Journal of Applied Computing Research, 4(1):35-48
January-June 2014
 - doi: 10.4013/jacr.2014.41.04

A hybrid sender and receiver-based routing protocol for Wireless Sensor Networks[1]

Gabriel Novy, Max Machado, Fátima Duarte-Figueiredo, Raquel Mini

Pontifícia Universidade Católica de Minas Gerais. Rua Walter Ianni, 255, São Gabriel, 31980-110, Belo Horizonte, MG, Brazil

gabriel.novy@ifmg.edu.br, maxm@pucminas.br, fatimafig@pucminas.br, raquelmini@pucminas.br

Abstract. In Wireless Sensor Networks (WSNs), sensor nodes detect environment events and send them to sink nodes, which are responsible for processing these events. Due to the reduced of the nodes, the biggest restriction in a WSN is related to power consumption. Sender-based and receiver-based communication protocols each have their own advantages and disadvantages in certain scenarios. Since a WSN can undergo alterations in time, a protocol that is able to adapt to environmental conditions can increase network lifetime. This paper presents a hybrid routing protocol that operates according to sender-based and receiver-based approaches. The protocol was implemented using the NS-2 simulator and compared to sender-based and receiver-based approaches operating on their own. The results showed that the hybrid protocol, compared to sender and receiver-based approaches, achieves delivery rates close to 100%, performing 2.9 times less transmissions for each packet delivered. These gains demonstrate the contribution of the proposed algorithm, which reduces the number of transmissions, allowing the WSN to have a longer survival time.

Keywords: hybrid sender and receiver protocol, Wireless Sensor Networks, routing protocol, simulations.

Introdution

WSNs are composed by sensor nodes and a few sink nodes. Sensor nodes carry out event detection in the environment and the sink nodes have the main functions of receiving, processing and transmitting data captured by the network to a base station. Other important function of the base station is to disseminate interest to the WSN. Communication between nodes happen through hops and is responsible for the most of a node's power consumption (Pottie and Kaiser, 2000). The sensor nodes in WSNs are smaller-sized devices, with reduced processing, memory, communication and power capabilities. Considering that the functioning of a WSN requires power, the network's lifetime is directly linked to how long nodes can function on their batteries. Also considering that the communication between nodes is the most power-hungry activity, reducing intra-node communication is fundamental to increase a network's lifetime.

The Medium Access Control (MAC) layer is responsible for controlling medium access, carrying out collision detection and correction and controlling a node's operational cycle and duty cycle. The operation cycle is the division of time into periodic intervals of node activity and node idleness. During the activity periods, the node remains with its radio on, in order to transmit and receive packets, and to listen to the network. During the moments in which the node remains idle, its radio is switched off to save power. The percentage of time during which a node remains active in an operation cycle is defined as the duty cycle (Ye *et al*., 2002).

The network layer protocols are developed with the goal of determining which neighbor should perform the forwarding of a packet. However, the transmission of a packet in WSNs runs into several problems when a path must be found between the source node, the one that detected the event, and the sink node. Problems like a node being unable to recog-

[1] This work has been partially supported by CNPq and Fapemig.

nize other nodes in the network, mobility of neighboring nodes, mobility of the sink node, faults in neighboring nodes, the use of duty cycles, among other situations, make routing in WSNs a complicated task. Still, network layer protocols should be able to deliver data in spite of all the potential problems that could afflict the network.

An important question when designing data communication protocols is how the network layer determines what node will be the next in the route. Two possible approaches to make this choice are the sender-based and the receiver-based approaches. A routing protocol is classified as sender-based if, when a node receives a packet, it selects a subset of its neighbors to forward the packet to. Next, the node sends the packet to the chosen neighbors. When a chosen neighbor receives the packet, it repeats the choice and forwarding process. On the other hand, a routing protocol is classified as receiver-based if, when a neighbor receives a packet, it decides whether or not it should be forwarded. In this case, the node sends the packet to all of its neighbors. Each of the neighbors then chooses whether to continue forwarding the packet or not.

A fundamental difference between the two packet-sending approaches is the knowledge of the neighboring nodes set. To select a neighbor, a node must be aware of its presence, and discovering neighbors comes with an associated cost, because discovery packets must be exchanged between the nodes. The receiver-based approach requires no such knowledge about neighbors. However, since nodes are unaware of their neighbors, many packet retransmissions may be performed, until a neighbor is able to forward the forwarding. Each approach is best suited for a different scenario. For example, when there are few packets to be transmitted, the receiver-based approach is ideal. The cost associated with neighbor discovery may not compensate for the reduction in the number of retransmissions present in the receiver-based approach. On the other hand, when there is a high packet traffic, the knowledge of neighbors becomes advantageous, because the number of retransmissions in the receiver-based approach represents a high cost to the network. Another example is the failure of communication between nodes. The sender-based approach chooses a neighbor to forward the packet. If there is a communication failure (which can occur for various reasons, such as interference, packet collision, movement or node death), the routing will be compromised. This problem is circumvented by the receiver-based approach in which the packet is sent to all neighbors. That way, if the node does not receive the package due to a communication failure, another node can perform routing.

Since the environment in which a WSN is present may experience changes, it would be best for the network to operate with a protocol capable of adapting itself to changes that may happen during a network's lifetime. The identification of the current scenario and the change in the protocol's operating mode to a mode that is able to work efficiently brings improvements and greater life expectancy to the network. There are hybrid protocols in the literature using a combination of the approaches with and without maintenance of the routing table. In both approaches, it is necessary for the node to know the neighbors. However, it is not always possible to know the neighbors. The moving of the nodes and duty cycles make the network very dynamic. The neighbors arise all the time and other ceases to exist. The maintenance cost of knowledge of the neighbors can cause a large overhead in network energy consumption. The goal of this work is designing a hybrid routing protocol capable of operating according to both sender and receiver-based approaches. In other words, scientific contribution consists of combining sender and receiver approaches for the routing problem. The proposed protocol evaluates, in real time, the state of the network and switches between the two approaches. For the elaboration of the protocol, the scenarios for which each approach (sender-based and receiver-based) has the best performance were identified. This was used as base to model the behavior of the hybrid protocol, helping the protocol when switching between the two strategies.

The rest of this work is structured as follows: the second section presents a discussion about the most relevant related work. The third section describes the proposed protocol. The fourth section contains the simulation results. Finally, the last section concludes this work and discusses future work prospects.

