Poster Abstract: Cost Efficient On–line Hop Count Routing Strategies for Wireless Sensor Networks

Michele Rossi†, Ramesh R. Rao‡, Michele Zorzi⋆

***Abstract*— In this abstract we present novel on–line routing strategies to achieve cost/energy efficient data forwarding in wireless sensor networks. Our algorithms are suitable for the cases where data packets have to be transmitted through multi-hop forwarding techniques to a central unit (the *sink*) and the aim is to realize the data delivery in a cost efficient manner. In our framework, each sensor node is characterized by a cost which is used to represent the status of the sensor (energy, queue, etc.) as well as its suitability to be selected as a relay node for data forwarding. In addition to node costs, node hop counts are accounted for as a rough representation of the underlying topology and are used to drive the forwarding process toward efficient solutions. We then compare the proposed on–line HC routing algorithms against globally optimal solutions.**

I. Introduction

One of the most challenging problems in wireless sensor networks (WSNs) is to provide energy efficient solutions for data forwarding so as to prolong the network lifetime. In particular, we seek to achieve robust and long-lived WSNs. Due to the energy/computation constraints characterizing micro-sensors, it is desirable to obtain such a goal by means of very simple algorithms.

The goal of our work is to discuss some design methodologies for data forwarding in WSNs. These algorithms, in order to be used in sensor networks should involve a small number of operations. For example, classic pro-active or reactive routing algorithms, proposed for Ad Hoc networks, are not a good choice in such a scenario as they need a substantial exchange of information among nodes to update routing tables. In sensor networks, due to energy constraints, this approach is not usable; instead, it is better to route packets based on *localized* information [1], where the next hop is decided, at every node, based on a *local* view of the network status. A typical approach in that sense is given by geographic routing schemes [2], where the next hop is decided based on the node position, on the position of the sink and on the coordinates of the nodes in the first–order (within range) neighborhood. In this work, we investigate on-line local forwarding techniques where hop counts are used instead of geographical coordinates in order to assess the direction to be followed to forward data. In addition, properly defined costs are used to represent the internal state of every node (possibly including link qualities). Costs and hop counts are therefore exploited as the local information to be considered to make routing decisions.

†Corresponding author, e-mail: mirossi@gmail.com. Department of Engineering, University of Ferrara via Saragat 1 – 44100, Ferrara, Italy. ‡University of California San Diego, UCSD. 9500 Gilman Drive, La Jolla, CA 92093-0436, US. ⋆Department of Information Engineering, University of Padova, via Gradenigo 6/B – 35131, Padova, Italy.

II. System Model

We model the network as a directed weighted graph $\mathcal{G} = (N, A)$, consisting of a set N of nodes. $|N| = m$ is the cardinality of N, that is composed by $m-1$ nodes and one special node (the *sink*) whose function is to gather and process network messages. The set A is a set of ordered pairs (i, j), where $i, j \in N$. The pair (i, j) is referred to as the link connecting node i with node j. The link (i, j) exists if i and j are within transmission range. In order to keep the analysis as general as possible, we do not specify here any propagation/connectivity model. In fact, our analysis is based on neighboring sets, i.e., on sets of nodes within coverage that verify certain properties. As the network may be highly dynamic, these sets may vary between subsequent forwarding actions and are therefore dependent on many factors such as connectivity model and node sleeping features. It is then reasonable to obtain these sets *on–demand* when forwarding decisions have to be actually made. For what concerns the cost associated with a link it can be represented, for instance, by a function related to the energy required to transmit an information bit from node i to node j, but other factors can also be taken into account, such as the node failure probability and/or the amount of traffic at node j, i.e., the state of its queue. In our investigation, we do not propose a specific cost model, so as to keep our results as general as possible. We define a path $\mathcal{P}$ from node s to node d as an ordered list $\mathcal{P} = \{s, r_1, r_2, \ldots, r_n, d\}$, where nodes s and d are referred to as the source and the destination node, respectively. Here, we only consider the loop–free oriented paths connecting node s to node d. The nodes r_i, $i \in \{1, 2, \ldots, n\}$ are referred to as relay nodes. The cost $C(\mathcal{P})$ associated with a path $\mathcal{P}$ is given as follows[1]

$$C(\mathcal{P}) = c_{sr_1} + \sum_{i=1}^{n-1} c_{r_i r_{i+1}} + c_{r_n d} \tag{1}$$

Choosing an additive cost function as the path cost criterion is reasonable as additive metrics arise in many settings. For instance, end-to-end delay, delay jitter, maximum total residual energy and reliability (logarithms) all correspond to the sum of link weights.

