Localized Network Layer Protocols in Wireless Sensor Networks Based on Optimizing Cost over Progress Ratio

Ivan Stojmenovic, University of Ottawa

Abstract

We present a simple framework for designing network layer protocols for sensor networks including localized routing, broadcasting, area coverage, and so on. The framework is general enough and is applicable to a variety of problems, network assumptions, and optimality criteria. Our simple framework is based on optimizing the ratio of the cost of making certain decisions (e.g., selecting a forwarding neighbor for routing) to the progress made in doing so (e.g., reduction in distance to destination). We show how to apply this general guideline for the design of hop count, power awareness, maximal lifetime, beaconless and physical-layer-based routing, minimal energy broadcasting, sensor area coverage, and multicasting protocols. Moreover, we show that in the case of routing, the best known strictly localized position-based techniques are, in almost all cases, special cases of the described general cost-to-progress ratio paradigm.

Recent technological advances have enabled the development of low-cost, low-power, and multifunctional sensor devices. Large collections of these devices organized in multihop sensor networks have the potential to revolutionize the way humans can "sense" the physical world. Applications of sensor networks are envisioned primarily for monitoring the environment (e.g., motion, target tracking, fire detection, chemicals, temperature) or as embedded systems (e.g., biomedical sensor engineering, smart homes). Sensors may measure distance, direction, speed, humidity, wind speed, soil makeup, temperature, chemicals, light, vibrations, motion, seismic data, acoustic data, strain, torque, load, pressure, and so on.

Sensors provide radically new communication and networking paradigms. They have small size, low battery capacity, small processing power, limited buffer capacity (thus routing tables, if used at all, must be small), and a low-power radio. They can be static and placed to monitor an area, or mobile (e.g., when attached to robots, soldiers, or vehicles). These nodes are autonomous devices with integrated sensing, processing, and communication capabilities. Sensor networks consist of a large number of sensor nodes that collaborate together using wireless communications with an asymmetric many-to-one data transfer model — sensors will typically send their data to a specific node called the sink node or monitoring station, which collects the requested information. Nodes in a sensor network are generally densely deployed. Thousands of sensors may be placed, mostly at random, either very close to or inside a phenomenon to be studied. Once deployed, the sensors are expected to self-configure into an operational wireless network. The limited energy budget at the individual sensor level implies that in order to ensure longevity, the transmission range of individual sensors is restricted. In turn, this implies that wireless sensor networks must be multihop.

In addition to traditional network layer problems such as routing, broadcasting, and multicasting, sensor networks impose their own specific challenges at the network layer. Because of their low power and low traffic demands, activity scheduling (deciding which sensors are active and which sleeping, and the rules of status changes) is a major issue for prolonged network life. To provide monitoring capability, it is important to place or select them so that the monitored area is covered as much as possible. This is known as the sensor area coverage problem, deciding which sensors need to remain active so that a certain area is minimally but fully monitored by the active sensors. There are also a variety of object location, path exposure, data dissemination, gathering, fusion, and other relevant issues.

The sensors in most cases use position information in their decisions. The availability of position information was widely recognized in the research community as highly desirable for the proper operation of the sensor network; however, it is a nontrivial problem, and the precision of the location information may impact the performance of communication protocols. There are a variety of position determination (or localization) protocols, with a variety of message complexities and position accuracies.

Protocol Design Criteria

The selection of a routing protocol for a certain network depends on the information available at the network nodes and the communication overhead that can be tolerated. Table 1 presents a taxonomy of existing routing approaches in this respect. In networks where each node is provided with full topology information, and which is stable over time, such as interconnection networks (e.g., hypercubes and meshes) used

0890-8044/06/$20.00 © 2006 IEEE

Centralized	Shortest (weighted) path	Interconnection networks
Distributed	Bellman-Ford	Internet
Localized — flooding	Route discovery	Dynamic ad hoc networks
Localized — path based	Greedy, Cost/progress ratio	Sensor networks

Table 1. *Communication overhead based taxonomy of routing protocols.*

for parallel computing, the shortest (possibly weighted) path algorithm can be applied to route a message from any source to any destination. Large networks with reasonably stable nodes over time, where autonomous nodes do not know the full network graph (e.g., the Internet), require a *distributed* routing approach. For this type of network, the Bellman-Ford protocol is mostly applicable. According to this protocol, nodes periodically exchange routing tables that contain the first hop toward "strategic" destinations and corresponding costs with neighbors in order to update their own tables. Each node checks whether the cost of routing via a given neighbor is smaller than that currently recorded in its routing table, and updates its next hop toward the destination if so.

