

## Research Article

# Heuristic Approach to Select Opportunistic Routing Forwarders (HASORF) to Enhance Throughput for Wireless Sensor Networks

Yousaf Bin Zikria,<sup>1</sup> Summera Nosheen,<sup>2</sup> Jin-Ghoo Choi,<sup>1</sup> and Sung Won Kim<sup>1</sup>

<sup>1</sup>Department of Information and Communication Engineering, Yeungnam University, Gyeongsangbuk-do, Gyeongsan-si 712-749, Republic of Korea

<sup>2</sup>School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6140, New Zealand

Correspondence should be addressed to Sung Won Kim; swon@yu.ac.kr

Received 13 November 2014; Revised 11 February 2015; Accepted 20 February 2015

Academic Editor: Bo-Wei Chen

Copyright © 2015 Yousaf Bin Zikria et al. 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.

Biological schemes provide useful resources for designing adaptive routing protocols for wireless sensor networks (WSNs). The key idea behind using bioinspired routing is to find the optimal path to the destination. Similarly, the idea of opportunistic routing (OR) is to find the least number of hops to deliver the data to the destination. Numerous routing schemes have been proposed in WSNs while targeting various performance goals, such as throughput, delay, and link quality. Recently, OR schemes have come onto the scene in comparison with the traditional routing algorithms. The performance of OR schemes, however, highly depends on the selection of forwarder nodes. In this paper, we consider a chain network topology, where nodes are separated by an equal distance. The throughput of the chain network is analyzed mathematically, and based on the analysis results, a heuristic algorithm is proposed to choose the forwarder nodes. We evaluate the performance of the proposed Heuristic Approach to Select Opportunistic Routing Forwarders (HASORF) by using the ns-2 simulator and compare it with previous schemes, such as random routing, Extremely Opportunistic Routing (ExOR), and Simple Opportunistic Adaptive Routing (SOAR). The empirical results show that our proposed scheme achieves the best performance among them.

## 1. Introduction

Biological systems have intrinsic appealing characteristics [1]. These characteristics lead to different levels of motivation from the biological system toward the origin of different approaches and algorithms. Opportunistic routing (OR) is similar to a bioinspired mechanism [2], where one of the goals is to reach the destination using the shortest path. The advent of bioinspired systems has generated many contributions and inspired systems instigated from natural systems and their application in WSNs. Applications for WSNs include wildlife monitoring, cold chain monitoring, glacier monitoring, rescue of avalanche victims, cattle herding, geographical monitoring, monitoring of structures, vital sign monitoring, ocean water and ocean bed monitoring, monitoring of fresh water quality, tracking vehicles, sniper localization, volcano monitoring, and tunnel monitoring. The underwater sensor network, normally built on ultrasound, is also one of

the applications for WSNs [3, 4]. Examples of real-world projects with wireless sensor networks are bathymetry [5], ocean water monitoring [6], ZebraNet [7], cattle herding [8], bird observation on Great Duck Island [9], grape monitoring [10], rescue of avalanche victims [11], and neuRFon [12].

OR is vigorous and well suited to WSNs. In WSNs, nodes have irregular connectivity and accessibility to the neighbor nodes for packet forwarding is disrupted. The neighbor sensors can be exploited as opportunistic elements for packet forwarding. OR is based on the idea of geographic routing. It exploits location information, but the selection of the forwarder nodes may vary according to the protocol used. In WSNs, one of the challenges is to design a routing protocol to meet the communication demands in less time. Moreover, in WSNs, forwarding packets to the intended destination is in multihop fashion, where intermediate nodes help packets reach the destination. One of the key applications of sensors is the wireless personal, or body, network for

health monitoring. It aims to improve existing healthcare and monitoring services. In a short time, there has been immense development in the number of various wearable health monitoring devices, ranging from simple pulse monitors, activity monitors, portable Holter monitors, and cardiac monitors [13, 14].

