized as follows. In Section 2 we review some major related solutions proposed in the literature. Then, we detail our contribution in Section 3. We evaluate its performance by simulation in Section 4. The results we obtained are discussed in section 5. Finally, we conclude the paper and give some perspectives in Section 6.

## Related Work

The need of both scalability and energy efficiency has always been a major issue in ad hoc networks. In recent years, several distributed solutions that combine *clustering*-based and power control-based strategies have been proposed to that end in the literature.

CLUSTERPOW by Kawadia and Kumar (2003) is certainly the most famous protocol of this category. To increase the spatial reuse authors proposed to rely on the minimum range of nodes involved in each packet transmission. This strategy forces each node to opt for multi-hop transmissions and implicitly allows the creation of as many clusters as a node has ranges. CLUSTERPOW does not require a leader to be elected but highly depends on the underlying routing protocol.

Manousakis and Baras (2003) proposed a simulated-annealing based heuristic to minimize nodes inter and intra-cluster transmission ranges in ad hoc networks. However, this solution is not suitable for dense ad hoc networks such as WSNs. The same shortcoming can be noticed in many MANET-oriented solutions such as the one proposed by Chiasserini *et al.* (2004).

Cardei *et al.* (2006) proposed IP-HRA for heterogeneous networks composed of some *supernodes* in charge of building clusters. To this end, each *supernode* has to broadcast its ID in its  $k$ -hop neighbourhood. Nodes that receive such messages will join the closest sender. After being affiliated, each node must optimize its range using an integer linear program either solved by a greedy heuristic or using the LMST protocol by Li *et al.* (2005). This strategy allows to reduce the self-organization latency but does not help to neither minimize the number of clusters (many isolated CHs) nor guarantee the network connectivity.

Vural *et al.* (2013) adopted a probabilistic strategy enabling each member of a *backbone* to optimize its range. The latter probability is calculated from the needed CH density and the euclidian distance from the sink. Unfortunately, this strategy ignores the existence of an intra-cluster topology. With protocol TTTC (Two-Tiered Topology Control) Hameed *et al.* (2014) proposed a similar strategy.

In order to minimize the *backbone* size nodes' ranges are in contrast bounded by a threshold value. The process is initiated by the sink which builds a Connected Dominating Set (CDS) from a spanning tree. Indeed, in each cluster the CH builds such a tree to make members

adjust their ranges according to their farthest neighbours. Although this protocol helps to reduce nodes' physical degrees, this strategy is not scalable since the *backbone* construction process is launched by the sink. Furthermore, TTTC ignores link quality whereas the latter can cause energy losses.

Hu and Han (2014) proposed a probabilistic-based *clustering* scheme inspired from LEACH (Heinzelman *et al.*, 2002) to optimize CHs' ranges and cancel interference. To this end, the transmission ranges of the CHs are determined by their positions from the sink and their residual energies. However, this solution is not really scalable since it requires all the CHs to be located in the sink's one-hop neighbourhood.

In order to minimize interferences through spatial reuse Liu *et al.* (2015) investigated a Poisson law-based *clustering* process. Each cluster is tessellated into  $n$ -layer annuli. Each time a transmitter selects one receiver at a certain layer it has to adjust accordingly its range. This scheme helps the receiver to achieve a low outage probability. The transmission range minimization problem is formulated as a convex optimization one. Regrettably, these authors do not discuss the *clustering* process. Furthermore, the deployment requires a deterministic scheme.

Unlike strategies that are commonly used, Tseng *et al.* (2015) suggested through the Green *Clustering* Algorithm (GCA) the construction of a Relative Neighbour Graph (RNG) before creating the dominating set. In order to avoid having isolated CHs at the borders of deployment zone, nodes located near this area have the priority of becoming CH. This strategy aims to guarantee load balancing and uniform cluster size. However, RNG construction requires nodes to know their exact positions. Unfortunately, euclidian distance is not always easy to estimate especially in hostile environments. Furthermore, CH selection criteria ignore residual energy. This is detrimental to network lifetime maximization.

Vinutha *et al.* (2017) proposed a hidden markov model-based scheme to adjust nodes' ranges. After calculation of its weight, each node informs its neighbours. Node with the lowest weight proclaims itself CH and requests neighbours to join its cluster. After the cluster formation, each CH sends probe packets with different transmission power levels to all the members. All reply-packets will help the CH to determine its initial transmission power level through the hidden markov model. In each cluster, the CH monitors the link quality when members start communicating with one another. CH recalculates and modifies the transmission power of a node when link quality goes below a defined threshold. This strategy may be too demanding for the CH.

Jameii and Maadani (2016) proposed CCALA (Connectivity Control Algorithm based on Learning Automata) a learning automata-based synchronous protocol. The latter consists of two main phases:

*Clustering* phase and transmission radius updating phase. In the *clustering* phase, nodes are elected as CHs on the basis of their degrees and residual energy while other become members. After the *clustering* phase, each sensor node adjusts its transmission range according to its connectivity and residual energy using a learning automaton. Actions correspond to the different ranges. At the beginning of each round, an action (power level) is selected then network feedback is determined considering node's degree and *unbalanced energy consumption* in the cluster. The latter energy is calculated in terms of the maximum and the minimum residual energy of sensor nodes in the cluster. Reward or penalty for the selected action is obtained respectively if the unbalanced energy consumption is higher or lower than a defined threshold. This strategy does not guarantee redundant links elimination. Furthermore, the unbalanced energy consumption calculation does not take account of the amount of energy lost by neighbours when gathering information about their residual energies.

Li *et al.* (2017) investigated a fuzzy logic-based strategy to adapt clusters to a nonuniform power control scheme.

The initial broadcasting power is adjusted in order to optimize the number of CHs. The latter are selected according to a combination of criteria including node degree, residual energy and euclidian distance from the sink. After election, a CH invites its neighbours to join the newly created cluster. This strategy helps to balance energy consumption in the network but the power control scheme is applied only to CHs.

Zhu *et al.* (2015) suggested to explicitly take account of the link quality. The resulting protocol named Hybrid Distributed Hierarchical Agglomerative (H-DHAC) and inspired by LEACH (Heinzelman *et al.*, 2002) is based on two hierarchical *clustering* distance evaluation techniques commonly used in datamining and bioinformatics namely, Unweighted pair-Group Method with arithmetic Averages (UPGMA) and Weighted Pair Group Method with arithmetic Averages (WPGMA) (Romesburg, 2004). Authors propose to use data (spatial, temporal, quantitative, or qualitative) related to both nodes and link topological properties (degree, distance, quality...). Therefore, after neighbour discovery, each node must calculate a *resemblance matrix* containing the degree of similarity with each neighbour. When nodes are equipped with a localization device this degree is denoted by the euclidean distance. By contrast, in the absence of a localization means, binary values are used (0 when the link exists and 1 otherwise). If a node has the lowest ID, it proclaims itself CH and sends an invitation message in its neighbourhood. Only the closest nodes will reply affirmatively. As soon as the cluster is built, members adjust their ranges using the ATPC protocol (Lin *et al.*, 2016). This strategy does not require nodes to know their exact locations while allowing through ATPC a packet level power control. However, it

does not take account of node's residual energy as a CH selection criterion. This can only negatively impact load balancing and network lifetime. Moreover, the cluster merging process (when the size is below a threshold) is energy costly since it requires to constantly update the resemblance matrix obtained when using UPGMA.

## Motivation and Objectives

Building a minimum spanning tree requires the use of a topological property-based weighting metric (*number of hops, euclidean distance, link quality...*) for edges. Any realistic energy efficient solution requires at least a link quality-based weighting metric; yet, from link quality point of view the graph induced by the network is directed. Therefore, it is impossible to use traditional construction schemes (Prim, 1957; Kruskal, 1956) to preserve link symmetry. It is also mandatory not to increase the *hop-stretch* factor in each created cluster. In other words, it is necessary to always keep each member to at most  $k$ -hops ( $k \geq 1$ ) from its Cluster Head (CH). Strategies that are commonly used in the literature fail to guarantee both link symmetry and link quality while preserving a good *hop-stretch* factor after CH's reelection.

Surprisingly, to the best of our knowledge, such a solution that asynchronously combines both *k-hop clustering* ( $k \geq 1$ ) and power control techniques does not exist in the literature.

Therefore, we aim at proposing a localized asynchronous *k-hop clustering* and power control protocol that is able to:

- Eliminate redundant links in terms of a metric that helps to find a good trade-off between quality and length (distance)
- Preserve connectivity and link symmetry while keeping in each cluster the CH-to-member distance to at most  $k$ -hops ( $k \geq 1$ )
- Minimize nodes' transmission ranges and neighbour tables
- Minimize the message overhead
- Maximize network lifetime

## Proposed Solution

We detail in this section our solution referred to as CONTRACT (Cluster-based Optimal Neighbour-level Transmission Range Assignment for the Communication Topology). The latter solution is a localized and asynchronous message passing protocol.

The rationale behind CONTRACT is to use a *stigmergy*-based (Grassé, 1959; Dorigo *et al.*, 2000; Khuong *et al.*, 2016) process that builds a tree composed of only symmetric and low-latency links. From the intra-cluster topology point of view, the result is a *k-hop*

diameter-bounded spanning tree rooted at the CH; while from the inter-cluster topology perspective, the resulting graph has no cycle in nodes'  $(k+1)$ -hops neighbourhood and no adjacent CHs.

We assume that:

- Each node has a unique identifier (ID)
- Nodes have a relative knowledge of their positions and can locate their neighbours using schemes such as RSSI (Received Signal Strength Indicator) and SINR
- Nodes have an efficient link quality estimation protocol
- Nodes' mobility is due to environment instability
- The radio channel is lossy
- Maximum number of retransmission attempts is known as a parameter
- The process takes place in the plan
- Nodes' interconnection is modeled by an Unit Disc Graph (UDG)

### Neighbourhood Information Analysis

This process is based on the underlying neighbour discovery protocol. The latter is assumed to be energy-efficient and able to reveal one-hop symmetric neighbours. This protocol must also help any node to inform its neighbours about some of their topological characteristics (ID, speed, degree, state...). After neighbour (re)discovery, each node  $u$  must make sure that all the conditions required for the creation of a new cluster are met, namely:

- no CH in its  $(k-1)$ -hop neighbourhood;
- $m$  ( $m \geq 0$ ) free nodes in its  $k$ -hop neighbourhood wishing to join its cluster. If  $m \geq 1$ , there must be between  $u$  and these  $m$  nodes at least one path of which length is denoted by  $l$  such as  $l \leq k$ .