Ad hoc networks are envisioned in conference, rescue and military scenarios, with nodes being personal communication devices. The bandwidth limitations of the wireless channel and the dynamic nature of ad hoc networks, where mobile nodes frequently join and leave the network, pose new challenges for routing protocols. Distributed protocols require significant communication overhead to be considered for using in such networks. The protocols need to be *localized*, where nodes make decisions solely based on the information available from their neighbors. Although some protocols appear localized, an extensive message exchange with neighbors, especially for maintenance reasons, amounts to collection and use of global information (e.g., Bellman-Ford type of routing protocols).

In dynamic ad hoc networks, a localized approach based on on-demand route discovery by flooding destination request packets is appropriate. However, power and bandwidth limitations, reduced computational capabilities, wireless channel characteristics (omnidirectional antennas and communication on a single common channel), and the dynamic nature of sensor networks require the design of network layer protocols satisfying a number of further properties under a general localized paradigm. It is extremely (power and bandwidth) inefficient to use flooding as a routing scheme in sensor networks, if a solution that provides a route competitive to the shortest (weighted) path is available. For sensor networks, path-based solutions like those discussed in this article are therefore the only viable routing approach.

Localized protocols can be further divided according to the amount of information required and the overhead in the construction and maintenance phases. This is especially important for network layer problems that inherently affect all nodes in the network, such as broadcasting and sensor area coverage. The amount of required information is related to the *message complexity*, which can be defined as the average number of transmitted messages per sensor node in a protocol. In a *strictly localized protocol*, all information processed by a node is either local or global in nature, but obtainable in short constant time by querying only the node's neighbors or itself. In other words, only a small bounded number of message exchanges with neighbors is allowed. Strictly localized protocols may need some information that is part of their input (e.g., destination position in a routing protocol) but cannot use structures that are global in nature (e.g., information on which outgoing link belongs to the minimum spanning tree). Cost over progress-ratio-based protocols, described in this article, belong to the class of strictly localized protocols.

For a number of network layer problems, a frequent solution is to introduce additional parameters, not present in the problem formulation, as part of the solution protocol. The protocol performance then depends on the selected parameter values. In most cases the best values of these parameters depend on global network conditions, which may be beyond the knowledge available to tiny sensors. A typical approach is to use thresholds in solutions. The effect of thresholds is to eliminate certain options in the protocols believed to lead to suboptimal solutions. However, such elimination also contributes to increased failure of the protocol. In the scenarios considered so far, we have shown that solution approaches that avoid thresholds achieve a much better trade-off between success rate and communication overhead.

In this article we introduce a simple solution concept that is applicable to new assumptions and scenarios. Specifically, we propose design guidelines based on optimizing the ratio of operation cost (measured in the context of the problem statement) and progress made by applying the concept (e.g., reduction in distance to route destination, or coverage achieved in broadcasting or sensor area coverage problems). Several special cases of this proposed design guideline have already been described in the literature. The goal of this article is to describe the concept as a general framework, provide a "top" view of some existing protocols, apply it to several additional problems, and derive some existing optimization criteria by simpler means.

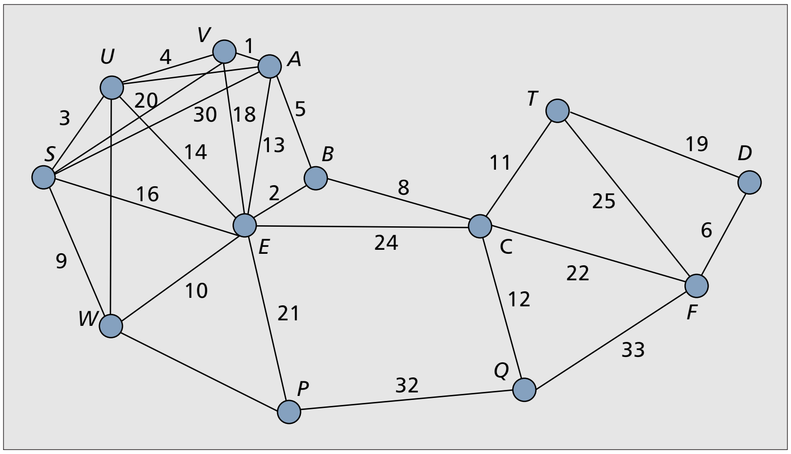
In the next section we discuss the cost to the progress ratio framework of the routing task, using a variety of metrics: hop count, power, reluctance, delay, and expected hop count. In the subsequent section we show that the framework is more general, by applying it in the context of broadcasting, multicasting, and sensor area coverage problems.

Routing in Sensor Networks

We first address the routing task when nodes are equipped with position information. Each node is assumed to know the position of itself, all its neighbors, and destination D. In addition, local knowledge at each node includes the cost of each of its links to neighboring nodes. The position of neighbors may be gained via periodic exchange of "hello" messages. In sensor networks the destination is normally a sink whose position is flooded to all sensors prior to their reporting.

