

## Research Article

# SWR: Smartness Provided by Simple, Efficient, and Self-Adaptive Algorithm

**Mujdat Soy Turk and Deniz Turgay Altılar**

*Computer Engineering Department, Istanbul Technical University, Maslak, 34469 Istanbul, Turkey*

Correspondence should be addressed to Mujdat Soy Turk, msoy turk@itu.edu.tr

Received 15 July 2011; Accepted 20 October 2011

Academic Editor: Yuhang Yang

Copyright © 2012 M. Soy Turk and D. T. Altılar. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Resource limitations of sensor nodes in wireless sensor networks (WSN) bound the performance on its implementations. Main concern becomes utilizing these limited resources (CPU, memory, bandwidth, battery) as efficient as possible. Their efficiency is mostly affected by the applied routing algorithm, which carries gathered data to inclined/intended destinations. In this paper, a novel routing algorithm, stateless weight routing (SWR), is proposed. The SWR differs from other protocols in many ways. Major feature of the SWR is its simplicity. It is a completely stateless protocol without requiring any network or neighborhood information for routing. This feature decreases packet transmissions and energy consumption dramatically. For reliability, data flows to the sink node over multiple paths. Moreover, nodes have the ability of recovering from voids. Nodes process each packet independently and apply an adaptive approach according to the current conditions. These mechanisms are part of the applied simple routing algorithm, the SWR. The resultant of these features assures flexibility and smartness at nodes and in the network. Therefore, topological changes have a little effect on data packet transmissions. Performance evaluation of the proposed approach shows that the SWR is scalable for WSNs whose topology change instantly and frequently as well as remain stationary.

## 1. Introduction

Properties of the wireless sensor networks (WSNs), such as energy constraints, limited availability of resources, communicating over unreliable wireless links, and ad hoc property, preclude scalable solutions. Since scalability is mainly affected by the implied routing protocol, many studies and efforts have been put to solve this well-known problem. Among the previously proposed approaches in the literature, geographical routing protocols that make use of location information for routing are more scalable than the others [1–6]. There are essential drawbacks of geographical approaches which can be listed as follows: requiring complex calculations at nodes, depending on the MAC-layer (e.g., IEEE 802.11), being susceptible to local-minima, possessing inability to go around voids, and reliability. Moreover, frequently changing topologies harden the problem mentioned above [6–8].

As the sensor node technologies take place in the market, demands on diverse application areas of WSNs draw on some requirements such as being able to work on different topolo-

gies, staying alive as long as possible without disruptions. Unfortunately, solutions aimed to cover these objectives introduce complex calculations at nodes and require information exchange between nodes which may even spread all over the network. Moreover, some approaches introduce solutions which require use of system-wide parameters that could not be defined clearly and/or deterministically, thus always require man-in-the-loop.

In this paper, a novel stateless data flow approach and routing algorithm, namely, stateless weight routing (SWR) for ad hoc and sensor networks is proposed. It is a stateless and reactive routing protocol utilizing the geographic location information for routing. Nodes do not have to be aware of either local or global topology information. Routing is achieved without keeping tables. Routes are constructed on-demand and spontaneously without requiring any neighborhood information. This approach avoids the beacon messaging and advertising. The delay and complex calculations encountered in geographical routing protocols and the communication overhead encountered in beacon-based

approaches are eliminated in the proposed approach. Moreover, the proposed approach is flexible since it is able to adapt itself according to the network dynamics.

In the SWR, a new metric called weight is introduced that is derived from nodes' own positions to be used in routing process. Weight value simplifies the routing algorithm and makes data flow spontaneously and simultaneously over multiple paths. Multiple paths provide reliability, eliminate the void problem substantially, and provide more robust routes including the shortest path. To keep energy-limited nodes out of the route, the decision to transmit includes QoS parameters such as power-left at the node.

The proposed algorithm:

- (i) provides scalability since neither routing tables nor beaconing is used,
- (ii) simplifies the routing process by designing an appropriate algorithm, which utilizes a weight metric,
- (iii) decreases calculations, delay, and resource requirements (such as processor and memory) at nodes since a weight metric is used instead of time consuming operations on routing tables,
- (iv) decreases energy consumption by:
  - (a) not beaconing,
  - (b) considering the remaining energy levels at nodes,
  - (c) limiting the number of relaying nodes,
- (v) provides reliability by exploiting multiple paths and recovering from voids,
- (vi) executes routing process completely in the network layer, independent of the MAC layer underneath.

The key contribution of the SWR is eliminating the communication overhead and energy consumption required in topology learning approaches. The SWR utilizes resources allowing simultaneous data flows over multiple paths rather than prior topology learning and path construction. The SWR is a self-healing algorithm that a failure in the network or links or nodes does not affect on-going data communications and data flows. With a smart approach, nodes make their own decision according to their own current conditions. Simulations prove that the SWR is scalable even for large-scale networks.

In the next section, related work is given. Design goals are given in Section 3. In Section 4, the proposed routing algorithm, SWR, is described. Analysis of the proposed algorithm is given Section 5. Performance evaluations are given in Section 6. In the final section, the paper is concluded.

## 2. State of the Art

Routing without tables can be achieved by using location information of the nodes retrieved from GPS or by applying a localization algorithm. In such protocols, which are called geographical routing protocols, nodes know their actual or relative positions with respect to a reference point and share

this information with their immediate neighbor nodes for routing process [1–6, 9–11]. Geographical routing protocols are more scalable with respect to conventional routing protocols due to only local topology information kept at nodes and no or less update overhead.

The taxonomy for position-based/geographical routing algorithms for ad hoc networks is given in [1, 2]. Surveys of the proposed protocols are given in [3–5] and [12]. Formerly proposed position-based/geographical routing protocols use greedy approaches by utilizing either distance or angle as metric [13–18]. In greedy approaches, there is a possibility that they may not find the route due to the constraint of using only local topology knowledge, even if there is a path to destination that can be found with global topology knowledge [19–22]. Besides that, beaconing-based greedy approaches consume excessive energy because of beaconing and introduce control traffic overhead. Furthermore, as the topology changes, providing proactively local topology knowledge reduces the performance and the scalability. Therefore, stateful protocols are not suitable for these types of networks [2]. However, stateless (table-free) protocols are not affected too much from the topological changes and network dynamics, but they use broadcasting to find routes as in flooding which wastes resources. They use MAC-layer integrated approaches to achieve this and introduce delay [10]. MAC-layer integrated approaches make them dependent to the MAC-layer used. In addition to these, some of the geographical routing protocols are prone to void problem. Routing algorithms should be able to cope with the void problem. Approaches and algorithms for void problem in geographical routing in sensor networks are well defined in [19–21].

Six well-known stateless routing protocols providing better performances among all others are described in this section: GPSR [23], SPEED [24], CBF [25], IGF [26], GDBF [11], and DDB [2]. GPSR [23] requires a priori local topology information. Nodes broadcast periodically the beacon messages independent of data packets to provide local topology information. Receiving neighbor nodes update their neighborhood tables accordingly. On a transmission need, best next node is selected by calculating the distances of neighbor nodes. Beaconing introduces communication overhead and consumes energy. Continuous table updating introduces processing overhead and buffers overflow due to periodic beaconing. While the GPSR tries to find the shortest path, it may experience the local minima problem. SPEED [24] is very similar to GPSR. However, it provides real-time communication and recovers from voids.

In contention-based forwarding scheme (CBF) [25], forwarding nodes select the next-hop through a distributed contention process using biased timers. All nodes which receive a packet check if they are closer to the destination than forwarding node and set their timers according to the progression toward the destination. Best suitable nodes respond in advance suppressing the other nodes. Forwarding node selects the best candidate node as next node from the responding nodes set. In this approach, next node selection phase introduces greater delay and energy consumption on route construction phase with respect to greedy approaches. Moreover, rebroadcast decision is based on RTF/CTF

(Request to Forward/Clear to Forward) packets and timers, which are completely processed in MAC-layer. Used energy models are not defined in [25], and it does not consider the energy efficiency.

Implicit geometric forwarding (IGF), which is introduced in [26], is very similar to CBF. As in CBF, it integrates the routing protocol with the MAC-layer, namely, the IEEE 802.11 protocol. It uses RTS and CTS packets with some modifications and additional functionalities. IGF defines a forwarding region which is a destination-directed sector. Each node in the forwarding region sets a response timer regarding the weighted sum of the distance to the destination, the remaining energy, and a random value and competes to acknowledge to the sender. First, acknowledging a node suppresses the others. However, the values of the set timers induce essential amount of delay due to existence of a number of nodes in the forwarding region. Consequently, a packet holder may have to wait for a long time before hearing an acknowledgement back. For the case of absence of a candidate forwarding node, IGF proposes two methods. In the first one, MAC layer informs the network layer to increase the range of transmissions. In the second one, although the use of backpressure method is proposed, no implementation details are given. As IGF induces delay in addition to delay encountered in collision resolution, it is completely dependent on MAC layer IEEE 802.11 protocol and is bounded by the use of modified RTS/CTS packets.

Another beaconless position-based routing protocol that guarantees the delivery of the packets, namely, guaranteed delivery beaconless forwarding (GDBF), is proposed in [11]. GDBF protocol selects appropriate next node by means of RTS/CTS packets. Forwarding a node broadcasts the RTS packet to its neighbors and the neighbor nodes compete with each other to forward the packet and set a timeout depending on their suitability. After timeout, nodes send CTS back to the forwarding node by using the suppression technique. Forwarding node decides one of the neighbor nodes as next node and forwards the message to it in a greedy manner. In case of a failure in this greedy mode such as reaching to local minima (no CTS response), guaranteed delivery is provided by the recovery mode. The drawbacks of the GDBF are as the same as the ones in the CBF. On the other hand, GDBF is a solely MAC-layer solution for the routing.

CBF, IGF, and GDBF are very similar to each other in terms of next node selection. A different approach which is called dynamic delayed broadcasting Protocol (DDB) is proposed in [2]. DDB allows nodes to make locally optimal rebroadcasting decisions by dynamic forwarding delay (DFD) and allows the nodes that have higher retransmission probability to rebroadcast first by suppressing the transmissions of other nodes. However, it cannot avoid multiple transmissions and introduces delay. On the other hand, during each receive operation; nodes have to recalculate/adjust their timers, which is computationally complex. Since packet scheduling is achieved in the MAC layer, on each receive operation an additional MAC-Network-MAC inter-layer communication is required to reschedule the packet transmission. A scheduled packet can even be dropped

following a number of calculations and scheduling operations, which are costly in terms of time and energy.

These beaconless and stateless algorithms introduce MAC-layer-involved solutions for routing. They rely on MAC layer and utilize IEEE 802.11 protocol functions for routing decisions. Note that timing and packet scheduling are the functions of MAC layer while decisions of broadcast, multicast, and unicast are the functions of network layer. In well-defined communication architecture, routing and node addressing should be independent from the MAC layer functions. Combining routing function with MAC layer introduces overhead and makes the routing protocol dependent to the MAC scheme used within the system.

