A Hierarchical LLC Routing Algorithm for WSNs

Cesare Alippi, Romolo Camplani, Manuel Roveri

Politecnico di Milano, Dipartimento di Elettronica e Informazione, Milano, Italy
{alippi, camplani, roveri}@elet.polimi.it

Abstract — In WSNs and tiny cooperating wireless robots, unit and communication faults (permanent or transient), energy availability for the units and environmental changes may modify the communication network topology over time. Traditional hierarchical routing algorithms combine adaptability to changing environments with energy-aware aspects. In this paper we propose a $k$-level hierarchical extension to Low-Power Localized Clustering (LLC). For the first time, we take into account the effects introduced by finite unit bandwidth on the routing capabilities of the proposed algorithm. A second novel content of the paper is that, differently from the existing algorithms, the proposed solution guarantees a uniform distribution of alive units in the deployment area. This feature is particularly appealing since it is associated with the QoS of the network over time. The effectiveness of the suggest approach has been validated with a large experimental campaign.

I. INTRODUCTION

Development of smart routing algorithms is a must for granting effective communications in large scale wireless networks with adaptation ability being the key issue combined with energy-aware aspects. To deal with this issue self-organizing routing algorithms have been specifically developed. In WSNs, these algorithms grant adaptability to environmental changes by providing a quick reaction to topological modification at the network level and, particularly interesting, can support an easy node add-on and removal modality with a tunable frequency. In this direction, routing algorithms suggested in [1] [2] [3] [4] [5] aim at reducing the overall consumption of the network energy defined as the sum of residual energies of nodes [6]. These power-aware algorithms share a common basic philosophy: the network topology is partitioned into clusters. Each cluster is ruled by a cluster head that coordinates the cluster nodes, e.g., by receiving messages from the cluster nodes with a single hop communication modality. Each cluster head is then directly connected to the base station with a star topology and sends acquired data through a single hop communication mechanism.

Among these algorithms, we focus on LEACH [1], LC [2], LLC [3] for their adaptability to changing environments (making them effective in the dynamic insertion or deletion of nodes) and for their scalability (in terms of number of nodes). LEACH, which was one of the first hierarchical routing algorithms proposed in the literature, is based on a simple election mechanism in which each node decides autonomously to become cluster head according to a threshold (that changes over time).

Differently, in LC the election of the cluster heads is not taken autonomously by each node but is based on the amount of messages exchanged among nodes during the election phase. Moreover, LC uses information about residual energy of nodes in the election mechanism. LLC, which is an extension of LC, reduces the impact of messages overhead by imposing a maximum value $x$ on the percentage of nodes that may become cluster heads w.r.t. the total amount of nodes (the parameter must be fixed at design time). The probability that node $j$ is considered in the election mechanism at time $t_r$ is proportional to the residual energy $e_j$:[3]:

$$p_j = \frac{e_j t x}{e (t - t_r)}$$ (1)

where $e$ is the initial energy of the node and $t$ is the expected network lifetime. Unfortunately, the considered routing algorithms (as other algorithms present in the literature) assume the unrealistic hypothesis of infinite (or very large) bandwidth: this is a main issue since it is hardly satisfied in many monitoring applications. Moreover, selection of the algorithm parameters might be critical. In fact, some parameter configurations proposed for LEACH, LC and LLC, such as cluster radius or cluster heads percentage, are not realistic for most applications due to the unknown topology of the deployment area and to finite bandwidth issues. An example of adaptive cluster radius estimation for LLC can be found in [7].