Cost to the Progress Ratio Routing Framework

We now explain the general framework for our localized routing scheme. Suppose that each edge has a cost measure. The cost measure depends on the assumptions and metrics used, while progress measures the advance toward the destination. In the next section we study specific cost measures such as *hop count*, *power*, *reluctance*, *power_reluctance, delay*, and *expected hop count*. Figure 1 shows an example with costs listed. If global knowledge is available at each node, the optimal (shortest weighted) path from source S to destination D would be $SUVABCFD$ (the total cost $3 + 4 + 1 + 5 + 8 + 22 + 6$ is minimal among total costs of all paths between S and D). However, we assume that each node has only local information. Thus, source S has five choices to forward a packet to D, neighbors U, V, A, E, and W, and no knowledge of other nodes in the network except D. What is the best choice for S,



■ Figure 1. *A sensor network with costs at each edge.*

based on such local information? We argue that the cost of the selected link should be evaluated against the progress made toward the destination. We measure the progress at the reduction in distance to destination of the current node to the distance of its neighbor. That is, ratios $3/(|SD|-|SU|)$, $20/(SD|-|SV|)$, $30/(|D|-|SA|)$, $16/(|SD|-|SE|)$, and $9/(|SD|-|SW|)$ are compared, and the minimal one is selected. The rationale for the method is that it attempts to minimize the total cost by favoring neighbors closer to the destination (thus reducing the number of hops) and with smaller costs, with a suitable simple formula to choose a "winner" at each step.

The selected neighbor applies the same criterion to forward further. That is, node C, currently holding the packet, will forward it to neighbor A, closer to destination D than itself (Fig. 2), which minimizes *cost(CA)/progress(A)*, the *ratio of cost over progress*.

Consider, as an alternative, a threshold-based approach. Suppose that edges with costs larger than $t = 20$ are ignored, and each node selects the neighbor closest to D among those with cost below 20. In our example the optimal route would then be *SUVABCTD*, which has larger cost. Moreover, deletion of such links may disconnect the network. For instance, if the threshold is $t = 18$, D becomes disconnected from S. Thus, both total cost and success rate may suffer. Furthermore, such a threshold-based approach has a parameter t whose best value may not be derived based on local knowledge. We are striving to design competitive parameterless schemes as part of our framework.

Note that only neighbors closer to the destination than the current node are considered for forwarding, to ensure some progress at each step. If no such neighbor exists, the packet is dropped; alternatively, a recovery scheme is applied, such as face routing described in [1].

All routing protocols based on the cost-to-progress ratio can be improved by applying the *iterative progress* method (proposed independently in [3, 6]) as follows. Suppose that current node C selected neighbor A based on a certain criterion (e.g., the cost-to-progress ratio considered here), while the overall goal is to minimize the total sum of costs over a route. If there is another neighbor B such that $cost(CB) + cost(BA) < cost(CA)$, it is more beneficial to forward the packet to neighbor B instead. This search can be repeated iteratively until no improvement is possible. Note that this verification is performed at C without message exchanges with neighbors, since C is aware of the position of all its neighbors (and destination). This position information provides distance information between nodes, which uniquely determines a power metric.

Greedy Routing

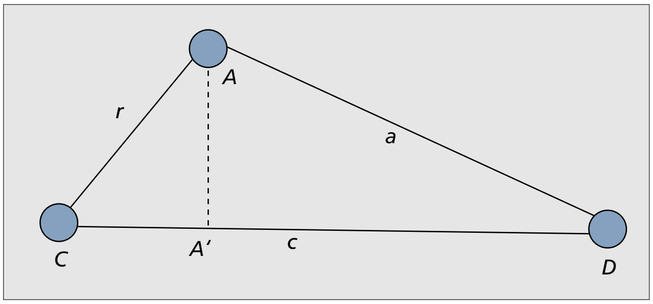
Depending on the context and cost and progress measures applied, this general framework leads to specific protocols. Among derived instances are also several well-known protocols. A simple special case of our general framework is *greedy* routing in sensor networks by Finn [2] for the generally accepted unit disk graph (UDG) model (two nodes communicate if and only if the distance between them is at most R, where R is the transmission radius, equal for all nodes). Let C be the node currently holding the message, D be the destination node, A the considered forwarding neighbor, $|CD| = c$, $|AD| = a$, and $|CA| = r$ (Fig. 2). The cost metric normally used in UDG is the hop count. Therefore, the cost of going to any neighbor is one hop. The progress made by forwarding from C to A is $c - a$, which is the difference between the distances to the destination from the current node and the neighboring node. Therefore, the protocol will minimize $1/(c - a)$, or maximize $c - a$; that is, the neighbor closest to the destination would be selected. For example, in Fig. 1 source S forwards the packet to neighbor A that is closest to destination D. Greedy routing continues, creating route *SABCFD*. A shortest path algorithm would yield route *SECFD* with fewer hops, but its application requires global knowledge at each node.