It is noteworthy to mention that a node is free when it is not affiliated to any cluster. Therefore, to make sure that the second condition aforementioned is met, node  $u$  will proclaim itself CH and broadcast a BEST message in its  $k$ -hop neighbourhood. The latter message contains its *fitness score* calculated using Equation 1; whereas, after neighbour (re)discovery a node becomes automatically a CH upon noticing that it is isolated:

$$score = a \times \left( \frac{n-1}{n} \right) + \beta \times \left( \frac{E_r}{E_i} \right) + \delta \times (1 - CC) + \gamma \times S(t) \quad (1)$$

$$CC = \frac{\sum_{i=1}^n \sum_{j=1}^n w_i \times w_j \times \sigma_{ij}}{\left( \sum_{i=1}^n \sum_{j=1}^n w_i \times w_j \right) \times \max_{i,j}(\sigma_{ij})} \quad (2)$$

$$S(t) = \begin{cases} 1, & \text{if } p = 0 \\ e^{-\lambda t}, & \text{otherwise} \end{cases} \quad (3)$$

$$MTBD = \frac{\sum_c^p (\tau_c^{(a)} - \tau_c^{(d)})}{p} \quad (4)$$

$$\lambda = \frac{1}{MTBD} \quad (5)$$

$$a + \beta + \delta + \gamma = 1, \beta > \max(\alpha, \delta, \gamma) \quad (6)$$

Parameter  $n$  denotes the number of neighbours,  $E_r$  is the remaining energy,  $E_i$  is the initial energy and  $CC$  the weighted *clustering* coefficient (Miyajima and Sakuragawa, 2014) of a node.  $w_i$  denotes the weight assigned to neighbour  $i$  and  $\sigma_{ij}$  the one assigned to node  $j$  by node  $i$ .  $S(t)$  is node's *instantaneous instability*.  $MTBD$  denotes the Mean Time Between two consecutive Departures from clusters.  $\tau_c^{(a)}$  is the arrival time in cluster  $c$  and  $\tau_c^{(d)}$  the departure time from the same cluster.  $p$  is the number of clusters to which the node has belonged so far.  $\lambda$  is the rate of departures from a cluster;  $\alpha$ ,  $\beta$ ,  $\delta$ , and  $\gamma$  are four weighting coefficients.

Note that weight  $w_i$  assigned to link  $i$  is related at least to its quality.

### Cluster Formation

Cluster formation process is initiated by a not isolated self-proclaimed CH as soon as it broadcasts a BEST message. The latter process requires that the two conditions mentioned in the previous section are met. The first condition can be trivially verified but the second one needs a more complex scheme especially when  $k > 1$ .

Indeed, this kind of verification means finding the  $p$  ( $p \geq 1$ ) best disjoint paths in the graph induced by the  $k$ -hop neighbourhood of the self-proclaimed CH. This issue is a well-known NP-hard problem in graph theory (Karp, 2011; Matthias and Pfeiffer, 1993). We propose an ant colony-based distributed heuristic to solve this problem. Our scheme uses *stigmergy*, a paradigm inspired from the pheromone-based communication system of several social insects like foraging ants (Grassé, 1959; Dorigo *et al.*, 2000; Khuong *et al.*, 2016).

Formally, a pheromone  $\Phi$  is a vector such as  $\Phi = (id, q, f, \rho)$ ; where  $id$  denotes the CH's ID (the nest),  $q$  is the remaining quantity of the pheromone,  $f$  the information carried by the pheromone (its type) and  $\rho$  denotes the evaporation rate per unit of time. For instance,  $\Phi_{ij}$  and  $\Phi_{ji}$  denote the pheromones deposited by an ant (a message) on the path between two food areas (the potential members)  $i$  and  $j$  respectively in  $ij$  and  $ji$  directions.

Ants must comply with the following rules:

- A food area can be discovered and marked by only one ant
- Any ant must search an area located not more than  $k$ -hops from the nest
- Ants have a specific pheromone for each phase (food areas prospection, return to the nest, food collection, territory marking)
- Ants only react to pheromones laid by their nestmates

These metaphoric rules are also referred to as *random proportional transition rules*. They correspond to those commonly applied in all flooding protocols in order to minimize the message overhead (*do not forward the same message twice, do not return a message to its sender, each message has a bounded lifetime...*).

The whole cluster formation process requires three types of pheromones (BEST, AGREE, TREE) for the intra-cluster connectivity construction and only one pheromone (BRIDGE) for neighboring clusters interconnection.

At the end of such a process each member will get a unique status namely, *Cluster-Head*, *Ordinary Member* or *Gateway* respectively denoted by *CH*, *OM* and *GW*. By contrast, all unaffiliated and stable nodes will have a *Free (FN)* status. Note that a node with a Gateway status has at least one neighbour not belonging to its cluster.

#### *Building the CH-to-Member Communication Paths*

After broadcasting a BEST message a node enters into the Self-Proclaimed Cluster Head (SPCH) state and sets a back-off timer for  $t_{best}$  units of time.

Upon receiving a BEST message any free node  $u$  must make sure that it has not handled a copy of this message yet. If so, node  $u$  must deposit a BEST pheromone on the link to the sender of the BEST message then decrease by 1 unit the time-to-live ( $ttl$ ) value of the message.

If a BEST message's  $ttl > 0$ , node  $u$  has to forward it to its neighbours or delete it otherwise.

#### *Building the Member-to-CH and the Member-to-Member Communication Paths*

Upon receiving a BEST message, a free node  $v$  must also compute its fitness score according to Equation 1 and compare it to the SPCH's score piggy-backed in the BEST message. If its score is the lowest, node  $v$  must join the SPCH  $u$ . Therefore, node  $v$  sends to the SPCH  $u$  a AGREE message through its *default communication link* i.e. the link with the BEST pheromone. Indeed, by default, AGREE messages are forwarded following the reverse path taken by messages BEST. Node  $u$  then starts its  $t_{agree}$  timer and enters into Potential Member (PM) state.

Before forwarding a AGREE message, a Potential Member must deposit a AGREE pheromone on the link that leads to the sender. However, such a message must be deleted by nodes that are already affiliated to another CH or those that have rejected the SPCH's authority. It is worth noting that an isolated CH behaves like a Free node. These rules help to have in the final intra-cluster topology paths consisting of only the hitherto free nodes.

Note that, the reception of at least one AGREE message allows the self-proclaimed CH to implicitly check that the second condition for the creation of a cluster with more than one member is met (see previous section).

In order to optimize the trade-off between length and quality, a potential member can make the *default communication link* compete against another one e.g. a link that sent a copy of a BEST message with a larger  $ttl$ . To this end, the latter link will also receive a BEST pheromone deposit. Therefore, during AGREE messages forwarding process, packets will choose according to a certain probability (discussed later) either the *default communication link* or the challenging link.

Upon its  $t_{agree}$  timer expiration the potential member will choose the link with the largest quantity of BEST pheromone as its definitive *default communication link*.

#### *Defining Roles and Cluster's Limits*

Upon its  $t_{best}$  timer expiration a SPCH must enter into CH state. If it has received at least one AGREE message, it sends a message TREE to the Potential Members (PM) via the links with pheromones AGREE. The latter will be replaced by pheromones TREE on all the intra-cluster communication paths.

Upon receiving a TREE message before its  $t_{agree}$  timer expiration a potential member becomes a fully-fledged member i.e., its status shifts from PM to OM.

For security purpose, Potential Members will accept only TREE messages sent by the former self-proclaimed CH. Likewise, to preserve link symmetry a PM accepts only messages TREE that are delivered through link with a BEST pheromone.

After becoming an Ordinary Member (status OM), a node must forward a TREE message to its neighbours avoiding to give it back to its sender. By contrast, the sender of a TREE message will receive a TREE-ACK message as a reply.

Upon receiving a TREE-ACK message, a node must deposit a TREE pheromone on the sender's link to replace the AGREE pheromone.

#### *Building External Communication Paths*

Nodes located at the border of a cluster (those with at least one neighbour affiliated to another cluster) obtain a Gateway status (GW). They are henceforth in charge of the cluster interconnection process; to this end, they

broadcast a BRIDGE-REQ message including the newly elected CH's ID. Only nodes that have a Gateway status and are not the sender's clustermates can respond to a first time received BRIDGE-REQ message (local verification).

Any valid bridge creation request is thus forwarded to the CH (Hierarchical verification). The interconnection is authorized if no link already exists between the two clusters. Upon receiving a bridge creation permission (BRIDGE-OK) a gateway sends a BRIDGE-ACK message to the requesting gateway. The latter deposits a BRIDGE pheromone on the link to the BRIDGE-ACK message's sender then responds with a BRIDGE message. The neighbour-gateway will also lay a BRIDGE pheromone after the reception of the BRIDGE message.

This double verification (local and hierarchical) before creating a bridge between two clusters aims at minimizing the number of redundant links in the inter-cluster topology.

#### Direct Affiliation to a Cluster

After neighbour (re)discovery, when node  $u$  finds that it is located at less than  $k$ -hops from one or several CHs, it broadcasts a JOIN-REQ message in its 1-hop neighbourhood for the gateways then starts a back-off timer denoted by  $t_{join}$ . If any gateway  $v$  (node with a GW status) finds that this JOIN-REQ message is valid it will reply to node  $u$  with a JOIN-ACK message including its CH's ID and a parameter  $hp_v$ , denoting the distance (in terms of number of hops) from its CH. Any gateway  $v$  considers that a JOIN-REQ message is valid if  $hp_v < k$ . Node  $u$  chooses the gateway  $v$  which was the first one to reply (JOIN-ACK) and cancels its  $t_{join}$  timer; then, node  $u$  becomes a new member by sending a JOIN message to gateway  $v$ , updates its variable  $hp_u$  such as  $hp_u = hp_v + 1$  and deposits on link  $(u; v)$  a BEST pheromone. Node  $v$  does the same on link  $(v; u)$  upon receiving the JOIN message sent by  $u$ . Both nodes  $u$  and  $v$  updates therefore their statuses from OM to GW if at least one of their neighbours is not a clustermate. If node  $u$  becomes a gateway it will initiate a bridge creation process as discussed in the previous section. If node  $v$  becomes an Ordinary Member it will deletes all the BRIDGE pheromones on its links to the neighbouring clusters.

#### Transmission Range Minimization

In order to minimize its transmission range a cluster member must eliminate from its neighbour table all the links that have neither a TREE or a BRIDGE pheromone. Then the latter node has to adjust its range to the power required to preserve the quality of the communication link with its farthest neighbour.

#### Pheromone Deposit and Update Rules

A packet forwarded at time  $t$  by node  $i$  will follow a given pheromone denoted by  $\Phi_{ij}$  laid on the link to

neighbour  $j$  according to a certain probability denoted by  $\pi_{ij}(t)$  and calculated using Equation 7:

$$\pi_{ij}(t) = \begin{cases} \frac{[\Phi_{ij}^{(g)}(t)]^a \times w_{ij}^b}{\sum_{l \in \Upsilon(i)} [(\Phi_{il}^{(g)}(t))]^a \times w_{il}^b}, & \text{if } j \in \Upsilon(i) \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$\Phi_{ij}^{(g)}(t) = [\Phi_{ij}^{(g)}(t-1)] + [\hat{\theta} - \psi_{ij}(t)] \quad (8)$$

$\Phi_{ij}^{(g)}(t)$  denotes the quantity of pheromone deposited at time  $t$  by node  $i$  on the link to neighbour  $j$ .  $\Upsilon(i)$  is the set of links on which node  $i$  has deposited such a pheromone.  $w_{ij}$  denotes the *visibility* of the link to neighbour  $j$  i.e., the inverse of the number of hops from the self-proclaimed CH to node  $i$  via  $j$ ;  $a$  and  $b$  denote two weighting coefficients such as  $a+b = 1$ .

Equation 8 relates to the accumulation process of a given pheromone. It helps to calculate the remaining quantity at time  $t$  of the given pheromone deposited by the node  $i$  on link to neighbour  $j$  after the attempt of sending a packet via neighbour  $j$ ;  $\hat{\theta}$  is the maximum number of attempts authorized by the underlying application.  $\psi_{ij}(t)$  is the number of failures experienced by node  $i$  at time  $t$  after trying to send a packet to neighbour  $j$ .