We further assume that the cost c_{ij} does not depend on node i, i.e., $c_{ij} = c_j$, $\forall\ j \in N$. This is, of course, a simplifying assumption that can be removed in future research. This assumption is reasonable when all nodes transmit with the same constant power. Hence, no power control is accounted for here. A possible model for the cost function could be, for instance, related to the residual energy at every node. Consider node $j \in N$ and let its residual and initial energy be $E_{res}(j)$ and $E_{init}(j)$,

[1]Under the assumption of additive cost function, see [3].

$D \leftarrow$ sink node;
$k \leftarrow$ current node;
repeat
  $\mathcal{N}(k)^{<} = \{i \in \mathcal{N}(k)$ s.t. $HC(i) < HC(k)\}$;
  $i^\star = \text{argmin}_{i \in \mathcal{N}(k)^{<}} c_{ki}$;
  Break ties arbitrarily. $k \leftarrow i^\star$;
until $k = D$;

Algorithm 1: Greedy forwarding algorithm.

respectively. If $E_{init}(i) = E_{init}(j) = E_{init}$, $\forall\ i,j$, the cost associated to transmitting to node j could therefore be written as $c_j = 1 - E_{res}(j)/E_{init}$, where $0 \leq E_{res}(j) \leq E_{init}$ and $c_j \in [0,1]$. Of course, the lower the residual energy, the higher the cost of choosing that node as the relay. Observe that the cost model is itself very important since it strongly influences the properties of the solutions found by the cost–based routing algorithms that will be discussed in the sequel.

III. Hop Count Routing Algorithms

A. Greedy Path Selection Algorithm (GREEDY)

In addition to node costs, in the following we consider node hop counts. We say that a given node has hop count (HC) equal to $i > 1$ if the minimum number of hops (transmissions) for a packet to get to the sink from that node is equal to i. A first routing scheme is reported in Alg. 1, and is based on the same concepts as in [4], with the difference that here the next hop is selected thanks to node HC rather than using geographical coordinates. When a data packet is generated at a source node, say node k, a neighbor set $\mathcal{N}(k)^{<}$ is obtained from $\mathcal{N}(k)$ by picking the nodes in $\mathcal{N}(k)$ with a lower hop count value with respect to the HC at node k (HC(k)). In this way, we select the next hop among the nodes that lead to the maximum advancement toward the sink. Moreover, we pick the next hop from the set $\mathcal{N}(k)^{<}$ by selecting the lowest cost node in the set. This second action is performed with the aim of minimizing the overall path cost. In the following sections, we propose a refined greedy forwarding algorithm, where some statistical knowledge regarding the costs of the nodes placed in the second–order neighbor (two hops away) is considered to improve forwarding decisions.

B. Routing as a Sequential Decision Problem

We formulate the routing problem as a sequential decision problem, where at every stage a node has to select a specific action, i.e., the best node to act as relay for the current packet. In particular, we are interested in *on–line* routing algorithms, where forwarding decisions are made based on local knowledge and on some statistical information regarding the second-order (two hops away) neighborhood of the current node. With the term *local knowledge*, we mean here the knowledge of the costs of those nodes within radio range. We assume that the currently occupied node is node $i \in N$, that its hop count is HC$(i) = n$ and that the forwarding process is at stage $t \geq 0$, $t \in \mathbb{N}$ where the time evolves by one unit every decision round. We define $\mathcal{N}(i)^{<}$ and $\mathcal{N}(i)^{=}$ as the set of nodes within range of node i with hop count $n-1$ and n, respectively. The problem to be solved by the decision maker is therefore to assess the best node to be selected to act as relay among the nodes in sets $\mathcal{N}(i)^{=}$ and $\mathcal{N}(i)^{<}$. Nodes in set $\mathcal{N}(i)^{>}$ are excluded *a priori* since, in normal operational conditions, they do not lead to satisfactory solutions.[2] We refer to $j^t_{n-1} \in \mathcal{N}(i)^{<}$, $j^t_n \in \mathcal{N}(i)^{=}$ and to c^t_{n-1}, c^t_n as the minimum cost nodes[3] in sets $\mathcal{N}(i)^{=}$ and $\mathcal{N}(i)^{<}$ and their associated costs, respectively. We finally refer to *forwarding cycle* for hop count n as the forwarding history between the first selection of a node with HC equal to n and the first selection of a node with HC equal to $n+1$.

