Smaller Connected Dominating Sets in Ad Hoc and Sensor Networks based on Coverage by Two-Hop Neighbors

François Ingelrest, David Simplot-Ryl, Ivan Stojmenovic

Smaller Connected Dominating Sets in Ad Hoc and Sensor Networks based on Coverage by Two-Hop Neighbors

François Ingelrest and David Simplot-Ryl
IRCICA/LIFL, Univ. Lille 1
CNRS UMR 8022, INRIA Futurs, France
{Francois.Ingelrest,David.Simplot}@lifl.fr

Ivan Stojmenović
Computer Science, SITE
University of Ottawa, Canada.
ivan@site.uottawa.ca

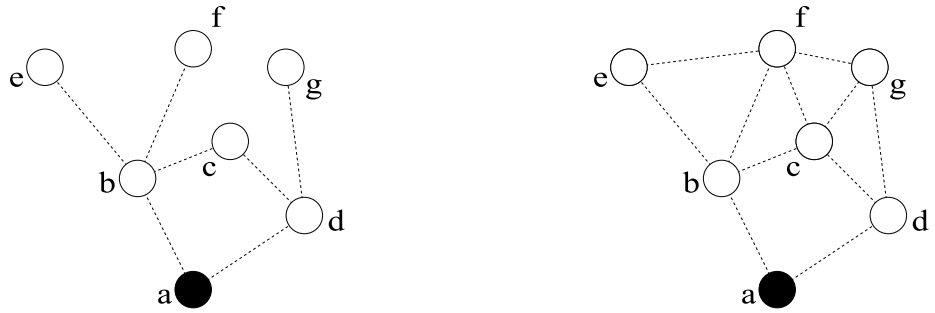
***Abstract*— In this paper, we focus on the construction of an efficient dominating set in ad hoc and sensor networks. A set of nodes is said to be dominating if each node is either itself dominant or neighbor of a dominant node. Application of such a set may for example be broadcasting, where the size of the set greatly impacts on energy consumption. Obtaining small sets is thus of prime importance. As a basis for our work, we use a heuristic given by Dai and Wu for constructing such a set. Their approach, in conjunction with the elimination of message overhead by Stojmenović, has been recently shown to be an excellent compromise with respect to a wide range of metrics. In this paper, we present an enhanced definition to obtain smaller sets in the specific case where 2-hop information is considered. In our new definition, a node u is not dominant if there exists in its 2-hop neighborhood a connected set of nodes with higher priorities that covers u and its 1-hop neighbors. This new rule requires the same level of knowledge used by the original heuristic: only neighbors of nodes and neighbors of neighbors must be known to apply it. However, it takes advantage of some topological knowledge originally not taken into account, that may be used to deduce communication links between 1-hop and 2-hop neighbors. We provide the proof that the new set is a subset of the one obtained with the original heuristic. We also give the proof that our set is always dominating for any graph, and connected for any connected graph. Two versions are considered: with topological and positional information, which differ in whether or not nodes are aware of links between their 2-hop neighbors that are not 1-hop neighbors. An algorithm for locally applying the concept at each node is described. We finally provide experimental data that demonstrates the superiority of our rule in obtaining smaller dominating sets. A centralized algorithm is used as a benchmark in the comparisons. The overhead of the size of connected dominating set is reduced by about 15% with the topological variant and by about 30% with the positional variant of our new definition.**

I. Introduction

Wireless networking has become an essential part of new technologies, allowing nomadic users to keep in touch with their family or their office using miscellaneous devices such as laptops, PDA's or smartphones. The most deployed technology, known as WiFi, is still very restrictive as users must be within the range of a correctly configured access point. Densely populated area like airports or train stations may easily be equipped with needed infrastructure, but it is not the case for other areas, where multi-hop wireless links may be desirable. Multi-hop wireless ad hoc and sensor networks have been widely studied recently. They are composed of a set of hosts operating in a self-organized and decentralized manner, which can communicate together using a radio interface. As in any wireless network, transmission ranges are limited due to propagation path loss, health and energy considerations. Thus, each node must act alternately as a terminal and a router, depending on the needs of the system, leading to a cooperative multi-hop routing.

Energy conservation is one of the most challenging problems in ad hoc and sensor networks because batteries have very limited capacities. Two particular important problems are activity scheduling and broadcasting. In activity scheduling problem, some nodes decide to turn off their radio equipment to preserve energy, but should have at least one active neighbor to collect messages for them or take over some sensing tasks. In broadcasting problem, one host needs to send a particular message to all the other ones in the network. Broadcasting is applied for route discovery [1], synchronization, alarming and other operations. In a straightforward solution to this problem, hosts only need to blindly relay packets once to their neighborhood. However, this leads to the well-known broadcast storm problem [2]: while consuming a lot of energy, this method does not even ensure a complete coverage of the network due to multiple collisions. Connected dominating sets may be used to solve these two problems.