Related work

A large number of hybrid MAC layer protocols use the slot reservation and idle listening approaches to regulate medium access. STC-MAC (Tan *et al*., 2009) groups network nodes into regions. In each region, a node is selected to be the master node and mediate communication between nodes. Communication inside the region uses the slot reservation strategy, while communication between master nodes is done using the idle listening solution. Differently from STC-MAC, IH-MAC (Arifuzzaman *et al*., 2011) decides which approach should be used based on packet latency. The protocols proposed in Bithas *et al*. (2012) and Zheng *et al*. (2013) determine which the employed strategy based on network packet flow will be. The slot reservation strategy is used when there is an elevated packet flow. Otherwise, medium access is done through idle listening.

Hybrid network protocols usually adopt approaches with and without routing tables to make the neighbor selection decision. For example, the ZRP protocol (Haas and Pearlman, 2001) uses routing tables for regions close to the node and route requests when a packet must be delivered to a node outside of the region. As a proposal to improve upon ZRP, SHARP (Ramasubramanian *et al*., 2003) changes region dimensions according to network packet flow. The protocol proposed in Safdar *et al*. (2012) is another ZRP extension in which region dimensions are altered using the sink node's mobility speed as base. Multi (Figueiredo *et al*., 2007) is a hybrid network protocol that uses a flow prediction mechanism to alter how it works. With a reduced flow, the route request approach is used to send a packet. When flow is elevated, a routing table infrastructure is created to send the packets. The protocol proposed in Chen *et al*. (2010) makes use of node reservation to increase network lifetime. Active nodes use communication with routing tables. When it is necessary to use idle nodes, communication with them is carried out using the route request strategy. Another hybrid protocol is the QB-LEACH (Gnanambigai *et al*., 2014). It is a hierarchical protocol that uses LEACH protocol approach to build a hierarchical network together with the approach of the Q-DIR protocol for transmission. The sending node divides the environment in fourth quadrants and run a flooding in the quadrant in which the sender and the sink node are located. Another protocol, presented in Won and Stoleru (2015), seeks to reduce the sum of the lengths of paths generated in the multicast sending packet. Next to that, the protocol evaluates the network topology to find holes, and then find ways to circumvent this problem. Hybrid network protocols that employ communication with and without routing tables aim at reducing overhead caused by route updates. However, they all use the knowledge of their neighbor in their structures. But not always energy expenditure in the neighbor's discovery process is rewarded. For example, in scenarios with low flow of messages, the cost of finding may exceed the cost of sending packets. In scenarios with high movement of nodes or use of duty cycles, at any time neighbors cease to exist and new neighbors arise. The maintenance of neighbors' knowledge in scenarios with low flow of messages and alternance of neighbors can be fatal to the network. Therefore, in scenarios such as these, it is interesting to use an approach which does not require knowledge of the neighbors to accomplish the shipping package.

Hybrid sender and receiver-based protocol

In every routing protocol, a packet's next hop is determined in a specific way. In the proposed protocol, a set of neighboring nodes is chosen so that they may be the only candidates to forward a packet. The proposed protocol always chooses the same set for the two approaches to prevent packets to follow different routes when they are forwarded.

In the proposed protocol, independently of the chosen communication approach (sender-based or receiver-based), the definition of neighbors that can be candidates to perform packet forwarding is always the same. First of all, this is done because the routes followed by the packets must be similar. If the packet forwarding paths found by the two approaches turn out to be very different, path lengths can significantly influence the number of necessary transmissions. The second reason to use the same policy for both strategies when defining candidate nodes is that the nodes closest to the sink nodes will always be chosen. In this case, packets will tend to follow a straight line path between the source node and the sink node, and a straight line path is the shortest possible path.

To determine who the candidate neighbors will be, it is first necessary to draw a virtual line between the current node, which will send the packet, and the sink node. In this line, a virtual point located at a distance r from the node sending the packet is determined. After the virtual point is selected, every node that received the packet and is at a distant less

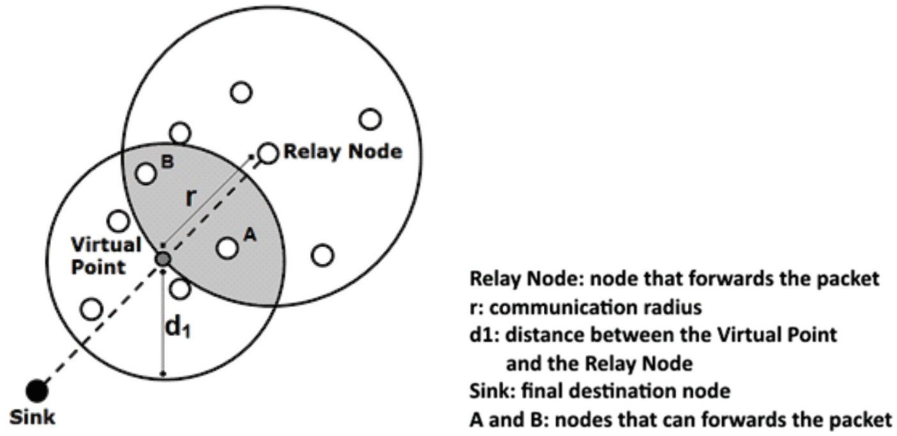


Figure 1. Example of candidate neighbors to forward a packet.

than or equal to d_1 from the virtual point will be candidates to forward the received packet. This policy is used both in the sender-based approach and in the receiver-based approach. However, in the sender-based approach, one of these candidate nodes will be selected to forward the packet. For the receiver-based approach, the packet will be sent to all of the neighbors. Figure 1 presents an example scenario. In this case, only nodes A and B, which are in the filled area, will be able to forward the packet. Still, it is important to analyze d_1. Is important that $d_1 \leq r$. If the distance d_1 is greater than r, (radius of the relay node), the next node chosen to forward the packet can be further from the sink (which is the final destination) than the actual node, that is forwarding the packet. An important observation is that the greater r and d_1 are, the larger the number of candidate neighbors.

The proposed hybrid protocol uses a policy for each communication approach. When the sender-based approach is used, a node must know who its neighbors are so that it can choose one of them to forward a packet. To discover the neighbors, a node sends update packets with information about its coordinates, the virtual point's coordinates, its duty cycle and what state of the operation cycle it is presently in. When receiving an update packet, neighbors will check, based on the virtual point, whether or not they are a candidate neighbor. Neighbors that are candidates to perform packet forwarding will also send update packets in response, containing their own information. By making these transmissions, neighbors are not only providing nodes with their information, but also requesting the same information from their neighbors. With that, when a node decides to discover its neighbors, in practice, it triggers a neighbor discovery process that propagates until it reaches the sink node.