In traditional routing, packets are forwarded on a hop-by-hop basis via predetermined intermediate nodes to the destination, based on previously established routes and statistics. The next-hop node can be selected by a simple shortest-path algorithm, or by more complicated optimizations, such as considering the channel conditions and the performance of the network links. The wireless channel is inherently broadcast in nature, and all the nodes within range of the sender can overhear the transmission. OR [15–20] takes advantage of this by choosing intermediate nodes as helper nodes to transmit data to the destination. Therefore, they are called forwarders in opportunistic routing. The selection of the forwarder nodes is the most important task in OR. The literature [17, 19–22] has proven that OR enhances throughput in multihop wireless networks.

The packets can be received by multiple nodes within range. Among the nodes that receive the packet, the node closest to the destination should be the one that forwards the packet. OR can choose multiple forwarders from among the intermediate nodes. However, in denser networks, there is a cost penalty from choosing too many forwarders. Choosing a small number of forwarders can minimize the potential cost of intra- and interpath collisions. However, if the number of forwarders is large, collisions can become frequent [19, 23].

The existing OR schemes leverage the forwarder selection mechanism of expected transmission (ETX) count [24]. Choosing the forwarder(s) is critical to overall performance of the network. The routing protocol can be subdivided into three parts: route discovery, packet forwarding, and route maintenance. In OR, selection of forwarding nodes is part of route discovery. The wrong selection of the forwarder nodes severely impacts the overall performance of OR. The impact on multihop transmission is much more devastating than for single-hop transmission. In this research, we focus on the first task, that is, route discovery, and specifically on forwarder selection. The selection of the intermediate nodes as forwarders is still an unexplored research area. The main objective of our study is to enhance system throughput and devise a new approach to selecting forwarders. For this purpose, we first analyze and evaluate equal, or uniform, distance node placement in the network and then determine the impact of forwarder selection on throughput. Thereafter, on the basis of the system model, we propose a heuristic approach to selecting the forwarders. We then compare the results with Extremely Opportunistic Routing (ExOR), random routing, and Simple Opportunistic Adaptive Routing (SOAR) forwarder selection schemes. The results indicate improvement in system throughput.

The rest of the paper is organized as follows. Section 2 briefly explains the related work. Section 3 presents the system model. In Section 4, we describe our proposed Heuristic Approach to Select Opportunistic Routing Forwarders (HASORF). Section 5 provides the details of the simulation

environment and discusses the results. Finally, Section 6 concludes the paper.

## 2. Related Work

In this section, we discuss the different opportunistic routing protocols while focusing on the forwarder selection methodology employed therein. Biswas and Morris [16] proposed ExOR. It is the most popular opportunistic routing protocol and a pioneer in exploiting the broadcast nature of wireless communications. In ExOR, the sender must include in the header of each packet the forwarder list, prioritized by closeness to the destination. Thus, a candidate forwarder set is predetermined by each sender. Simple Opportunistic Adaptive Routing [18] tries to solve the lack of support for multiple simultaneous flows in ExOR by introducing an explicit forwarding responsibility. Similar to ExOR, SOAR has a predetermined list of candidate forwarders based on the ETX metric. The forwarder list is included in the packet header and is also prioritized by closeness to the destination. SOAR performs better than ExOR. However, that is not due to the forwarder selection algorithm but the implicit duplicate transmission avoidance mechanism based on diverging route prevention. Furthermore, it also implements a selective and piggybacked acknowledgement mechanism for higher performance.

Chachulski et al. [22] presented media access control (MAC-) independent opportunistic routing and encoding (MORE), which resembles ExOR in many ways. Both protocols implement a predetermined candidate selection process based on the ETX metric. Moreover, both include a forwarder list in the packet header, prioritize the forwarder nodes by the distance to the destination, and limit the candidate size to reduce overhead. The main difference between MORE and ExOR is that each packet sent by MORE is a coded packet. Yuan et al. [25] presented the resilient opportunistic mesh routing (ROMER) protocol, which introduces a credit-based forwarding scheme similar to that of SOAR. The assigned credits are equal to the minimum cost from source to destination, that is, the shortest-path cost. Thus, ETX is the de facto standard employed in all opportunistic routing protocols explained above. However, frequent ETX measurement involves significant overhead in networks, and the links with a lower ETX may lead to a higher loss rate, yielding low throughput. In this paper, we propose a new forwarder selection scheme that does not include any extra overhead but substantially increases overall system throughput. Our scheme uses distance as a selection metric for forwarder nodes and maintains higher throughput compared to the other schemes.