In a connected dominating set (CDS), each node either belongs to CDS or has a (1-hop) neighboring node in CDS. In an activity scheduling solution, only nodes from CDS may remain active. To reduce the set of relaying nodes in a broadcasting task, only nodes marked as dominant have to act as routers to relay the broadcasting packet, so that the broadcasting is performed by retransmitting by as few nodes as possible. An efficient distributed algorithm, known as the generalized self-pruning rule [3], has been proposed to compute CDS by using only local 1-hop information. In this paper, we propose an improvement to this rule: while requiring the same level of knowledge at each node, our enhanced rule elects fewer nodes as dominant. No additional message



(a) Topological information.

(b) Positional information.

Fig. 1. Positional information may be used at node a to determine links between its 2-hop neighbors.

exchanges are required to apply it, and needed information may be obtained by using simple beacon messages. We also give theoretical proofs that the set of nodes elected by our algorithm is always dominant and connected, and provide experimental data demonstrating the superiority of our method.

The remainder of this paper is organized as follows: in the next section, we provide the definitions needed by our network model, while in Sec. III a review of existing work is proposed. In Sec. IV we describe our algorithm and give the proof that the designated set is indeed always dominant and connected. We then provide in Sec. V experimental data for our new method and comparisons with existing work. We finally conclude in Sec. VI and give some directions for future work.

II. Preliminaries

The common representation of a wireless network is a graph $G = (V, E)$, where V is the set of vertices (the hosts, or nodes) and $E \subseteq V^2$ the set of edges giving the available communications: if a node v is a physical neighbor of a node u (v lies within the communication range of u and thus receives its messages), then there exists $(u, v) \in E$. If we assume that all nodes have the same communication range, denoted by R, then the set E is defined by:

$$E = \{(u, v) \in V^2 \mid \text{dist}(u, v) \leq R\},$$

$\text{dist}(u, v)$ being the Euclidean distance between nodes u and v. Each node u must be assigned a unique identifier $\text{id}(u)$ (this may be, for instance, IP or MAC address). We define the neighborhood set $\text{N}(u)$ of a node u as:

$$\text{N}(u) = \{v \in V \mid v \neq u \wedge (u, v) \in E\},$$

and the extended neighborhood set $\dot{\text{N}}(u)$ as:

$$\dot{\text{N}}(u) = \text{N}(u) \cup \{u\}.$$

The neighborhood function is naturally extended to sets of nodes: for a given subset $V' \subseteq V$, we have $\text{N}(V') = \bigcup_{u \in V'} \text{N}(u)$. The degree of a node u is simply its number of neighbors $|\text{N}(u)|$, while the density of the network is the average size of the extended neighborhood sets. We measure the distance between two nodes u and v in terms of number of hops, which is simply the minimum number of edges a message has to cross to travel from u to v.

A graph $G_D = (V_D, E_D)$, where $V_D \subseteq V$ and $E_D \subseteq E$, is dominant iff:

$$\forall u \in V \qquad \exists v \in V_D \mid v \in \dot{\text{N}}(u).$$

In simple terms, each vertex is either dominant or 1-hop neighbor of a dominant node. The set E_D is the subset of E that contains only edges between two dominant neighbors:

$$E_D = \{(u, v) \in V_D^2 \mid \text{dist}(u, v) \leq R\}.$$

We assume that each node is aware of its 2-hop neighbors. This is achieved in two rounds of HELLO messages. First, each node informs its neighbors about its existence (and position, if this information is available). Next, each node sends message to all its neighbors informing about its 1-hop neighbors (nodes from which HELLO message in the first round was received). In a mobile ad hoc network, each node (regularly or based on its mobility) emits additional HELLO messages, to maintain 2-hop information. When a node u receives from a node v such a message, u adds v to its neighborhood table, or updates the entry if it was already there. Too old entries are regularly removed from the table, as corresponding nodes have not signaled themselves recently.

If Euclidean distances between neighbors are needed, a straightforward method to obtain them is to let nodes add their position in their beacon messages. Positions may simply be acquired by using a location system such as the GPS (*Global Positioning System*). Other methods may be used, like deducing distances to neighbors by measuring the reception power of messages. In the graph representing the 2-hop knowledge at each node, there exists a difference between topological and positional information that may be used. If positional information is available, each node may conclude, based on their locations, whether two of its 2-hop neighbors (which

are not 1-hop neighbors) are neighbors themselves. Such a conclusion cannot be made without position information (that is, based solely on topological information), and therefore no edge between such neighbors is assumed. Fig. 1 illustrates the difference between these two assumptions, considering node a. With topological information, in 1(a), the 2-hop neighbors $\{c, e, f, g\}$ are assumed to not be directly connected, while it is not the case with positional information in 1(b). However, node c is always known to be a common neighbor of nodes b and d because it appears in both neighborhood lists.

III. Related Work

As stated in Sec. I, the easiest method for broadcasting a packet is to have all nodes act as routers and relay it at least once to their neighborhood: this method is known as *blind flooding*. However, such a simple behavior has huge drawbacks: too many packets are lost due to collisions between neighboring nodes (this can lead to only a partial coverage of the network) and far too much energy is consumed.

A possible method to reduce the energy consumption is to determine a set of nodes, such that if only those nodes act as routers, the broadcasting is still achieved. A dominating set of vertices V_D is suitable for this task, as long as it is connected (there exists a path in G_D between any two vertices). Once such a set has been obtained, the broadcasting process becomes obvious:

- The source node sends the packet to its neighborhood.
- Each dominant node that receives it acts as a router and forwards it to its neighbors.
- Each non-dominant node simply drops it.