After discovering its neighbors, when a node decides to forward a packet using the sender-based approach, it searches its neighbor list looking for a neighbor that is currently active. If all of the neighbors are sleeping, the chosen neighbor will be the one that is programmed to awake first. The packet is then configured with the address of the selected neighbor and sent to the MAC layer. Since the used communication approach is sender-based, the packet will be sent via unicasting. If the selected neighbor is awake, the packet is transmitted immediately. If not, a transmission is scheduled for when the neighbor wakens. When receiving a packet, the neighbor immediately sends back an ACK packet to confirm reception and avoid future retransmissions.

When the receiver-based approach is used, there is no discovery infrastructure. Therefore, the node cannot choose a neighbor to send the packet to. Instead, the node determines that the packet must be sent via broadcasting, and then the packet is sent to the MAC layer. Since it does not know who its neighbors are, the MAC layer will determine the maximum number of retransmission attempts to be done and will send the packet. However, in the middle of the retransmission process, the node can receive a confirmation packet from a neighbor that may continue the forwarding process. If the acknowledgement packet is received, retransmissions are cancelled. If the maximum retransmission attempts are completed and no confirmation packet is received, the packet is dropped.

In the hybrid protocol, the node must decide which communication strategy to use. The sender-based strategy has an elevated neighbor discovery cost. However, on average, a packet's retransmission cost is reduced. The receiver-based approach does not have an associated neighbor discovery cost. However, since it has no knowledge about the neighbors, a node cannot determine when neighbors are awake. It then has to perform several retransmissions until neighbors awake and receive the packet.

Taking only retransmission costs into account, the sender-based approach always has the upper hand compared to the receiver-based strategy, since it has a fixed and reduced number of transmissions that must be done to send a packet. However, the sender-based approach suffers from the mandatory neighbor discovery cost. Considering all of the previous information, whenever the neighbor infrastructure is available, the sender-based approach should be used. So, what a node must determine is the appropriate moment in which to conduct neighbor discovery and build the neighbor infrastructure.

The proposed hybrid protocol is able to produce an estimate of the cost of using each strategy to transmit all of the packets. To do that, the protocol assumes that nodes are aware of the packet sending rates and the period of time during which it will conduct data collection. So, the approximate cost of using a sender-based approach is given by the neighbor discovery cost added to the packet sending cost, as shown in Equation 1. The neighbor discovery cost is given by the inverse of the node's duty cycle (*cycle*) multiplied by the amount of candidate neighbors (n_{nei}). The number of update packets is inversely proportional to the node's duty cycle, because if the node is only awake during short periods of time, the probability of a packet being received is reduced. Due to that, the number of update packets sent must be bigger to increase the probability that they will be received. On the other hand, the cost of sending packets estimates the amount of transmissions that can be performed so every packet will be sent. The number of packets to be sent is determined by the ratio of data collection time (Δ_c) and the packet sending rate ($packet_{rate}$). The packages sending rate is a piece of information that can be provided by application. Each application has a requirement of packet sending rate. Constant represents the estimated cost to send a packet, which, for most of the cases, happens with only one transmission, plus the transmission of the ACK packet to confirm reception.

$$cost_{sender} = ([\frac{1}{cycle}] \times n_{nei}) + (2 \times \Delta_c \times packet_{rate}) \qquad \text{Equation 1}$$

The estimated cost of using the receiver-based approach is given only by the estimated packet transmission cost. This cost is shown in Equation 2, where *cycle* is the node's current duty cycle, (n_{nei}) is the number of candidate neighbors, (Δ_c) is the period of time in which packets are sent and $packet_{rate}$ is the rate with which packets are sent. The inverse of the product of duty cycle and the number of candidate neighbors ($[\frac{1}{cycle \times n_{nei}}]$) determines the average number of retransmissions that must be done until a candidate neighbor receives a packet. The larger the number of candidate neighbors, the smaller the number of necessary retransmissions. The retransmission amount is then added to constant , representing the ACK packet that will be sent by the neighbor selected to forward the packet. The ratio of the data collection period to the packet sending rate ($\Delta_c \times packet_{rate}$) determines how many packets will be sent.

$$cost_{receiver} = ([\frac{1}{cycle \times n_{nei}}] + 1) \times (\Delta_c \times packet_{rate}) \qquad \text{Equation 2}$$

The packet transmission process in the receiver-based approach is similar to the neighbor discovery process in the sender-based approach. Since there is no knowledge about neighbors, the packet must be retransmitted several times. However, in the neighbor discovery process, packets are retransmitted as many times as it is stipulated. In the receiver-based approach, initially, the stipulated retransmission amount is the same, that is, the inverse of the duty cycle. But, during the retransmission process, the first candidate neighbor to receive the packet will respond with an ACK packet, and will cause the transmitting node to cancel its retransmission process, thus avoiding unnecessary transmissions. The amount of neighbors directly affects the number of retransmissions that must be performed until the first candidate neighbor receives a packet. The greater the number of neighbors, the fewer the number of retransmissions that will have to be performed, because there will be a higher probability that a neighbor is awake during transmissions.

Once there is knowledge about the neighbors, it is always better to use the sender-based approach, due to the constant and reduced number of retransmissions to deliver a packet. Therefore, the important decision point of the hybrid protocol is to decide when to perform neighbor discovery. The protocol works according to Figure 2.

When sending a packet, nodes check whether they know their neighbors. In case they do, the sender-based approach is chosen. During the sending process using the sender-based approach, communication flaws may be detected. In case an error occurs, the communication strategy is changed to the receiver-based approach. However, when sending a packet, if the node does not know its neighbors, the cost of using each approach is calculated. If the cost to use the sender-based approach turns out to be smaller than the receiver-based cost, the neighbor discovery process is initiated and the packet is sent using sender-based approach. If the cost is not advantageous, the receiver-based approach is used instead.

Simulations results

This section evaluates the simulations results. The subsection "Simulation Scenarios" presents the configurations of the simulations scenarios. The next subsections evaluate the total transmissions, the delivery rate, the transmission per delivery packet and, finally, the last subsection evaluates the latency.

Simulation scenarios

This work uses simulation to implement and evaluate the proposed protocol. Simulations were used due to the high financial cost of implementing a WSN with hundreds of nodes. The goal of the simulations is to evaluate the hybrid protocol's performance compared to pure sender-based and receiver-based solutions. Network Simulator 2 v. 2.32 (NS2, 2014) was our simulator of choice for the tests. This simulator was chosen because it is widely used in academia in simulations of various types of networks.