As for Equation 9, it relates to the evaporation process of a given pheromone deposited by node  $i$  on the link to neighbour  $j$ :

$$\Phi_{ij}^{(g)}(t+1) = [1 - \Phi_{ij}^{(p)}(t)] \times \Phi_{ij}^{(g)}(t) \quad (9)$$

$$\Phi_{ij}^{(p)}(t) = 1 - e^{-\frac{\ln\left(\frac{q_e}{q_0}\right)}{T_e}} \quad (10)$$

$\Phi_{ij}^{(g)}(t)$  is the quantity of the given pheromone deposited by node  $i$  on the link to neighbour  $j$  at time  $t$ ; where  $\Phi_{ij}^{(p)}(t)$  is the evaporation rate of that pheromone. The latter is defined by node  $i$  knowing that  $\Phi_{ij}^{(p)}(t+1) = \Phi_{ij}^{(p)}(t)$ .

Equation 10 helps to calculate the evaporation rate of a given pheromone when its evaporation *deadline*  $T_e$ , its theoretical quantity at that date denoted by  $q_e$ , and its initial quantity  $q_0$  are known. The latter quantities are given as parameters.

Note that  $q_0 > 0$  and  $q_e \in ]0; 1]$ .

#### Cluster-Head Reelection

In order to balance the loads, CHs are elected for a duration denoted by  $t_{service}$  set as a parameter by the underlying application. However, when its residual energy (*communication budget*) reaches a threshold value (e.g. 90% of the initial value at the beginning of its *mandate*), it gives up its CH status and becomes a free node (*CH to FN*).

It will refrain from attempting to dominate its neighbourhood during a *probationary period*. During this space of time denoted by  $t_{prob}$ , whenever it becomes free, a former CH can only try to join a neighbouring cluster. Unless it is forced to create a cluster of which it would be the only member.

Equation 10 helps the different member of a newly created cluster to determine the evaporation rate of the TREE and BRIDGE pheromones from the CH's service time denoted by  $t_{service}$ .

Therefore, after neighbour (re)discovery when a node finds that the quantity of all its TREE pheromones has reached the threshold value it becomes free.

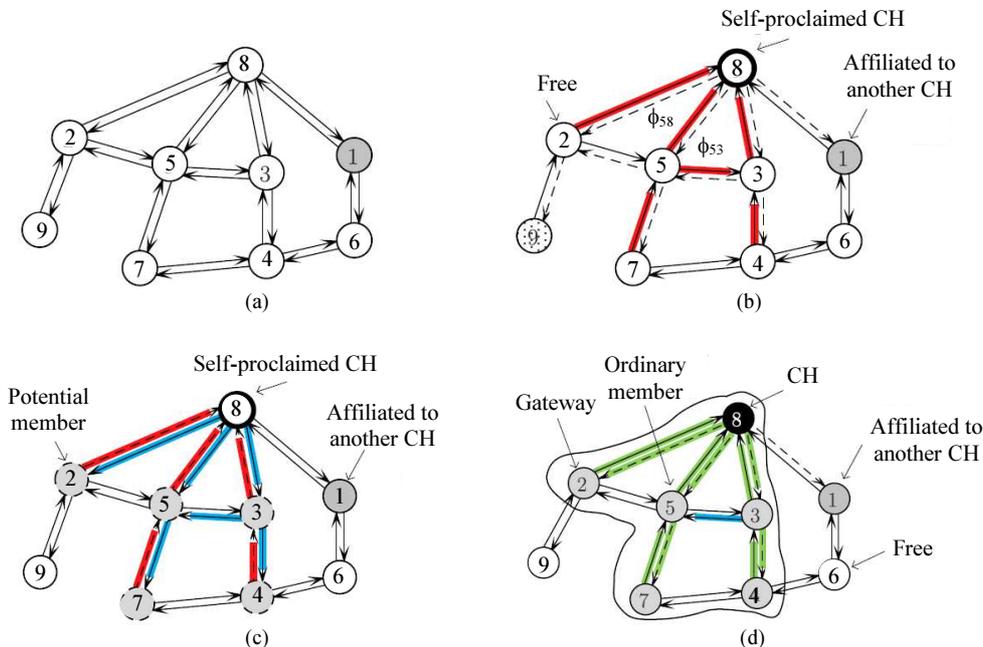
The latter node will attempt to join the closest cluster or may possibly create a new cluster.

### Illustrative Example

Figures (1a-1d) depict the main phases of the cluster formation process. Figure 1a, after analyzing its neighbourhood, node 8 considers that it has to build a 2-hop cluster, by broadcasting a BEST message. Figure 1b, free nodes whose scores are lower than that of node 8 deposit a BEST pheromone on the link to the neighbour which sent the BEST message then forward a copy to their other neighbours. Node 1 is not free therefore it deletes the BEST message. Node 5 received from node 8 a copy of BEST message that came after the one forwarded by node 3; but node 5 noticed that the link to node 8 is better (in terms of quality and number of hops from the CH) than the *default communication link* to node 3. In order to break tie between these two links,

node 5 deposits on them respectively, BEST pheromones  $\Phi_{58}$  and  $\Phi_{53}$ . Node 9 is still free but its score is higher than the one of node 8. Therefore, node 9 refuses to join node 8 and therefore, does not deposit any pheromone. After two hops, copies of BEST message reach nodes 4 and 7 with  $t_{tl} = 0$ , so they cannot be forwarded.

Figure 1c, All the Potential Members of the new cluster send to node 8 their AGREE messages via links bearing the BEST pheromones. Note that, upon receiving a copy of AGREE message from a neighbour, a Potential member must deposit a AGREE pheromone on the link to that neighbour. Node 5 decides to keep pheromone  $\Phi_{58}$  since the link on which it was deposited is the one that has forwarded the largest number of packets. Figure 1c, node 8 has received at least one AGREE message; hence, it can build a cluster with more than one members and become a CH. Therefore, Node 8 sends a TREE message to the Potential Members via links that bear a AGREE pheromone. Upon receiving a TREE message, each Potential Member becomes an Ordinary Member then changes the BEST pheromones into TREE pheromones before replying with a TREE-ACK message. The latter will help the receiver to also change its AGREE pheromones into TREE pheromones. Since the AGREE pheromone  $\Phi_{35}$  has not yet evaporated, node 5 will receive the TREE message sent by neighbour 3 but will not acknowledge it because the BEST pheromone  $\Phi_{53}$  was deleted. AGREE pheromone  $\Phi_{35}$  will eventually evaporate since it will no longer be reinforced.



**Fig. 1:** Illustration of CONTRACT's strategy for a 2-hop cluster formation process: (a) The graph induced by node 8's 2-hop neighbourhood (b) node 8 becomes a self-proclaimed CH and causes BEST pheromones deposits (c) AGREE pheromones deposit process (d) TREE pheromones deposit process (see online version for colors)

Nodes 2 and 4 opt for a Gateway status after having found that respectively node 9 and 6 do not belong to their cluster. As a CH, node 8 behaves like any gateway. Unlike nodes 2 and 4, node 8 can immediately initiate a bridge construction process with node 1 since it has already joined another CH but not with node 9 or 6 who are still free. After neighbour (re)discovery, node 9 will notice that it is now located at 2-hops from a CH (i.e. node 8); it may join node 8's cluster via gateway 2. By contrast, node 6 which is now located at 3-hops from node 8 may be forced to create its own cluster and become an isolated CH.

Note that from a metaphoric point of view, pheromones are deposited by ants (i.e. messages); but from a technical perspective, as illustrated by the example we have just discussed, pheromones are actually laid by nodes by updating some information in their neighbour tables.

It is also worth mentioning that CONTRACT uses a scheme referred to as *Sink-As-Cluster-Head* (Jain and

Reddy, 2014; Diédié *et al.*, 2016). Therefore, the BEST messages broadcasted by the sink in its  $k$ -hop neighbourhood are treated as those originating from any sensor-node; except that by convention the sink's score is set to its maximum value, namely 1.

Algorithms 1-4 formally describe the cluster formation, the direct affiliation, and the Cluster-Head reelection processes.

Let:

- $N(u)$  be the set of node  $u$ 's neighbours
- $NG(u)$  be the set of nodes with gateways (GW) and CHs statuses that node  $u$  has discovered in its 1-hop neighbourhood
- $Er(u)$  be node  $u$ 's residual energy
- $\Upsilon(u)$  be the set of links on which node  $u$  has deposited a pheromone
- $C(u)$  be the set of node  $u$ 's clustermates

**Algorithm 1:** Affiliation process of node  $u$

---

**Require:**  $E_{thr}, t_{discov}, \alpha, \beta, \delta, \gamma, t_{service}, k, t_{hello}, t_{join}, t_{best}, t_{bridge}, t_{agree}$

```

1:  $Er(u) \leftarrow$  Estimate_residual_energy()
2: while ( $Er(u) > E_{thr}$ ) do                                ▷ Residual energy is enough
3:   if (Current_time() = delay_DISCOV) then
4:     Broadcast HELLO ( $id_u, ch_u, state(u), hp_u$ ).        ▷ 1-hop discovery
5:     delay_HELLO  $\leftarrow$  Current_time() +  $t_{hello}$ 
6:   end if
7:   if (Current_time() = delay_HELLO) then
8:     if ( $N(u) = \emptyset$ ) then
9:        $state(u) \leftarrow$  CH
10:    else
11:      if ( $\Upsilon(u) = \emptyset$ ) then
12:         $state(u) \leftarrow$  FN
13:      end if
14:       $NG(u) \leftarrow$  Look_for_gateways( $N(u)$ )
15:      if ( $NG(u) \neq \emptyset$ ) then
16:        if  $state(u) = FN$  then
17:          Send JOIN( $id_u, state(u)$ ) to  $v, \forall v \in NG(u)$ 
18:          delay_JOIN  $\leftarrow$  Current_time() +  $t_{join}$ 
19:        end if
20:        if  $state(u) = GW$  then
21:          Send BRIDGE-REQ( $id_u, state(u), ch_u, hp_u$ ) to  $v, \forall v \in NG(u)$ 
22:          delay_BRIDGE  $\leftarrow$  Current_time() +  $t_{bridge}$ 
23:        end if
24:      else
25:         $score(u) \leftarrow$  Calculate_score( $\alpha, \beta, \gamma$ )      ▷ Equation 6
26:        Send BEST( $id_u, score(u)$ ) to  $v, \forall v \in N(u)$ 
27:         $state(u) \leftarrow$  SPCH
28:        delay_BEST  $\leftarrow$  Current_time() +  $t_{best}$ 
29:      end if
30:    end if
31:    delay_DISCOV  $\leftarrow$  Current_time() +  $t_{discov}$ 
32:  end if
33:  Receive message from  $v$ 
    
```

```

34: switch do
35:   case HELLO
36:     Send HELLO-ACK ( $id_u, ch_u, state(u), hp_u$ ) to  $v$ 
37:   case HELLO-ACK
38:      $N(u) \leftarrow$  Update_neighbour_table( $id_v, ch_v, state(v), hp_v$ )
39:   otherwise
40:     Handle_affil_messages( $v, message, k, N(u), NG(u)$ )    ▷ Algorithm 2
41:     Adjust_range( $N(u), C(u)$ )
42:   end switch
43:   Handle_delays()                                         ▷ Algorithm 4
44:    $Er(u) \leftarrow$  Estimate_residual_energy()
45: end while