These stateless algorithms integrate MAC layer to select forwarding node based on calculated timer values to schedule packets. This approach introduces a computational overhead in MAC/Network layer. Moreover, scheduling is triggered whenever a node receives a new packet. Consequently, timing and scheduling changes very frequently. Worse is that scheduling algorithm reduces the performance of the network layer by disturbing it for the sake of routing. Moreover, there are some unconsidered cases in these protocols, such as unforeseeable erroneous timer setting, deceive of the validity of applied timer value in collision resolution, and unpredictable terminations of nodes. In these protocols, timer is generally set according to node's geographical position and a number of node and system-wide-parameters. The idea is to find a "forwarding node" which is supposed to be best among all candidates considering the given set of parameters. Although the selected node may be considered as the optimal "forwarding node," contention based problems has to be rediscussed in detail to verify the optimality. When a node receives a packet, packet encounters a delay in buffers. This delay includes queuing delay and processing delay and varies according to nodes' current conditions including in-buffer situation, processing ability, out-buffer situation—current medium conditions which affect the propagation of a packet, and collision resolution delay in MAC layer. Therefore, a better node that is expected to calculate shorter time may process the packet later than others. In addition, when the packet is passed to the MAC layer, depending on the state of out-buffer of the node, packet may not be transmitted immediately and encounters delay along with possible delay in collision resolution. These all affect the timers calculated at nodes and make them inaccurate. Therefore, selected forwarding node may not be the expected "optimal" node. Added delay according to the timer in packet scheduling is an addition to the delay encountered in collision resolution in MAC layer. It increases the overall delay during transmissions. Moreover, the proposed stateless protocols' performance is sensitive to the node terminations and nodes' unpredictable come-ups and go-downs. Making data flow over a single path is prone to failures at any time, which is very common in frequently changing topologies. Some of these protocols are also prone to void problem. Routing algorithms should be able to cope with the void problem as well.

### 3. Design Goals

Five essential design goals are considered in this research, which are:

- (i) providing simple and efficient architecture that comprises of flexibility, rapid deployment, low configuration-setup-management, and high data reception rate and accuracy,
- (ii) providing reliability for a number of different severe circumstances and conditions such as low density deployment, existence of voids, node terminations, link failures, and frequent topology changes,
- (iii) producing an uncomplicated, robust and scalable algorithm that works with modest resources of disposable nodes,
- (iv) enabling nodes to adapt themselves to changing network conditions such as variable network density, void or congestion occurrences, and link or node failures,
- (v) providing a steady delivery speed where the end-to-end delay is proportional to the distance between the source and destination.

According to defined design goals given above, the proposed approach satisfies the following design objectives.

**3.1. Stateless Architecture.** Physical limitations are the main constraint of wireless sensor networks. To cope with this constraint, the SWR algorithm is designed to minimize the use of resources such as memory and processor. Neither routing tables nor any information on topology are not kept at nodes yielding memory requirement to be kept at minimum. The SWR does not require routing table, beaconing, or any information exchange that is unaffordable in other protocols. This approach also reduces the processing overhead at nodes. Complex calculations existing in stateless approaches are not required in the SWR. Thus, CPU requirements are kept at minimum as well. The SWR is designed as a very simple algorithm that can be executed at sensor nodes having the lowest possible processing capability.

**3.2. Reliability.** Routing algorithms that use a single path to the destination suffer from route breakages and packet losses. Using multiple-paths provides reliability but introduces communication overhead to build and maintain these paths up to date. In the SWR, data flow spontaneously and simultaneously over multiple-paths without excessive energy consumption and communication overhead. Path construction, maintenance and update are not required in the SWR. The algorithm is simple and inherits the benefits of flooding. Moreover, mechanisms to recover from voids and congestion are developed as defined below.

**3.3. Void Avoidance and Recovery.** Local-minima problem dictates that greedy algorithms may fail to find a path to the destination even when one or more paths actually exist. These algorithms may also experience void problem. Only few of those [19–21] propose solutions for local-minima

problem and to recover voids. Although the SWR does not suffer from local-minima problem, it utilizes a void recovery algorithm to pass around large void areas [27] if and only if such a problem is not overcome implicitly by the use of multiple paths.

**3.4. Robustness to the Congestion.** Reactive protocols and stateless protocols construct routes on demand. Dynamic nature of these protocols may cause fluctuations in network traffic, which may also introduce additional congestion. Moreover, coping with congestion introduces additional overhead and reduces performance. Routing algorithms in WSN generally use the backpressure mechanism to cope with congestion. In the SWR, the above-mentioned void recovery and avoidance approaches inherently solve the congestion problem.

**3.5. Scalability.** As the size of the network increases, the performance degrades dramatically in terms of throughput, reliability, delay, and energy-efficiency. The SWR algorithm presents a scalable solution for WSNs by avoiding table-keeping at nodes, avoiding information exchange between nodes, and utilizing locally implemented approaches to avoid/recover from problems such as void and congestion.

**3.6. Large-Scale Applicability and Multiple-Sink Usage.** Using multiple sinks has been proposed as a feasible approach to overcome scalability issues in large-scale WSNs in the literature [28, 29]. However, most of the protocols are based on previously proposed ones developed for single sink networks. Therefore, adapted such multisink protocols inherit the deficiencies of single sink network protocols such as scalability. However, the SWR algorithm can be used with either multiple sinks or a single sink without any adaptation or modification in the routing algorithm at all.

**3.7. Self-Adaptation.** Protocol design considerations for link problems, communication environment, node failures, and mobility require management. Management solutions include information exchange between neighbor nodes or dissemination of control information throughout the networks. Solutions may include observation of the designed network and manipulation on the parameters to provide suggested performance. The SWR algorithm provides this key feature by adapting itself dynamically to changing conditions with no user involvement and no information exchange between nodes. Note that there is no control packet in our protocol design. Nodes take their own decisions by considering the current network conditions and the parameter values via previously received data packets. This feature, in addition to the others given above, makes the SWR a unique routing algorithm among the ones designed for WSNs.

**3.8. Traffic Load Balancing.** Traffic load balancing is another challenging requirement in wireless sensor networks. As known balancing, the traffic load at nodes prolongs network-wide connectivity. The SWR algorithm uses spontaneous data flow approach in which a node is participated in routing process according to the availability of its own resources and

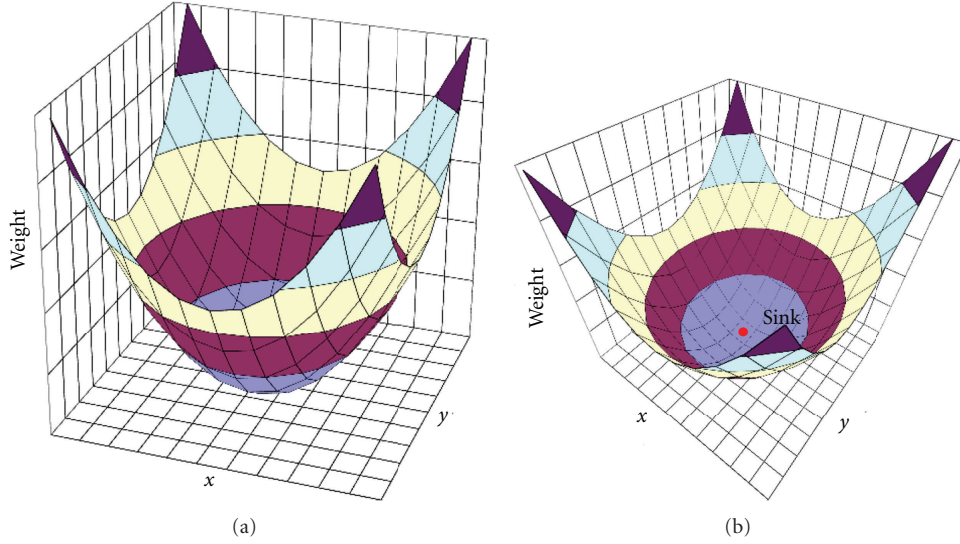


FIGURE 1: Weight metric using only Euclidean-distance provides a spontaneous flow toward the sink.

the parameters received within the data packet. Therefore, the data may flow from source to destination over different paths. Note that congestion avoidance provided with the SWR supports load balancing as well.

**3.9. Localized Behavior.** System-wide operations, such as beaconing or topology, update messaging that yields high message traffic and reduce the performance of the networks. Also, problems such as congestion and void have to be sorted out locally to preserve scalability. The SWR algorithm uses localized distributed operations to deal with such problems. Each node makes its own decision in an isolated manner according to its own knowledge and availability of resources. Note that the SWR algorithm does not use periodic beaconing which also contributes to the localized behavior.

#### 4. Stateless Weight Routing Algorithm

The stateless weight routing (SWR) is a stateless and reactive routing protocol that utilizes the geographic location information for routing. Routing tables and local/global topology information are not kept at nodes. Nodes do not exchange information prior to sending packets, and they do not even need to know the identities of their neighbors. Routes are constructed on-demand spontaneously without requiring any neighborhood information. Routing is achieved with aid of *weight* values of nodes ( $w_i$ , for node  $i$ ), which is derived from the geographical positions of the node and a number of parameters (1). These parameters may belong to either the node itself (parameters <sub>$i$</sub> , such as energy left at the node), overall network system (parameters<sub>network</sub>, such as network's instantaneous situation), or a combination of these two sets of parameters

$$w_i = \text{location}_i + \text{parameters}_i + \text{parameters}_{\text{network}} \quad (1)$$

For the case in which only the location comprises of the weight function, the weight value indicates the square of the

Euclidean distance to the sink node. Regardless of the content of the weight function, nodes away from the sink node have usually greater weight values with respect to closer ones, whereas the sink has always a weight value of 0. A generic weight diagram only considering the distance from the sink is shown in Figure 1, where the sink is positioned in the center of the area. This approach provides spontaneous natural data flows from nodes toward the sink when data is sent according to weight values of nodes. Weight values are not exchanged between nodes separately, but inserted into data packets. The use of weight metric makes the routing process simple and minimizes delay, energy consumption, and processing requirements at nodes in routing decision phase.

**4.1. Making Data Flow over Multiple Paths.** The SWR utilizes multiple simultaneous paths on data transmissions. The source node inserts its weight value into the packet and broadcasts. When a node receives a packet, it compares its own weight value with the weight value in the packet. Any node whose weight value is smaller than the transmitting node's weight value rebroadcasts the packet after replacing *Sender ID* and *Sender Weight* fields with its own values. The packet is dropped otherwise. In order to limit both the number of transmissions and the number of multiple paths, a *Threshold* value in terms of weight metric is used. Only the nodes have weight difference greater than the threshold value can rebroadcast the packet. By this way, nodes closer to transmitting node are avoided to rebroadcast. Rebroadcasting nodes are those that make more advances toward the destination. As seen in Algorithm 1, Euclidian distances to neighbor nodes are not calculated. Rather, only the weight value of the sending node that is retrieved from the packet header and the node's own weight value that is already known are compared.  $w(i)$  defines the weight of node  $i$ .

The *Threshold* value in terms of weight, which is also inserted into packet sent, gets value between 0 and  $r$ , where  $r$  is the transmission range known by all nodes. *Threshold* value

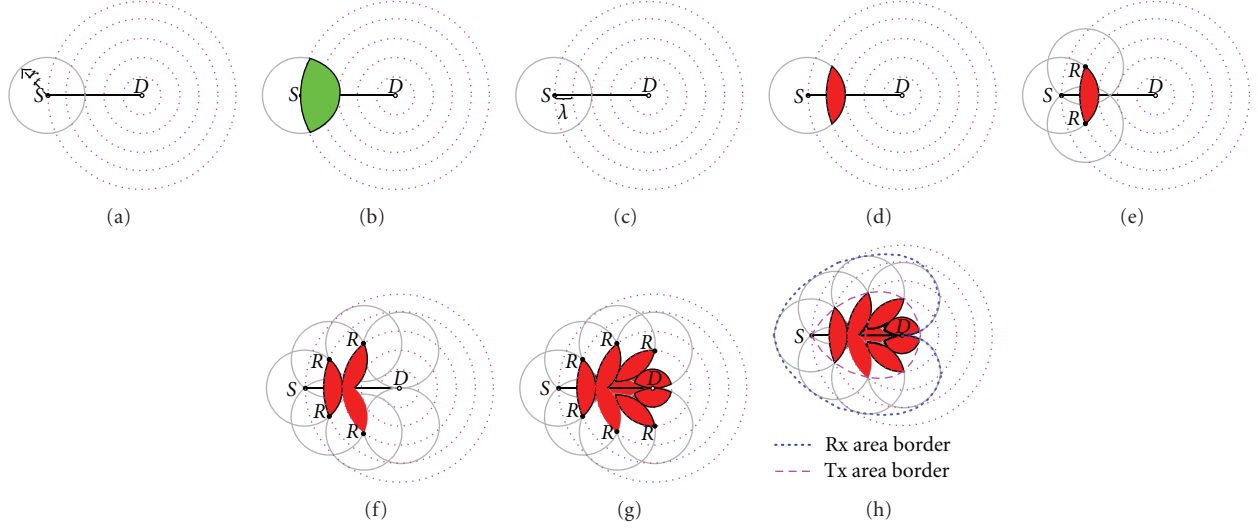


FIGURE 2: Possible transmission and receive areas between the source node  $S$  and destination node  $D$ .

```

if  $((w(\text{sender}) - w(i)) \geq \text{Threshold})$  then
    rebroadcast;
else
    drop the packet;

```

ALGORITHM 1: Pseudo code of data flow algorithm for node  $i$ .

is used for three purposes: to reduce energy consumption by regulating the number of transmitting nodes; to adjust the number of possible multiple paths; and to recover from voids. The *Threshold* value limits the number of retransmissions by avoiding relatively closer nodes to retransmit. This approach has two favorable outcomes: first of all, avoiding the closer nodes to transmit reduces the energy consumption; second of all, making more nodes to transmit rather than only the farthest one, which is prone to the link failures, constructs more robust paths.