## 3. System Modeling

In this section, we consider a chain network shown in Figure 1, where  $N + 1$  nodes are separated by the same distance. Node 0 acts as the source  $s$ , and node  $N$  becomes destination  $d$ . The nodes from 1 to  $N - 1$  are intermediate nodes and can be potential forwarders  $f$ . Forwarders can help

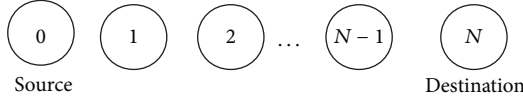


FIGURE 1: System model.

the packets from the source reach the destination. We assume that one or more nodes can play the role of forwarder node.

The key metric to choose the forwarder nodes is ETX from previous schemes, which prioritizes the nodes by closeness to the destination, because successful transmission probability decreases with a greater distance. When the distance between transmitter and receiver is long, it is likely to have more packet loss and more retransmissions. Thus, if we choose a node close to the destination as a forwarder node, we will experience poor performance in transmissions from the source to the forwarder, whereas transmissions from the forwarder to the destination are successful with high probability. This forwarder selection scheme is inefficient at fully utilizing bandwidth and achieving greater throughput. In the sequel, we analyze the chain network to obtain some intuition into how to choose the best forwarder to attain optimal throughput.

We begin our analysis from the definition of throughput in the considered system; that is,

$$\text{Throughput} := \frac{M \times S}{Q_1 + Q_2 + \dots + Q_M}, \quad (1)$$

where  $M$  is the number of total transmitted packets,  $S$  is the packet size, and  $Q_i$  is delivery time for the  $i$ th packet to reach the destination. We assume that the next packet is not transmitted from the source node until the current packet reaches the destination successfully. We can further simplify the equation by the law of large numbers [26] since  $Q_i$  are independent and identically distributed (i.i.d) random variables:

$$\begin{aligned} \text{Throughput} &= \frac{M \cdot S}{Q_1 + Q_2 + \dots + Q_M} \\ &= \frac{S}{(Q_1 + Q_2 + \dots + Q_M)/M} = \frac{S}{E[Q]}, \end{aligned} \quad (2)$$

where  $E[Q]$  denotes the expectation of random variable  $Q$ . Since the packet size  $S$  is constant, throughput is inversely proportional to the number of transmissions from source to destination of an arbitrary packet; that is,

$$\text{Throughput} \propto (\text{Expected delivery time of each packet})^{-1}. \quad (3)$$

We can decouple the total delivery time of a packet as the sum of two partial delivery times, that is,  $T_{s \rightarrow f}$ , from source to any forwarder, and  $T_{f \rightarrow d}$ , from the forwarder with the packet to the destination, that is,

$$\text{Delivery time of a packet} = T_{s \rightarrow f} + T_{f \rightarrow d}. \quad (4)$$

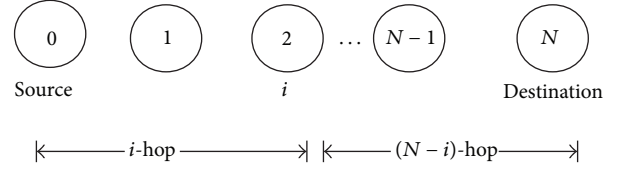


FIGURE 2: Single-forwarder network topology.

By taking expectations on both sides, we can obtain the expected delivery time as

$$\begin{aligned} \text{Expected delivery time of each packet} &= E[T_{s \rightarrow f}] \\ &\quad + E[T_{f \rightarrow d}]. \end{aligned} \quad (5)$$

We are now prepared to derive the throughput of the considered network. We consider two cases here: in Case 1, the number of forwarders is one, and in Case 2, the number of forwarders is two. However, our analysis can be extended to more forwarders in a straightforward manner.