An alternative measure of progress is the distance $|CA'|$, where A' is the projection of A on line CD (Fig. 2). This method is known as MFR (which stands for most forward within radius), proposed by Takagi and Kleinrock [2]. In all subsequent examples, this progress measure can replace the $c-a$ measure, which will be used by default. Note that in these and other protocols described here, only neighbors closer to the destination than the current node are considered. This provides a loop-free protocol, while a recovery scheme (to guarantee delivery) when such a neighbor does not exist is described by Bose, Morin, Stojmenovic, and Urrutia [1]. For completeness, we note that the only known remaining routing protocol, based on discussed assumptions, is directional routing [2] where the current node forwards the packet to one or more neighbors making the smallest angle with respect to the direction of the destination. However, they were shown to have inferior hop count (compared to greedy routing) and are not loop-free [2].

Power-Aware Routing

The next example is localized power-aware routing, first studied in [11]. The power needed to send a packet from C to A is proportional to $r^\alpha + c$, where α is power attenuation factor ($2 \le \alpha \le 6$), $r = |CA|$, while c is a constant ($c > 0$). Constant c accounts for the energy needed to run electronic circuits at transmitter and receiver and minimal signal strength for correct signal reception. This power measure can be used as a cost measure in our general protocol. Therefore, the neighbor that minimizes $(r^\alpha + c)/(c - a)$ will be selected [6]. This means that the selected neighbor minimizes the power spent per unit of progress made in terms of getting closer to the destination.

In [11] it is shown that if additional nodes can be placed at desired locations, the optimal forwarding distance is $(c/(\alpha - 1))^{1/\alpha}$. This is used to derive a formula for minimal power $v(a)$ for routing between two nodes at distance a. The neighbor that minimizes $r^\alpha + c + v(a)$ is then selected [11] (Fig. 2). The optimal forwarding distance can also be confirmed using the cost-progress ratio concept. The cost of transmitting a packet at distance r is $r^\alpha + c$, while the progress made toward the destination is r. The ratio $(r^\alpha + c)/r$ is optimal for exactly



■ Figure 2. *Current node* C *selects the best neighbor* A *in localized routing schemes.*

the same forwarding distance as in [11], and the proof is simpler (using the standard calculus method of finding the root of the first derivative). Therefore, the existing algorithm [11] can also be described as a different application of the cost-to-progress ratio framework.

Power-aware routes may drain energy from certain nodes. It is therefore desirable to consider instead the *maximal lifetime routing* problem, where the goal is to maximize the number of routing tasks the network can perform. This definition does not provide a clear measure of optimality. Two such measures considered in [11] are *reluctance* (proposed by Singh, Woo, and Raghavendra [11]) and *power_reluctance*. Reluctance corresponds to the willingness of a node to participate in routing. Nodes with more energy are more eager to assist, while nodes with less remaining energy show more reluctance to do so. As a particular choice for the reluctance measure $f(A)$ of node A, the inverse of the normalized (i.e., maximum energy corresponds to 1) remaining energy can be used. The algorithm then selects neighbor A that minimizes $f(A)/(c - a)$. If a reluctance metric is used, nodes need to include the information on their remaining energy in their "hello" messages. Note that the algorithm in [11] minimizes instead $f(A)(R + a)$, where R is the transmission radius.

Somewhat better results are obtained when *power_reluctance* cost measures $f(A)(r^\alpha + c)$ [11] is used. This leads to the protocol that selects the neighbor minimizing $f(A)(r^\alpha + c)/(c - a)$ [6]. Such a choice avoids the use of parameters (combining separate power and reluctance measures with certain parameter weights), and experimental data in [11, 6] show that such a parameterless choice is not inferior to a number of attempted parameter-based combined measures. These experiments also show competitive performance of the described localized protocols with respect to the "optimal" shortest-weighted-path-based solutions, which require global knowledge at nodes to be applied. Experimental data in [6, 11] also show similar performance of cost-to progress-ratio-based schemes to other localized parameterless power-aware localized schemes [11] that are more sophisticated.

Bit-Rate-Aware Routing

Consider now the case where nodes transmit with the same transmission energy, but can adjust bit rate (number of bits transmitted per unit time) based on the distance of the selected neighbor. Note that such a bit rate adjustment already exists in medium access protocol standards such as IEEE 802.11. However, the equipment supporting it normally provides only several discrete bit rate levels. The transmission energy is normally fixed at E, and the energy needed to transmit one bit is proportional to $r^\alpha + c$. Thus, the bit rate with a neighbor at distance r is proportional to $E/(r^\alpha + c)$. Because of existing threshold reception power, closer neighbors may use higher bit rates (the selection is, in existing equipment, done even automatically to the fastest available bit rate for communication between two nodes). The transmission time (which is inversely proportional to the bit rate) for a packet of fixed length is then proportional to $r^\alpha + c$. We have used here the constant value c, although more careful analysis at the physical layer may reveal its dependence on the selected bit rate. Interestingly, the problem of minimizing the time needed for routing with adjustable bit rate and fixed transmission power appears to produce the same routes as that used for power-aware routing.