In this paper we propose the $X$-LLC algorithm which extends LLC by considering a $k$-level hierarchy for the sensor nodes and takes into account bandwidth constraints. The algorithm guarantees a more uniform distribution of alive nodes in the monitoring area yet aiming at further minimizing the energy consumption of the whole network. The basic tree-like structure is such that sensor nodes send data to the first level cluster heads that, in turn forward them (together with their acquisitions) to the 2-nd level cluster heads up to the $k$-the level where, finally, cluster heads send data collected over the tree to the base station. The structure of the cluster heads is hierarchical: a cluster head belongs only to one level of clusters and is associated only to one upper level cluster head. The introduction of intermediate levels between simple nodes and base
station reduces the energy consumption since the distance between cluster heads of different levels is always smaller than that between traditional cluster heads and base station (short range activity). A further significant advantage of a multilevel solution is that cluster heads can envisage compression techniques with a further power consumption reduction.

In the suggested algorithm identification of clusters and election of cluster heads (at the different levels) is performed with a simple -yet effective- low power consuming distributed algorithm which provides the network with high adaptation ability to the changing environment. A second novel content of the paper is that, differently from the existing literature, we take into account effects posed by finite node bandwidth which traditional algorithms unrealistically assume to be infinite. The bandwidth limit is critical in WSNs nodes up to the point that available routing algorithms are unfeasible on current day node technology for a large class of applications.

Cooperative robots systems and swarm robots share common networking problems of WSNs (e.g., robot and communication failures, environmental changes, energy savings). The proposed routing algorithm can thus be envisaged also in such systems to increase the adaptivity to changes as well as granting an elongated life by reducing the power consumption of the routing protocol.

The structure of the paper is as follows. Section II presents the extension to LCC with the multi level hierarchical architecture with bandwidth, power consumption and load balancing constraints. Experimental results are given in Section III.

II. A K-LEVEL LOW ENERGY LOCALIZED CLUSTERING

In this Section we extend LLC to a k-level hierarchical algorithm to improve network energy management; selection of LLC derives from the fact that it generally outperforms LEACH and LC in terms of the network lifetime [3] and [8]. In addition to the multi-level hierarchical extension, the algorithm we propose also considers node bandwidth limits to bound selection of the network parameters.

A. Set-up Model

We can safely assume that the antenna of a generic WSN node generates an isotropic spherical electromagnetic field [1].

The base station is central to the environment to better exploit communication coverage and we assume that nodes are uniformly distributed within the deployment area with a superficial distribution density \( \delta \).

A finite number \( W \) of transmission power levels are usable by the nodes (usually 8/10 levels in commercial units) and fixed by technological constraints. Denote by \( P_w \) and \( R_w \) the \( w \)-th power level and the associated transmission radius of the communication neighbour according to the first order communication model, respectively; for power level \( W \) we have the maximum transmission power \( P_W = P_{MAX} \) and the maximum transmission radius \( R_W = R_{MAX} \). Since \( P_w \leq P_{w+1} \), \( \forall w \in 1, \ldots, l_p \) we have that \( R_w \leq R_{w+1} \).

The expected number of nodes within the reachable environment is \( N_0 = \frac{l_p \delta \pi R_{MAX}^2}{\delta} \); for larger networks the above must be intended for each subnetwork communicating each other with a multi-hop communication.

B. X-LLC: the Proposed Routing Algorithm

We propose an autonomous and adaptive hierarchical routing algorithm, named X-LLC, that is a multi-level extension of LLC. The proposed multi-level extension is particularly appealing for reducing the energy consumption of LLC since it allows a short range transmission activity. Moreover, the proposed algorithm grants adaptability to changing environments (since it automatically provides quick reaction to topological changes in the network by instructing new clustering configurations) and is autonomous in the routing decision (the routing algorithm is not centralized and nodes perform local decisions to select the optimal path route).

Traditional hierarchical routing algorithms consider only one level of cluster heads. Here, we propose to extend the LLC algorithm so has to have \( k \geq 1 \) intermediate levels of cluster heads by considering \( k \) distinct LLC election phases.