**3.1. Single/One Forwarder.** We denote the forwarder node as node  $i$ . So, the hop distance from source to forwarder is  $i$ , and the hop distance from forwarder to destination is  $N - i$ , as shown in Figure 2.

We first introduce some notations. Given distance  $x$  between transmitter and receiver,  $p(x)$  denotes packet success probability, which is assumed to be differentiable and monotonically decreases with  $x$ . Note that the assumptions on  $p(x)$  are reasonable and practically mild. It is straightforward to see that when packet success probability is  $p(x)$ , the average delivery time or the average number of transmissions is given as  $1/p(x)$ . So, recalling that the forwarder is located at the  $i$ th position, we can see that

$$E[T_{s \rightarrow f}] = \frac{1}{p(i)}, \quad E[T_{f \rightarrow d}] = \frac{1}{p(N-i)}. \quad (6)$$

If every packet is delivered to the destination via the forwarder node, the average delivery time is  $1/p(i) + 1/p(N-i)$ . However, we notice that packets reach the destination without the help of the forwarder node with probability  $p(N)$ . So, in the single forwarder case, the average delivery time from source to destination is written as

$$T(i) = p(N) \cdot 1 + (1 - p(N)) \left( \frac{1}{p(i)} + \frac{1}{p(N-i)} \right) \quad (7)$$

and throughput is given as

$$\text{Throughput} \propto (T(i))^{-1}. \quad (8)$$

We temporarily relax hop count  $i$  as real numbers. Let us take the differentiation of  $T(i)$  with respect to  $i$  and set it to zero to find the minimum value. It is easy to find that  $T(i)$  is minimized (or throughput is maximized) at  $N/2$ ; that is, the optimal forwarder is the node midway between source and destination. This intuition is crucial to developing the forwarder selection algorithm in the next section.

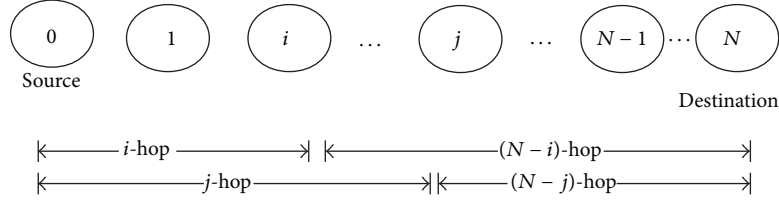


FIGURE 3: Two-forwarder network topology.

3.2. *Two Forwarders.* In this scenario, we select two intermediate nodes as forwarders (Figure 3). Let node  $i$  and node  $j$  be designated as the forwarders. We can calculate expected delivery time from source  $s$  to forwarder  $f$  as

$$E[T_{s \rightarrow f}] = \frac{1}{1 - (1 - p(j))(1 - p(i))}. \quad (9)$$

Then, we calculate the delivery time from forwarder  $f$  to destination  $d$ . We here need to consider two cases. To that end, we define the term “effective forwarder” as the node that successfully receives a packet.

- (1) Node  $j$  is the effective forwarder, which happens with probability  $p(N - j)$ :

$$E[T_{f \rightarrow d}] = \frac{p(j)}{[1 - (1 - p(j))(1 - p(i))]} \quad (10)$$

- (2) Node  $i$  is the effective forwarder, which happens with probability  $p(N - i)$ :

$$E[T_{f \rightarrow d}] = \frac{(1 - p(j))p(i)}{[1 - (1 - p(j))(1 - p(i))]} \quad (11)$$

Now the delivery time from forwarder  $f$  to destination  $d$  is given as

$$E[T_{f \rightarrow d}] = \frac{p(j)}{[1 - (1 - p(j))(1 - p(i))]p(N - j)} + \frac{(1 - p(j))p(i)}{[1 - (1 - p(j))(1 - p(i))]p(N - i)}. \quad (12)$$