```

---

**Algorithm 2:** Affiliation messages handling for node  $u$

---

**Require:**  $v, message, k, N(u), NG(u)$

```

1: switch  $message$  do
2:   case JOIN-REQ
3:     if ( $NG(u) \neq \emptyset$ ) then
4:       Send JOIN-ACK( $id_u, state(u), ch_u, hp_u$ ) to  $v, \forall v \in NG(u)$ 
5:     end if
6:   case JOIN-ACK
7:     if ( $state(u) = FN$ ) then
8:       delay_JOIN  $\leftarrow$  0
9:        $state(u) \leftarrow OM$ 
10:       $ch_u \leftarrow ch_v$ 
11:       $hp_u \leftarrow hp_v + 1$ 
12:       $\Upsilon(u) \leftarrow$  Deposit_pheromone( $v, TREE$ )           ▷ update pheromone table
13:      Send JOIN ( $id_u, state(u)$ ) to  $v$ 
14:      if ( $|N(u)| \neq |C(u)|$ ) then
15:         $state(u) \leftarrow GW$ 
16:        Send BRIDGE-REQ ( $id_u, state(u), ch_u, hp_u$ ) to  $w, \forall w \in NG(u) \setminus v$ 
17:      end if
18:    end if
19:   case JOIN
20:     if ( $(state(u) \neq CH) \wedge (|N(u)| = |C(u)|)$ ) then
21:        $state(u) \leftarrow OM$ 
22:     end if
23:      $\Upsilon(u) \leftarrow$  Deposit_pheromone( $v, TREE$ )           ▷ update pheromone table
24:     if ( $|N(u)| = |C(u)|$ ) then
25:        $\Upsilon(u) \leftarrow$  Delete_pheromones( $BRIDGE$ )         ▷ update pheromone table
26:     end if
27:   case BRIDGE-REQ
28:     if ( $state(u) = GW$ )  $\wedge$  (Check(BRIDGE-REQ)) then
29:       Send BRIDGE-REQ ( $id_v, state(v), ch_v, hp_v$ ) to  $ch_u$ 
30:     end if
31:     if ( $state(u) = CH$ )  $\wedge$  (Check(BRIDGE-REQ)) then
32:       Send BRIDGE-OK ( $id_w, state(w), ch_w, hp_w$ ) to  $v$ 
33:     end if
34:   case BRIDGE-OK
35:     if ( $state(u) = GW$ ) then
36:       Send BRIDGE-ACK ( $id_u, state(u), ch_u, hp_u$ ) to  $w$ 
37:     end if
38:   case BRIDGE-ACK
39:     if ( $state(u) = GW$ ) then
40:        $\Upsilon(u) \leftarrow$  Deposit_pheromone( $v, BRIDGE$ )       ▷ update pheromone table

```

```

41:     Send BRIDGE ( $id_u, state(u), ch_u, hp_u$ ) to  $v$ 
42:     end if
43:     case BRIDGE
44:     if ( $state(u) = GW$ ) then
45:          $Y(u) \leftarrow Deposit\_pheromone(v, BRIDGE)$       ▷ update pheromone table
46:     end if
47:     otherwise
48:         Handle_cluster_formation_messages( $v, message, k, N(u)$ )      ▷ Algorithm 3
49: end switch
    
```

---

**Algorithm 3:** cluster formation messages handling for node  $u$

---

**Require:**  $v, message, k, C(u)$

```

1: switch message do
2: case BEST
3:     if ((( $state(u) = FN$ )  $\wedge$  ( $message.ttl > 0$ ))  $\vee$  (( $state(u) = CH$ )  $\wedge$  ( $|C(u)| = 0$ ))) then
4:          $score(u) \leftarrow Calculate\_score()$       ▷ see Equation 1
5:         if ( $score(u) < message.score$ ) then
6:              $state(u) \leftarrow PM$ 
7:              $Y(u) \leftarrow Deposit\_pheromone(v, BEST)$       ▷ update pheromone table
8:              $hp_u \leftarrow message.ttl$ 
9:              $ch_u \leftarrow message.ch$       ▷ Self-proclaimed CH's ID
10:            Send AGREE( $id_u, score(u), ch_u$ ) to  $v$ 
11:             $delay\_AGREE \leftarrow Current\_time() + t_{agree}(u)$ 
12:        end if
13:    else
14:        if (( $state(u) = PM$ )  $\wedge$  ( $message.ttl > 0$ )  $\wedge$  ( $message.ch = ch_u$ )) then
15:             $message.ttl \leftarrow message.ttl - 1$ 
16:            Send message to  $w, \forall w \in N(u) \setminus \{v\}$       ▷ forward the message
17:        end if
18:    end if
19: case AGREE
20:     if ((( $state(u) = PM$ )  $\vee$  ( $state(u) = OM$ ))  $\wedge$  ( $message.ch = ch_u$ )) then
21:          $message.ttl \leftarrow message.ttl + 1$ 
22:          $\hat{\Phi} \leftarrow \arg \max_{\Phi_{ux} \in Y(u)} (Find\_pheromone(\Phi_{ux}, BEST))$ 
23:          $w \leftarrow Choose\_destination(\hat{\Phi})$ 
24:         Send message to  $w$       ▷ forward the message
25:          $Y(u) \leftarrow Deposit\_pheromone(v, AGREE)$       ▷ update pheromone table
26:     end if
27:     if (( $state(u) = SPCH$ )  $\wedge$  ( $message.ch = id_u$ )  $\wedge$  ( $message.ttl \leq k$ )) then
28:          $C(u) \leftarrow C(u) \cup \{v\}$       ▷ List of potential members
29:     end if
30: case TREE
31:     if (( $state(u) \in \{PM, OM, GW\}$ )  $\wedge$  ( $message.ch = ch_u$ )) then
32:          $state(u) \leftarrow OM$ 
33:         ( $\hat{\Phi}$ )  $\leftarrow \arg \max_{\Phi_{ux} \in Y(u)} (Find\_pheromone(\Phi_{ux}, BEST))$ 
34:         if ( $Choose\_destination(\hat{\Phi}) = v$ ) then
35:              $Y(u) \leftarrow Deposit\_pheromone(v, TREE)$       ▷ update pheromone table
36:             Send TREE-ACK ( $id_u$ ) to  $v$ 
37:         end if
38:         ( $\hat{\Phi}$ )  $\leftarrow \arg \max_{\Phi_{ux} \in Y(u)} (Find\_pheromone(\Phi_{ux}, AGREE))$ 
39:         Send message to  $Choose\_destination(\hat{\Phi})$       ▷ forward the message
    
```

```

40:   end if
41:   case TREE-ACK
42:     ( $\hat{\Phi}$ )  $\leftarrow$  arg max  $(\text{Find\_pheromone}(\Phi_{uv}, \text{AGREE}))$ 
          $\Phi_{ux \in \Upsilon(u)}$ 
43:     if  $((\text{message.ch} = ch_u) \wedge (\text{Choose\_destination}(\hat{\Phi}) = v))$  then
44:        $\Upsilon(u) \leftarrow \text{Deposit\_pheromone}(v, \text{TREE})$   $\triangleright$  update pheromone table
45:     end if
46:   end switch
    
```

---

**Algorithm 4:** affiliation timers handling for node  $u$

---

**Require:**  $\alpha, \beta, \delta, \gamma, t_{service}, t_{best}$

```

1: switch Current_time() do
2:   case delay_BEST
3:     if  $(C(u) \neq \emptyset;)$  then
4:       Send TREE( $id_u, ch_u, t_{service}$ ) to  $v, \forall v \in N(u)$ 
5:     end if
6:      $state(u) \leftarrow CH$ 
7:      $delay\_SERV \leftarrow \text{Current\_time}() + t_{service}$   $\triangleright$  service time duration
8:   case delay_JOIN
9:      $score(u) \leftarrow \text{Calculate\_score}(\alpha, \beta, \gamma)$   $\triangleright$  Equation 6
10:    Send BEST( $id_u, score(u)$ ) to  $v, \forall v \in N(u)$ 
11:     $state(u) \leftarrow SPCH$ 
12:     $delay\_BEST \leftarrow \text{Current\_time}() + t_{best}$ 
13:   case delay_AGREE
14:     if  $(state(u) = OM)$  then
15:       if  $(|N(u)| \neq |C(u)|)$  then
16:          $state(u) \leftarrow GW$   $\triangleright$  becoming a gateway
17:         Send BRIDGE-REQ ( $id_u, state(u), ch_u, hp_u$ ) to  $v, \forall v \in NG(u)$ 
18:       end if
19:     else
20:        $state(u) \leftarrow FN$ 
21:        $ch_u \leftarrow -1$ 
22:        $hp_u \leftarrow -1$ 
23:     end if
24:   case delay_SERV
25:      $state(u) \leftarrow FN$   $\triangleright$  end of the CH service time
26:      $ch_u \leftarrow -1$ 
27:      $hp_u \leftarrow -1$ 
28: end switch
    
```

---

## Performance Evaluation

To verify and validate our solution we conducted some simulations campaigns using the OMNeT++ simulator version 4.6 (Andras, 2016). We compared our results with those obtained applying 3 protocols recently proposed in the literature, namely, TTTC by Hameed *et al.* (2014) GCA by Tseng *et al.* (2015) H-DHAC by Zhu *et al.* (2015). Parameters we used are summarized in Table 1 and 2. The values of link quality metrics (PRR, SINR, LQI) were randomly and uniformly varied each 7s (simulated time) (Baccour *et al.*, 2013; Bas and Ergen,

2012; Boano *et al.*, 2010). Furthermore, we used the energy consumption model proposed by Heinzelman *et al.* (2002).

### Quality of the Topology (QoT)

This experiment aimed at evaluating the ability of each protocol to build a good topology. Strictly speaking, we deployed a series of networks composed of sensors randomly and uniformly distributed and a sink with a fixed position. The experiment was stopped when all the nodes became affiliated and adjusted their ranges.

**Table 1:** Simulation parameters

Parameter	value
Deployment zone	1000 m × 1000 m
Number of sensors	100 to 1000
Position of the Sink	(450;200)
sensors' transmission range	{15;35;54;70;83;98;117;127} m
sink's transmission range	250 m
sensors' initial energy	0.2 J
Self-discharge per second	0.1 μJ
$E_{elec}$	50 nJ/bit
$ef_s$	10 nJ/bit/m <sup>2</sup>
$e_{amp}$	0.0013 nJ/bit/m <sup>4</sup>
$d_0$	87 m
Length of messages	2000 bits
CH's service time	0.5 s

**Table 2:** Parameters for link quality dynamics

Quality	PRR <sup>a</sup>	SINR <sup>b</sup> (dB)	LQI <sup>c</sup>
Excellent	1	]30; 40]	]106; 255]
Good	]0.75; 1[	]15; 30]	]102; 106]
Mean	]0.35; 0.75]	]5; 15]	]80; 102]
Bad	[0; 0.35]	[0; 5]	[0; 80]

<sup>a</sup>Packet Received Rate

<sup>b</sup>Signal to Interference plus Noise Ratio

<sup>c</sup>Link Quality Indicator

In order to assess the effect of the number of sensors (network size) on the different metrics we varied the latter number from 100 to 1000 in steps of 100. These experiments were repeated 100 times for each population of nodes. All the results were averaged with a 95% confidence interval.