C. One-Step Ahead Prediction Routing Techniques

Consider a generic forwarding step $t \geq 0$, $t \in \mathbb{N}$ and consider that the packet at time t is at node i with HC$(i) = n$ and that time 0 corresponds to the instant when the current *forwarding cycle* has started. At time t the *decision maker* (node i) has to choose a forwarding action, i.e., whether the packet has to be forwarded to node j^t_{n-1} or to j^t_n. We define the action set and the decision maker's current state as $\mathcal{A}_t = \{a^t_{n-1} = j^t_{n-1}, a^t_n = j^t_n\}$ and $X_t = (c^t_n, c^t_{n-1})$, respectively. Moreover, we assume that if action $a(t) \in \mathcal{A}_t$ is chosen when in state X_t, $t \geq 0$, a cost $C(X_t, a(t)) \geq 0$ is incurred. For any routing policy π, the total expected cost incurred over time when X is the initial state is defined as [5]

$$V_\pi(X) = E_\pi\left[\sum_{n=0}^{+\infty} C(X_n, a(n)) | X_0 = X\right] \quad (2)$$

Moreover, let $V(X) = \inf_\pi V_\pi(X)$ be the minimum expected cost under any policy [5]. We say that a policy π^* is optimal if $V_{\pi^*}(X) = V(X)$, $\forall X$. The optimal policy is determined by the following optimality equation [5]

$$V(X_t) = \min_{a(t) \in \mathcal{A}_t}\left[C(X_t, a(t)) + \int_{\mathcal{D}_X} V(X_{t+1})\, \mathrm{d}F(X_{t+1})\right] \quad (3)$$

where X_t and X_{t+1} are the current and the next state, respectively, $C(X_t, a(t))$ is the cost incurred at the current decision step t, the term $\int_{\mathcal{D}_X} V(X_{t+1})\, \mathrm{d}F(X_{t+1})$ accounts for the average cost incurred in future decisions, $\mathcal{D}_X$ is the domain set of X_{t+1} and $F(X_{t+1})$ is the cdf governing the state for the next forwarding step. Our forwarding process can therefore be modeled as an optimal stopping problem, where at the generic step t the decision maker can either decide to continue ($a(t) = a^t_n$) or stop ($a(t) = a^t_{n-1}$). In what follows, we discuss a first possible (and simple) way to model the costs associated with this decision process. In this first approach if $a(t) = a^t_n$, a cost $C(X_t, a^t_n) = c^t_n$ is paid and the cycle is continued, while if $a(t) = a^t_{n-1}$ the cycle is ended with a final cost $C(X_t, a^t_{n-1}) = c^t_{n-1}$ and the integral $\int_{\mathcal{D}_X} V(X_{t+1})\, \mathrm{d}F(X_{t+1})$ is zero because once the cycle has ended all the future costs are zero by definition. Keeping this in mind

[2] This has been verified by extensive simulations and is also supported by previous studies.

[3] In the case where there are multiple nodes with the same minimum cost in one of the two sets, we indifferently refer to one of them as they are, by definition, equivalent. Moreover, without losing generality, we consider $c_{ij} = c_j$, i.e., costs only depend on the receiving node.

and considering Eq. (3), we may define the following set

$$\mathcal{B}_1 = \Big\{ X_t : C(X_t, a_{n-1}) \le C(X_t, a_n) + \int_{\mathcal{D}_X} C(X_{t+1}, a_{n-1})\, \mathrm{d}F(X_{t+1}) \Big\} \quad (4)$$

this set contains the states for which stopping is at least as good as continuing for one more period and then stopping. The policy that stops the first time the process enters the set $\mathcal{B}_1$ is called *one–stage look–ahead policy*. Set $\mathcal{B}_1$ simplifies to

$$\mathcal{B}_1 = \Big\{ X_t : c_{n-1}^t - c_n^t \le \mathcal{E} \Big\} \quad (5)$$

where

$$\mathcal{E} = E[c_{n-1}^{t+1}] = \int_0^1 c_{n-1}^{t+1}\, \mathrm{d}F(c_{n-1}^{t+1}) \quad (6)$$

is the expected minimum cost among nodes with hop count $n-1$ at stage $t+1$.[4] Therefore, the one–stage optimal policy tells us to stop at the instant in which set $\mathcal{B}_1$ is entered for the first time, i.e., at time t we should select node j_{n-1}^t and end the cycle only if $c_{n-1}^t - c_n^t \le \mathcal{E}$. Note that the one–stage policy is only locally optimal, whereas in general this policy does not correspond to the globally optimal behavior. Global optimality and related policies are the object of our current research. However, in the next section, we will show that this routing scheme is able to highly improve the performance of the greedy solution presented in Section III-A.