In conclusion, the expected delivery time from source to destination is given as

$$T(i, j) = \frac{1}{1 - (1 - p(j))(1 - p(i))} + \frac{p(j)}{[1 - (1 - p(j))(1 - p(i))]p(N - j)} + \frac{(1 - p(j))p(i)}{[1 - (1 - p(j))(1 - p(i))]p(N - i)} \quad (13)$$

and throughput is

$$\text{Throughput} \propto (T(i, j))^{-1}. \quad (14)$$

## 4. HASORF

Algorithm 1 is the proposed algorithm to choose a single OR forwarder. We assume that the distance between two consecutive nodes is equal. Our heuristic algorithm works as follows. At the start, it gets the distance between source and destination (line 1). After that, it gathers the total number of nodes in the network (line 2). It creates a candidate list by excluding the source and the destination (line 3). Next, it assigns node identities and positions of candidates (line 4). Afterwards, if the number of forwarders is 1 (line 5), it determines the potential forwarder by getting half of the distance (line 6). Then, it searches the candidate list to match the distance and gets the identity of the matched node (lines 7-8). Thereafter, it chooses this node as the forwarder node to transmit the packet from source to destination (line 9).

Algorithm 2 is the proposed HASORF algorithm when the number of forwarders is more than one. It follows the same procedure as Algorithm 1 until it gets the source and destination distance and the total number of nodes in the network. After that, it creates a candidate list and stores node identities and positions. Next, it calculates the total number of forwarders in the network (line 1). Then, it creates the candidate list according to the number of forwarders and selects the intermediate nodes as forwarders. Next, it searches to match the distance and node ID positions of the candidates (lines 4–8). Later, these candidates act as forwarders (line 9) and help the source to deliver packets to the destination.

## 5. Performance Evaluation

To validate our proposed scheme, we conduct an extensive set of simulations by using the network simulator, ns-2.30 [27]. We use a chain topology [28, 29] for our experimentation. The packet size is set to 1000 bytes, and the interface queue is configured as 50 packets. The physical data rate is 216 Mbps, and the physical basic rate is 54 Mbps. The simulation area is a square 1000 m  $\times$  1000 m. We focus on improving the end-to-end throughput of the network by choosing the best forwarder nodes. Throughput is the total number of bytes received per second at the destination, measured in megabits per second. Table 1 shows the key design choices of different forwarder selection schemes in opportunistic routing.

We consider two network topologies to conduct the simulation. In the first network, there are eight nodes, as shown in Figure 4. It contains six intermediate nodes that can be forwarders. The source transmits data to the destination by choosing some intermediate nodes as the forwarders.

```

(1) Get Distance = getdistance
(2) Get Total Nodes = get_nodes
(3) Assign Candidate_list = Exclude_S_D(getnodes)
(4) Assign Forwarder_List_id_Pos = get_nodes.id_pos(Candidate_list)
(5) if No_of_Forwarders == 1
(6)   Assign Candidate = round(Distance/2)
(7)   for i = 0 to length(Forwarder_List_id_Pos)
(8)     Match Forwarder_list_id_pos[i] == candidate
(9)     Assign Forwarder = Forwarder_list_id_pos[i]
(10)  end for
(11) end if

```

ALGORITHM 1: HASORF (when number of forwarders is 1).

```

(1) Assign No_of_Forwarders == get_forwarders
(2) Assign j = No_of_Forwarders + 1
(3) for i = 0 to No_of_Forwarders
(4)   Assign candidate[i] = getdistance(round(i/j))
(5)   for k = 0 to length(Forwarder_List_id_Pos)
(6)     Match Forwarder_list_id_pos[k] == candidate[i]
(7)     Assign Forwarder[i] = Forwarder_list_id_pos[k]
(8)   end for
(9) end for

```

ALGORITHM 2: HASORF (when number of forwarders is more than 1).

TABLE 1: Key design choices.