Each election phase \( i \) elects cluster heads of the \( i \)-th level. Iteratively, the \( i \)-th level cluster heads participate to the election of the \( i + 1 \)-th level cluster heads. The process iterates up to the \( k \)-th level. Extension to \( k \) levels implies that cluster heads at level \( i \) - 1 send data to cluster heads at level \( i \) with a star network topology, level 0 are sensor nodes while cluster heads of the last \( k \) level send packets directly to the base station. Cluster heads, in addition to communication and data, also provide sensorial data acquisition.

The suggested X-LLC algorithm is given in Algorithm 1 while the nomenclature of the symbols is presented in Table I. Being X-LLC a distributed algorithm, the routing algorithm is executed by each node when participating to the election.

Each node generates a random value \( u \) between 0 and 1 (Step 2) that is compared (Step 3) with a threshold \( P_{ijr} \) defined in (1): if \( P_{ijr} \) is greater than \( u \), the node becomes a candidate cluster head and participates to the election phase; otherwise it stays silent (Step 12) until the election process terminates. Each candidate node broadcasts an advertisement in the environment.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N_w )</td>
<td>total number of nodes</td>
</tr>
<tr>
<td>( S_i )</td>
<td>the set of nodes that may participate to elections at the ( i )-th level</td>
</tr>
<tr>
<td>( n_{jir} )</td>
<td>the ( j )-th node in ( S_i ) at election round ( r )</td>
</tr>
<tr>
<td>( w_{jir} )</td>
<td>the residual energy of ( n_{jir} )</td>
</tr>
<tr>
<td>( P_{ijr} )</td>
<td>probability that ( n_{jir} ) participate to the election mechanism (computed with Equation 1)</td>
</tr>
<tr>
<td>( R_w )</td>
<td>the transmission radius at the ( w )-th transmission power level</td>
</tr>
<tr>
<td>( t_{election} )</td>
<td>the time required for the procedure completion</td>
</tr>
</tbody>
</table>

TABLE I

SYMBOLS USED IN ALGORITHM 1
message with transmission power $P_w$ and, consequently, covers a spatial neighborhood of radius $R_w$. Then, each candidate node collects the advertisement messages coming from the other candidate nodes in the neighborhood (Step 7) and sets a promotion timer $T_h$ that is function of the amount of received messages and the node residual energy (Step 19). When $T_h$ expires the candidate node becomes a cluster head at level $i$ and broadcasts an advertisement message with transmission power $P_w$. If $T_h$ has not yet expired and a candidate node receives an advertisement message (coming from a candidate node that becomes cluster head) (Step 18), it interrupts the promotion timer and waits until the election process terminates (Step 19). Nodes that become cluster heads at the $(i)$-th level participate to the election of the $(i+1)$-th level cluster heads. Cluster heads not elected at the $(i)$-th level simply remain $(i-1)$-th level cluster heads. The election phase is then iterated up to the $k$-th level. To allow transmissions to higher distances, the value of the transmission power $P_w$ has to increase with $i$. Selection of $P_w$ and $i$, which depends on technological constraints and is application-dependent, can be taken by suitably mapping the number of levels available for the transmission power $W$ to the number of cluster heads levels $k$.

The association phase, which starts after the completion of the election process, is composed by $k$ specific association sub-phases which are performed in a top-down fashion: starting from the base station to simples nodes. In the first association phase, the $k$-th level cluster heads associate themselves to the base station which will send back their TDMA table. This process iterates up to the associated $(k-1)$-th level cluster heads. This process iterates up to the sensor nodes level.