QoS Routing

The routing scenarios considered so far did not involve the transport layer (i.e., we considered only one routing task at a time). Suppose now that there is ongoing traffic, and consider how to choose the best forwarding neighbor in few scenarios. In quality of service (QoS) routing, the goal is to minimize the time needed to send a message from source to destination. The cost in this case is the *delay* experienced if a message is sent to a particular neighbor before the message is forwarded (delay includes queuing delay and the time to forward the message). The current node may select the neighbor that minimizes $delay/(c - a)$. This problem is considered in [3], where the proposed solution minimizes $delay/|CA'|$. A similar idea is used as part of the SPEED protocol [4], which proposes to compute $delay/(c - a)$ for each neighbor, and use them as weights in a probabilistic decision. However, if no node has this ratio over a predetermined threshold, a scheme that avoids congested areas is applied.

Consider now the case of variable bit rates in scenarios with congestion. The delay then consists of access (queuing) delay and transmission delay. The latter is determined by the ratio of packet size and selected bit rate. Therefore, the selected neighbor, using the cost-over-progress scheme, is the one that minimizes $(access\_delay + packet\_size/bit\ rate)/(c - a)$.

QoS routing for multimedia traffic may involve both time and bandwidth limitations. Bandwidth limitations can be handled in two ways. One is to ignore links with insufficient bandwidth for given traffic demand, and select any of the remaining ones based on another criterion (e.g., delay). This provides good solutions for the task at hand, but may leave reduced choices for other routing tasks if "tiny" links (links with small difference between available and requested bandwidths) are overused. It is also a threshold-based solution, which may fail just because a particular link for a certain time had somewhat lower bandwidth. A better metric might be to use the inverse of the normalized remaining bandwidth. The delay can be approximated with the number of hops taken, due to a similar processing requirement at each node, and fixed overall capacity. The cost is then $1/bandwidth$, and the selected neighbor is one that maximizes $bandwidth(c - a)$.

Routing with a Realistic Physical Layer

We have considered so far the UDG model. Consider now a more realistic physical layer model. Instead of merely using the transmission radius as in the UDG model, the physical, medium access control (MAC), and network layers share the information about a bit and/or packet reception probability $p(r)$ as a function of distance r between nodes and other environmental variables. We assume that all nodes use the same transmission power for sending messages. The MAC layer reacts to this probabilistic reception information by adjusting the number of acknowledgments and/or retransmissions. It was demonstrated (by Schmitz, Torrent-Moreno, Hartenstein, and Efelsberg at IEEE WLN 2004) that signal strength fluctuations have a significant impact on ad hoc network performance metrics, sometimes "outperforming" the impact of node mobility. Therefore, the hop count metric does not properly reflect the cost involved. The expected hop count

(EHC) needs to be used instead of the hop count, which measures all the (re)transmissions and possibly acknowledgments on each hop.

There are several possible assumptions regarding acknowledgments. If acknowledgments are not used, $EHC(r) = 1/p(r)$, which measures the expected number of retransmissions for a message to be correctly received. If acknowledgments are used, it can be shown that EHC, when the optimal number u of acknowledgments is used ($u \approx 1/p(r)$), is approximately proportional to $1/(p(r)(1 - (1 - p(r))^u))$ [7] (the constant of proportionality depends on the relative size of an acknowledgment with respect to the packet size), which can be used as a value for $EHC(r)$. Our framework also includes the case of variable packet sizes at the MAC layer, for which the corresponding value for $EHC(r)$ can be derived and used in the same routing protocol (ongoing work by authors of [6]). Therefore, under a variety of particular assumptions, one can apply the general cost-over-progress ratio scheme, where current node C will forward the message to the neighbor that minimizes $EHC(r)/(c - a)$, where $EHC(r)$ is measured accordingly.

The proposed routing algorithms and several others are simulated in [7], using a model without collisions from other traffic and with the ideal MAC layer (where each link has cost exactly as in the EHC metric). The experimental results clearly show that the shortest path algorithm with the hop count metric performs poorly when it is evaluated with the new metric, since long links with many retransmissions are preferred. The ideal routing protocol is obviously the shortest weighted path algorithm, with EHC as the cost of each link. This protocol provides optimal routes if nodes are provided with full (global) network information (positions of all nodes and consequently the EHC for every link in the network). The simulations show very competitive performance (closely matching the performance for sufficiently dense networks) of localized routing protocols compared to the ideal shortest weighted path solution, which requires global network knowledge (and therefore unacceptable communication overhead for its maintenance) at each node. The experiments also show that the localized cost-over-progress-based routing protocol is superior to the localized threshold-based greedy protocol for any selection of threshold. In the latter protocol the message is sent to the neighbor, closest to the destination among neighbors that are at distance $\leq tR$, where t is a parameter threshold and R is a reference distance (in [7], R is determined by $p(R) = 0.5$). For instance, the best value for threshold t in [7] appears to be near $t = 1.25R$. For this t, the success rate is lower by over 20 percent for densities below 10, while the EHC is over three times larger for densities over 10; the other metric in both cases still does not show better data.