Routing	Forwarder selection metric	Coding
Opportunistic	ExOR/ETX	No
Opportunistic	SOAR/ETX	No
Opportunistic	MORE/ETX	Yes
Opportunistic	ROMER/credit-based (similar to ETX)	No
Opportunistic	HASORF/distance	No
Opportunistic	Random	No

Figure 5 compares the throughput of HASORF, ExOR, random, and SOAR forwarder selection schemes. The random forwarder selection scheme chooses any intermediate node as a forwarder, whereas ExOR picks up the nodes closest to the destination. SOAR also selects nodes closest to the destination. Our suggested scheme selects the forwarders on the basis of distance. The results show that HASORF outperforms EXOR, random, and SOAR forwarder selection algorithms. EXOR and SOAR perform poorly because both ExOR and SOAR choose the forwarders based on the ETX metric. Note that both schemes attain almost the same results. The random forwarder selection scheme performs better than ExOR and SOAR because the selected forwarders turn out to give better throughput. The results also show that throughput increases with more forwarder nodes. However, more forwarders lead to more overhead. So, we restrict the maximum number of forwarders to three.

Figure 6 shows the throughput improvement of HASORF over ExOR. Initially, both forwarder schemes attain small

throughput. However, as the number of forwarders increases, throughput increases and more data is delivered successfully. It is clearly seen that our proposed scheme has comparatively better performance than ExOR.

Figure 7 illustrates the percentage improvement of HASORF over the random forwarder selection scheme. The throughput gain is about 24.5% to 122.61% for different numbers of forwarders.

Figure 8 illustrates the percentage throughput gain of HASORF over SOAR. With a single forwarder, both forwarder selection schemes fail to achieve higher throughput. Nevertheless, it shows improvement as the number of forwarders increases. However, our proposed forwarder selection scheme is still able to select the best forwarder and achieves better throughput than SOAR. Overall, in all cases, HASORF shows consistent results and secures greater throughput than the SOAR forwarder selection scheme.

The second network topology for our simulation is presented in Figure 9, which contains 14 nodes, including source and destination. The forwarders can be chosen from among the intermediate nodes according to the associated forwarder selection scheme.

Figure 10 illustrates throughput for varying numbers of forwarders in different forwarder selection schemes. SOAR attains better throughput than ExOR, but the gain is not significant. Both forwarder selection schemes are similar, and the only difference is that the forwarder nodes should be located on or nearby the shortest path in SOAR. The random forwarder selection performs worst among all the considered schemes. However, in the first experimental setup, it performs

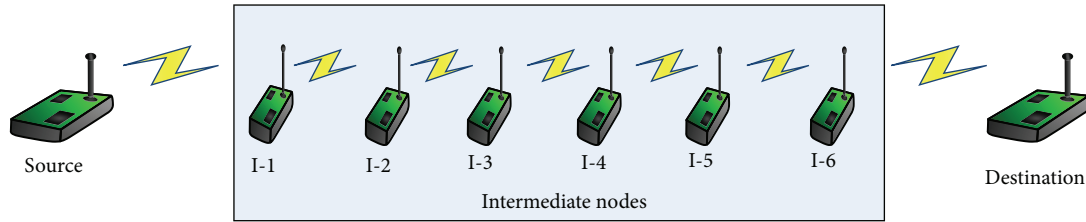


FIGURE 4: Network topology 1.

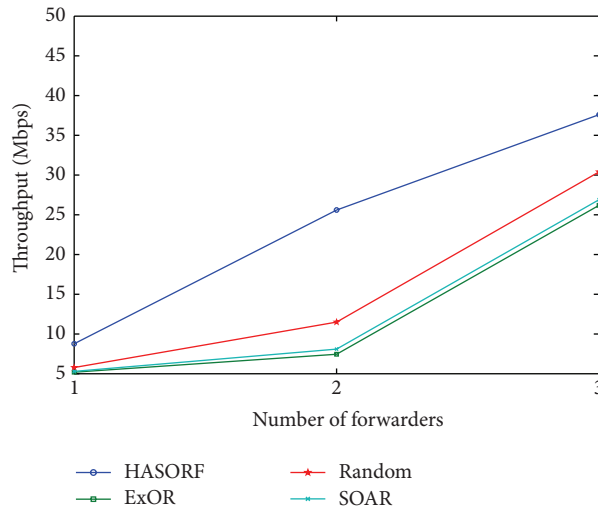


FIGURE 5: Number of forwarders and throughput for network topology 1.