IV. Results

In this section we compare the performance of the routing algorithms presented above. As a reference model, we consider a random topology network, where nodes are placed according to a planar Poisson process with node density $\lambda_n = \lambda \pi R^2$, whereas R is the constant node transmission range. We consider a unit disk connectivity model, i.e., two nodes can communicate iff their distance is within R. However, it is worth observing that the schemes discussed here can work for any topology setting as λ and the connectivity model just translates into different neighboring sets. In the following numerical results, we consider $\lambda_n = 15$, $R = 1$ and nodes are randomly placed on a square area of $16R \times 16R$. Moreover, we focus on the quality of the path from a source node with HC $= 8$. Every node is assumed to have a good estimate of the expected minimum cost $\mathcal{E}$ (Eq. (6)) related to the second order neighborhood of the current node and node costs are uniformly distributed in their definition set. This is assumed here with the aim of understanding how close to the optimal solution we can get with HC policies. The next results are therefore valid from a theoretical point of view (cost estimates are error free) and give us an indication of the maximum achievable gains with respect to geographical and greedy routing. As will be shown in the following, the performance gain is good and thereby encouraging to proceed with further research in this direction. In addition to the hop count based schemes discussed above, we consider an idealized geographic routing algorithm where we subdivide the relaying area into a number N_r of priority regions, according to the related advancement toward the destination [2]. In this scheme, the relay is always the lowest cost node within the non-empty region with the highest priority, i.e., the lowest cost node leading to the maximum advancement toward the destination. In Fig. 1, we plot the probability that the cost of on–line routing algorithms exceeds the cost of the optimal path by ΔC, where ΔC is expressed as a percentage of the optimal path cost (computed by means of an off-line standard optimization procedure). In particular, we show the performance of the geographic routing scheme (GEO) and of the on–line routing schemes reported in Section III-A (GREEDY) and Section III-C (that we name here as Statistically assisted Greedy Routing, SGR). Clearly, geographical routing performs worse than both GREEDY and SGR. Further, it is not clear how the number of geographical regions N_r can be chosen in order to improve the optimality in terms of cost. Apparently, the maximum geographical advancement is not a good strategy to select low cost paths. How to couple geographical advancements and costs in a more effective strategy than subdividing the forwarding area into priority regions (N_r) is the objective of our current research. From this figure, it is also clear that the exploitation of one–step ahead cost predictions makes SGR perform significantly better than GREEDY.

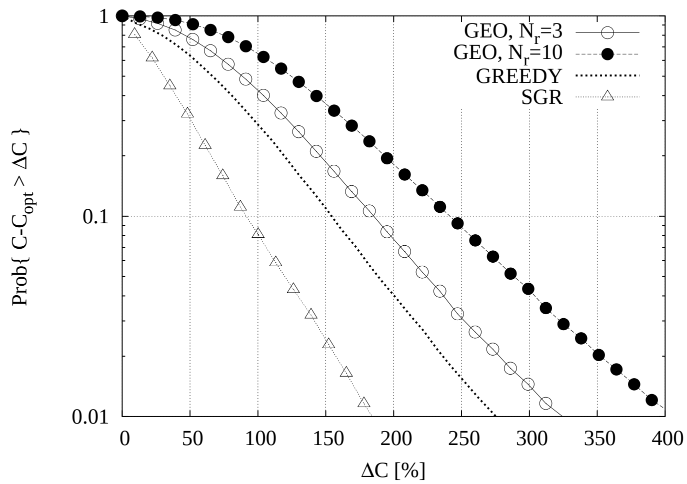

Fig. 1. Complementary distribution function of the cost difference between on–line routing strategies and the (off–line) non–dominated optimal cost solution.

[4] We restrict the relay selection to the minimum cost nodes in sets $\mathcal{N}(\cdot)^{=}$ and $\mathcal{N}(\cdot)^{<}$.

References

[1] T. Melodia, D. Pompili, and F. I. Akyildiz, "Optimal Location Topology Knowledge for Energy Efficient Geographical Routing in Sensor Networks," in *Proceedings of IEEE Infocom*, Hong Kong, P.R., China, Mar. 2004.

[2] M. Zorzi and R. R. Rao, "Geographic Random Forwarding (GeRaF) for Ad Hoc and Sensor Networks: Multihop Performance," *IEEE Trans. on Mobile Computing*, vol. 2, pp. 337–348, Oct-Dec 2003.

[3] Roch Guérin and Ariel Orda, "Computing Shortest Paths for Any Number of Hops," *IEEE/ACM Trans. Networking*, vol. 10, no. 5, pp. 613–620, Oct. 2002.

[4] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," in *Proceedings of ACM MOBICOM*, Boston, Massachusetts, US, Aug. 2000, pp. 243–254.

[5] S. M. Ross, *Introduction to Stochastic Dynamic Programming*. Academic Press, 1983.