Besides this simple method, dominating sets can also be used as part of a more sophisticated broadcasting mechanisms [4].

Another application of connected dominating sets introduced in Sec. I is activity scheduling, especially in sensor networks. In this problem, only a subset of nodes needs to keep active its radio equipment to form a backbone. The latter is used to forward messages from some sensors to the sink. Nodes that do not belong to the backbone may turn off their radio equipment, thus saving a lot of energy. When an event occurs in their neighborhood, they just have to turn on their radio module to transmit an alert to their dominant neighbors.

The simplest connected dominating set is the unit graph itself, and using such a set for broadcasting is the same as performing a blind flooding. The problem of computing the smallest possible connected dominating set is known to be a NP-complete problem [5], [6] and requires a global knowledge of the network topology, thus many centralized and distributed approximated algorithms for constructing efficient sets have been proposed. We will describe only one here, which is quite simple, efficient compared to others, and easy to describe. It is a centralized algorithm proposed by Guha and Khuller in [7] as follows. Each node is initially colored as white. The densest node in the graph is then colored as black, and all its neighbors as grey nodes. Then, iteratively, while there exists some white nodes, the grey node with the largest number of white neighbors is selected, colored as black node, and all its white neighbors as grey nodes. Ties can be resolved by using some keys (identifiers). At the end, the set of black nodes is a connected dominating set, and its size is a good estimate of the limits one can reach with a localized heuristic. Some localized efficient heuristics have also been proposed and may thus be applied in decentralized networks.

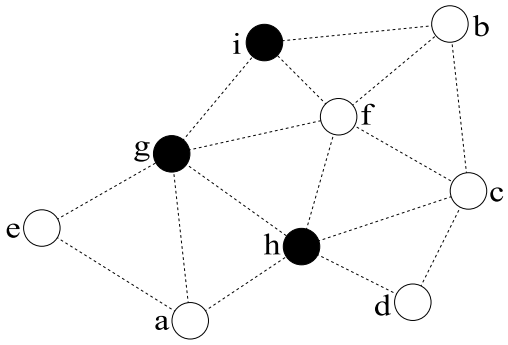
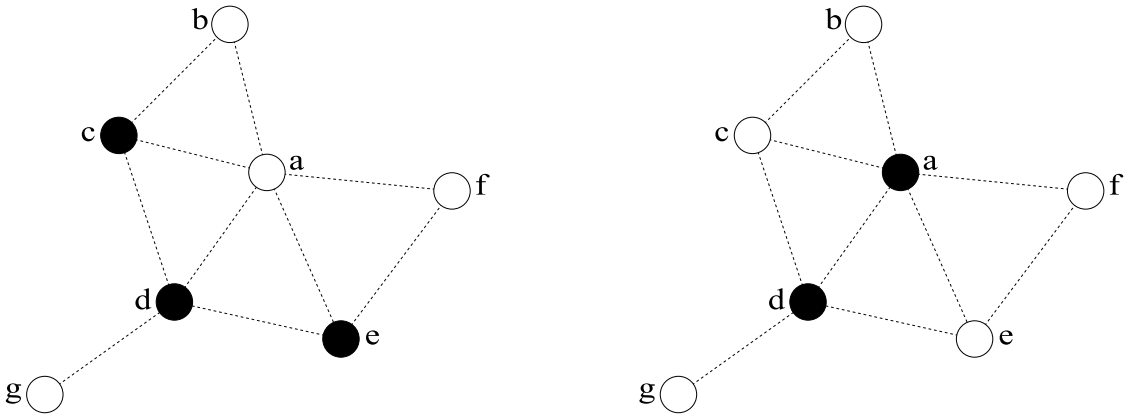


Fig. 2. Dai and Wu's heuristic: $V_D = \{g, h, i\}$.

Wu and Li proposed in [8] an algorithm that has been later improved in term of message overhead in [9], [10]. We describe here the latter because it requires no messages once 2-hop neighborhood information is available. A node is referred to as *intermediate* if it has at least two neighbors not directly connected. A node u is covered by a node $v \in \mathrm{N}(u)$ if $\mathrm{N}(u) \subseteq \mathrm{N}(v)$ and $\mathrm{key}(v) > \mathrm{key}(u)$. Nodes that are not covered by any neighbor are called *inter-gateway* nodes. A node u is covered by two connected nodes $v \in \mathrm{N}(u)$ and $w \in \mathrm{N}(u)$ if $\mathrm{N}(u) \subseteq (\mathrm{N}(v) \cup \mathrm{N}(w))$, $\mathrm{key}(v) > \mathrm{key}(u)$ and $\mathrm{key}(w) > \mathrm{key}(u)$. Inter-gateway nodes that are not covered by any pair of connected neighboring nodes become *gateway* nodes.