Figures 2a-g depict examples of logical topologies we obtained during such experiments for each protocol on a network composed of 300 sensor-nodes and one sink.

Besides, usual *clustering* metrics (mean number of clusters, mean cluster size, mean number of adjacent CHs), we also took into consideration :

#### Mean Clustering Coefficient

The *clustering* coefficient of a node  $u$  denoted by  $CC(u)$  is obtained using Equation 11 (Watts and Strogatz, 1998); where  $e$  denotes the links that actually exist between its  $n$  neighbours ( $n > 0$ ). The mean value is obtained by averaging each node's coefficient. This metric is a good estimator of the link density:

$$CC(u) = \frac{2 \times e}{n \times (n-1)} \quad (11)$$

#### Mean Weighted Clustering Coefficient

The weighted *clustering* coefficient of a node  $u$  is obtained using Equation 2 (Miyajima and Sakuragawa, 2014). It is aimed at evaluating the protocol's ability to delete redundant links according to some criteria. We

chose *link quality* as the main weighting criterion. The latter was estimated using *the triangle metric* by Boano *et al.* (2010). The mean value is obtained by averaging each node's coefficient.

#### Mean Topological Coefficient

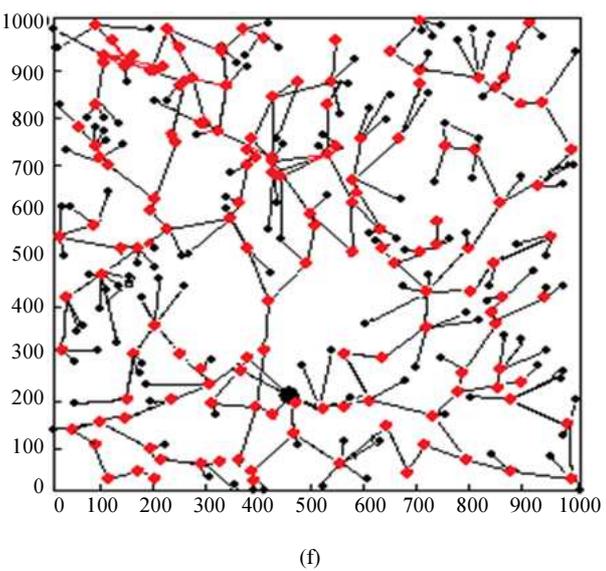
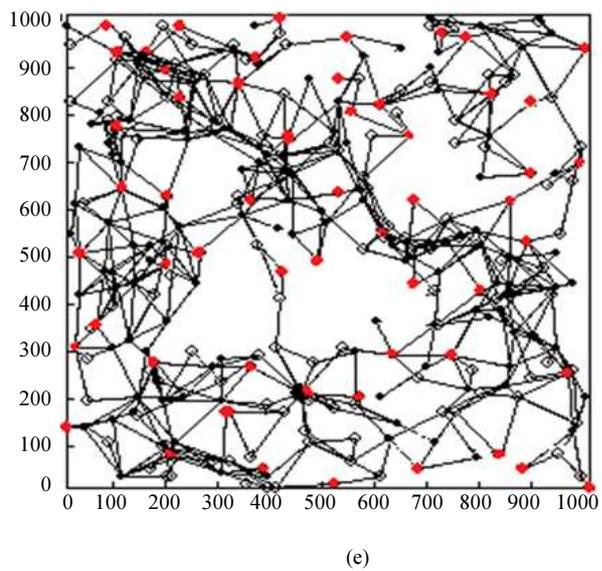
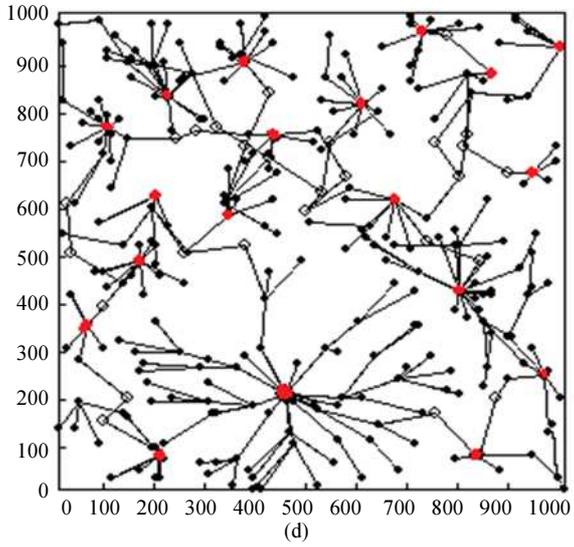
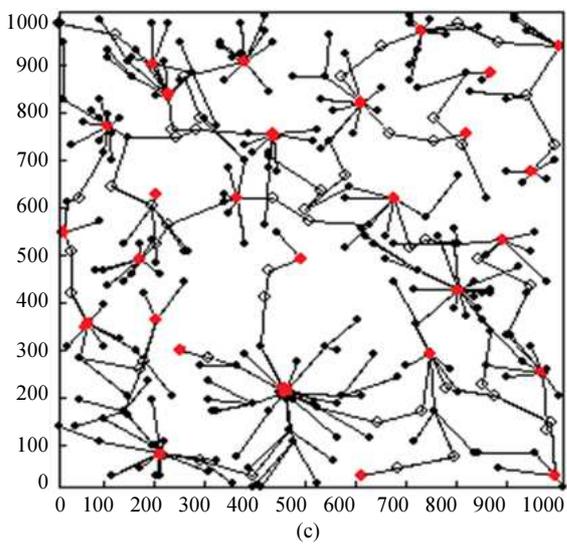
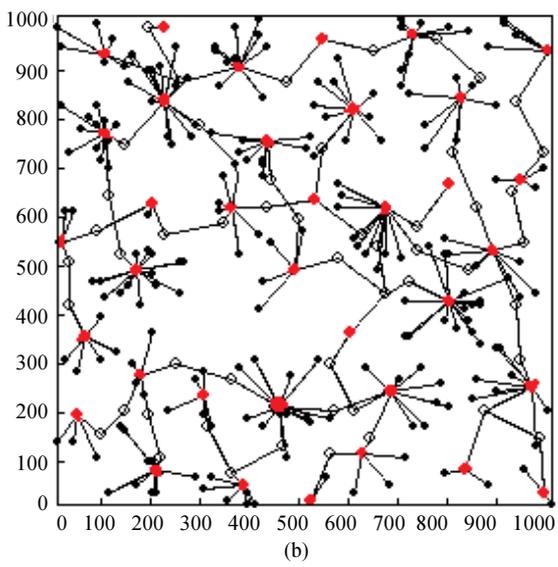
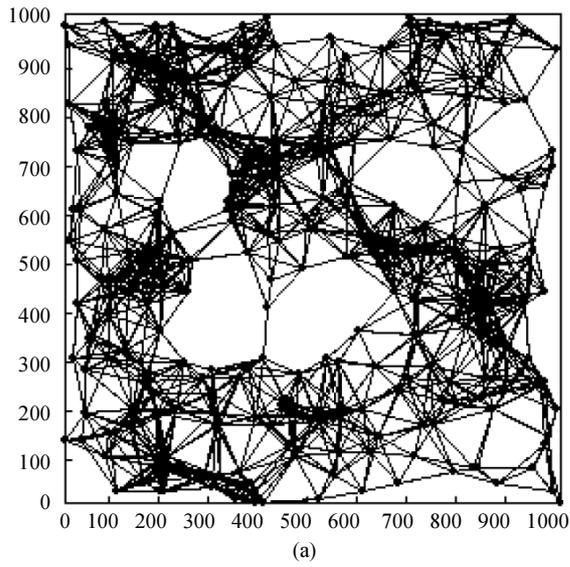
Let  $G = (V, E)$  the graph induced by the communication topology, where  $V$  denotes the set of nodes and  $E$  the set of links. The topological coefficient of a node  $u$  denoted by  $TC_u^{(k+1)}$  in its  $(k+1)$ -hop neighbourhood  $N_u^{(k+1)}$  is calculated using Equation 12 (Stelzl *et al.*, 2005):

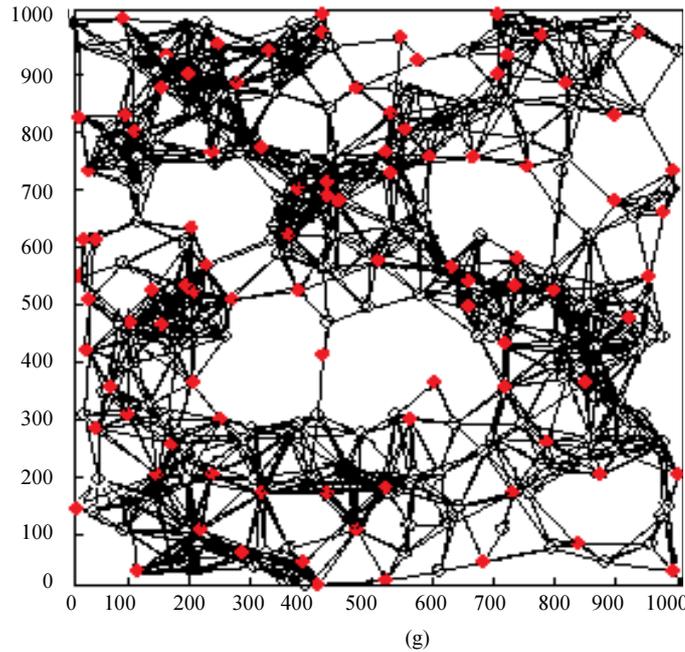
$$TC_u^{(k+1)} = \begin{cases} \frac{\sum_{v \in N_u^{(k+1)}} \eta_{u \rightarrow v}}{|N_u^{(k+1)}|^2} & , \text{if } |N_u^{(k+1)}| > 0 \\ 0 & , \text{otherwise} \end{cases} \quad (12)$$

$$\eta_{u \rightarrow v} = \begin{cases} |V_{u \rightarrow v}| + 1, & \text{if } (u, v) \in E \\ |V_{u \rightarrow v}|, & \text{otherwise} \end{cases} \quad (13)$$

with  $V_{u \rightarrow v} = \{w \in N_u^{(k+1)} : \{(u, w), (v, w)\} \in E \times E\}$

This topological coefficient helps to estimate the number of cycles that exist in a node's  $(k+1)$ -hop neighbourhood. The mean value is obtained by averaging each node's coefficient.





**Fig. 2:** Examples of logical topologies built by each of the evaluated protocols from the logical topology induced by a network composed of 300 sensor-nodes randomly and uniformly deployed on a  $1000 \times 1000$  m<sup>2</sup> area of interest. Black points represent the ordinary members, the red one are the CHs and the empty circles denote the gateways. The sink is located at (450;200) (see online version for colors) (a) The original topology (NO-TC) (b) CONTRACT( $k=1$ ) (c) CONTRACT( $k=2$ ) (d) CONTRACT( $k=3$ ) (e) GCA( $k=1$ ) (f) TTTC( $k=1$ ) (g) H-DHAC( $k=1$ )

#### Mean Logical Node Degree

The logical degree of a node denotes the size of its neighbour table. This metric aims at evaluating the protocol's ability to eliminate the 1-hop redundant logical links. The mean value is obtained by averaging each node's degree.