**4.2. Coverage Area.** Figure 2 explains the covered area on transmissions. Assume that source node  $S$  has a data packet to send to the destination node  $D$  (Figure 2(a)). If conventional geographical routing algorithms were used, on the transmission of  $S$ , the nodes positioned in the shaded area between the  $S$  and  $D$  would be candidate retransmitting nodes (Figure 2(b)). In conventional greedy algorithms and stateless algorithms, *selection of next retransmitting node* schemes is used. In our approach, there is no such node selection scheme running at the sender. A node decides to relay or to drop the packet itself. The number of candidate retransmitting nodes is reduced inherently by using the threshold value as seen in Figures 2(c) and 2(d). All these candidate nodes can retransmit the received packet applying the Algorithm 1. Figures 2(e)–2(g) shows the covered area after successive transmissions until the destination  $D$  is covered. The covered area in Figure 2(g) shows the *worstcase* scenario since the outermost edge nodes ( $R$ ) are selected as the retransmitting nodes.

In Figure 2(h), the covered area is bordered with a dashed line to show the maximum possible transmission area, while the area bordered with dots shows the possible area that is affected after these transmissions (reception area). Mathematical model of the presented approach and its analysis are presented in Section 5.

**4.2.1. Reducing Transmissions by Adjusting Threshold Value.** The threshold value is adjusted to save energy by limiting the number of retransmitting nodes. Figure 3 shows the covered area after multiple successive transmissions between the source node  $S$  and the destination node  $D$  when Algorithm 1 is applied with different threshold values. Threshold values 50% and 85% of  $r$  are used in Figures 3(a) and 3(b), respectively. As seen, increasing the threshold value reduces the number of candidate retransmitting nodes.

**4.2.2. Adjusting Threshold Value in Dense Topology.** The default threshold value can be adjusted according to the node density in the network. In dense networks, the threshold value can be set to be high by default to limit the retransmitting nodes. In non-dense networks, the default threshold value can be set to be small value to allow enough nodes to participate in data flow. In dense topologies, sleep scheduling is a typical approach to reduce transmissions and energy consumption. Sleep scheduling can also be applied to the presented approach but it is not covered within the scope of this paper.

**4.3. Data Packet Transmissions.** If a node has a data to send the sink, it inserts its identification number, current packet sequence number, and the intended destination (here it is sink node)'s identification number into the appropriate fields. Relaying nodes do not change these three values until the packet reaches its destination. The sender inserts also its identification number and the current weight value into the *Sender ID* and *Sender Weight* fields, respectively, where these values are changed by the relaying nodes to their own values.

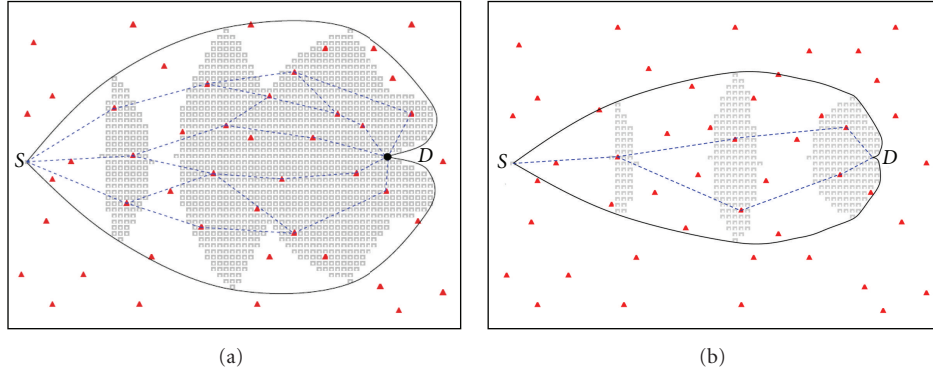


FIGURE 3: Possible retransmission area and candidate nodes and paths.

*Threshold* field is set to system-wide default value which is actually 50% of  $r$ , but can be changed according to the network dynamics. Then the node broadcasts the packet. Actually, the packet is passed to the MAC layer to be sent to the addressed nodes. Relaying intermediate nodes only make changes on *Sender ID* and *Sender Weight* fields on their transmissions. In case of a void, each relaying node also makes its own decision to change the *Threshold* value according to its needs. This approach is described in Section 4.5.

**4.4. Reliability.** Using multiple paths is the most reliable method to convey the data to the destination. In MANET and WSN, paths should be constructed on-demand due to frequently changing topology and propagation conditions. However, almost all of the proposed protocols in the literature [30–33] have a route construction phase and these multiple paths are constructed before sending the data packet. In these approaches, generally the packet is sent over the primary path. In case of a route failure, the packet is sent over the alternate path. Switching from the failed path to the alternate one introduces an additional delay. Failed paths cause packet drops and retransmissions, exacerbating the delay. If all paths known in-advance failed, a new route recovery or route reconstructed is required, which increases the delay longer.

Reliability in the SWR is provided by using multiple paths. Contrary to the known protocols in the literature, these on-demand paths are not constructed ahead of packet sending. Algorithm 1 spontaneously constructs simultaneous multiple paths while the packet is on the way toward the sink. Data packet is simultaneously carried over every constructed path. Information about paths is not kept at nodes for future use. Keeping such path information is unnecessary in frequently changing topology and introduces overhead and delay as depicted above. Therefore, comparing to the protocols using multiple-paths, the SWR provides the minimum delay.

The use of simultaneously active multiple paths provides a continuous connection in the case of a broken path. While the spontaneous dynamic data flow is regenerated at the broken link of the broken path, the data packet, meanwhile, is carried over other paths. This approach naturally eliminates the problem of a possible breakage in end-to-end communication. Moreover, simultaneous establishment of multiple

dynamic data flows naturally isolates problems triggered by a single node failure. Thus, route breakages are not observed in the SWR.

The number of the paths depends on the distance (length in hops) and the applied threshold value. As the distance increase, the number of the constructed paths increases. As a result, for the same source-destination pairs, data packets may follow different paths. Considering the construction and use of multiple-paths, the SWR provides high reliability.

Multiple paths exploited in data flow are the braided multiple paths. They overlay with each other in some part and utilize the advantages of the best path. One design measure of SWR is providing guaranteed delivery. Simultaneous data flow over multiple paths in SWR substantially achieves this goal. In case of a failure in data flow, void recovery algorithm (Algorithm 2) is invoked. If a node experiences void, it rebroadcasts the packet with a reduced threshold value. This approach allows the sender's neighbors, which have the same or smaller weight values, to integrate into flow process. Further step for on-going void problem is using a fake weight value instead of actual weight value of the sender. This second approach allows all neighbors to participate in flow process. These two approaches increase the covered area of the data packet sent, building more paths on-demand. Actually, void recovery is not a separate process; it is inherently done in route construction. Although, the void recovery is only invoked for the data flow that encounters the void, the data packet continues to flow over other paths, meanwhile. Figure 4 illustrates the constructed multiple paths and sub-paths in void recovery. In Figure 4(a), there are  $k$  different braided multiple paths from source to destination, which are constructed spontaneously and simultaneously. If a node on a possible path cannot forward the packet (node 2,  $n_2$ , on path 1,  $P_1$ , in Figure 4(a)), alternate subpaths are constructed spontaneously and simultaneously as shown in Figure 4(b). These subpaths may overlap as well with other paths, as defined above.

**4.5. Void Avoidance and Recovery.** Void and coverage hole problems and related studies in the literature are summarized in [19–21]. Only a few routing protocols in the literature, propose solutions to recover from voids. The methods used in these solutions include the use of multiple paths and

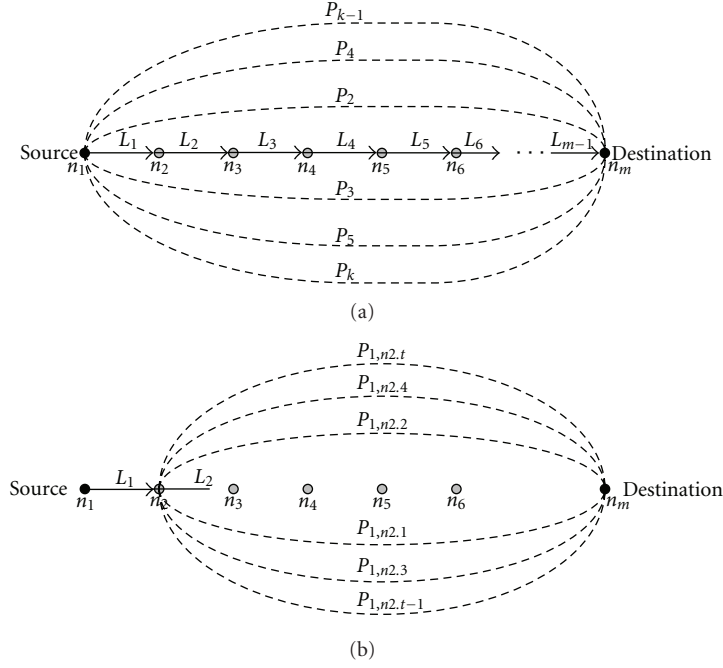


FIGURE 4: Multiple paths exploited in SWR are braided paths. Void recovery algorithm provides construction of new paths.

```

if (Threshold > 0) then
  set Threshold to 0 (zero);
  rebroadcast;
if (the packet cannot be relayed) then
  set the  $w_{\text{sender}}$  to  $w_{\text{sender}} + w'$  in header;
  rebroadcast;

```

ALGORITHM 2: Void avoidance algorithm.

alternating paths, retransmissions, broadcasting, flooding or localized flooding, and discovery of the voids and boundary of voids. Unfortunately, these protocols present poor performance, especially when the topology changes frequently. Reasons of performance decrease can be explained as follows. These methods are highly dependent on the use of information exchange between nodes due to topology information requirement at nodes to recover from voids. Success, effectiveness, and responsiveness of these methods depend on the frequency and reliability of the retrieved topology information. Therefore, a tradeoff exists between the provisions of the topological information and the accuracy of this information [7, 16, 22]. Frequently exchange of information consumes energy and introduces communication overhead. On the other hand, infrequent information exchange causes the nodes have unreliable topology knowledge.

Moreover, some protocols such as the stateless geographical routing protocols propose solutions to be implemented at the MAC layer and generally have local-minima problem. Recovery from voids using such protocols is too complex and costly. With respect to these methods, simple and efficient void avoidance and recovery methods are provided in the

SWR. The proposed methods are peculiar to the SWR and guarantee the delivery of data to the destination.