This rule has been further improved in term of number of dominating nodes by Dai and Wu [3]: they proposed a more general rule where coverage can be provided by an arbitrary number of connected 1-hop neighbors. A modification of this *generalized self-pruning rule* has been proposed by Stojmenović in [11] in order to avoid similar message exchanges between neighbors. A node u is covered by a set of 1-hop neighbors A_u if A_u is connected, $\mathrm{N}(u) \subseteq \mathrm{N}(A_u)$ and if each node in A_u has a higher key than u. It has been further computationally simplified by Carle and Simplot-Ryl [12] as follows. First, each node checks if it is intermediate, that is, whether it has at least two neighbors not directly connected. Then each intermediate node u constructs a subgraph G_{h} of its 1-hop neighbors with higher keys. In the graph composed by $\mathrm{N}(u)$, each node which has a lower key than u is removed, as well as the corresponding edges. The resulting subgraph is denoted by G_{h}. If the latter is empty or disconnected then u is in the dominating set. If G_{h} is connected but there exists a neighbor of u which is not neighbor of any node from G_{h}



(a) key$(u) = u$.

(b) key$(u) = \{|\mathrm{N}(u)|, u\}$.

Fig. 3. Applying Dai and Wu's heuristic with different kinds of priorities.

then u is in the dominating set. Otherwise u is covered and is not in the dominating set. Dijkstra's shortest path algorithm can be used to test the connectivity (it is performed locally at each node). Non-intermediate nodes are never dominant. This rule is illustrated by Fig. 2, where black nodes are dominant, and identifiers of nodes are used as keys using the lexicographical order for comparisons. Nodes $\{d, e\}$ are not intermediate because they do not have unconnected neighbors, they are thus not dominant. Graphs G_{h} of nodes $a(\{e, g, h\})$, $b(\{c, f, i\})$, $c(\{d, f, h\})$ and $f(\{g, h, i\})$ are all connected, and cover neighbors with lower priority. These nodes a, b, c and f are thus not dominant. Only nodes g, h and i change their status to dominant. Finally, $V_D = \{g, h, i\}$.

The key of a node represents its priority, and it is assumed to be unique for each node. A simple priority can be the identifier of the node, but it may also be any collection of values with the aim of increasing the efficiency of the dominating set. For example, in [13] the proposed key for a node u is:

$$\mathrm{key}(u) = \{\mathrm{energy}(u), \mathrm{degree}(u), \mathrm{id}(u)\}.$$

This means that nodes with higher energy level have a larger probability to be elected as dominant. If the energy levels are equal for two nodes, then the second key, degree, is used for comparison. Finally, if there is a tie with the degree as well, the identifier is used. Fig. 3 illustrates the differences obtained when using different kinds of keys: in 3(a), the keys are the identifiers of nodes, while in 3(b) the degree is used a the primary key and the identifier as the secondary one. Some other keys were later proposed and studied in [14].

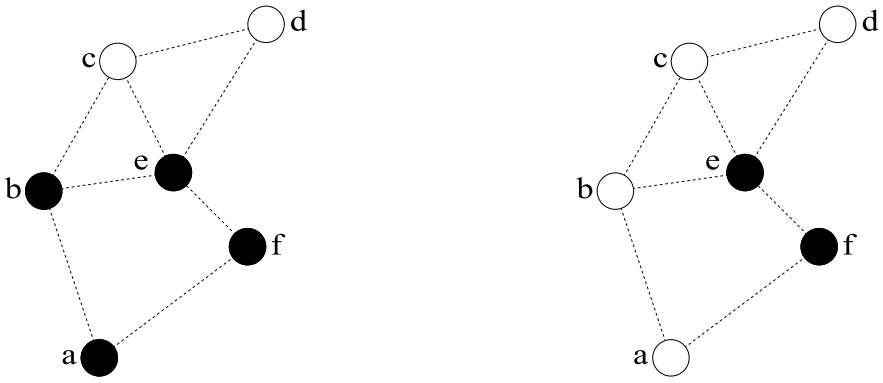
Liu, Pan and Cao recently proposed in [15] an iterative localized algorithm for connected dominating sets, improving the concept of [8] in terms of size of connected dominating sets, but at the expense of additional messages between neighboring nodes. The principle is to have nodes exchange messages with their neighbors (there are exactly 5 messages exchanged) in order to decide whether they should be dominant, using information received from their neighbors. At each step, each node that decides not to be dominant becomes passive; otherwise it is active and reevaluates this decision in the next round. There can be 6 messages exchanged if each node wants to know which of its neighbors are dominant. The authors claimed that this process prevents 'illegal simultaneous' removals from the dominating set, which may disconnect it. The experimental performances show that the computed set is efficient, but the communication overhead and the synchronization needed make it more difficult to apply in a distributed environment. Furthermore, beacon messages are also needed for the first step to take place.

In [16], authors proposed a performance comparison of various protocols for computing backbones in ad hoc networks, including the previously cited protocols. They measured miscellaneous parameters, like the computation complexity (needed time to create the backbone), the backbone size or even the energy consumption per node in order to determine which protocol suits the best to ad hoc networks. They concluded that Wu and Li's algorithm used in conjunction with the variant by Stojmenović, Seddigh and Zunic [9] is an excellent compromise with respect to all the considered metrics, and overall far superior than any other approach that exists in literature. This study completely justifies our approach on working on this concept to propose a more efficient one.