**Algorithm 1 electClusterHeads**: $i, r, n_{jir}$

1: $m = 0$
2: $u = rand \{U(0,1)\}$
3: if $p_{jir} \geq u$ then
4:   enableTimer($T_h$);
5:   send ($msg_{ADV}, R_w$);
6: while $isNotExpired(T_h)$ do
7:   if receiveMessage ($msg_{ADV}$) then
8:      $m = m + 1$
9:   end if
10: end while
11: else
12:   sleep ($t_{election}$)
13:   return
14: end if
15: $T_h = calculatePromotionTimer (m, e_{ir})$
16: enableTimer($T_h$);
17: while $isNotExpired(T_h)$ do
18:   if receiveMessage ($msg_{ADV}$) then
19:      sleep ($t_{election}$)
20:   end if
21: end while
22: if $i < k$ then
23:   electClusterHeads ($i + 1, r, n_{jir}$)
24: else
25:   associateNodes: ($i + 1, r, n_{jir}$)
26: end if

**Fig. 1.** Hierarchical structure of the sensor network

$X$-LLC provides a remarkable advantage in terms of transmission energy consumption (once compared with traditional hierarchical algorithms) since reduces the cluster size in terms of radius and, hence, number of nodes. Moreover,

- each cluster head rules over a small -possibly balanced-number of nodes;
- cluster heads forward collected information to a cluster head at higher abstraction level instead of sending them directly to the base station;
- the transmission range of simple nodes can be reduced w.r.t. the LLC one: the transmission requires less power and the inter-cluster interference decreases.

A larger number of clusters (or, equivalently, a smaller cluster size) is always profitable from the energetic point of view but at the expense of an increased bandwidth overhead for transmission management. Determination of the optimal number of levels for a given application depends on the characteristics of the deployment, the nature of nodes, the available bandwidth and energy as we will see in the next Section.

### C. Bandwidth limitations in a generic $k$-level hierarchy

At the end of the association phase the units of the wireless sensor network are organized as an $n$-ary balanced tree (see Figure 1), where $n$ is defined by the network designer. Traditionally, the value of $n$ is selected with a trial-and-error approach. Here we investigate the effects of a limited bandwidth on the choice of parameter $n$ hence reducing the feasible values it can assume. Let $B_M$ and $b$ be the maximum bandwidth (in kbps) for each unit (sensor nodes, cluster heads and base station) and the throughput (in kbps) derived from the sensing activity, respectively.

As shown in Figure 1, sensor nodes dispatch the acquired data to the 1-st level cluster heads. Assume that the throughput of each sensor node is $T_0 = b$. Then, each 1-st level cluster head forwards the received data to the upper level cluster head together with its acquired data: the throughput of each 1-st level cluster head is then $T_1 = (n+1)b$. This process iterates up to $k$-level cluster heads that receive data from the $(k-1)$-level cluster heads and send them to the base station (together with
their own acquired data). Hence, the throughput of the \( k \)-th
level cluster heads is \( T_k = n T_{k-1} + b \). The throughput of the
\( i \)-th level cluster heads becomes:

\[
T_i = nb \cdot \frac{1 - n^i}{1 - n} + b. \tag{2}
\]

Due to the hierarchical structure of the sensor network, \( T_i \)
measures all the traffic passing through each cluster head level
\( i \). In other words, \( T_i \) represents the sum of throughputs of all
the sensor units (sensor nodes and cluster heads of lower levels)
of a sub-tree rooted in a \( i \)-th level cluster head.

Being \( T_i \) the throughput of the sensor units at the \( i \)-th level,
the network has to guarantee a suitable bandwidth to grant cor-
correct data transmission to the \((i+1)\)-th level. In real applica-
tions, the bandwidth available to sensor units is finite and de-
deps on technological constraints (i.e., Mica[9] and Tmote-
sky[10] have a bandwidth of 250kbps, Mica2 38.4kbps).

By considering bandwidth a constraint immediately arises:
the incoming bandwidth of the sensor units of the \( i \)-th level
must be larger than the throughput of the sensor units of the
\((i-1)\)-th level (the bandwidth available to sensor units of the
\((i-1)\)-th level must satisfy throughput needs). When this con-
straint is not satisfied the network cannot properly work since
packets are lost before reaching the base station. Assume,
without loss of generality, that the network adopts a TDMA
as MAC layer (reasonable assumption as also pointed out in
the literature, e.g., see [1]) and let \( b_0 \) (kbps) be the overhead
(in terms of bandwidth usage) caused by the synchronization
message of the TDMA (whose manager is implemented in all
cluster heads and in the base station). The incoming bandwidth
of the cluster heads and base station is reduced to \( B_M - b_0 \).