Note that [7] also proposed a protocol that minimizes the sum of EHC to a neighbor plus the ideal EHC from neighbor to destination. The latter is obtained by calculus, but can also be derived by a cost-to-progress ratio argument. Details are omitted for space limitations. Note also that [7, 10] stated that cost-over-progress-ratio-based routing can be applied for arbitrary cost measure.

Greedy routing in UDGs was shown to be a special case of our general framework. However, it can also be considered a special case of threshold-based design, with transmission radius serving as the threshold. We have shown that the cost-to-progress ratio paradigm provides better generalization than a threshold-based approach, which simply decides what the best new threshold (generalization of the transmission radius-notion) for greedy routing is. Therefore, when hop count (in UDG) is replaced by EHC (in a realistic physical layer), it is not sufficient to merely clarify what the transmission radius (new threshold for greedy routing) is in order to design an optimal localized routing protocol. That is, we argue that the proposed cost-to-progress ratio framework is a natural generalization for greedy routing when the UDG is replaced by a realistic physical layer. Moreover, we argue that there is no need to introduce an additional parameter (threshold t in the above protocol) into the routing protocol, since the threshold-based protocol has inferior performance (to our cost-to-progress framework) for all threshold values.

An example of the cost-to-progress-ratio-based routing paradigm is discussed in [12]. Progress is measured by the MFR method, while cost measures local throughput on a link, and incorporates spectral efficiency and a modulation coding scheme. The ratio is referred to as *information efficiency*.

Beaconless Routing

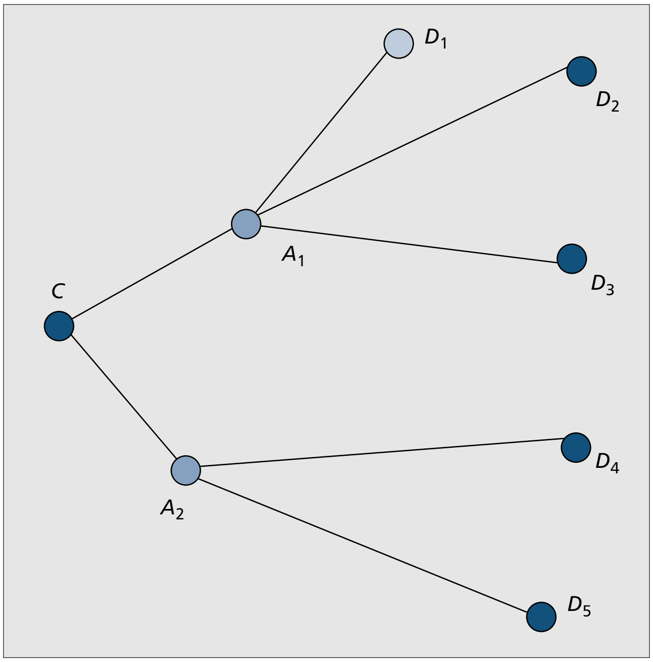
The described position-based routing schemes assume that each node knows the position of all its neighbors. This is not necessary because an adjustment can be made if neighbor knowledge is not available. One can apply the idea of beaconless routing using the variant described in [13], where Zorzi proposed to avoid duplicate forwarding in the previously proposed beaconless routing schemes by applying the ready to send/clear to send (RTS/CTS) MAC scheme. The current node sends an RTS signal instead of a message, and waits for a node to respond with a CTS signal. Neighbors closer to the destination will respond sooner, based on an appropriately set timeout for the response. If several responses are received, the node selects one that appears to be best for forwarding, and then sends the packet to that neighbor directly. This procedure can be adapted to the various described scenarios by modifying the criteria for selecting the best forwarding neighbor and appropriate timeout. The timeout can be based on corresponding formulas already described for selecting the best forwarding neighbor. For physical layer considerations, the messages may need to be sent a few times in both directions in order to select the best neighbor.

Physical-layer-based recovery schemes and beaconless routing may be combined into a single scheme to produce beaconless routing with guaranteed delivery (of course, this claim is subject to some obvious conditions since the performance is always probabilistic). A simple solution is that the current node C solicits responses from all neighbors and selects the proper one after gaining neighbor knowledge. We are designing a more optimal solution that is not reported here due to space constraints.