IV. Dominating sets based on coverage by two-hop neighbors

In this section, we give an enhanced definition for computing a connected dominating set over a connected graph by using only 2-hop topological or positional information. Note that a very general definition for computing such sets was given by Wu and Dai in [17].

We also provide proofs that the set obtained with this new definition is:



(a) Original rule. (b) Enhanced rule.

Fig. 4. Dai and Wu's heuristic and its enhanced variant. Identifier of nodes gives their priority, using the lexicographical order.

- a subset of the one obtained with Dai and Wu's heuristic,
- always dominating for any graph,
- always connected for any connected graph.

We finally provide an efficient algorithm to apply this rule in a practical context.

A. Description

Our new definition of dominating sets is based on the observation that the method described by Dai and Wu [3] requires 2-hop topological knowledge, because nodes need to know their neighbors and the neighbors of their neighbors, and that this knowledge could be better used by applying some enhanced concepts. To illustrate this, let us consider Fig. 4(a) where the generalized self pruning rule has been applied using the lexicographical order to determine the priority of nodes. The node a has been marked as dominant because it has two neighbors $\{b, f\}$ not covered by any set of neighbors with higher priority. In fact, a is itself covered by $\{b, e, f\}$ although e is not a 1-hop neighbor, and a could be marked as passive as illustrated in 4(b): the set of black nodes would remain connected and dominant. While e is not a direct neighbor of a, this does not prevent the latter from verifying whether any of its neighbors are neighbors of e, or whether $\{b, e, f\}$ are connected, since e appears in the list of neighbors sent to a by its 1-hop neighbors, therefore such conclusion can be made. Similarly in 4(b), node b concludes that it is not dominant since all its neighbors $\{a, c, e\}$ and itself are covered by its connected higher key 2-hop neighbors $\{e, f\}$.

Therefore, our new definition of the dominating set V_D may be described as follows:

$$\forall u \in V \qquad u \notin V_D \Leftrightarrow \tag{1}$$
$$\exists A_u \subseteq \dot{\mathrm{N}}(u)^2 \setminus \{u\} \begin{cases} \forall v \in A_u,\ \mathrm{key}(v) > \mathrm{key}(u) \\ A_u \text{ connected} \\ \dot{\mathrm{N}}(u) \subseteq \dot{\mathrm{N}}(A_u) \end{cases}$$

In other words, an intermediate node u is not dominant if there exists in its 2-hop neighborhood a connected set A_u of nodes with higher priorities, such that each neighbor of u either belongs to A_u or is a neighbor of a node in A_u (i.e., A_u covers $\dot{\mathrm{N}}(u)$).

Note that, when topological information is used, u is not aware of possible links between its 2-hop neighbors, and therefore may declare the set disconnected although in reality it may be connected (refer to Fig. 1). This can be avoided if nodes are able to determine their location: they can add it to their beacon messages, and thus links between 2-hop neighbors will be part of the knowledge of nodes. This variant is considered from an experimental point of view in Sec. V.

B. Proof of inclusion

Theorem 1: The dominating set V_D computed with our new definition is a subset of the one obtained with the generalized self-pruning rule.

Proof: In the generalized self-pruning rule, a node u is marked as not dominant if there exists a connected subset of $\mathrm{N}(u)$ composed by higher priority neighbors such that $\dot{\mathrm{N}}(u)$ is covered by this subset. As $\mathrm{N}(u) \subseteq \dot{\mathrm{N}}(u)^2 \setminus \{u\}$, if there exists such a set in $\mathrm{N}(u)$, then it also exists in $\dot{\mathrm{N}}(u)^2 \setminus \{u\}$. We can thus deduce that nodes marked as not dominant by the generalized self-pruning rule are also marked as not dominant by our new definition, which can only remove more nodes from the dominating set. ■

This proof demonstrates that our new definition cannot generate a larger set V_D than the one obtained with the generalized self-pruning rule.

C. Proof of dominance

Theorem 2: For any given graph $G = (V, E)$, each node $u \in V$ either belongs to V_D or is a neighbor of a node $v \in V_D$.

Proof: Assume that the set V_D is not dominating. Let u be a node which is not in the dominating set and has no dominant neighbor. Node u is covered by A_u, set of 2-hop neighbors with higher key values than u. Let v be the node with the highest key value in $\mathrm{N}(u)$. Node v has higher priority

than u because of the existence of A_u and the need for at least one node from A_u to be 1-hop neighbor of u. Node v is not dominant because u is not covered by V_D. It is thus covered by a set A_v. Therefore u is neighbor of a node w from A_v. Node w is therefore neighbor of u and has higher key value than v, which is a contradiction with respect to the choice of v. ■

D. Proof of connectivity

Theorem 3: For any given connected graph $G = (V, E)$, there exists a path between any two vertices in the graph $G_D = (V_D, E_D)$ produced by our algorithm.