**4.5.1. Implicit Void Avoidance Approach.** One use of *Threshold* value is for void avoidance. Increasing the threshold value provides fewer nodes in number to be able to relay the data packets, and decreasing the threshold value provides more nodes in number to be able to relay the data packets (Figure 3). In case of void detection, the transmitting node decreases the threshold value allowing more nodes to apply the data flow algorithm (Algorithm 1) as depicted in Figure 5. Nodes can understand the existence of a void by the nonretransmission of the packet with the same parameters by the nodes that have lower weight values. Adjusting the threshold value adjusts the number of multiple paths. Implicitly, the void problem is eliminated substantially due to utilizing multiple paths. For the case of large gaps in the topology, a void elimination algorithm (Algorithm 2) is proposed to solve the void problem.

**4.5.2. Explicit Void Avoidance Algorithm.** Threshold value introduces limitations on data dissemination area. However, if the void-experiencing node cannot deliver the packet to the recipients due to large size of the void, an explicit void elimination approach is used. On encountering a void, the node executes the void elimination algorithm (Algorithm 2). The algorithm consists of two steps. In the first step, the algorithm tries to transcend the void by decreasing the threshold value to 0 (zero). Therefore, larger area can be covered to forward the packet. If the packet still cannot be forwarded due to void, the second step is performed. Transmitting node retransmits the packet with a weight value greater than its own weight embedded into the sender's weight field in the packet and the



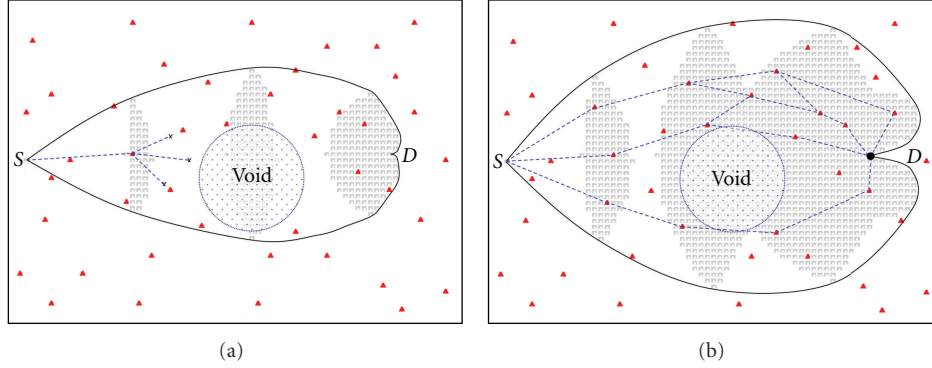


FIGURE 5: Void recovery in SWR.

threshold value set to 0 (zero). Assuming that  $w$  is the weight of the transmitting node and  $w'$  is the additional weight to be added where  $w' > 0$ , the new value for weight would be  $w + w'$ . By increasing the original value for weight, the transmitting node enforces the rearward nodes to participate into the routing. Therefore, a void can be passed by without any complex calculations.

## 5. Analysis of the SWR Algorithms

In this section, the analysis of the SWR algorithm in terms of the number of transmissions, energy consumption and reliability is presented. Compared to the other geographical routing algorithms, the SWR algorithm does not use beaconing, but utilizes multiple simultaneous paths on data transmissions. First, the transmissions and then the energy consumption are analyzed.

**5.1. Analysis of the Transmissions.** Geographical routing algorithms send beacons periodically to inform neighbors. At an instant time,  $1/f_{\text{beacon}} \times N$  beacon transmissions occur, where  $f_{\text{beacon}}$  is the beacon frequency and  $N$  is the number of nodes in the network. During a time interval  $T$ ,

$$\begin{aligned} \text{Total Transmissions} &= \text{Beacon Transmissions} + \text{Data Transmissions}, \\ \text{Total Transmissions} &= \left( \frac{T}{f_{\text{beacon}}} \times N \right) \\ &+ (\text{number of generated packets} \times \text{hop count}), \end{aligned} \quad (2)$$

where hop count is assumed as an average path length.

On the contrary, in the SWR, only data transmissions occur. The number of these data transmissions varies according to the distance between the source and destination, and the parameters applied in routing algorithm such as threshold value. It is not possible to calculate exact number because of spontaneous data transmissions over multiple paths. However, maximum number of transmissions can be found analytically by considering the covered area on transmissions (Figure 2(h)). Since each node can only transmit once, total

transmissions are equal to the covered area multiplied with node density ( $ND$ ). Figures 6 and 7 show the analytical representation of the covered area. This area is found approximately with finding the area of each triangle, starting from the triangle near source (1st triangle) and ending with the  $K$ th triangle close to sink

$$\text{Covered Area} = 2 \times \sum_{i=1}^K A_i, \quad (A_i \text{ is the triangle area } i), \quad (3)$$

where  $K$  is the amount of triangles that covers the area and can be found as

$$\begin{aligned} K &= \max \left\{ n \mid \left( \sum_{i=1}^n \theta_i \right) \leq 180^\circ, \right. \\ &\left. \cos \theta_i = \frac{(d_i - \lambda)^2 + d_i^2 - r^2}{2(d_i - \lambda)d_i}, \quad d_{i+1} = d_i - \lambda \right\}, \end{aligned} \quad (4)$$

where  $d_1$  is the distance between the source and the sink,  $r$  is the transmission range, and  $\lambda$  is the threshold value.

Total transmissions in the SWR is

$$\begin{aligned} \text{Total Transmissions} &= \text{Data Transmissions}, \\ \text{Total Transmissions} &= \left\{ \left( 2 \times \sum_{i=1}^K A_i \right) \times ND \right. \\ &\left. \times \text{number of generated packets} \right\}. \end{aligned} \quad (5)$$

The calculated area will always be greater than actual covered area. Gaps on the actual covered area (Figure 2(h)) are included in the analytical calculation (Figure 7). Comparison of the calculated and actual covered areas is given in Figure 8. Actual covered area has been found by simulations and considering the worst cases. Figure 8 shows that there is a great difference when there is a high threshold value. With high threshold value, far distant nodes will be selected. In this case, the gaps between the successive transmitting nodes will

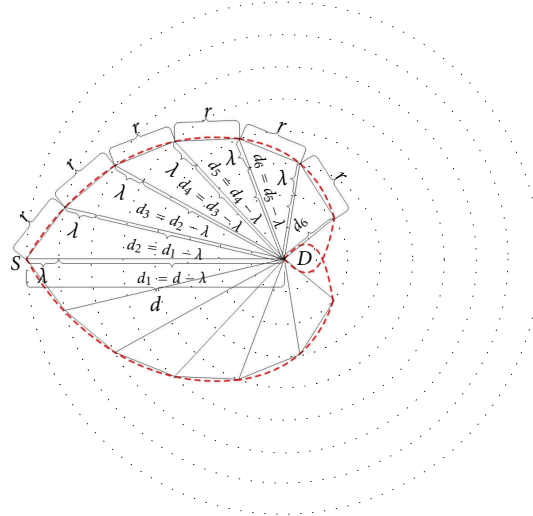


FIGURE 6: Covered area calculation for SWR.

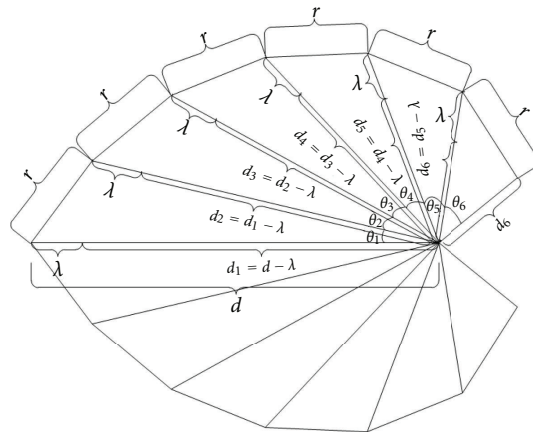


FIGURE 7: Covered area calculation for SWR. Transmission range circles are removed from Figure 6.

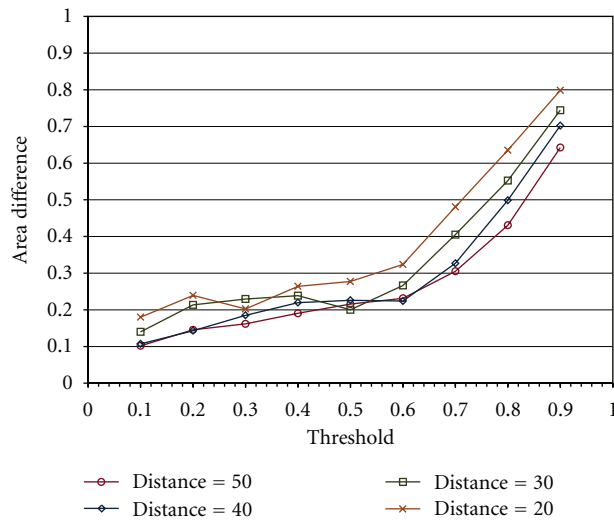


FIGURE 8: Comparison of covered area differences between the calculated area and the actual worst case covered area. In this comparison, distance between the source and destination pair varies between 20 m and the transmission range is  $r = 10$  m. Threshold value changed to observe the variations between calculated and actual covered area.

be greater. Since the calculation with triangulation includes these areas, the difference gets higher. Experiments support these results. It was observed in the SWR that number of transmissions are much less than the calculated ones. Compared to the other geographical routing algorithms, this outcome affects energy consumption at nodes gratifying the SWR algorithm.

*5.2. Analysis of Energy Consumption.* Routing algorithms in the literature propose approaches to minimize the energy consumption, but consider only energy consumption on transmissions. Energy consumptions on receptions, calculations, and sensing are not involved or modeled. Regretting these energy consumptions, especially energy consumption in receive process, outcomes unrealistic performance results. Related studies in energy consumption [34–37] emphasize that receive process consumes as much power as the transmission process. Then the total energy,  $E_{\text{total}}$  (6), consumed by a node at an arbitrary time is the sum of these energy requirements [34]. It is also defined in [34] that efficient sensing circuitries and computation algorithms reduce  $E_{\text{sensing}}$  and  $E_{\text{computation}}$  substantially. Therefore, they are considered as constant values

$$E_{\text{total}} = E_{\text{transmit}} + E_{\text{receive}} + E_{\text{computation}} + E_{\text{sensing}}. \quad (6)$$

On a transmission, transmitting node consumes the energy,  $E_{\text{transmit}}$ , and a receiving node consumes the energy,  $E_{\text{receive}}$ . If the transmitting node has  $n$  neighbors, the overall system consumes the energy,  $E_{\text{network}}$ , for one transmission;

$$E_{\text{network}} = (n \cdot E_{\text{receive}}) + (1 \cdot E_{\text{transmit}}). \quad (7)$$

If it is assumed that  $E_{\text{transmit}} \approx E_{\text{receive}}$ , the overall system consumes  $(n + 1)E_{\text{transmit}}$  for only one transmission. Neglecting such an amount of energy consumption causes unreliable system performance results. According to (7), beacon-based geographical routing protocols consume most of their energies in the beaconing processes.