Each simulation scenario was composed by 500 nodes randomly spread out in an area of 1,000 × 1,000 m^2. However, although nodes are distributed in a random fashion, the network is guaranteed to be connected, that is, there will always be a path between any two nodes in the network. The communication radius for every node is 100 meters. This value was chosen to ensure that it is a connected scenario, that is, there is always a path between any two network nodes. For every scenario, the number of nodes sending data is 10. All senders are located 900 *m* away from the sink node and are equidistant between them. The sink node is always located in the P(0,0) point of the scenario. The sender nodes are all at the same distance from the sink node because the distance interferes significantly in the evaluated parameters. We set the distance so it will not interfere the results.

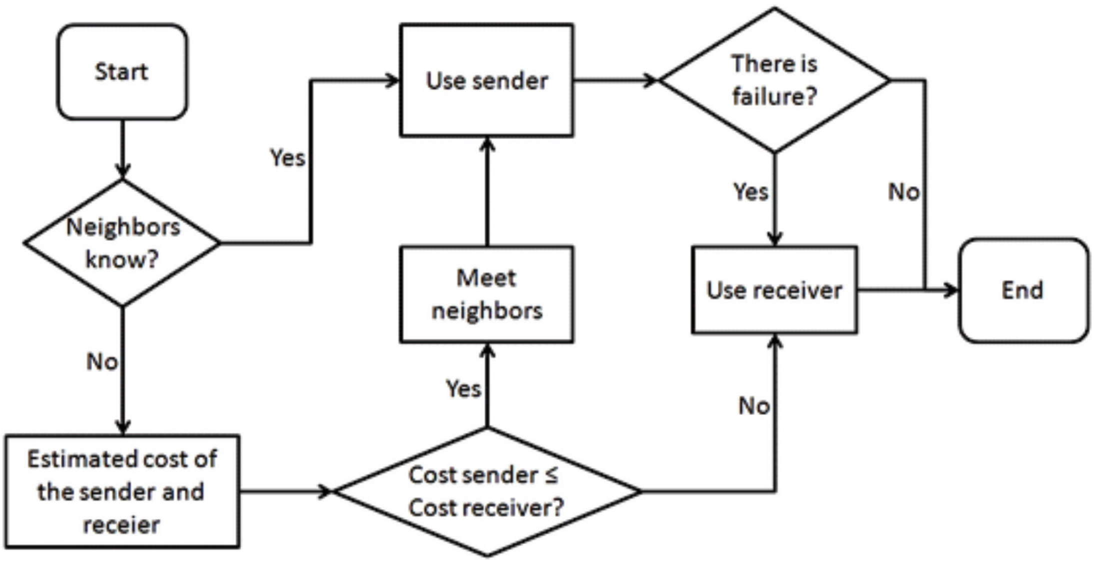


Figure 2. Hybrid protocol decision algorithm flowchart.

The previous section showed that sender-based and receiver-based approaches calculate a region close to a virtual point to determine who the candidate neighbors are and send a packet. This virtual point is located at a distance *r* from the node and is between the sender node and the sink node. From this virtual point, only nodes that are at a distance less than or equal to d_1 will be considered candidates to forward the packet. Empirical testing showed that, for the simulated scenarios, the ideal value for d_1 is 80 *m*. This value is ideal because it allows for a high packet delivery rate coupled with reduced transmission costs.

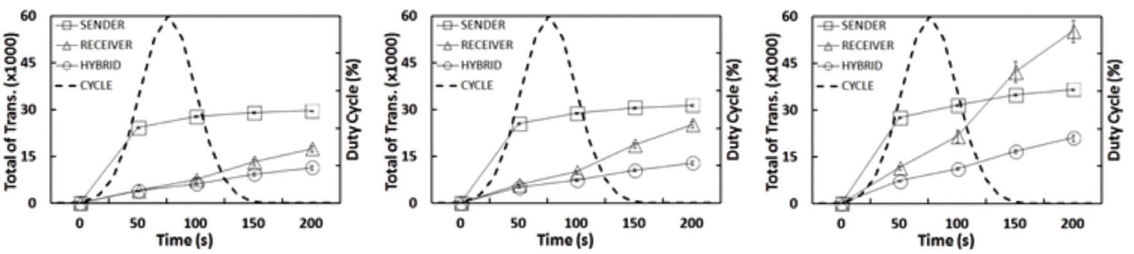
Simulations were split into two scenario sets: scenarios with no communication faults and scenarios with communication faults. For both of them, nodes' duty cycles vary with time, following a normal distribution with an average of 80 seconds and standard deviation of 20. For each set, simulations were conducted with different packet flow rates. The used rates were 0.67 *pkt/sec*, 1 *pkt/sec* e 2 *pkt/sec*. The second scenario set evaluates the impact of communication faults that are generated when nodes die out. The scenarios are the same as the first sets, but with communication faults. 10% of the nodes die during the simulations, following a normal distribution with an average of 80 seconds and standard deviation of 20. Simulation time was 200 seconds for all of the simulations. The presented charts are a result of an average between 30 simulations with a confidence interval of 95%.

The evaluation parameters used for comparisons between the protocols were the total number of packets sent, packet delivery rate, number of transmissions per delivered packet and average packet delivery latency. The total number of packets sent is directly linked to power consumption. The more packets are sent, the more energy is consumed. Since energy is the most important resource for a WSN, the network protocol must be capable of operating using the smallest possible number of packet transmissions. Packet delivery rates are directly related to a protocol's capability to find routes and overcome possible route problems. The number of transmissions per delivered packet shows the protocol's efficiency with respect to its utilization cost. A protocol with a reduced number of transmissions per delivered packet is an energy-efficient protocol. Finally, packet delivery latency is the total transmission time for a packet, from the source node to the sink node. Some applications are very latency-sensitive. For example, a WSN used to monitor a building to detect fires must be able to quickly report the detection of such event. Therefore, it is important for the network protocol to attain low latencies when delivering packets.

Total transmissions

This section presents the evaluation of the total transmissions performed by each approach by the end of the simulations. The proposed hybrid protocol seeks to reduce this number of transmissions, switching between sender or receiver-based approaches when most adequate.

Figure 3 presents the total transmissions over time in scenarios where no communication flaws occur. Each graph presents a scenario with a specific packet flow. In every scenario, the sender-based protocol has high transmission rates during the initial simulation moments. These transmissions originate in the initial neighbor discovery stage. In the beginning of the simulation, sender nodes send update packets requesting neighbors to send them information about their operation cycles and duty cycles. All neighbors receive these



0.67 *pkt/sec* **(a)** 1 *pkt/sec* **(b)** 2 *pkt/sec* **(c)**

Figure 3. Total transmissions for different flows without node death.