Proof: We may assume that nodes are removed one by one in ascending order of priority instead of simultaneous removal and will show that the removal of any node u preserves the connectivity. Let vuw be a path via node u. Node u is removed because of the set A_u, which is connected and covers its neighbors, as illustrated by Fig. 5. Moreover, no nodes of A_u has already been removed since A_u contains nodes with higher priority than u. This means that there are two nodes v' and w' from A_u so that v is neighbor of v', w is neighbor of w', and v' and w' are connected in A_u. This means that nodes v and w remain connected after removal of node u. We can thus deduce that a node u will never 'remove' itself from the dominating graph if there does not exist another path between any two components that are 'glued' together thanks to u. ■

All three proofs do not depend on the possible links between 2-hop neighbors. They are therefore valid for both topological and positional information. This does not mean that they will result in the same dominating set. On the contrary, they can differ, since these 'special' links may be used to make a set of neighbors with higher priorities connected.

E. Algorithm

It could seem at first that the algorithm used for Dai and Wu's heuristic described in Sec. III could also be used for our new definition: instead of computing the graph G_h using the 1-hop neighborhood, it could be computed using the 2-hop neighborhood (minus the links between the 2-hop neighbors in

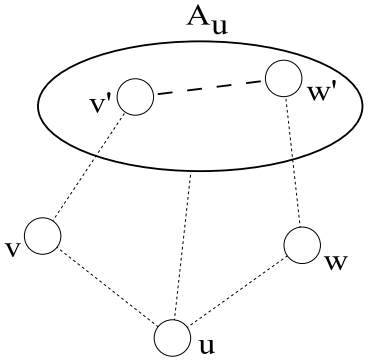


Fig. 5. Removal of vertex u does not lead to a loss of connectivity between vertices v and w.

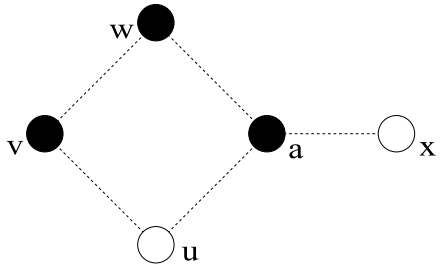


Fig. 6. A new algorithm must be used for the enhanced definition.

case of topological information). However this may not work correctly, as illustrated by Fig. 6: node u is not dominant because there exists a connected set $\{v, w\}$ which covers $\{a, u\}$. However, the set of nodes of higher priority within the 2-hop neighborhood $\{v, w, x\}$ is not connected. Using the original algorithm, u would have been marked as dominant.

We therefore describe here a new algorithm for testing whether or not node u will declare itself as being in dominating set as follows. This algorithm takes $O(k^2d^2)$ time, where d is the average density.

1) Create a graph G_{h} composed from the nodes of higher priority than u within the 2-hop neighborhood of u.
2) Find the connected components in G_{h}; this can be done by repeated application of Dijkstra's shortest path algorithm starting each time from an unseen node. Alternately, depth first search or breath first search protocols can be repeatedly applied to find all components.
3) If u and its 1-hop neighbors are covered by at least one component, then mark u as not dominant.

V. Performances Evaluation

In our simulation, we compared our enhanced definition with the original heuristic by Dai and Wu. We did not consider other methods because of their communication overhead for construction and maintenance and other significant drawbacks as verified in [16]. On the other hand, only beacon ('HELLO') messages are required for the two compared rules. We used a very efficient centralized algorithm by Guha and Khuller [7] as the benchmark in our comparisons. That algorithm is a good measure of the ability of localized algorithms to produce small connected dominating sets, and a good indicator of progress made among localized protocols.

We use the following abbreviations:

- CH: Centralized heuristic.
- ED: Enhanced definition.
- EDPOS: Enhanced definition with positioning information, this means that nodes are aware of links between their 2-hop neighbors.
- GSPR: Generalized self-pruning rule.

The parameters of the simulations are the following. The network is static and always composed of 500 nodes randomly distributed in a uniform manner over a square area whose size is computed in order to obtain a given degree. We define the latter as the number of nodes in a communication area. For

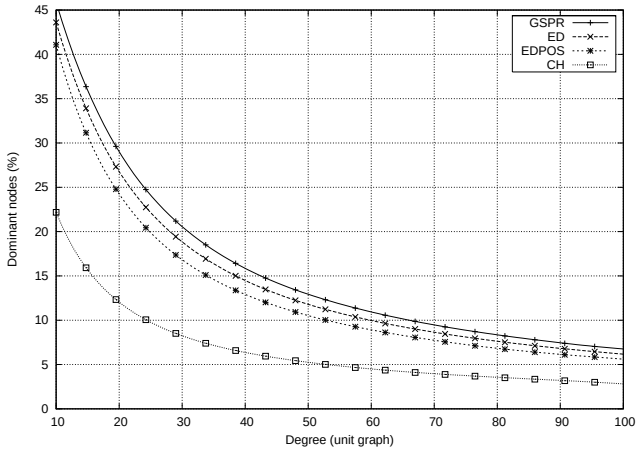


Fig. 7. Percentage of dominant nodes for varying degree.

each measure, we took the average value obtained after 500 iterations.