The energy consumption in a system, during a time period,  $T$ ,

$$E_{\text{network}}(T) = E_{\text{beaconing}} + E_{\text{events}}. \quad (8)$$

Note that  $E_{\text{beaconing}}$  is only consumed in beacon-based protocols. Considering the transmissions in (2), overall energy consumption becomes

$$\begin{aligned} E_{\text{network}}(T) &= \left\{ \left( \frac{T}{f_{\text{beacon}}} \times N \right) \right. \\ &\quad \left. + (\text{number of generated packets} \times \text{hop count}) \right\} \\ &\quad \times \{(n \cdot E_{\text{receive}}) + (1 \cdot E_{\text{transmit}})\}. \end{aligned} \quad (9)$$

The SWR protocol consumes energy only on data transmissions. Nodes that remain in the shaded area retransmit the received packet for only once. The maximum energy

consumption in data transmissions for the SWR can be found by multiplying (5) and (7)

$$\begin{aligned} E_{\text{network}}(T) &= \{\text{Covered Area} \times \text{ND} \\ &\quad \times \text{number of generated packets}\} \\ &\quad \times \{(n \cdot E_{\text{receive}}) + (1 \cdot E_{\text{transmit}})\}. \end{aligned} \quad (10)$$

$$\begin{aligned} E_{\text{network}}(T) &= \left\{ \left( 2 \times \sum_{i=1}^K A_i \right) \right. \\ &\quad \left. \times \text{ND} \times \text{number of generated packets} \right\} \\ &\quad \times \{(n \cdot E_{\text{receive}}) + (1 \cdot E_{\text{transmit}})\}. \end{aligned} \quad (11)$$

Comparing energy consumption (9) and (11), it is seen that, as the covered area in the SWR remains smaller than the network area, the SWR will consumes less energy than any beacon-based protocols. Compared to the any other protocol, for a generated data packet, the SWR consumes less energy when nodes in the covered area in the SWR are less than the total transmissions to transmit a packet including the control and management packets in other protocols. The SWR, on the other hand, carries the data packet on multiple paths to provide reliability. The SWR utilizes the energy to provide reliability rather than topology learning, route construction, and maintenance.

*5.3. Analysis of the Reliability.* There is a number studies in the literature that analyze the reliability issues in MANET and WSN. In [30], an analytical framework is developed to characterize the random behavior of a multihop path and derive path metrics to characterize the reliability of paths, modeling and analyzing the mean path duration, and the path persistence. Supporting results are provided in [31] with experiments. They characterize link reliability measures in an actual sensor network setting and analyze how reliable data transfer mechanisms impact overall path reliability. For the path  $P$  which consist of  $|P|$  links and where  $\mathcal{P}_{i,j}$  is the probability of a failure on a transmission link  $l(i,j)$ , the success probability,  $\mathbb{P}^{\text{success}}$ , is

$$\mathbb{P}^{\text{success}} = \prod_{\forall l(i,j) \text{ on path } P} (1 - \mathcal{P}_{i,j}^{\text{fail}}). \quad (12)$$

Simultaneous multiple paths increases the reliability as given below

$$\mathbb{P}_M^{\text{success}} = 1 - \prod_{m=1}^M \left( 1 - \prod_{\forall l(i,j) \text{ on path } P_m} (1 - \mathcal{P}_{i,j}^{\text{fail}}) \right). \quad (13)$$

Figure 9 shows the reliability for variable number of paths with respect to variable link failure probability. It is seen that using simultaneous multiple paths provides considerable higher reliability with respect to single path.

The bandwidth in WSN and MANET is also limited. Usage of a single path for routing may not provide the

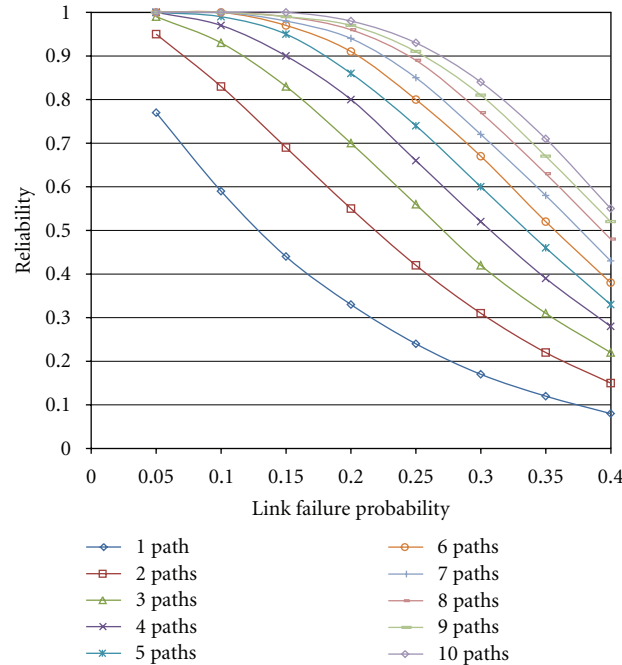


FIGURE 9: End-to-end reliability for variable number multiple paths with 5 hops.

required bandwidth for the current communication. In case of transmission of packets over multiple paths, the aggregate bandwidth of multiple paths may satisfy the bandwidth requirement of the current communication [38]. Therefore, a smaller end-to-end delay may be achieved [38].

## 6. Simulation and Results

In order to evaluate the performance of the SWR, it is compared with well-known benchmarking protocols. Performance results present efficiency of the proposed approach with respect to other protocols. A number of energy-related metrics are evaluated in detail as well as the impact of the node density. Moreover, there are some additional results to show the characteristics of the SWR.

**6.1. Compared Protocols.** The proposed approach is compared with known benchmark protocols. One of the benchmark of all geographical routing protocols is the Greedy Perimeter Stateless Routing (GPSR) protocol, which is also a stateless geographical routing protocol [23]. However, it uses neighborhood topology information for packet forwarding in greedy manner. GPSR collects the local topology (neighborhood) information by periodic beaconing messages. We used three beaconing periods for GPSR as 1 sec, 3 sec, and 6 sec. Shorter beaconing intervals increase the energy consumption in an effort to provide more accurate and up to date topology information. Different beacon periods yield different performance results even though they use the same routing approach. Therefore, to differentiate the results of GPSR for different beacon periods, naming strategy that includes the beaconing period such as “GPSR 1 sec” is used. The most well known routing algorithm is flooding. Actually

most of the routing protocols for WSN and ad hoc networks are the variants of flooding with some modifications and optimizations. Flooding is the simplest stateless routing protocol since it does not require any routing table. The original data packet traverses on every path in the network including the shortest one. We also compare the results with a real-time protocol called SPEED [24]. Another well-known energy efficient-stateless protocol, IGF [26] is compared with SWR to compare energy efficiency and other performance metrics. The results are also compared with an imaginary routing protocol, which is called *Virtual Optimal Routing* protocol. It is assumed in this protocol that all nodes always know the optimal path to the destination. Thus, it has not any routing overhead and the data packets are carried over optimal path towards the destination. Therefore, the transmissions and the energy consumption will remain minimal. Such a naturally optimal protocol provides a good mean to compare other protocols for various performance metrics such as energy consumption, lifetime, and remaining energy.

Protocols are experimented with two different scenarios as presented in Table 1. Parameters for the scenarios are subject to observe the performance of the system in different conditions. Nodes are randomly distributed in a well-defined topology [39]. Network is designed with the methodology defined in [40]. Randomly generated, UDP-based constant bit rate (CBR) traffic is used for evaluations. Nodes randomly generate 128 Byte payload packets with a probability of 0.05 packet/min. Packet generation frequency is increased to 1 packet/sec to observe the effects of load to the energy consumption. To provide the double range property [39], nodes have a sensing range ( $R_S$ ) of 50 meters and a transmission range ( $R_C$ ) of 100 meters ( $R_C/R_S = 2$ ).

TABLE 1: Scenarios used in simulations.

Scenario number	Number of nodes	Area (m <sup>2</sup> )	Density	
			Nodes per m <sup>2</sup>	Nodes per node coverage
Sc. 1	100	500 × 500	0.04%	13
Sc. no. 2-A	500	1000 × 1000	0.05%	16
Sc. no. 2-B	1000	1000 × 1000	0.1%	31
Sc. no. 2-C	10000	1000 × 1000	1%	314

6.2. *Performance Evaluation*. Different factors are considered to observe the effects on the performance of the system. In performance evaluations, in addition to the canonical metrics such as energy consumption in routing and effects on network lifetime, routing overhead, some additional metrics such as load of components are observed.

Two energy-related performance metrics in WSNs are the energy consumption and the lifetime of the system. Although some nodes might consume major portion of their energy, the system may continue to live by load balancing and selecting more powered (with respect to energy residual) nodes as retransmitting nodes. Therefore, the lifetime of the system may be prolonged by avoiding the transmissions through energy-limited nodes. If the energy level of the nodes is not considered, the system may fail due sudden energy depletion at some nodes even though most of the nodes may still keep major portion of their energies. Therefore, energy consumption and lifetime have to be investigated to understand the behavior of the system. Note that besides the energy level of the system, the remaining energy of individual nodes is also investigated.

This information gives clues on node redeployment strategies. Thus, the performance evaluations have been performed. To make it clear, the following definitions are made.

- (i) *System Energy*: the ratio of the cumulative energy of the nodes in the system at startup.
- (ii) *System Lifetime*: it is measured from startup until first failure occurrence on path construction from source to the sink.
- (iii) *Remaining Energy Level*: the ratio of the energy at an instant time to the energy at startup.

6.2.1. *Energy Consumption and System Lifetime*. As discussed in Section 5, nodes consume as much power in the receptions as in the transmission. In our experiments, we measure energy consumptions in both transmissions and receptions. Energy consumptions in sensing and computation are considered as negligible constant values as in [34].

Figure 10 shows the remaining system energy percentages against the applied routing algorithms in Scenario 1. The  $x$ -axis shows the elapsed simulation time in seconds. The  $y$ -axis shows the remaining energy levels of the system. Only the energy consumption on data packet transmission and routing processes are considered (including all transmissions and receptions).

System energies are measured until the lifetime of the system. The lifetime is considered as the first failure on finding

any route to the destination. GPSR with 1 sec beaconing, SPEED, and the flooding algorithm deplete the allocated system energy very quickly. Their lifetimes are very close: 124 seconds, 125 seconds, and 153 seconds for SPEED, GPSR (1 sec beaconing), and flooding, respectively, (Table 2). SPEED and GPSR deplete most of their energy for beaconing, while the flooding depletes its energy on routing process. The system energy of the GPSR protocol is slightly higher than flooding. However, flooding has longer lifetime because it uses all paths at once to reach the destination. Node terminations do not affect flooding if there is a path to the sink. In GPSR, energy consumption decreases and lifetime extends as the beaconing periods increase (e.g., GPSR 3 sec and GPSR 6 sec). However, they fail to live longer than beaconless protocols. Extending the beaconing period in GPSR avoids energy consumption at nodes; however, they fail to keep routing tables up to date. The SWR protocol survives when the simulation ends after 900 sec. Energy consumption in the SWR is close to the energy consumption in the optimal routing. In SWR, nodes consume their energy only in data packet transmissions. However, nodes in IGF consume their energy both on probe and data packet transmissions. Remaining system energy is lower than SWR. IGF fails to find routes at simulation time 531 though it has high remaining system energy. It fails to find route due to node terminations close to the sink node because of double transmissions of nodes on-the-route (one for probe reply and one for data packet).

In addition to the lifetime of the protocols, information about node terminations is presented in Table 2. Each protocol would fail to find routes after node terminations. However, their response to node terminations varies, and that affects their lifetime. In GPSR 1 sec, the paths are constructed until second 124. When the number of terminated nodes reaches 23, GPSR 1 sec fails to find routes to the sink. SPEED presents similar results with GPSR 1 sec. Although there are more terminated nodes in flooding (34 terminated nodes) than both SPEED and GPSR 1 sec, paths are constructed until second 153 since every possible path is tried in flooding even if there are many terminated nodes. Besides that in GPSR and SPEED, nodes closer to the sink deplete energy more quickly because roughly all paths toward the sink involve these nodes. Such a quick depletion thus composes a gap surrounding the sink. On the other hand, in flooding, nodes deplete their energy almost equally, because every node equally involves in routing. These results are presented in Figure 11.

It is better to analyze Figure 11 and Table 2 together. It is seen in Figure 11(a) that when the GPSR 1 sec fails to find any route at 125th second, the other nodes almost have depleted

TABLE 2: Comparisons of the protocols with respect to lifetime and node terminations.