#### Mean Physical Node Degree

The physical degree of a node denotes the number of neighbours within its range. This metric helps to assess the ability of the protocol to minimize nodes' ranges. The mean value is obtained by averaging each node's degree.

#### Mean Transmission Range

This metric helps to corroborate the mean physical degree and the mean logical degree.

#### Energy Efficiency and Network lifetime

This series of experiments was aimed at evaluating the amount of energy consumed on average during the communication topology construction process; in other words, the evaluated protocol's ability to increase network lifetime. Nodes were deployed in the same conditions as described in the previous section. We relied on 4 types of network lifetime definitions:

- First Node Dead (FND): Time elapsed from the end of the deployment till the disappearance of a sensor-node

- First Sink's Neighbour Dead (FSND): Time interval between the end of the deployment and the disappearance of a node in sink's  $k$ -hop neighbourhood
- ASND (All Sink's Neighbours Dead): Time elapsed from the end of the deployment till the disappearance of all the sink's  $k$ -hop neighbours
- LND (Last Node Dead): Period of time between the end of the deployment and the disappearance of all the sensor-nodes

In order to measure the effect of the network size on these metrics we varied the sensor-nodes population from 100 to 1000 in steps of 100. All of these experiments were repeated 100 times for each population. The results were averaged with a 95% confidence interval.

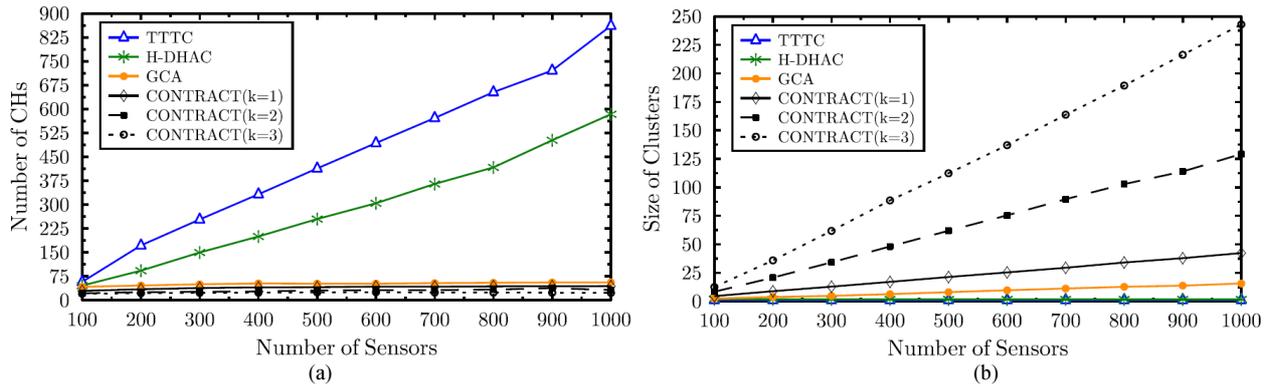
## Results and Discussion

In this section, we analyze and explain the results of the experiments we described in the previous sections.

#### Quality of Topology (QoT)

##### Mean Number of Clusters

Figure 3a shows that CONTRACT and GCA are the two protocols that build the topologies with the fewest clusters; e.g., for  $k = 1$ , this number varies respectively from 30 to 45 and from 43 to 56 proportionally to the network size. On the other hand, with H-DHAC and TTTC, the number of clusters is quite higher.



**Fig. 3:** Cluster properties vs. network size: (a) mean number of clusters; (b) mean cluster size

CONTRACT and GCA tend to create larger clusters; hence the low number of clusters.

An analysis of the mean cluster size provides further explanations.

#### Mean Cluster Size

Figure 3b shows some results that corroborate those related to the previous metrics. Indeed, TTTC and H-DHAC build small clusters (2 to 5 members on average). TTTC aims at building a *backbone* composed of the largest number of nodes (adjacent CHs) while with H-DHAC the self-proclaimed CHs tend to create clusters with a unique member in particular when link quality is bad. This is due to the fact that H-DHAC does not consider link quality as a CH selection criterion.

By contrast, the *clustering* process of GCA is aimed at constructing a Relative Neighbourhood Graph (RNG). Clusters are created by merging several cycles. This strategy results in the integration of many members; since the size of the clusters is not bounded. This is also the case with CONTRACT which indirectly limits the size of clusters by controlling the CH-to-member distance ( $k$ ).

However, CONTRACT builds larger clusters than those of GCA on average, because of the *Sink-As-Cluster-Head* strategy that was applied in the sink's  $k$ -hop neighbourhood (the *core network*).

Indeed, especially when  $k > 1$  this strategy helps to gather in the same cluster a large number of sensor-nodes under the control of the sink.

#### Mean Number of Adjacent CHs

Figure 4a shows that GCA and CONTRACT are the two protocols which help to avoid having in the final topology adjacent CHs. This is because the latter issue was explicitly addressed in these protocols. By contrast, H-DHAC and especially TTTC aim at constructing a *backbone* precisely composed of adjacent CHs.

Figure 4b suggests that all the protocols evaluated build topologies composed of very few isolated CHs even when node density is low. However, the number of isolated CHs are the lowest when using CONTRACT

and H-DHAC. This is due to the fact that this problem was explicitly addressed by both protocols.

Indeed, with CONTRACT and H-DHAC a process was designed to allow isolated CHs to join neighbouring clusters especially when they receive an invitation message from an ongoing election process.

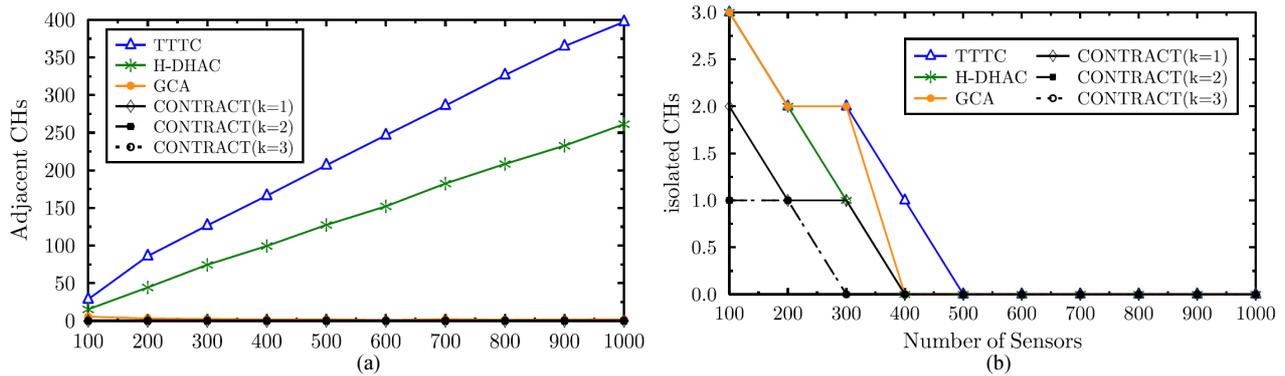
#### Mean Clustering Coefficient

Figure 5a suggests that CONTRACT allows to obtain the lowest mean coefficients, despite the increase of the size of the network and parameter  $k$ . In other words, CONTRACT is the protocol that best eliminates redundant links.

These results are due, on the one hand, to the integration of the *clustering* coefficient in CH selection criteria and, on the other hand, to the pheromone-based link marking strategy that helps to avoid having triads by preventing the delivery of two copies of the same message during the cluster formation process. H-DHAC is the protocol that eliminates the least redundant links as shown by the example depicted by Figure 2g. H-DHAC builds topologies that are close to the original graph (NO-TC). The reason for this performance is that H-DHAC uses a power control protocol namely, ATPC (Lin *et al.*, 2016), just to preserve link symmetry (in terms of quality) and not to explicitly try to minimize link redundancy.

#### Mean Weighted Clustering Coefficient

Figure 5b shows the ability of all the evaluated protocols to prune bad quality links; since the weighting criterion is related to link quality. CONTRACT has the best mean coefficients (0.2 on average) while other protocols' results are close to those of the original graph (NO-TC). These results are also due to the integration of the weighted *clustering* coefficient in the CH selection criteria. Unlike CONTRACT, other protocols struggle to prune bad and redundant links because none of these solutions have explicitly considered link quality while eliminating redundancies.



**Fig. 4:** CH's properties vs. network size: (a) mean number of isolated CHs; (b) mean number of adjacent CHs

### Mean Topological Coefficient

Figure 5c suggests that CONTRACT builds topologies with the lowest mean topological coefficients. In other words, CONTRACT is the protocol that best prevents the creation of cycles in nodes'  $(k + 1)$ -hop neighbourhood.

These results are due, on the one hand, to the fact that CONTRACT prevents the creation of triads inside the clusters; and on the other hand, to the rules applied in the cluster interconnection process. Indeed, the latter prohibit the creation of a link (bridge) between two gateways if such a link already exists between their respective clusters. Such a scheme does not exist in the other protocols in particular GCA which constructs a RNG; this type of graph allows only the pruning of the 1-hop redundant links and leave therefore several cycles in nodes'  $k$ -hop neighbourhood when  $k > 1$ .

### Mean Logical Degree

Figure 6a it can be observed that CONTRACT helps to build topologies with the lowest mean logical degrees regardless of network size or parameter  $k$ . In other words, CONTRACT helps best to minimize nodes' neighbour table. These results are also due to the stigmergy-based redundancies elimination strategy we used for the intra-cluster spanning tree construction process. By contrast, H-DHAC and TTTC help to obtain mean logical degrees close to those of the original graph (NO-TC). Indeed, power control in H-DHAC helps only to improve link quality and is not explicitly aimed at neighbour table size minimization.

As for TTTC, it is mainly aimed at building a *backbone* composed of CHs with small degrees. GCA is based on a relative neighbourhood graph and avoids having triads; however, the link recreation process at the end of the *clustering* process makes several redundant links to reappear in the final topology and therefore increase the logical degree.

### Mean Physical Degree

Figure 6b suggests that CONTRACT helps to build topologies where the mean physical degree remains low

despite the increase of network size and parameter  $k$ . These results are close to those of GCA which considers only link length. They are also due to the stigmergy-based strategy that helps to keep links providing the best trade-off between length and quality. A discussion of the mean transmission range will help to better understand these results.