We give in Fig. 7 the average percentage of dominant nodes obtained by applying the different schemes for varying degree between 10 and 100. Not surprisingly, the centralized greedy heuristic performs the best in obtaining small dominating sets for all ranges, and for high degree like 50, only 5.15% of nodes are marked as dominant. As theoretically proven in previous section, our enhanced definition always performs better in giving smaller dominating sets in the average case. For the value of 20 nodes per communication area, only 26% of nodes are elected as dominant, against 28.19% for the generalized self-pruning rule. As expected, using positioning information brings even better results, and only 23% of nodes are then marked as dominant. For higher density (50) the percentage of dominant nodes even decreases down to respectively 11.56% and 10.32%.

We provide in Fig. 8 the overhead of the localized heuristics over the centralized one. Considering degree 30 in 8(a), GSPR has an overhead equal to 151%: this means that this heuristic elects 151% more nodes as dominant than CH. For the same degree, ED has an overhead of 129%, this is a difference of 22%. With positioning information, EDPOS scores only an overhead of 105%, the difference being equal this time to 46%. In summary, as illustrated in 8(b), the overhead of the size of connected dominating set was reduced by about 15% with the topological variant and by about 30% with the positional variant of our new definition, with respect to the generalized self prunning rule. These percentages appears rather stable with respect to the network densities, for dense networks.

We finally give in Fig. 9 a comparison of the protocols in terms of the dominant graphs they produce. In 9(a) is given the average degree of the CDS's (this is the average number of dominant neighbors per dominant node). Not surprisingly, this value is dependent on the average size of the CDS's: the smallest is the set, and the smallest is the average degree, so CH obtains the best results and GSPR the highest ones. An interesting remark is that the degree of the produced CDS's is relatively constant for varying degree of the unit graph. The degree of the localized algorithms varies between 3 and 4, while the average degree of the centralized heuristic is around 2. We also consider in 9(b) the average length of the edges between two dominant neighbors divided by R. Once again, this value is relatively stable and does not really depend on the degree of the unit graph. The three localized schemes obtain nearly the same results, while CH has higher values. This can be easily explained: in this heuristic, at each step, the node with the highest number of 'non-covered' neighbors is chosen, and we can expect this value to increase with the distance between the nodes.

VI. Conclusion

In this paper, we have presented an enhanced definition for computing a dominating set in ad hoc and sensor networks, using as a basis a work from Dai and Wu when 2-hop information is considered. We have proved that our rule gives a subset of the one obtained thanks to the original heuristic. We have also prover that this subset is always dominating, and connected for any connected graph. We finally provided experimental results which demonstrate the superiority of our rule over the original one in electing fewer nodes as dominant. This is especially interesting in the dynamic networks we considered, where fewer dominant nodes induces greater energy savings.

As future research associated to this paper, we would like to consider specific usage of our heuristic (e.g., broadcasting) in a realistic environment, where mobility might be involved. We believe that the gain obtained thanks to our enhanced heuristic may be emphasized in such environment, compared to an algorithm that requires additional messages exchange that could get lost due to mobility. Maintenance of connected dominating structure in the presence of moving nodes is also a nontrivial operation that should be specifically studied.

Acknowledgments

This work was partially supported by a grant from CPER Nord-Pas-de-Calais/FEDER TAC COMDOM, INRIA research action IRAMUS, CNRS National platform RECAP and NSERC.

References

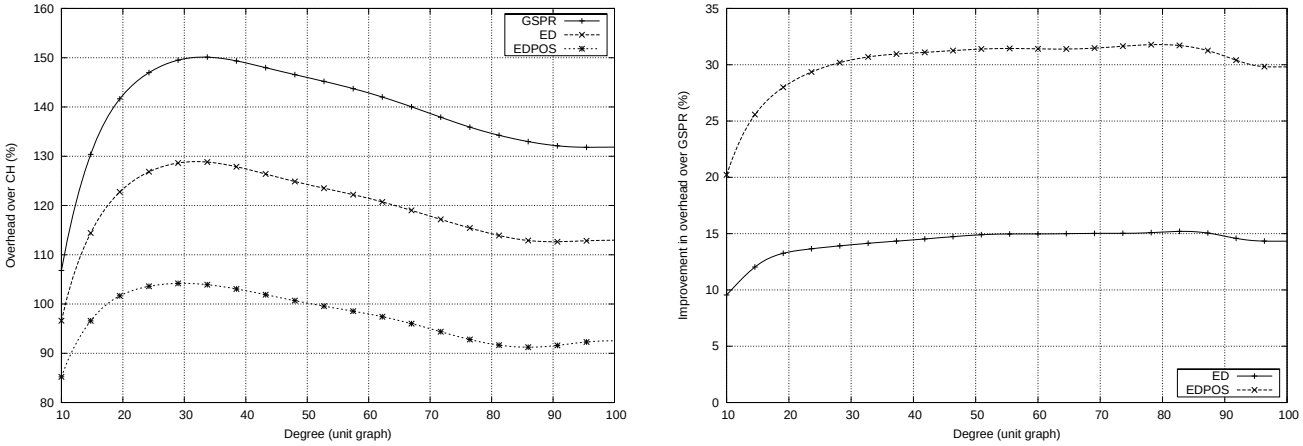
[1] D. Johnson, D. Maltz, and Y. Hu, "The dynamic source routing protocol for mobile ad hoc networks (DSR)," Internet Draft, draft-ietf-manet-dsr-10.txt, July 2004.