Routing protocols	Flooding	GPSR 1 sec	GPSR 3 sec	GPSR 6 sec	SPEED	IGF	SWR
Average system lifetime	153 sec	125 sec	331 sec	571 sec	124 sec	531 sec	>900 sec
Time of the first node termination	109 sec	106 sec	268 sec	481 sec	104 sec	498 sec	NONE in 900 sec
Average number of terminated nodes on destination unreachable	41	23	24	15	29	8	NONE in 900 sec

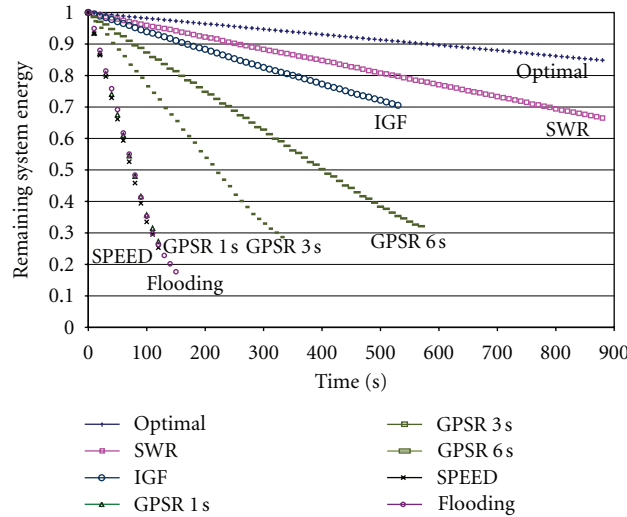


FIGURE 10: Remaining system energy levels of the protocols in Scenario 1.

their energies. Energy consumption has been diffused all over the system. If the system lifetime has been prolonged a few more seconds, almost all of the nodes would terminate. Although the remaining nodes in flooding have higher energy levels than the remaining nodes in GPSR 1 sec (Figure 11(a)) at this moment, similar results to GPSR 1 sec are observed in flooding. Due to flooding, all nodes participate equally to the routing process. This makes the nodes have almost equal energy levels. Similar results are observed in GPSR 3 sec and GPSR 6 sec except the extended lifetime due to beacon period extension as seen in Figures 11(c)–11(e). In SWR, nodes have higher energy levels and longer lifetime. In IGF, due to probing on data packet transmissions, nodes deplete their energies earlier than SWR. In SWR energy levels of the nodes are close to the optimal. The reason is that energy is consumed only in data packet transmissions in these protocols. Node terminations do not occur in these two protocols in the simulation course of 900 s.

**6.2.2. Effects of Node Density.** Routing protocols should perform well on both densely and sparsely deployed environments, however, fall beyond this expectation. They suffer from the node density due to introduced overhead and more energy consumptions at nodes. Scenario 2 is used to compare protocols in dense environments. Number of the nodes per

unit area is increased to observe the effects of node density. Node density ratios are 0.05%, 0.1%, and 1% to total area; in terms of numbers of nodes 500, 1000, and 10000, respectively. In this scenario, to be able to observe and compare the effects of node density, unlimited energy is given to each node. Other parameters are same as in Scenario 1. Results of energy consumption with different node densities are presented in Figures 12(a)–12(c) for node densities 0.05%, 0.1%, and 1%, respectively.

The  $x$ -axis shows the elapsed simulation time in seconds. The  $y$ -axis shows the system-wide energy consumption in joules. Only the energy consumption related with the routing processes (transmissions and receptions) are considered. Note that the  $y$ -axis is in logarithmic scale. It is clear that as the node density increases, energy consumption increases (Figures 12(a)–12(c)). Similar results are observed for all protocols. GPSR 1 sec and SPEED present almost identical, so they overlaps in Figures 12(a)–12(c). The difference due to beaconing in GPSR 1 sec, GPSR 3 sec, and GPSR 6 sec is seen clearly. Unsurprisingly, there is gap between the beacon-based protocols and beaconless protocols. The difference between these protocols considering the energy consumption in the logarithmic scale emphasizes the excessive energy consumption in beacon-based protocols and the energy-efficiency of beaconless protocols. Beaconless protocols outperform the others. Though the SWR uses multiple paths, it has

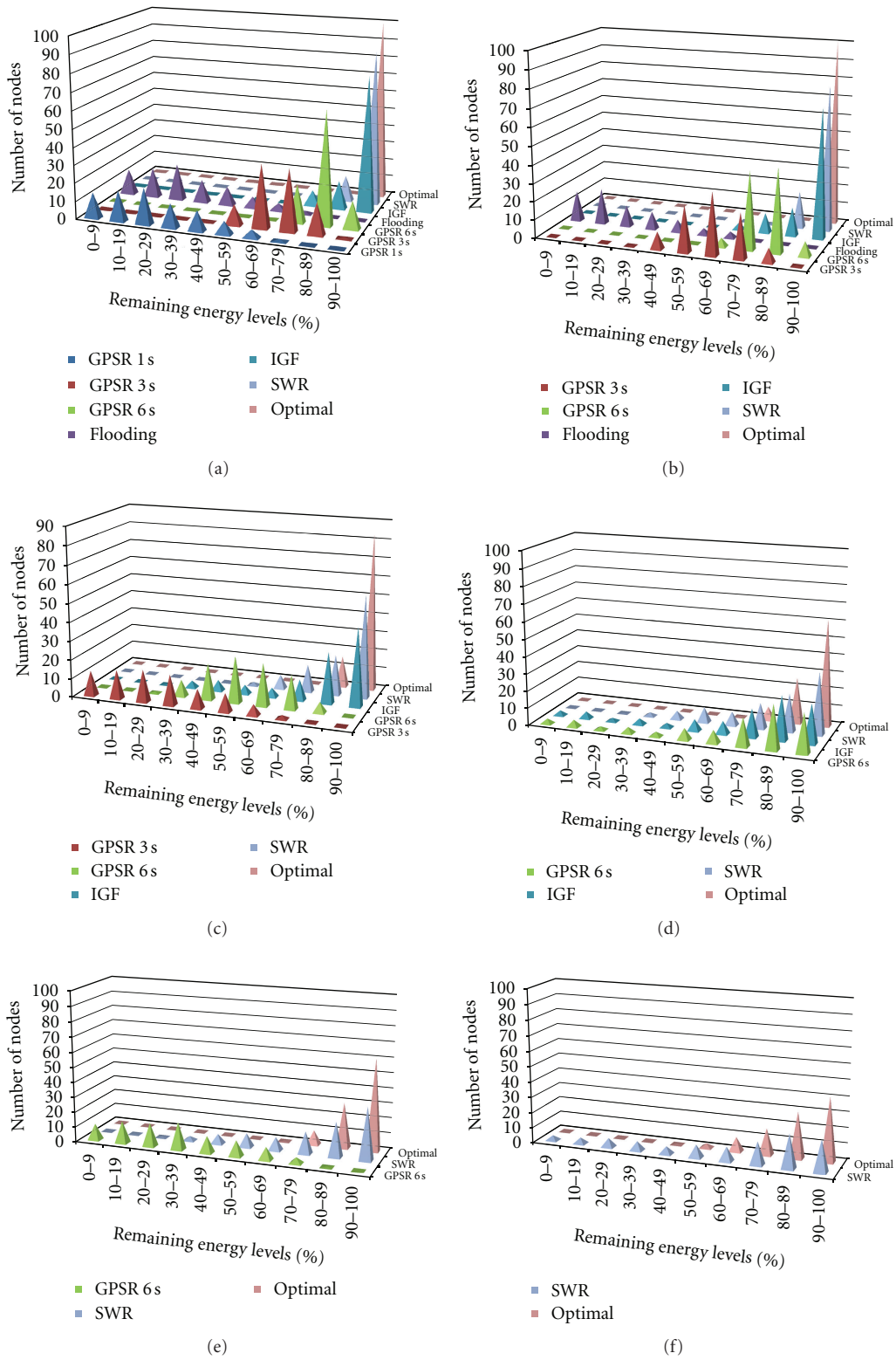


FIGURE 11: Remaining energy levels of the nodes when (a) "SPEED" fails to find a route at time 124 sec, (b) "GPSR 1 sec" fails to find a route at time 125 sec, (c) "Flooding" fails to find a route at time 153 sec, (d) "GPSR 3 sec" fails to find a route at time 331 sec, (e) "IGF" fails to find a route at time 531 sec, and (f) "GPSR 6 sec" fails to find a route at time 571 sec.

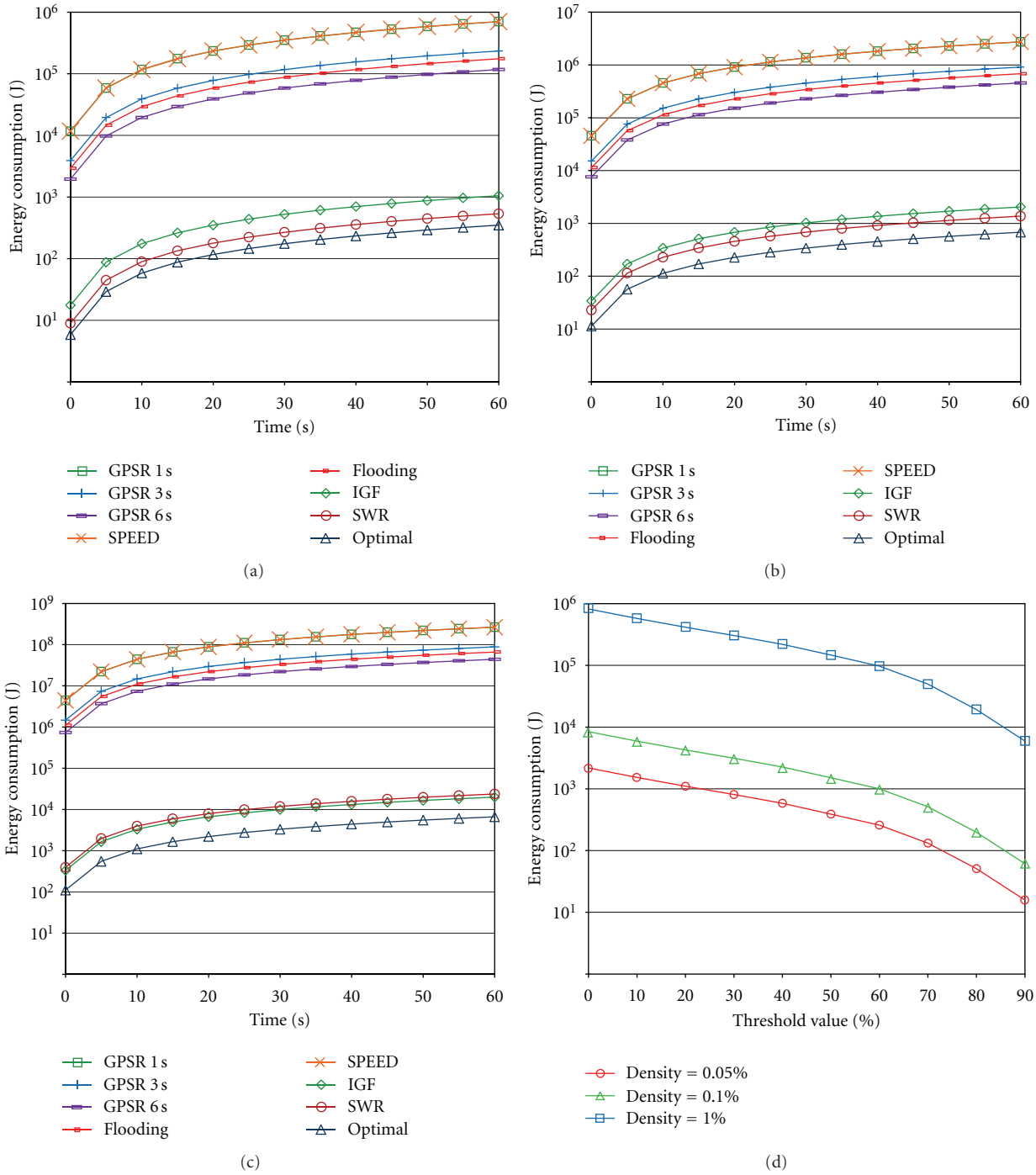


FIGURE 12: Energy consumption of protocols with different node densities: (a) 0.05% nodes/m<sup>2</sup>, (b) 0.1% nodes/m<sup>2</sup>, (c) 1% nodes/m<sup>2</sup>, and (d) effects of the threshold value and the node densities to the energy consumption in SWR.

low energy consumption even in high density networks. Effects of node density to SWR protocol are not more than other protocols. In fact, the effect remains at a moderate level. On the other hand, the amount of consumed energy in SWR can be decreased by adjusting the threshold value (Figure 12(d)). Increasing the threshold value reduces the number of retransmissions, as described in Section 4. In this respect, SWR presents a great opportunity and flexibility

with respect to other protocols to adjust the threshold value according to network density and environmental parameters.