### Mean Transmission Range

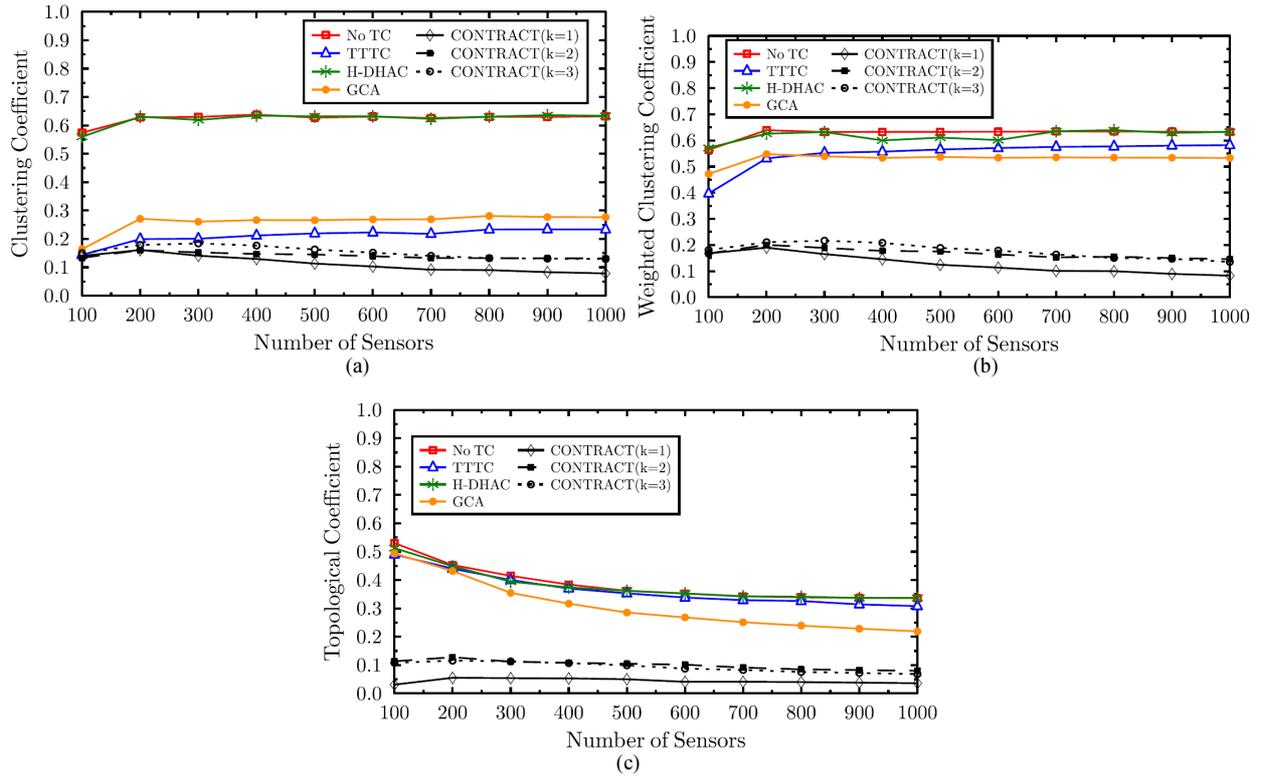
The results shown in Figure 6c illustrate that length does not correlate with quality. Indeed, the minimum transmission range of a node is only related to the euclidian distance (length) from its farthest logical neighbour. When the link metric is the length like with GCA, the mean transmission range decreases in terms of network size. However, when the link metric is related to both length and quality, like with CONTRACT, network size and parameter  $k$  have less influence on the mean transmission range.

While eliminating redundancies and preserving connectivity CONTRACT does not look for the shortest transmission range but rather endeavours to find the useful one. In other words, CONTRACT tries to provide to each node the transmission range that can help it to have low delay and energy-efficient communications.

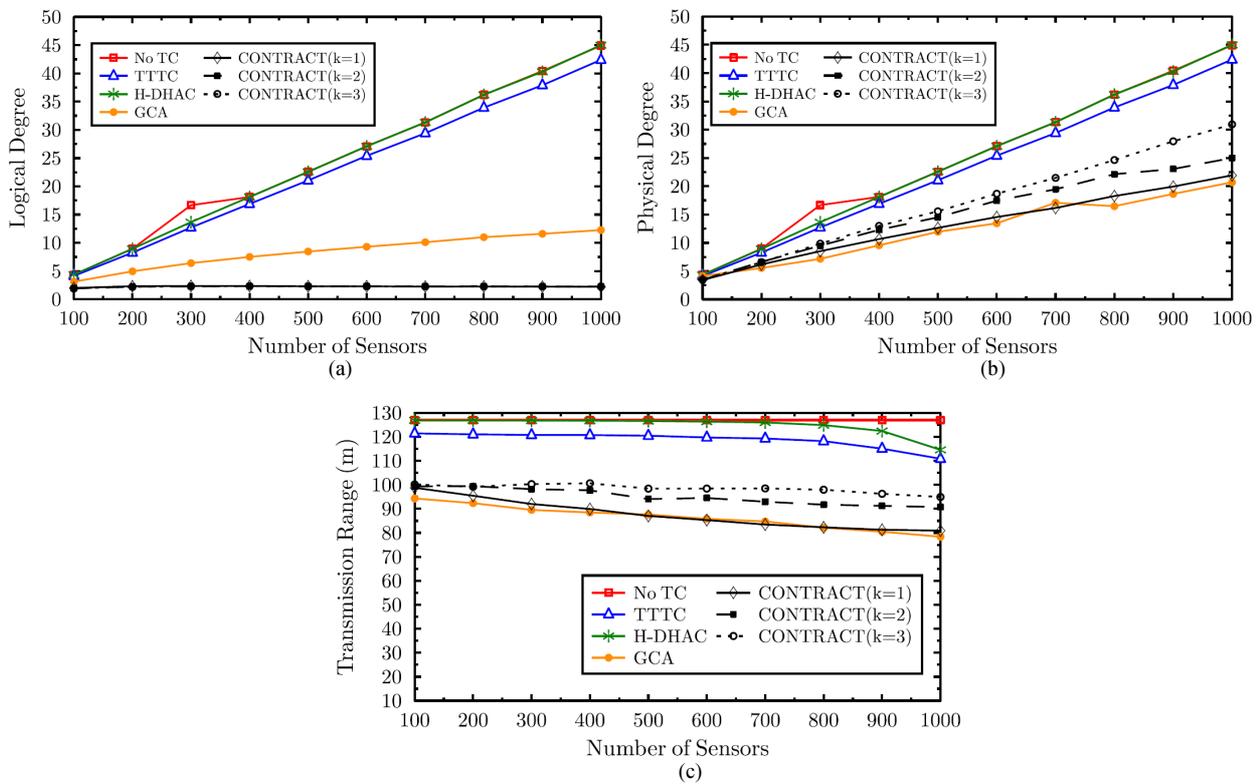
### Energy Efficiency and Network Lifetime

Figures 7a and 7b show that regardless of the protocol used, network lifetime is inversely proportional to network size. Indeed, when their physical degrees increase nodes tend to consume a lot of energy while communicating with their neighbours. However, CONTRACT is the protocol that obtains the longest lifetimes regardless of the definition. These results are due to the quality of the topologies built by CONTRACT as discussed in the previous section.

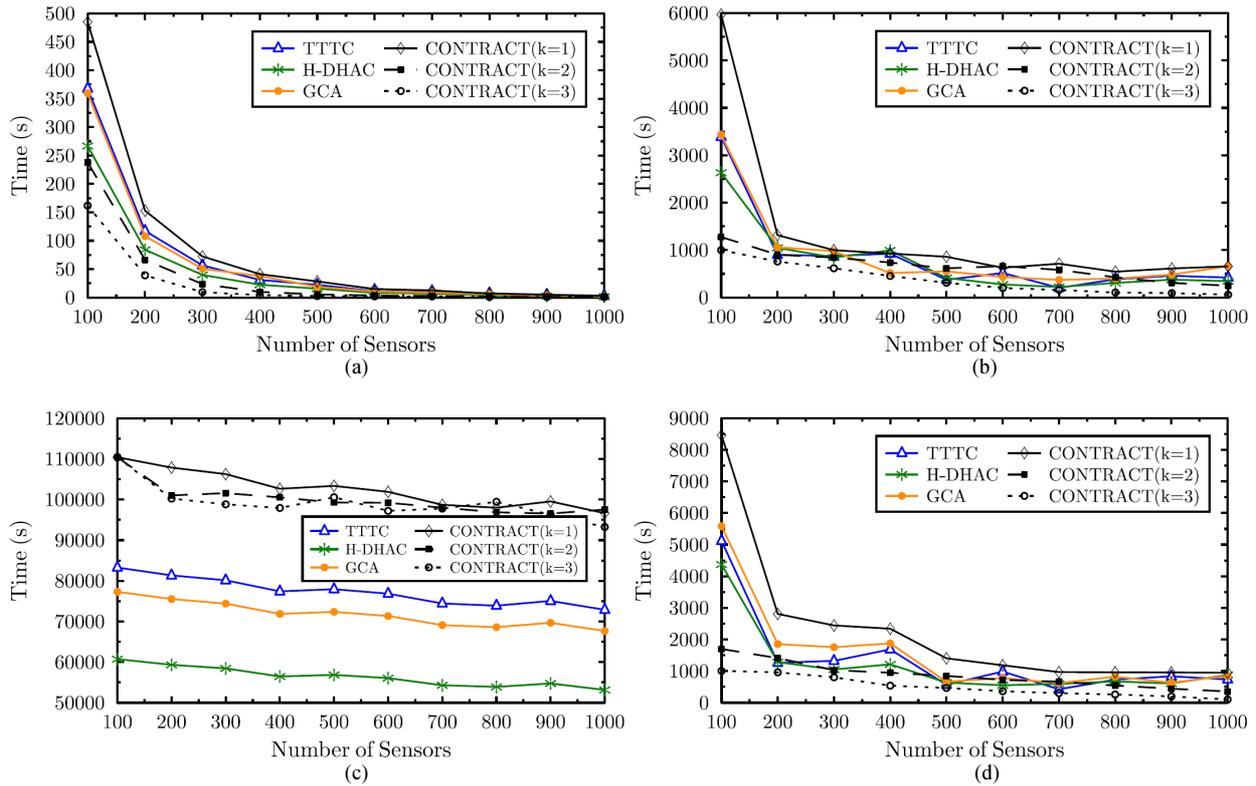
The integration of residual energy into the CH selection criteria, cycles elimination in nodes'  $(k+1)$ -hop neighbourhood, the choice of links allowing a good trade-off between length and quality etc. are the factors that contribute to energy waste minimization. However, the results discussed in the previous section proved that CONTRACT outperforms other protocols according to these metrics.



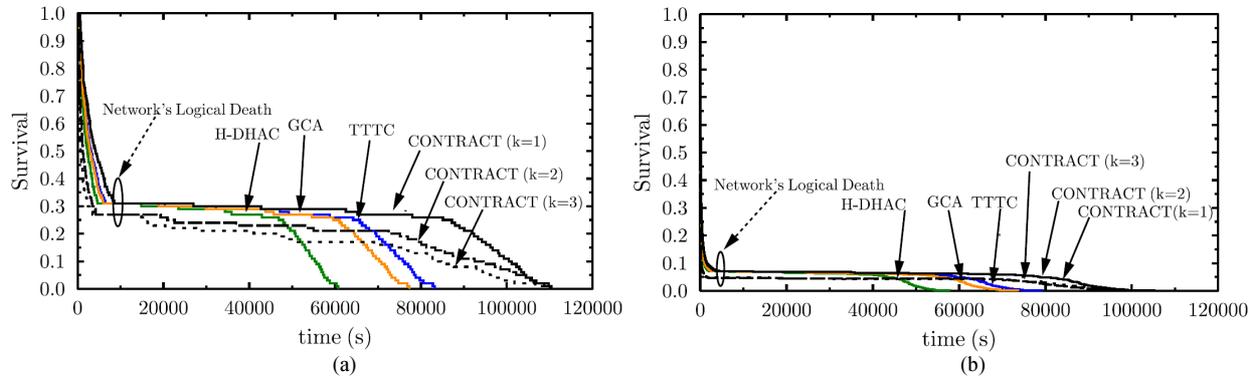
**Fig. 5:** Topology sparseness vs. network size: (a) mean *clustering* coefficient; (b) mean weighted *clustering* coefficient; (c) mean topological coefficient



**Fig. 6:** Node degree vs. network size: (a) mean logical degree; (b) mean physical degree; (c) mean transmission range



**Fig. 7:** Network lifetime vs. network size: (a) until First Node Dies (FND); (b) until First Sink's Neighbour Dies (FSND); (c) until Last Node Dies (d) until All Sink's Neighbours Die (ASND)



**Fig. 8:** Node survival analysis: (a) network size  $n = 100$ ; (b) network size  $n = 500$

Futhermore, as depicted in Figures 7a and 7b it can be network lifetime is influenced by parameter  $k$  since it impacts among other things, the cluster size (Figure 3b). This is particularly true in the *core network* i.e. the sink's neighbourhood (see Figures 7b and 7d). Hence, it worth building and maintaining in this area only a 1-hop cluster when using the *Sink-As-Cluster-Head* scheme.