[2] S. Ni, Y. Tseng, Y. Chen, and J. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom)*, Seattle, USA, August 1999.

[3] F. Dai and J. Wu, "Distributed dominant pruning in ad hoc networks," in *Proceedings of the IEEE International Conference on Communications (ICC)*, Anchorage, Alaska, May 2003.

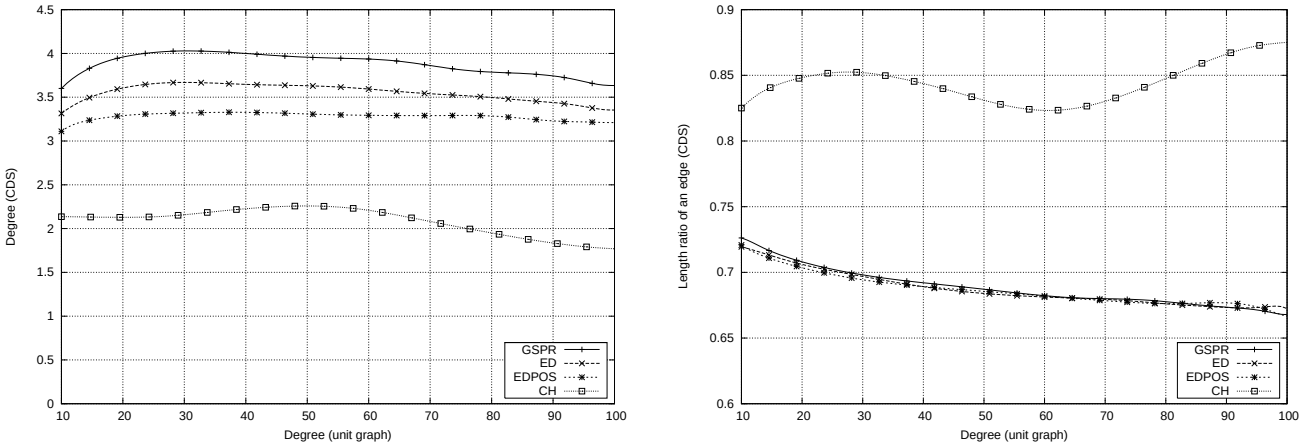
[4] F. Ingelrest, D. Simplot-Ryl, and I. Stojmenović, "Optimal transmission radius for energy efficient broadcasting protocols in ad hoc and sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 6, pp. 536 – 547, June 2006.

[5] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman, 1979.



(a) (b)

Fig. 8. Overhead compared to the centralized heuristic.



(a) (b)

Fig. 9. Comparison of the produced connected dominating sets.

[6] A. Amis, R. Prakash, T. Vuong, and D. Huynh, "Max-min D-cluster formation in wireless ad hoc networks," in *Proceedings of the IEEE Infocom*, Anchorage, Alaska, April 2001.

[7] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374 – 387, April 1998.

[8] J. Wu and H. Li, "On calculating connected dominating sets for efficient routing in ad hoc wireless networks," in *Proceedings of the ACM International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM)*, Seattle, USA, August 1999.

[9] I. Stojmenović, M. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 12, no. 1, pp. 14 – 25, January 2001.

[10] I. Stojmenović, "Comments and corrections to 'dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks'," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 11, pp. 1054 – 1055, November 2004.

[11] I. Stojmenović and J. Wu, *Mobile Ad Hoc Networking*. IEEE Press, August 2004, ch. Broadcasting and Activity Scheduling in Ad Hoc Networks, pp. 205 – 229, (S. Basagni and M. Conti and S. Giordano and I. Stojmenović, eds.).

[12] J. Carle and D. Simplot-Ryl, "Energy efficient area monitoring by sensor networks," *IEEE Computer Magazine*, vol. 37, no. 2, pp. 40 – 46, 2004.

[13] J. Wu, B. Wu, and I. Stojmenović, "Power-aware broadcasting and activity scheduling in ad hoc wireless networks using connected dominating sets," *Wireless Communications and Mobile Computing*, vol. 4, no. 1, pp. 425 – 438, June 2003.

[14] J. Shaikh, I. Stojmenović, and J. Wu, "New metrics for dominating set based energy efficient activity scheduling in ad hoc networks," in *Proceedings of the International Workshop on Wireless Local Networks (WLN)*, Bonn, Germany, October 2003.

[15] H. Liu, Y. Pan, and J. Cao, "An improved distributed algorithm for connected dominating sets in wireless ad hoc networks," in *Proceedings of the International Symposium on Parallel and Distributed Processing and Applications (ISPA)*, Hong Kong, China, December 2004.

[16] S. Basagni, M. Mastrogiovanni, A. Panconesi, and C. Petrioli, "Localized protocols for ad hoc clustering and backbone formation: A performance comparison," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 4, pp. 292 – 306, June 2006.

[17] J. Wu and F. Dai, "Broadcasting in ad hoc networks based on self-pruning," *International Journal of Foundations of Computer Science*, vol. 14, no. 2, pp. 201 – 221, April 2003.