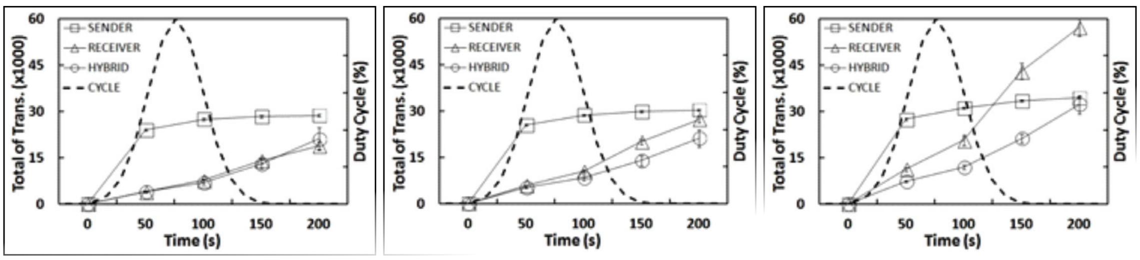
requests. However, only candidate neighbors will respond to them. Candidate neighbors then repeat the process so that they can gather information about their own neighbors. After the initial discovery stage, transmission numbers for the sender-based protocol rise less steeply, due to the reduced number of transmissions required to deliver a packet.

For the receiver-based protocol, transmission numbers rise more acutely and constantly. This is due to the fact that receivers perform more retransmissions than senders to be able to deliver a packet. The curve's inclination is different for the 50 – 100-second and the 100 – 150-second intervals, because during these stages the duty cycle rises to 100% and gets reduced again to 1%. When duty cycles are reduced, the number of retransmissions necessary to deliver a packet is elevated. During the 50 – 100 interval, duty cycles are increasing, therefore, the number of transmissions tends to be smaller. For the 100 – 150 interval, duty cycles decrease again, sharply increasing the number of necessary retransmissions and, consequently, augmenting the curve's inclination.

Although receiver-based approaches retransmit more than sender-based ones on average, sender-based solutions do not always perform fewer transmissions than receiver-based ones, due to the initial neighbor discovery stage. In scenarios with elevated message flows, sender-based strategies are more appropriate, because the cost to discover neighbors is mitigated by the savings that stem from the reduced packet retransmissions. This situation can be observed in Figure 3c. When packet flows are lower, however, the discovery costs, although constant, no longer compensate for the reduction in the number of retransmissions due to the smaller number of packets that need to be sent. For such scenarios, the receiver-based approach, in spite of the elevated retransmission numbers, has an inferior total transmission number, which makes it the most adequate approach, as can be seen in Figures 3b and 3a.

In all of the scenarios, the hybrid protocol performs fewer transmissions than both the sender-based and the receiver-based approaches. Such a behavior can be explained by the ability of the protocol to determine whether neighbor discovery is worth it, based on message flow and duty cycles. Nodes with a reduced message flow will opt to use the receiver-based approach, while nodes with a high packet flow will prefer the sender-based strategy. It is important to emphasize that, for scenarios where sender nodes send packets at reduced rates, there can be nodes that experience high flow levels. A node that belongs to many paths will receive several different flows. The sum of these flows will make that node experience elevated flow rates, and, therefore, opt to use the sender-based strategy. Another factor that leads the hybrid protocol to have smaller transmission numbers is the fact that it can conduct neighbor discovery at any time. The sender-based protocol must have information about neighbors before it can start sending packets. It triggers the discovery process during low duty cycles, which causes discovery packets to need to be retransmitted several times. The hybrid protocol can choose to use the receiver-based strategy when duty cycles are low and initiate neighbor discovery during high duty cycles. That way, retransmissions for discovery packets can be significantly reduced.

Figure 4 shows total retransmissions over time in scenarios in which communication faults occur. The sender-based protocol has a very similar behavior, concerning transmission numbers, to its behavior in scenarios without node death. However, the total number of transmissions is slightly reduced. In the



0.67 *pkt/sec* **(a)** 1 *pkt/sec* **(b)** 2 *pkt/sec* **(c)**

Figure 4. Total transmissions for different flows with node death.

sender-based protocol, an awake or about-to-be-awake neighbor is selected to perform the forwarding. The packet is then sent to this selected neighbor. In case the node does not receive a reception confirmation, some retransmissions are carried out. If they, too, are unsuccessful, the packet is discarded. This way, the next nodes in the path fail to receive it and, therefore, do not retransmit it, causing total transmissions to decrease.

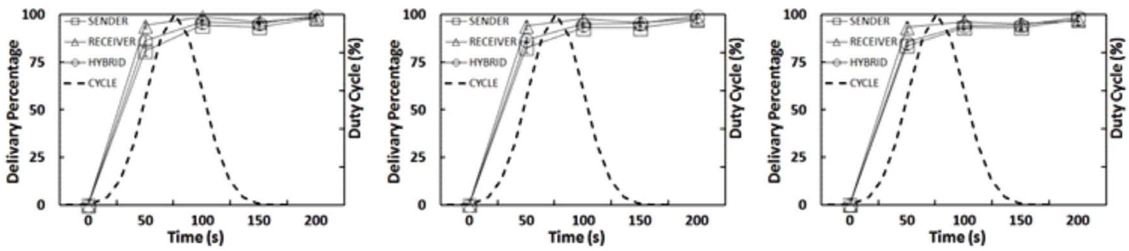
Contrary to the sender-based approaches, transmissions for the receiver-based approach tend to increase. In the receiver-based protocol, the first candidate node to wake up receives the packet and proceeds to forward it. When faults occur, this candidate may not receive the packet. The current node keeps retransmitting until it finds another candidate node to perform the forwarding. So, the number of retransmissions is high, causing the total number of transmissions for the receiver-based solution to increase.

In scenarios with communication flaws, the hybrid protocol presents the most variation in total transmission number compared to scenarios with no communication flaws. The increase in transmission number occurs due to the use, in the same node, of both communication approaches when a fault is detected. Initially, if the node opts to use a sender-based approach, it must have knowledge about its neighbors, generating retransmission costs from discovery packets. From the moment a node obtains information about its neighbors, it will always opt for the sender-based strategy. However, when communication faults take place, the receiver-based approach is used. In this case, the retransmission cost from the sender-based approach until a fault is detected and the retransmission cost from the receiver-based approach until some other neighbor receives and forwards the packet increases.