Nodes' Kaplan-Meir survival curves depicted in Figure 8a and 8b show that CONTRACT is also the protocol that best increases nodes' survival capacity. In other words, CONTRACT is among these protocols the

one that depletes the least nodes' energy. These results are due to the schemes we applied such as CH service time limitation, number of elections minimization (especially in the *core network*), adjacent or isolated CHs minimization, trade-off between link length and link quality that minimizes signaling and helps to provide each node with an energy-efficient and delay minimization transmission range. As shown in the previous discussions, such features do not exist in the other evaluated protocols. GCA of course also allows CH service time limitation, but CONTRACT imposes a

probationary period to CHs after the expiration of their mandates and during the selection process, considers each node's *instantaneous instability* (i.e. its probability of being a stable CH after a number of mandates).

## Conclusion

In order to effectively minimize energy losses in a dense wireless sensor network, we decided to eliminate link redundancies by operating on both logical and physical topologies. To that end, we proposed CONTRACT an asynchronous localized message passing protocol that combines respectively two well-known connectivity control approaches, *clustering* and power control based on *stigmergy*, a paradigm for ant colony optimization and metaheuristics conception. This solution helps to avoid common topological defects (adjacent CHs, isolated CHs,...), prevents early energy holes in the vicinity of the sink, keeps the CH-to-member distance to at most  $k$ -hops ( $k \geq 1$ ), and provides each node with the transmission range that guarantees minimum delay, energy-efficiency and minimizes signalization. The combination of these features increases nodes' survival capacity and prolongs network lifetime.

For future work, we plan to extend this solution by addressing the problems of fault tolerance and communication channel overhearing minimization.

## Acknowledgement

The authors feel grateful to the anonymous reviewers for their valuable comments and suggestions to improve the quality of this paper and would like to thank them from the bottom of their hearts.

## Author's Contributions

**Gokou Hervé Fabrice Diédié and Boko Aka:** Contributed to design the research plan and writing the manuscript.

**Michel Babri:** Coordinated the data-analysis and contributed to the writing of the manuscript.

## Ethics

The authors declare that there is no ethical issues that may arise after the publication of this manuscript.

## References

Andras, V., 2016. <https://omnetpp.org>  
Baccour, N., A. Koubaa, C. Noda, H. Fotouhi and M. Alves *et al.*, 2013. Radio Link Quality Estimation in Low-Power Wireless Networks. 1st Edn., Springer Science and Business Media, ISBN-10: 3319007742, pp: 147.

Bas, C.U. and S.C. Ergen, 2012. Spatio-temporal characteristics of link quality in wireless sensor networks. Proceedings of the IEEE Wireless Communications and Networking Conference, Apri. 1-4, IEEE Xplore Press, Shanghai, China. DOI: 10.1109/WCNC.2012.6213950  
Boano, C.A., M.A. Zuniga, T. Voigt, A. Willig and K. Romer, 2010. The triangle metric: Fast link quality estimation for mobile wireless sensor networks. Proceedings of the 19th International Conference on Computer Communications and Networks, Aug. 2-5, IEEE Xplore Press, Zurich, Switzerland. DOI: 10.1109/ICCCN.2010.5560118  
Cardei, M., M.O. Pervaiz and I. Cardei, 2006. Energy-efficient range assignment in heterogeneous wireless sensor networks. Proceedings of the International Conference on Wireless and Mobile Communications, Jul. 29-31, IEEE Xplore Press, Bucharest, Romania. DOI: 10.1109/ICWMC.2006.44  
Chiasserini, C.F., I. Chlamtac, P. Monti and A. Nucci, 2004. An energy-efficient method for nodes assignment in cluster-based ad hoc networks. Wireless Netw., 10: 223-231.  
Clementi, A.E.F., P. Penna and R. Silvestri, 1999. Hardness results for the power range assignment problem in packet radio networks. Proceedings of the 3rd International Workshop on Approximation Algorithms for Combinatorial Optimization Problems: Randomization, Approximation, and Combinatorial Algorithms and Techniques, Aug. 8-11, Springer Berlin Heidelberg, pp: 197-208.  
Diédié, H.G., M. Babri and B. Aka, 2016. Efficient cluster-based self-organization scheme for connectivity control in wireless sensor networks. J. Commun., 11: 632-643. DOI: 10.12720/jcm.11.7.632-643  
Dorigo, M., E. Bonabeau and G. Theraulaz, 2000. Ant algorithms and stigmergy. Future Generation Comput. Syst., 16: 851-871. DOI: 10.1016/S0167-739X(00)00042-X  
Garey, M.R., D.S. Johnson and R. Sethi, 1976. The complexity of flowshop and jobshop scheduling. Math. Operat. Res., 1: 117-129. DOI: 10.1287/moor.1.2.117  
Grassé, P.P., 1959. La reconstruction du nid et les coordinations interindividuelles chez *Bellicositermes natalensis* et *Cubitermes* sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites. Insectes Sociaux, 6: 41-80.  
Hameed, Z., K.W. Lim and Y.B. Ko, 2014. On the interplay between *clustering* and power control in multihop wireless networks. Proceedings of the 11th International Conference on Computer Systems and Applications, Nov. 10-13, IEEE Xplore Press, Doha, Qatar. DOI: 10.1109/AICCSA.2014.7073214

- Heinzelman, W.B., A.P. Chandrakasan and H. Balakrishnan, 2002. An applicationspecific protocol architecture for wireless microsensor networks. *IEEE Trans. Wireless Commun.*, 1: 660-670.
- Hu, S. and J. Han, 2014. Power control strategy for *clustering* wireless sensor networks based on multi-packet reception. *IET Wireless Sensor Syst.*, 4: 122-129.  
DOI: 10.1049/iet-wss.2013.0108
- Jain, A. and B.V.R. Reddy, 2014. Sink as cluster head: An energy efficient *clustering* method for wireless sensor networks. *Proceedings of the International Conference on Data Mining and Intelligent Computing*, Sept. 5-6, New Delhi.  
DOI: 10.1109/ICDMIC.2014.6954261
- Jameii, S.M. and M. Maadani, 2016. Intelligent dynamic connectivity control algorithm for cluster-based wireless sensor networks. *Proceedings of the 11th International Conference for Internet Technology and Secured Transactions*, Dec. 5-7, IEEE Xplore Press, Barcelona, pp: 416-420.  
DOI: 10.1109/ICITST.2016.7856744
- Karp, R.M., 2011. Computational complexity of combinatorial and graph-theoretic problems. *Theoretical Comput. Sci.*
- Kawadia, V. and P.R. Kumar, 2003. Power control and *clustering* in ad hoc networks. *Proceedings of the Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies*, Mar. 30-April 3, IEEE Xplore Press, San Francisco.  
DOI: 10.1109/INFCOM.2003.1208697
- Khuong, A., J. Gautrais, A. Perna, C. Sba and M. Combe *et al.*, 2016. Stigmergic construction and topochemical information shape ant nest architecture. *National Acad. Sci.*, 113: 1303-1308.  
DOI: 10.1073/pnas.1509829113
- Kruskal, J.B., 1956. On the shortest spanning subtree of a graph and the traveling salesman problem. *Am. Math. Society*, 7: 48-48. DOI: 10.2307/2033241
- Labrador, M.A. and P.M. Wightman, 2010. *Topology Control in Wireless Sensor Networks: With a Companion Simulation Tool for Teaching and Research*. 1st Edn., Springer Science and Business Media, Dordrecht, ISBN-10: 1402095856, pp: 209.
- Li, J., X. Hou, D. Su and J.D.D. Munyemana, 2017. Fuzzy poweroptimised *clustering* routing algorithm for wireless sensor networks. *IET Wireless Sensor Syst.*, 7: 130-137.
- Li, M., Z. Li and A.V. Vasilakos, 2013. A survey on topology control in wireless sensor networks: Taxonomy, comparative study and open issues. *Proc. IEEE*, 101: 2538-2557.
- Li, N., J.C. Hou and L. Sha, 2005. Design and analysis of an MST-based topology control algorithm. *IEEE Trans. Wireless Commun.*, 4: 1195-1206.
- Lin, S., F. Miao, J. Zhang, G. Zhou and L. Gu *et al.*, 2016. Atpc: Adaptive transmission power control for wireless sensor networks. *ACM Trans. Sensor Netw.*, 12: 1-31.
- Liu, C.H., B. Rong and S. Cui, 2015. Optimal discrete power control in poisson-clustered ad hoc networks. *IEEE Trans. Wireless Commun.*, 14: 138-151.
- Manousakis, K. and J.S. Baras, 2003. *Clustering* for transmission range control and connectivity assurance for self configured ad hoc networks. *Proceedings of the IEEE Military Communications Conference*, Oct. 13-16, IEEE Xplore Press, Boston. DOI: 10.1109/MILCOM.2003.1290311
- Matthias, M. and F. Pfeiffer, 1993. On the complexity of the disjoint paths problem. *Combinatorica*, 13: 97-107.
- Miyajima, K. and T. Sakuragawa, 2014. Continuous and robust *clustering* coefficients for weighted and directed networks.
- Prim, R.C., 1957. Shortest connection networks and some generalizations. *Bell Syst. Technical J.*, 36: 1389-1401.
- Romesburg, C., 2004. *Cluster analysis for researchers*. Lulu.com.
- Stelzl, U., U. Worm, M. Lalowski, C. Haenig and F.H. Brembeck *et al.*, 2005. A human protein-protein interaction network: A resource for annotating the proteome. *Cell*, 122: 957-968.
- Tseng, C.C., K.C. Ting, H.C. Wang, F.C. Kuo and L.H. Chang, 2015. Construction and analysis of a green clustered architecture for RNG-based wireless ad hoc networks. *Int. J. Ad Hoc Ubiquitous Comput.*, 19: 62-2.
- Vinutha, C.B., N. Nalini and M. Nagaraja, 2017. Cluster-based adaptive power control protocol using hidden Markov model for wireless sensor networks. *Int. J. Electron.*, 104: 968-981.
- Vural, S., P. Navaratnam and R. Tafazolli, 2013. Transmission range assignment for *backbone* connectivity in clustered wireless networks. *IEEE Wireless Commun. Lett.*, 2: 46-49.
- Watts, D.J. and S.H. Strogatz, 1998. Collective dynamics of 'small-world' networks. *Nature*, 393: 440-442.  
DOI: 10.1038/30918
- Zhu, J., C.H. Lung and V. Srivastava, 2015. A hybrid *clustering* technique using quantitative and qualitative data for wireless sensor networks. *Ad Hoc Netw.*, 25: 38-53.