Figure 12(d) shows the energy consumption with different threshold values for SWR. The x-axis shows threshold values applied in routing algorithm for SWR. Note that the y-axis shows the system-wide energy consumption in logarithmic scale. The amount of energy saving is very high when a higher threshold value is used. For example, for the density



0.05%, the amount of energy consumed with threshold value 0.5 is 388 joules, and with threshold value 0.9 is 16. There is a 95% energy saving with these values. When the density of the network increases, it becomes more noticeable. For the density 0.1%, the amount of energy consumed with threshold value 0.5 is 1509 joules, and with threshold value 0.9 is 61 joules. For the density 1%, the amount of energy consumed with threshold value 0.5 is 146706 joules, and with threshold value 0.9 is 5956 joules. The amount of energy saving is 96% with these values.

*6.2.3. Effects of Range and Threshold to Energy Consumption.* In SWR, energy consumption can be reduced by adjusting the threshold value as shown in Figure 12(d). As defined in Section 4.3, threshold value can be adjusted momentarily and independently in a distributed manner at each node for different reasons according to the current needs of the node. It depends on the current condition or the event a node experiencing, such as void recovery, requirement of higher reliability or guaranteed delivery, and urgent or real-time data transmissions. After the completion of the event, the threshold value can be readapted to the default value to save energy. These adaptations occur in a distributed manner independently at nodes which require these adaptations and without any administrative manipulation outside the network. Each node itself intrinsically decides to increase or decrease the threshold according to its current conditions. On the other hand, to satisfy some performance metrics, for example, reliability, the threshold value can be set to a new default desired value, which is based on the preknown information about the network-wide requirements. It can be reset again to the predefined default threshold value as needs according to the network-wide requirements as described above.

Another parameter which affects the energy consumption is the range of the transmissions. Increase in transmission range causes more nodes to receive transmissions, which cumulatively increases the system-wide energy consumption. On the other hand, range increase causes shorter path constructions, which causes a reduction on the number of transmissions and receptions. Therefore, the effect of range to the energy consumption is examined. Threshold value in SWR affects the reliability and energy consumption. Thus, effects of these two parameters range and threshold are examined. In WSNs, there are some approaches that adjust the transmission range of the transmitter according to the known distance of the receiver. In SWR, nodes do not have any information about the topology nor neighborhood nodes. Usage of adaptive transmitter is needless in SWR. Transmission range is fixed. What a node may need to change is only the threshold value in SWR.

Effects of range and threshold values are shown in Figures 13(a) and 13(b). Figure 13(a) shows the relation between relay nodes coverage, range, and threshold value. The *x*-axis shows the *applied threshold value*, and the *y*-axis shows the *relay node coverage reduction*. Relay node coverage defines the nodes are the candidates to be involved in relaying a packet between a source-destination pair. These relay nodes are located in an area shaped similar to the one in Figure 2(h).

Of the nodes in the transmission range, SWR protocol allows only those nodes which have lower weight values to relay the data. The number of these nodes is dependent on the applied threshold value. Therefore, change in threshold value changes the covered area, in other words, changes the number of relay nodes.

Figure 13(a) shows the relay node coverage relationship for a source-destination pair 100 meters away from each other. For 90 meters transmission range (tx range = 90), data is relayed in 2 hops. With a threshold value of 10%, the covered area is reduced 72% with respect to the area covered with threshold value 0%. For transmission range of 80 meters (tx range = 80), data is again relayed in 2 hops. With a threshold value 10%, the covered area is reduced 33% with respect to the area covered with threshold value 0%. Other transmission ranges (tx range = 70, 60, 50, 40, 30, 20, 10) show similar results with respect to the transmission range of 80 meters (tx range = 80) for threshold value 10%. Secondly, transmission ranges between 70 meters and 10 meters present close reduction values for the same threshold values. However, transmission ranges of 90 meters and 80 meters present a better reduction in area coverage. The reason is that with high transmission ranges in close distances between the source and the destination, some unnecessary part of the topology is covered. In other words, the data is relayed to some far way nodes from the sinks. Applying a threshold value prevents the far way nodes from the sink to be a relay node. When a smaller transmission range is used, the distance between the source and the destination is divided more equally. This is similar to occupy a square shape area with smaller square shape areas.

It is clear that increasing the threshold value reduces the covered area by relay nodes. It should be pointed out that there is a great coverage area reduction (between 77–93%) even with a 50% threshold value. Increasing the threshold value higher than 50% causes less reduction in coverage area. In simulations, it is found out that 50% threshold value provides a high reliability. Therefore, 50% threshold value is selected as default parameter.

The inference related with range and threshold value described above is seen clearly in Figure 13(b). In this figure, effects of range and threshold value are shown together. These are the results for 100 meters distance between the source and the destination. As seen, as the threshold value increases, the energy gain increases. However, energy gain gets higher as the transmission range increases. The reason of the erratic part for ranges 90 and 80 is as described above.

*6.2.4. Energy Consumption per Data Delivery.* We also measure the energy consumption on transportation of one data packet from source to destination node. Multiple receptions at the destination for the same data packet are counted as one successful delivery. Table 3 shows the comparative energy consumptions for different path lengths for Scenario 1. In flooding, equal amount of energy is consumed for each path length, since the packet is flooded within the network each time. However, for all other protocols, the number of transmissions and receptions varies according to the path length.

TABLE 3: Comparisons of the protocols with respect to energy consumption per data delivery.

Routing protocols	Energy consumption (joule)					
	1 hop	2 hop	3 hop	4 hop	Arithmetic average	Average by simulation
Flooding	1545.6	1545.6	1545.6	1545.6	1545.60	1043.01
GPSR 1 sec	1563.8	1582	1600.2	1618.4	1591.10	1306.99
GPSR 3 sec	533.4	551.6	569.8	588	560.70	519.71
GPSR 6 sec	275.8	294	312.2	330.4	303.10	259.70
SPEED	1582	1618.4	1654.8	1691.2	1636.60	1341.45
IGF	54.6	109.2	163.8	218.4	136.50	121.90
SWR	18.2	63.7	109.2	163.8	88.73	104.40
Optimal	18.2	36.4	54.6	72.8	45.50	36.52

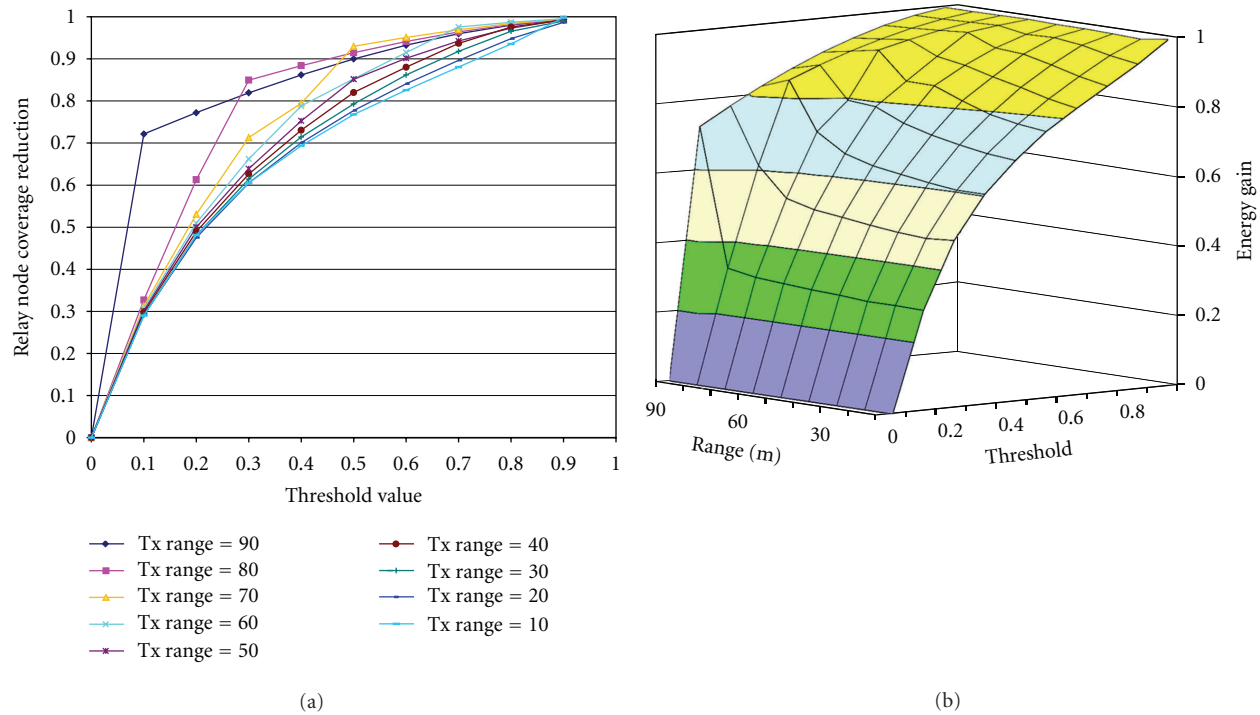


FIGURE 13: (a) Effects of range and threshold value on the relay nodes coverage in SWR. (b) Effects of threshold and range values on the energy consumption in SWR.

GPSR and SPEED use periodic beaconing to construct and update the neighborhood topology. They need this process to relay the data packets. Therefore, energy consumption for beaconing is considered and added to the energy consumption in GPSR and SPEED protocols. As expected, GPSR, SPEED, and flooding have higher energy consumptions. In GPSR and SPEED, beaconing consumes excessive energy. As the beaconing period extends, energy consumption reduces. For these protocols, packet forwarding consumes much less energy with respect to other protocols because shortest paths are constructed in GPSR and SPEED due to priori known topology. Beaconing generates the major energy consumption in these protocols.

Beaconless protocols consume less energy for routing. Compared to the energy consumption of these protocols, the order from high to low is IGF and SWR. Energy consumption in IGF for one data packet is higher than SWR because of

its probe packets before sending data packets. Energy consumption in SWR is higher than the Optimal Routing, but data is sent over multiple paths in SWR providing reliability. SWR provides reliability and minimum delay with shortest path by using multiple transmissions and utilizing multiple paths, while there is only one path prone to the failures in all other protocols except flooding. Energy consumption in SWR is very low and better than all protocols except the Optimal Routing.

*Average during Simulation* values are the averages of the results obtained in Scenario 1. There is difference between the *Average during Simulation* and *Arithmetic Average*. There are two reasons for this difference. Arithmetic Average is the average of energy consumptions in different path lengths. But the number of constructed paths for different path lengths cannot be the same during the simulations. This affects the average energy per path. Secondly, protocols continue to find

TABLE 4: Comparisons of the protocols with respect to transmissions and receptions per data delivery.