Delivery rate

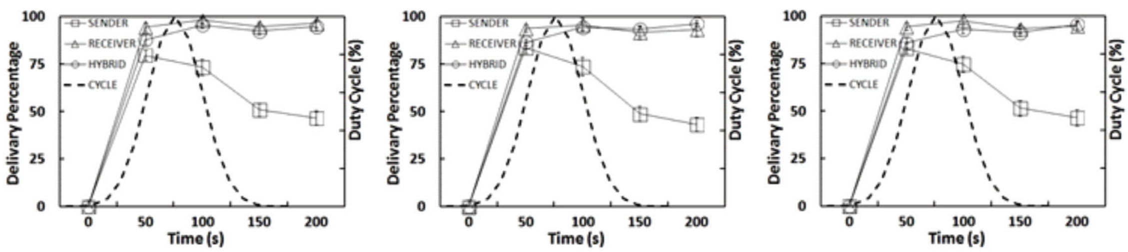
In this section, packet delivery rates for the compared approaches are evaluated. Packet delivery rate is directly linked to protocol efficiency. A protocol that has low delivery rates may not be reliable.

Figure 5 presents average packet delivery rates, over simulation time, in scenarios with no communication faults. Each graph presents a scenario with a specific packet flow. In spite of the high delivery rates for all three protocols (sender-based, receiver-based and hybrid), in scenarios with communication faults, presented in Figure 6, the sender-based



0.67 *pkt/sec* **(a)** 1 *pkt/sec* **(b)** 2 *pkt/sec* **(c)**

Figure 5. Delivery rates for different flows without node death.



0.67 *pkt/sec* **(a)** 1 *pkt/sec* **(b)** 2 *pkt/sec* **(c)**

Figure 6. Delivery rates for different flows with node death.

protocol suffers from a significant drop in delivery rates, reaching figures inferior to 50%. In the sender-based protocol, a node is chosen to forward the packet. If communication with the chosen node fails, the packet is discarded and never reaches its destination. The receiver-based protocol can circumvent this limitation, because if communication with a node is unsuccessful, another node will receive the packet and continue the forwarding process. The hybrid protocol has slightly inferior delivery rates, in scenarios with faults, than the receiver-based protocol. But, since the confidence intervals overlap, it is valid to say the delivery rates are the same. The hybrid protocol can bypass the flaws because, when the sender-based strategy is used and a communication error is detected, it switches to the receiver-based approach so the packet can be successfully sent.

Transmissions per delivered packet

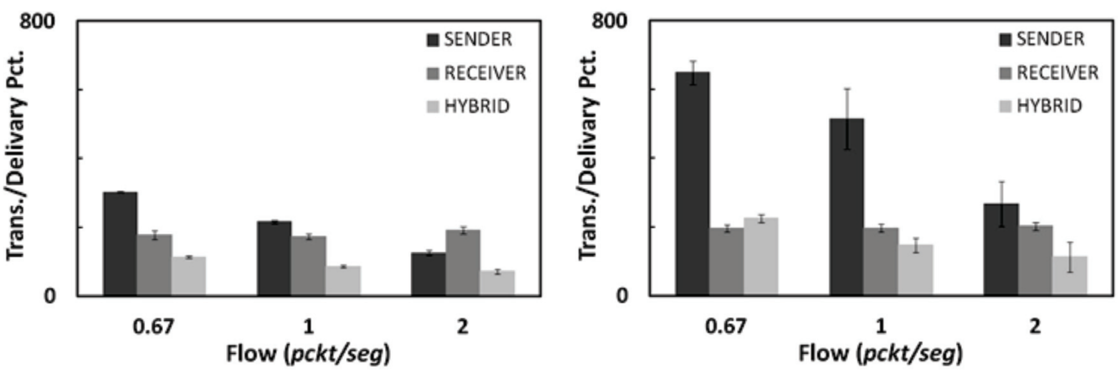
In this section, the number of transmissions performed by each protocol to deliver a packet is evaluated. Figure 7 displays the number of conducted transmissions for different flows and in scenarios with and without communication faults.

As packet flow decreases, the number of transmissions per delivered packets done by the sender-based protocol tends to get greater. This happens because the cost of neighbor discovery is constant, no matter how many packets must be sent. When the flow is elevated, the discovery cost is diluted in the large amount of packets that must be transmitted. For smaller packet flows, the neighbor discovery cost becomes proportionally more significant, causing the average number of transmissions per packet to increase, since there are fewer packets to be transmitted.

In scenarios with communication problems, the number of per-packet transmissions for the sender-based approach increases noticeably. This happens because, in scenarios prone to faults, the sender-based strategy may start not delivering packets. So, transmissions done for undelivered packets are also taken into account when calculating total transmission numbers. For this reason, the sender-based protocol performs more transmissions per delivered packet.

The receiver-based protocol has a similar cost for every flow value, in scenarios with or without communication faults. In this approach, transmissions will be carried out only for sending data packets, while sender-based approaches will use packet transmissions to conduct neighbor discovery. The number of retransmissions in a receiver-based solution is influenced by duty cycle and number of neighbors. So, for identical duty cycle and neighbor count, sending a packet belonging to a high flow or a low flow will result in the same number of transmission attempts.

The hybrid protocol has a smaller number of transmission per delivered packet, on average, than both sender and receiver-based solutions for almost all tested scenarios. This can be explained because the hybrid protocol can choose the most appropriate strategy to transmit packets at every moment. Each node has the autonomy to operate according to the approach it deems most adequate. Thus, the hybrid protocol can significantly reduce the number of transmissions required to deliver a packet.



(a) Scenarios with no communication faults

(b) Scenarios with communication faults

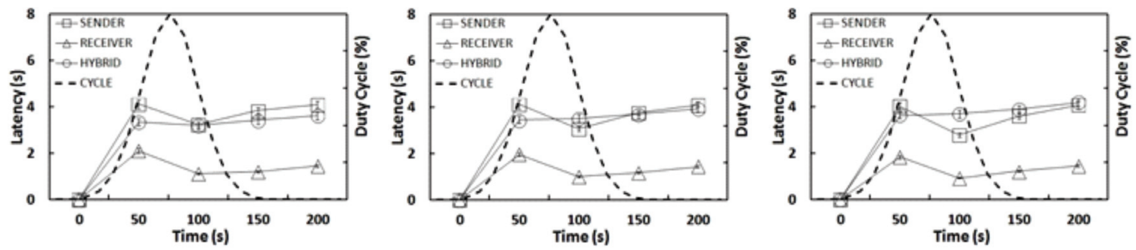
Figure 7. Number of transmissions per delivered packet for different packet flows.