As pointed in [11], \( k \)-th level cluster heads have the largest
bandwidth, while sensor nodes assume the lowest value of \( B_0 \):

\[
B_0 = \frac{B_M}{n^{k+1}} - b_0 \cdot \frac{n^{k+1} - 1}{n^{k+1}(n-1)}. \tag{3}
\]

As presented in (3) the number of levels \( k \) and the number of
nodes \( n \) for each cluster heavily affect the required bandwidth
for transmission at the \( k \)-th level cluster head. As expected,
by increasing either \( k \) or \( n \), the required bandwidth increases.
However, the node bandwidth is fixed by technological con-
straints at design time and, by fixing either \( k \) (or \( n \)), the coun-
terpart parameter can be computed according to (3).

The amount of available bandwidth w.r.t. the total number
of levels (base station, \( k \) levels of cluster heads and sen-
sor nodes) for the most important commercial sensor units
is presented in Figure 2. As expected, the available bandwidth
decreases quickly with the increase of the number of levels (in
this case \( n = 5 \) and \( b_0 = 1 \)).

Let us consider an \( i \)-th level cluster composed by \( n \) \( i \)-th level
cluster heads, the transmission radius \( R_i \) at level \( i \), and the
corresponding transmission power \( P_i \). In [11], we shown that:

\[
R_{i+1} = R_i \cdot \sqrt{n} \tag{4}
\]

under a first order radio model \( P_i \propto R_i^2 \).

We have that \( R_k = R_{\text{MAX}} \) as a boundary condition. By
using (4) it is possible to compute the \( R_i \) sequence and conse-
quently, the power needed for transmission.

III. SIMULATION RESULTS

To evaluate the different performance in terms of system
lifetime, energy consumption and alive nodes distribution on
different algorithms, we developed an ad-hoc simulator. We
considered a first order radio model and the same parameters
suggested in [8]. We fixed the packet size for data at 1024\( \text{bits} \);
while advertisement and synchronization were 40\( \text{bits} \) each.
A generic node was charged with an initial energy \( e = 1J \). We
uniformly deployed 100 nodes within a circular environment
of radius \( R_{\text{MAX}} = 30m \) centered in the base station. The
node bandwidth is 40\( \text{kbps} \), and \( b = 1\text{kbps} \). We also
assumed that each node sends a single data packet at each round.
Missing parameters (e.g., \( R_i \)) have been determined as sug-
gested in Section III by exploiting inter-relationships among
the algorithm parameters. For other algorithms such as LLC
or LEACH we used the parameters suggested in [8] and [3],
respectively. We selected \( k = 2 \) for X-LLC.

To compare performance we consider two indexes:

- the network life activity, defined as the percentage of alive
  nodes in a given round (with respect to the initial ones);
- the network residual energy, defined as:

\[
\frac{N_{T_{\text{act}}}(r)}{N_{T_{\text{act}}}^r} \cdot \frac{\sum_{i=0}^{\infty} e_i}{N_{T_{\text{act}}}^r}. \tag{5}
\]

were \( N_{T_{\text{act}}}(r) \) is the number of alive nodes at election
round \( r \), and \( e_i \) is the residual energy of the \( i \)-th node.

Results addressing the network life activity are given in Fig-
ures 3; the \( x \) axis represents the round (a round is composed
by an election procedure and one or more steady-state phases,
associated with data delivery (here, we have a single steady-state)). The y axis shows the number of alive nodes or the averaged node energy. We observe that LEACH outperforms, at the beginning, all other algorithms in terms of number of alive nodes. This can be explained by noting that in LEACH the election mechanism simply reduces to one advertisement message to be sent for each cluster head, while in other algorithms a larger number of messages is required. Nothing is free: overhead introduced is necessary to assure a good cluster quality (a uniform number of nodes in clusters).