Routing protocols	Number of transmissions and receptions			
	1 hop	2 hops	3 hops	4 hops
Flooding	100 tx/1004 rx	100 tx/1004 rx	100 tx/1004 rx	100 tx/1004 rx
GPSR 1 sec	101 tx/1016 rx	102 tx/1028 rx	103 tx/1040 rx	104 tx/1052 rx
GPSR 3 sec	34 tx/347 rx	35 tx/359 rx	36 tx/371 rx	37 tx/383 rx
GPSR 6 sec	18 tx/179 rx	19 tx/191 rx	20 tx/203 rx	21 tx/215 rx
SPEED	102 tx/1028 rx	104 tx/1052 rx	106 tx/1076 rx	108 tx/1100 rx
IGF	3 tx/36 rx	6 tx/72 rx	9 tx/108 rx	12 tx/144 rx
SWR	1 tx/12 rx	3 tx/42 rx	6 tx/72 rx	9 tx/108 rx
Optimal	1 tx/12 rx	2 tx/24 rx	3 tx/36 rx	4 tx/48 rx

routes even if there are some terminated nodes. However, their responses to node terminations vary in terms of lifetime and ability to find routes. For example, flooding has the ability to find routes even if there are many node terminations. In Scenario 1, as shown previously in Table 2, flooding continues to find routes even when there are many node terminations. Supporting results are observed in Table 4. Flooding always makes the same amount of transmissions and receptions if there are not any node terminations. However, node terminations during the simulation reduce the number of transmissions and receptions in flooding. Similar results are observed for beacon-based protocols, GPSR and SPEED. Terminated nodes reduce the system-wide energy consumption in beaconing. Depending on the beaconing period, results change for GPSR 3 sec and GPSR 6 sec.

Scalability can be considered through multiple perspectives. It can be evaluated with regard to an increase in the sensor network area, the density of the nodes, the traffic, and so forth. The impact of all issues is minimized in the virtual optimal algorithm. Simulations indicate that the SWR is not only a better performing algorithm compared to the others in the literature but also quite close to the virtual optimal one. Main reason is that the SWR consumes the energy only on transmissions where and when events occur. Periodic data packet transmissions can also cause performance degradation as observed in other protocols. Considering the event-based data packet transmissions, the scalability of the SWR is very close to *virtual optimal routing* protocol as shown in Figures 10 and 11 and Tables 3 and 4.

The SWR is essential and imperative for Mission-Critical applications which require high reliability. The SWR is also suitable for event-based data acquisition networks rather than periodic data acquisition networks.

## 7. Conclusions

In this paper, a novel stateless routing algorithm for WSN, the SWR, is proposed. The SWR differs from other proposed protocols in the literature in many ways. It is a completely stateless routing protocol that does not require any topology knowledge on data transmissions. As it provides reliability by conveying data over multiple paths, the SWR reduces

transmissions and energy consumption drastically by avoiding transmissions on topology learning and control information exchange. Implicit features of the SWR provide nodes to adapt current conditions and to recover from voids. Moreover, an explicit void recovery is proposed to recover from voids in case of a node cannot recover a void by the implicit approach. Each node makes its own decision in a smart way according to its own conditions. In the SWR, data packets spontaneously flow over simultaneous multiple paths. This approach provides utilization of the shortest available path. Stateless property of the SWR helps to reduce the delay at nodes. These features are provided with a simple algorithm that does not require much resource such as CPU and memory. Assuring the use of cheap and disposable nodes for WSNs applications, the SWR can be used in instantly topology changing networks as well as with stationary ones. These features make the SWR a unique one considering the current literature. The measurement of delay is considered as a future work.

## References

- [1] S. Giordano, I. Stojmenovic, and L. Blazevic, "Position based routing algorithms for Ad Hoc networks: a taxonomy," in *Ad Hoc Wireless Networking*, X. Cheng, X. Huang, and D. Z. Du, Eds., pp. 103–136, Kluwer Academic, Boston, Mass, USA, 2004.
- [2] M. Heissenbüttel, T. Braun, M. Wälchli, and T. Bernoulli, "Optimized stateless broadcasting in wireless multi-hop networks," in *Proceedings of 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, pp. 1–12, April 2006.
- [3] M. Mauve, J. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile Ad Hoc networks," *IEEE Network*, vol. 15, no. 6, pp. 30–39, 2001.
- [4] F. Araujo and L. Rodrigues, "Survey on position-based routing," Tech. Rep. TR-01, University of Lisbon, 2006.
- [5] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325–349, 2005.
- [6] M. Soyuturk and T. Altılar, "The challenges and the approaches for the geographic routing protocols in wireless sensor networks," in *Proceedings of the IEEE International Conference on Technologies for Homeland Security and Safety (IEEE TEHOSS '06)*, October 2006.

- [7] S. Dulman, T. Nieberg, J. Wu, and P. Havinga, "Trade-off between traffic overhead and reliability in multipath routing for wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking (WCNC '03)*, pp. 1918–1922, March 2003.
- [8] U. Monaco, F. Cuomo, T. Melodia, F. Ricciato, and M. Borghini, "Understanding optimal data gathering in the energy and latency domains of a wireless sensor network," *Computer Networks*, vol. 50, no. 18, pp. 3564–3584, 2006.
- [9] H. Fuessler, J. Widmer, M. Kasemann, and M. Mauve, "Beaconless position-based routing for mobile Ad-Hoc networks," Tech. Rep., Department of Computer Science, University of Mannheim, 2003.
- [10] M. Heissenbuttel and T. Braun, "A novel position-based and beacon-less routing algorithm for mobile Ad Hoc networks," in *Proceedings of 3rd Workshop on Applications and Services in Wireless Networks (ASWN '03)*, pp. 197–210, Berlin, Germany, 2003.
- [11] M. Chawla, N. Goel, K. Kalaichelvan, A. Nayak, and I. Stojmenovic, "Beaconless position based routing with guaranteed delivery for wireless ad-hoc and sensor Networks," *IFIP International Federation for Information Processing*, vol. 212, pp. 61–70, 2006.
- [12] H. Xiaoyan, X. Kaixin, and M. Gerla, "Scalable routing protocols for mobile Ad Hoc networks," *IEEE Network*, vol. 16, no. 4, pp. 11–21, 2002.
- [13] Y. B. Ko and N. H. Vaidya, "Location-aided routing (LAR) in mobile Ad Hoc networks," *Wireless Networks*, vol. 6, no. 4, pp. 307–321, 2000.
- [14] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A Distance routing effect algorithm for mobility (DREAM)," in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking of (MOBICOM '98)*, pp. 76–84, 1998.
- [15] E. Kranakis, H. Singh, and J. Urrutia, "Compass routing in geometric graphs," in *Proceedings of 11th Canadian Conference on Computational Geometry (CCCG '99)*, pp. 51–54, 1999.
- [16] T. Melodia, D. Pompili, and I. F. Akyildiz, "On the interdependence of distributed topology control and geographical routing in Ad Hoc and sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 3, pp. 520–531, 2005.
- [17] L. Savidge, H. Lee, H. Aghajan, and A. Goldsmith, "Event-driven geographic routing for wireless image sensor networks," in *Proceedings of the Proceedings of Cognitive Systems and Interactive Sensors (COGIS '06)*, March 2006.
- [18] S. Lee, B. Bhattacharjee, and S. Banerjee, "Efficient geographic routing in multihop wireless networks," in *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '05)*, pp. 230–241, May 2005.
- [19] N. Ahmed, S. S. Kanhere, and S. Jha, "The holes problem in wireless sensor networks: a survey," in *ACM SIGMOBILE Mobile Computing and Communications Review (MCR '09)*, pp. 4–18, April 2005.
- [20] Q. Fang, J. Gao, and L. J. Guibas, "Locating and bypassing routing holes in sensor networks," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, pp. 2458–2468, March 2004.
- [21] S. Chen, G. Fan, and J.-H. Cui, "Avoid "Void" in geographic routing for data aggregation in sensor networks," *International Journal of Ad Hoc and Ubiquitous Computing Archive*, vol. 1, no. 4, pp. 169–178, 2006.
- [22] D. Son, A. Helmy, and B. Krishnamachari, "The effect of mobility-induced location errors on geographic routing in mobile Ad Hoc and sensor networks: analysis and improvement using mobility prediction," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 233–245, 2004.
- [23] B. Karp and H.T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '00)*, pp. 243–254, Boston, Mass, USA, August 2000.
- [24] H. Tian, J. A. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: a stateless protocol for real-time communication in sensor networks," in *Proceedings of the 23rd International Conference on Distributed Computing Systems*, pp. 46–55, May 2003.
- [25] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile Ad Hoc networks," *Ad Hoc Networks*, vol. 1, no. 4, pp. 351–369, 2003.
- [26] B. Blum, T. He, S. Son, and J. Stankovic, "IGF: a state-free robust communication protocol for wireless sensor networks," Tech. Rep., University of Virginia, Charlottesville, Va, USA, 2003.
- [27] M. Soyuturk and D. T. Altılar, "Stateless data flow approach with void avoidance for wireless Ad Hoc and sensor networks," in *Proceedings of the 2nd International Symposium on Wireless Pervasive Computing (ISWPC '07)*, pp. 252–257, February 2007.
- [28] J. B. Schmitt, F. A. Zdarsky, and U. Roedig, "Sensor network calculus with multiple sinks," in *Proceedings of the Workshop on Performance Control in Wireless Sensor Networks (IFIP NETWORKING '06)*, Lecture Notes in Computer Science, pp. 6–13, Springer, Albacete, Spain, September 2006.
- [29] K. Yuen, B. Liang, and B. Li, "A distributed framework for correlated data gathering in sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 1, pp. 578–593, 2008.
- [30] N. Antunes, G. Jacinto, and A. Pacheco, "An analytical framework to infer multihop path reliability in MANETs," in *Proceedings of the International Conference on Measurement and Modeling of Computer Systems (ACM SIGMETRICS '10)*, pp. 323–332, 2010.
- [31] T. Korkmaz and K. Sarac, "Characterizing link and path reliability in large-scale wireless sensor networks," in *Proceedings of the 6th Annual IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '10)*, pp. 217–224, October 2010.
- [32] A. Tsigirigos and Z. J. Haas, "Analysis of multipath routing—part I: the effect on the packet delivery ratio," *IEEE Transactions on Wireless Communications*, vol. 3, no. 1, pp. 138–146, 2004.
- [33] W. Yang, X. Yang, S. Yang, and D. Yang, "A greedy-based stable multi-path routing protocol in mobile Ad Hoc networks," *Ad Hoc Networks*, vol. 9, pp. 662–674, 2011.
- [34] E. I. Oyman and C. Ersoy, "Overhead energy considerations for efficient routing in wireless sensor networks," *Computer Networks*, vol. 46, no. 4, pp. 465–478, 2004.
- [35] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 364–369, April 2005.
- [36] A. Caracas et al., "Energy-efficiency through micro-managing communication and optimizing sleep," in *Proceedings of 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '11)*, 2011.

- [37] S. Tozlu, "Feasibility of Wi-Fi enabled sensors for internet of things," in *Proceedings of the 7th IEEE International Wireless Communications and Mobile Computing Conference (IWCMC '11)*, 2011.
- [38] S. Mueller, R. P. Tsang, and D. Ghosal, "Multipath routing in mobile Ad Hoc networks: issues and challenges," in *Proceedings of the International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '03)*, vol. 2965 of *Lecture Notes in Computer Science*, pp. 209–234, 2003.
- [39] C. Avin, "Fast and efficient restricted delaunay triangulation in random geometric graphs," in *Proceedings of the Workshop on Combinatorial and Alg. Aspects of Networking (CAAN '05)*, 2005.
- [40] Y. Yu, B. Hong, and V. K. Prasanna, "On communication models for algorithm design in networked sensor systems: a case study," *Pervasive and Mobile Computing*, vol. 1, no. 1, pp. 95–121, 2005.