Application to Other Network Layer Problems

Broadcasting

We now apply the described concept to analyze some other network layer protocols and propose some basic solutions for them. One application is related to the concept of timeout being applied in several cases. For example, consider broadcasting, where a message is to be sent from one node to all other nodes in the network. If transmission radii of all nodes are fixed and the same, the cost of sending a broadcasting message is 1. The "progress" made by broadcasting is the number of neighbors covered by this transmission, but not covered by previous ones. That is, neighbors the current node knows (based on previously received packets) that have already received the packet are not counted. Let n be the number of such remaining neighbors. Each receiving node sets its timeout counter proportional to $1/n$. The node waits for the duration of a timeout counter before making its deci-



■ Figure 3. *Evaluating the candidate forwarding from* C *to* A_1 *and* A_2.

sion on whether or not to retransmit. The timeout is adjusted after each received packet. When the timeout counter expires, the node retransmits the message if and only if $n > 0$. This method is known as the *neighbor elimination* scheme [5, references therein].

In the minimum energy broadcasting problem, the transmission radii are adjustable, and the total sum of transmission energies applied at each node is to be minimized. If the power consumption model $r^\alpha + c$ is applied to transmission with radius r, and $c > 0$, it is not optimal to merely cover nearest neighbors while preserving connectivity. In [5] it is shown that if a node decides to retransmit, the optimal transmission radius is $(2c/(\alpha - 2))^{1/\alpha}$, increased if necessary to preserve connectivity. We derive the same conclusion here by a different proof, based on our general framework. The cost of transmitting a packet is proportional to $r^\alpha + c$, while the progress made is proportional to the area being covered (i.e., r^2). The ratio $(r^\alpha + c)/r^2$ is minimized for exactly the same target value, and the proof is simpler.

Sensor Area Coverage

A conceptually similar problem is sensor area coverage. Sensors should decide, in a localized manner, which of them should remain active and which should sleep so that the monitored area remains fully covered by active sensors. With the same sensing radii, the cost of each active sensor is 1, while the progress made by being active is proportional to the area A of region covered only by that sensor. This leads to the selection of timeout proportional to $1/A$, and activity at the end of timeout when $A > 0$. Timeout represents the time the sensor waits before making its own decision on whether or not to be active.

With variable sensing radii, the cost of a sensor remaining active is proportional to $r^\alpha + c$, where r is the selected sensing radius, while the progress made is proportional to the area A covered only by that sensor. A, however, depends on r (and active neighboring sensors). If there are no active neighboring sensors, the optimal radius is the one that minimizes $(r^\alpha + c)/r^2$ since A is then proportional to r^2. It again gives the same target radius as minimum energy broadcasting. If there are active sensors in the area, the calculation of A in terms of r is not straightforward; details of this sensor area coverage protocol will be provided in a future publication.

Multicasting

Consider now the multicasting problem, where a source node (sensor) wishes to send a packet to several destinations (sinks) with known positions. It is assumed that the number of such destinations is small, which is a reasonable assumption for a sensor reporting to several sinks. The closest protocol to the one described here is given in [8]. The solution in [8] considers total hop count as the metric to optimize, and distances from neighbors to destinations as part of the criteria to optimize. The final criterion has a parameter whose best value is to be separately determined. We describe here a solution that needs no parameter, by using different criteria made from the same ingredients. At the beginning, the source is responsible for sending the packet to all destinations. During the protocol, an intermediate node receives multicasting tasks that consist of forwarding the packet to a subset of original destinations (we say that the node is "responsible for" these destinations). Assume that node C, after receiving a multicasting task, is responsible for destinations $D_1, D_2, \ldots$, and it evaluates neighbors $A_1, A_2, \ldots$ for forwarding. The whole task could be sent to one neighbor only (e.g., if there is one closer to all destinations than C), or split among several neighbors, each with a subset of destinations to handle. Hop count is assumed to be proportional to distances.

Consider the case in Fig. 3 as an illustration of the general principle. The current total distance for multicasting is $T_1 = |CD_1| + |CD_2| + |CD_3| + |CD_4| + |CD_5|$. If C considers A_1 and A_2 forwarding nodes, covering D_1, D_2, D_3, and D_4, D_5, respectively, the new total distance is $T_2 = |A_1D_1| + |A_1D_2| + |A_1D_3| + |A_2D_4| + |A_2D_5|$, and the progress made is $T_1 - T_2$. The cost is the number of selected neighbors, which in the above example is two. Thus, the forwarding set $\{A_1, A_2\}$ is evaluated as $2/(T_1 - T_2)$. Among all candidate forwarding sets, the one with the optimal value of this expression is selected. Note that the number of expressions to evaluate grows with the number of neighbors or destinations. Details of this multicasting protocol and its relation with [8] are given in [9]. The problem can also be generalized to solve rate-based propagation, where different sinks should be served at different rates.

Conclusion

We describe a general framework for the design of localized position-based routing schemes. It is based on optimizing the ratio of a cost measure and a measure of progress. The same principle is also applied to broadcasting, sensor area coverage, and multicasting problems. We believe that the framework will be useful for solving a number of other network layer problems, or the same problems under different assumptions and optimality criteria.