Latency

This section evaluates accumulated packet latency for the studied approaches. We define, in this work, the latency as the time spent by the packet from the sending by sender node, passing by nodes present on the route, to be received by the sink node. Latency is strongly influenced by node duty cycle because, in scenarios with shorter duty cycles, neighbor nodes are awake during shorter periods of time. Latency is, therefore, elevated, because more time is wasted waiting for neighbors to become active so packets can be sent. When duty cycles are longer, latency tends to be shorter, because nodes remain awake longer and waiting times to forward packets are shorter.

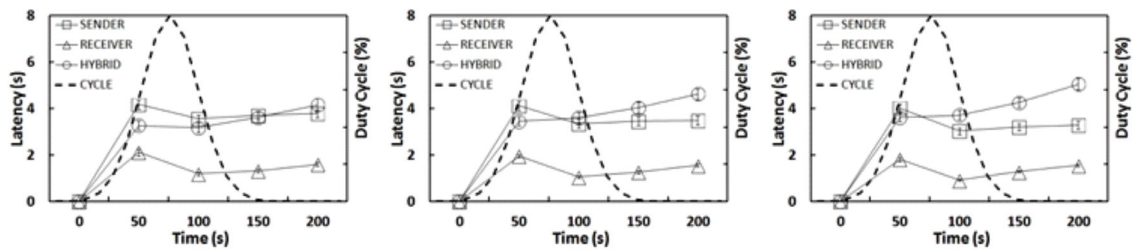
Figure 8 shows the average of accumulated packet latency over simulation time in scenarios with no communication faults. The sender-based protocol always selects an awake neighbor or the one programmed to awake the soonest. Meanwhile, the receiver-based protocol carries out many retransmissions. During these retransmissions, the awake neighbor or the soonest-to-be-awake neighbor forwards the packet. In the receiver-based protocol, while a node is performing a retransmission, it is still apt to receive new incoming packets. It is during these moments that the receiver-based protocol is able to reduce its latency. In the sender-based protocol, packets will only be transmitted when the elected neighbor is in its active state. In case some other neighbor awakens sooner to perform a transmission, it will not receive the data. In the receiver-based protocol, the first neighbor to be awake, be it because its duty cycle determines so or because it awoke to do a retransmission, can receive a new packet and carry on with the forwarding. The hybrid protocol has an average of accumulated latency similar to the sender-based approach, and, consequently, packet latency tends to be elevated. It is also important to note that as data flow decreases, fewer nodes use the sender-based strategy, causing average latency to decrease, as shown in Figure 8a.

Figure 9 shows that, in scenarios with communication faults, latency for the hybrid protocol is significantly increased. In the hybrid protocol, if there is knowledge about the neighbors, the sender-based approach is used. This way, there is a stand by time until the chosen neighbor wakes to do the transmission. However, if a transmission failure occurs and the transmission strategy is switched to receiver-based, there are also the accumulated retransmission latencies until a neighbor is



0.67 *pkt/sec* **(a)** | 1 *pkt/sec* **(b)** | 2 *pkt/sec* **(c)**

Figure 8. Latency for different flows without node death.



0.67 *pkt/sec* **(a)** | 1 *pkt/sec* **(b)** | 2 *pkt/sec* **(c)**

Figure 9. Latency for different flows with node death.

awake. So, in situations where faults occur, a packet transferred using the hybrid protocol accumulates latencies from both sender-based and receiver-based approaches, increasing average accumulated packet latency.

In this section, we present the simulation results analysis of the protocol developed. We also compared the performance of the hybrid protocol with sender and receiver-based approaches. It was concluded that the total number of transmissions for the hybrid protocol remains, in almost all scenarios, below of the sender and receiver-based approaches. The reduction in the number of transmissions is proportional to power consumption. Thus, we conclude that the hybrid approach can increase the network lifetime because it can save energy by reducing the number of transmissions.

As for the delivery rate, the hybrid approach maintains a high delivery percentage in all scenarios, being higher than the sender based approach, which has performance loss in scenarios with communication failures, and equal to the receiver based approach.

In the parameter number of transmissions by delivered packages is that can be seen the real gain of the hybrid protocol. In most scenarios, the protocol tends to perform a small number of transmissions to be able to deliver a packet. This is because the hybrid protocol can determine which approach is the best to be used at each moment.

Finally, in latency parameter, the hybrid protocol tends to have a high latency. This is because of the neighbors' discovery process and also because, in scenarios with failures, it uses both approaches to deliver the packages.

Conclusions and future work

The WSNs have as main challenge the reduction of energy consumption. The routing of packets has a strong impact on the network lifetime, since packet transmission is the process that consumes most energy in a node. Between the two routing approaches, the sender-based features a fewer retransmissions, but has a high cost to the discovery of neighbors. It is more interesting in scenarios with high packet traffic, reduced duty cycle and without fails of communication. On the other hand, the receiver-based approach performs a high number of retransmissions, but lacks the cost of neighbor discovery. It has better performance in scenarios with low packet traffic, high duty cycle and when there are communication gaps between nodes. Thus, each approach has better performance in specific situations. By operating in dynamic scenarios, it is important that a WSN uses a protocol able to adapt to environmental conditions.

This paper presented a proposition for a hybrid sender and receiver-based network protocol for WSNs. The protocol uses both approaches with the goal of reaching high packet delivery rates with the lowest possible retransmission number, which leads to a decrease in energy consumption.

Simulation results show that, in scenarios with variable duty cycles and absence of communication faults, the hybrid protocol performs until 2,9 times fewer transmissions in total compared to pure sender-based and pure receiver-based approaches used individually. In scenarios with variable duty cycles and communication faults, when message flow is elevated, the hybrid protocol does fewer transmissions than the sender-based strategy. When flow decreases, both tend to perform the same number of transmissions.

Concerning latency, the hybrid protocol performs worse than the receiver-based strategy and better than the sender-based approach in scenarios with variable duty cycles and no communication faults, because it uses both approaches when sending a packet. In scenarios with faults, the hybrid protocol experiences a slight increase in latency, because it accumulates sender-based latency up until a fault is detected and receiver-based latency before finally delivering the packet.

Regarding delivery rates, the hybrid protocol achieves percentages of delivery close to 100% in all scenarios. The receiver-based approach also achieves a high percentage of delivery, but at the cost of a large number of transmissions. In turn, the sender-based approach reduces its delivery rate by half when there are communication network failures.