However, this initial non-optimality is abundantly compensated both by an improved quality of service (more nodes remain active over time and are characterized by a uniform distribution over the environment) and a longer lifetime. LC, LLC and X-LLC, better exploit available energy by balancing the overhead from the very beginning of the network life; this does not happen with LEACH as can be seen in Figure 4.

X-LLC outperforms the envisaged algorithms, in particular, at round 20000, we have that it improves (Figure 3) over LLC at least for a 35% in terms of the number of alive nodes and a 12% in saved energy (Figure 4). X-LLC consumes more at the beginning, when many nodes are available but, after about 6000 rounds, outperforms the others. This phenomenon can be explained by considering that, initially, all nodes have the same energy level $e_i$ (and then, almost the same probability $p_i$ to be selected as clusterheads, according to (1)) and, as a consequence, a high number of them go under the election process. This produces an extra overhead due to the quantity of messages sent and received by nodes. In a long term perspective, the reduced transmission range assured by a X-LLC hierarchical structure, is more effective in terms of energy savings than other algorithms.

A uniform distribution of alive nodes over rounds (or, again, the absence of areas not covered by nodes) is an amenable property we would expect from an effective monitoring network. In fact, only in this way we can grant a quality of service over time, i.e., a proper monitoring of the environment.

In Figures 6(X-LLC), 7(LLC), 8(LC) and 9(LEACH) we present the distribution of alive nodes in case $r = 9000$ and in Figure 5 the initial node deployment. Algorithms experience a degradation in performance as function of $r$ (the number of alive nodes decreases). We believe that the non uniform distribution in the long run is associated with the excessive usage of nodes near to the base station which become cluster heads with high probability (due to (1)). Hence, these nodes will consume the available energy before the others causing a non-uniform distribution of the alive nodes.

Differently, X-LLC grants a uniform distribution of alive nodes over rounds (see Figure 6). This amenable behavior guarantees the proper functioning of the WSN; alive nodes acquire data from the area even if with a coarser spatial resolution. The reason of this behavior resides in the ability to reduce side effects caused by the election mechanism (the possibility that a node is elected depends on the amount of nodes in its neighborhood): X-LLC, by creating smaller clusters, supports the election of cluster heads everywhere in the environment, leaving to a uniform distribution of alive nodes. Differently, in LLC, LC, LEACH the averaged cluster size is larger than the X-LLC one and eligible nodes close to the boundary heavily suffer from bound effects (i.e., are elected as cluster nodes with lower probability).

IV. CONCLUSIONS

The paper presents an extension of the LLC routing algorithm by introducing a hierarchical structure in the network management (network nodes are clustered with a hierarchical approach) which, by exploiting the nature of the topology, allows us for improving adaptability, network lifetime and overhead load balance among clusters. The novelty of the proposed approach resides in a $k$ level hierarchical structure of cluster heads (the literature only considers one level of cluster heads) and considering the realistic case of finite bandwidth for nodes. This latter point is crucial in WSN nodes since available routing algorithms may become unfeasible in a large class of applications for technological constraints associated with a reduced bandwidth. Moreover, the proposed algorithm is particularly appealing for its ability to maintain a uniform distribution of alive nodes in the deployment area, feature associated with the monitoring QoS, where other routing algorithms introduce vast areas not covered by sensor nodes.

Since tiny cooperating and swarm robots share the same behavioral model, in terms of communications, with WSNs, what proposed has a larger validity; with minor modifications it can be extended also to such world.

Fig. 3. Network life activity, with $N_0 = 100$

Fig. 4. Network residual energy, with $N_0 = 100$

Fig. 5. Initial Deployment

Fig. 6. X-LLC

Fig. 7. LLC

Fig. 8. LC

Fig. 9. LEACH