One important advantage of the proposed framework is that it avoids using any parameter in the solution (other than those that are part of the problem statement). The simulations performed so far for some of the listed cases (e.g., [6] for power-aware routing, [7] for routing with a realistic physical layer) all confirm that the routing protocol remains highly competitive when parameters balance some important criteria, or perform much better when alternative protocols are threshold-based. Using thresholds as parameters in the design was shown to be inferior in all cases studied so far, because either some reasonable solutions are "thresholded," or solution becomes inefficient if thresholds are too "optimistic."

Acknowledgments

This research is partially supported by NSERC. The author greatly appreciates comments received by three reviewers and Chatschik Bisdikian that have significantly improved the clarity of the presentation.

References

[1] P. Bose *et al.*, "Routing with Guaranteed Delivery in Ad Hoc Wireless Networks," *ACM Wireless Networks*, vol. 7, no. 6, Nov. 2001, pp. 609–16.

[2] S. Giordano and I. Stojmenovic, "Position based Routing in Ad Hoc Networks, A Taxonomy," *Ad Hoc Wireless Net.*, X. Cheng, X. Huang, and D. Z. Du, Eds., Kluwer, 2003, pp. 103–36.

[3] C. Huang, F. Dai, and J. Wu, "On-Demand Location-Aided QoS Routing in Ad Hoc Networks," *Proc. Int'l. Conf. Parallel Proc.*, Montreal, Canada, Aug. 2004.

[4] T. He *et al.*, "A Spatiotemporal Communication Protocol for Wireless Sensor Networks," *IEEE Int'l. Conf. Distrib. Comp. Sys.*, May 2003; to appear, *IEEE Trans. Parallel and Distrib. Sys.*

[5] F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic, "Optimal Transmission Radius for Energy Efficient Broadcasting Protocols in Ad Hoc and Sensor Networks," *Proc. IEEE ICC*, 2004, to appear, *IEEE Trans. Parallel and Distrib. Sys.*

[6] J. Kuruvila, A. Nayak, and I. Stojmenovic, "Progress and Location Based Localized Power Aware Routing for Ad Hoc and Sensor Wireless Networks," *Proc. ADHOC-NOW*, LNCS 3158, July 2004, pp. 294–99; to appear, *Int'l. J. Distrib. Sensor Networks.*

[7] J. Kuruvila, A. Nayak, and I. Stojmenovic, "Hop Count Optimal Position Based Packet Routing Algorithms for Ad Hoc Wireless Networks with a Realistic Physical Layer," *IEEE Mob. Ad Hoc Sens. Sys.*, Oct. 2004; *IEEE JSAC*, vol. 23, no. 6, June 2005, pp. 1267–75.

[8] M. Mauve *et al.*, "Position-Based Multicast Routing for Mobile Ad Hoc Networks," TR-03-004, Dept. Comp. Sci., Univ. of Mannheim, Mar. 2003; ACM Mobihoc Poster, 2003.

[9] P. M. Ruiz and I. Stojmenovic, "Cost-efficient Multicast Routing in Ad Hoc and Sensor Networks," to appear, *Handbook on Approximation Algorithms and Metaheuristics*, T. Gonzalez, Ed., Chapman & Hall/CRC.

[10] I. Stojmenovic, A. Nayak, and J. Kuruvila, "Design Guidelines for Routing Protocols in Ad Hoc and Sensor Networks with a Realistic Physical Layer," *IEEE Commun. Mag.*, Ad Hoc and Sensor Networks Series, vol. 43, no. 3, Mar. 2005, pp. 101–06.

[11] I. Stojmenovic and X. Lin, "Power Aware Localized Routing In Ad Hoc Networks," *IEEE Trans. Parallel and Distrib. Sys.*, vol. 12, no. 10, Oct. 2001, pp. 1023–32.

[12] M. R. Souryal, B. R. Vojcic, and R. L. Pickholtz, "Information Efficiency of Multihop Packet Radio Networks with Channel-Adaptive Routing," *IEEE JSAC*, vol. 23, no. 1, 2005, pp. 40–50.

[13] M. Zorzi, "A New Contention-based MAC Protocol for Geographic Forwarding in Ad Hoc and Sensor Networks," *IEEE ICC*, Paris, France, 2004.

Biography

IVAN STOJMENOVIC (ivan@site.uottawa.ca) received a Ph.D. degree in mathematics. He has held positions in Serbia, Japan, the United States, Canada, France and Mexico. He has published over 200 different papers and edited three books on wireless, ad hoc, and sensor networks with Wiley and IEEE Press. He is currently an editor of several journals including *IEEE Transactions on Parallel and Distributed Systems.* He has recently guest edited special issues in several journals including *IEEE Computer* (February 2004), *IEEE Network* (July 2004), and *Wireless Communications and Mobile Computing* (Wiley).