The biggest gain in using the hybrid protocol is with respect to total packet transmissions, which is directly linked to energy consumption. The hybrid protocol can attain higher delivery rates performing fewer transmissions than the sender-based and receiver-based approaches. Thus, it is shown to be an appropriate option for dynamic scenarios in WSNs.

The major contributions of this work are the algorithm that determines which approach to use and the equations that estimate the cost of using each approach. However, we can say that both are only one element. The algorithm is based on the idea that it is necessary to know the future behavior of the network. After several studies we concluded that it is not possible to determine with certainty which approach to use without having any knowledge of the network future behavior. Even in a situation with a high duty cycle, when the cost of discovering its neighbors is reduced, it may not be appropriate in this case. If the number of packets to be transmitted is low, for example 2 or 3 packets, meeting neighbors will generate a large overhead on the network. And to determine which approach to use the algorithm is based on the results of the equations that estimate the transmission costs. These equations are based on parameters which directly affect the cost of transmission of both approaches and cost of neighbor discovery. But we understand that such estimates can be improved in order to have better results in other scenarios.

Simulations were used to implementation and evaluation of the proposed protocol. Because of the simplicity of the algorithm that determines which approach to use, we believe that its implementation in a real environment should not present major difficulties. However, it is important to perceive that the protocol assumes that some parameters are known by the node. So, it is necessary to investigate how to determine efficiently such parameters.

As future work, the d_1 value, used to determine which nodes are candidate to forward a packet, can be more thoroughly studied. For example, the value of this parameter can vary as a function of duty cycle. Another future work suggestion is the evaluation and adaptation of the hybrid protocol to scenarios that consider node mobility. A node could evaluate some parameters, like packet flow, speed and movement direction for itself and neighbors to determine whether it is adequate to use a sender-based or receiver-based transmission strategy. We believe that such evaluation should be made to determine the effectiveness and possible improvements in the protocol in different scenarios from the ones evaluated in this work.

References

ARIFUZZAMAN, M.; ALAM, M.; MATSUMOTO, M. 2011. A hybrid MAC with intelligent sleep scheduling for wireless sensor networks. *In*: Kaleidoscope 2011: The Fully Networked Human? - Innovations for Future Networks and Services (K-2011), Cape Town, 2011. *Proceedings*… ITU, p. 1-7.

BITHAS, P.; LIOUMPAS, A.; ALEXIOU, A. 2012. A hybrid contention/reservation medium access protocol for Wireless Sensor Networks. *In*: Globecom Workshops (GC WKSHPS), LOCAL, 2012. *Proceedings*… IEEE, p. 1669-1674. http://dx.doi.org/10.1109/GLOCOMW.2012.6477836

CHEN, J.-L.; MA, Y.-W.; HSU, Y.-M.; HUANG, Y.-M. 2010. Adaptive routing protocol for reliable wireless sensor networking. *In*: The 12th International Conference on Advanced Communication Technology (ICACT), Phoenix Park, 2010. *Proceedings*… **1**:358–363.

FIGUEIREDO, C.; NAKAMURA, E.; LOUREIRO, A.; RUIZ, L. 2007. An Event-Detection Estimation Model for Hybrid Adaptive Routing in WSNs. *In*: IEEE International Conference on Communications, Glasgow, 2007. *Proceedings*… IEEE, p. 3887-3894.

http://dx.doi.org/10.1109/ICC.2007.641

GNANAMBIGAI, J.; RENGARAJAN, N.; NAVALADI, N. 2014. A clustering based hybrid routing protocol for enhancing network lifetime of Wireless Sensor Network. *In*: 2nd International Conference on Devices, Circuits and Systems (ICDCS), Combiatore, 2014. *Proceedings*… IEEE, p. 1-4.

http://dx.doi.org/10.1109/ICDCSyst.2014.6926119

HAAS, Z.J.; PEARLMAN, M.R. 2001. The performance of query control schemes for the zone routing protocol. *IEEE/ACM Transactions on Networking*, **9**(4):427–438.

http://dx.doi.org/10.1109/90.944341

NS2. 2014. Network Simulator 2. Available at: http://isi.edu/nsnam/ns/. Accessed on: 15/10/2014.

POTTIE, G.J.; KAISER, W.J. 2000. Embedding the Internet: wireless integrated network sensors. *Communications of the ACM*, **43**(5):51–51.

http://dx.doi.org/10.1145/332833.332838

RAMASUBRAMANIAN, V.; HAAS, Z.J.; SIRER, E.G. 2003. SHARP: a hybrid adaptive routing protocol for mobile ad hoc networks. *In*: 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing, Mobihoc, New York, 2003. *Proceedings*… ACM, p. 303-314. http://dx.doi.org/10.1145/778415.778450

SAFDAR, V.; BASHIR, F.; HAMID, Z.; AFZAL, H.; PYUN, J.Y. 2012. A hybrid routing protocol for wireless sensor networks with mobile sinks. *In*: 7th International Symposium on Wireless and Pervasive Computing (ISWPC), Dalian, 2012. *Proceedings*… IEEE, p. 1-5.

http://dx.doi.org/10.1109/ISWPC.2012.6263665

TAN, M.; TANG, L.; CHANG, H.; TIAN, H. 2009. A Hybrid MAC Protocol for Wireless Sensor Network. *In*: 5th International Conference on Wireless Communications, Networking and Mobile Computing, Beijing, 2009. *Proceedings*… IEEE, p. 1-5.

http://dx.doi.org/10.1109/WICOM.2009.5302663

WON, M.; STOLERU, R. 2015. A Hybrid Multicast Routing for Large Scale Sensor Networks with Holes. *IEEE Transactions on Computers*, **PP**(99):1. http://dx.doi.org/10.1109/TC.2015.2409863

YE, W.; HEIDEMANN, J.; ESTRIN, D. 2002. An energy-efficient MAC protocol for wireless sensor networks. *In*: Twenty-First Annual Joint Conference of The IEEE Computer and Communications Societies, Infocom, 2002. *Proceedings…* IEEE, vol. 3, p. 1567-1576.

http://dx.doi.org/10.1109/INFCOM.2002.1019408

ZHENG, Q.; FEI, X.; TANG, T.; WANG, Y.; WANG, P.; LIU, W.; YANG, H. 2013. A hybrid, dynamic traffic-adaptive MAC protocol for Wireless Sensor Networks. *In*: 15th International Conference on Advanced Communication Technology (ICACT), Phoenix Park PyeongChang, Korea (South), 2013. *Proceedings…* **1:**574-578.

Submitted on October 17, 2014
Accepted on April 7, 2015