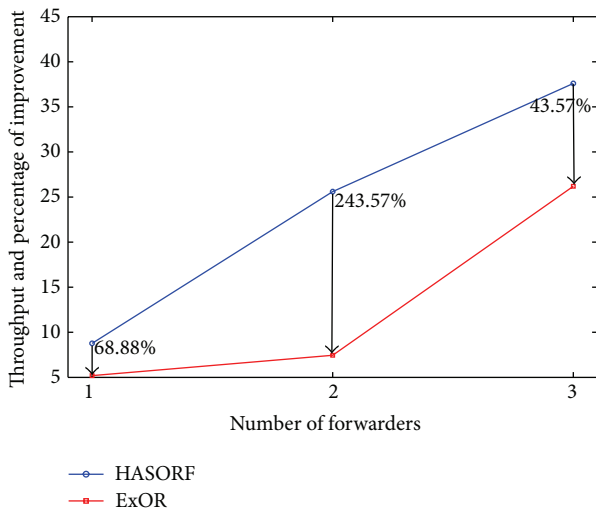


FIGURE 6: Percentage throughput improvement of HASORF and ExOR schemes for network topology 1.

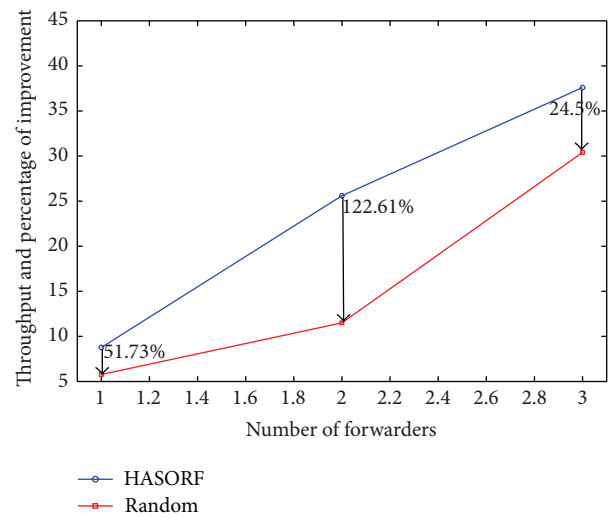


FIGURE 7: Percentage throughput improvement of HASORF and random schemes for network topology 1.

reasonably well. Hence, this scheme gives inconsistent results and is highly unpredictable. HASORF is the best scheme, because it shows consistent throughput and outperforms the other forwarder selection schemes.

Figures 11, 12, and 13 depict the percentage throughput improvement of HASORF over the ExOR, random, and

SOAR forwarder selection schemes, respectively. The results demonstrate that our proposed scheme performs better than any other scheme for throughput.

HASORF achieves the best throughput; that is, it transmits more bytes in less time. This indicates significant impact on the performance of WSNs. It not only delivers more data

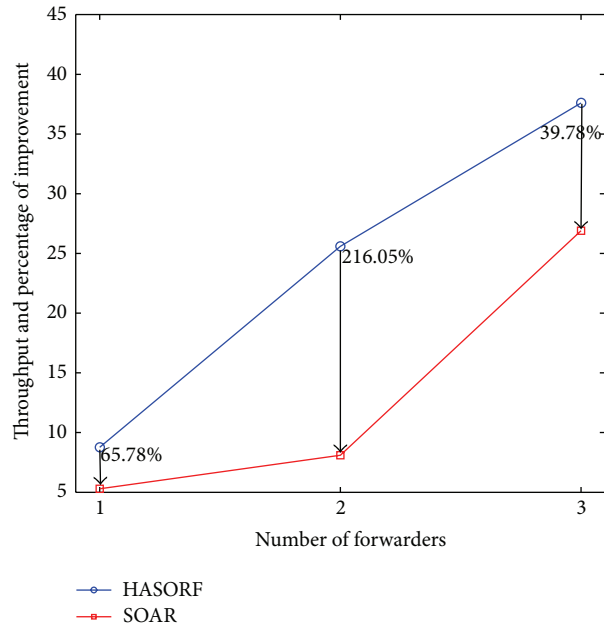


FIGURE 8: Percentage throughput improvement of HASORF and SOAR schemes for network topology 1.

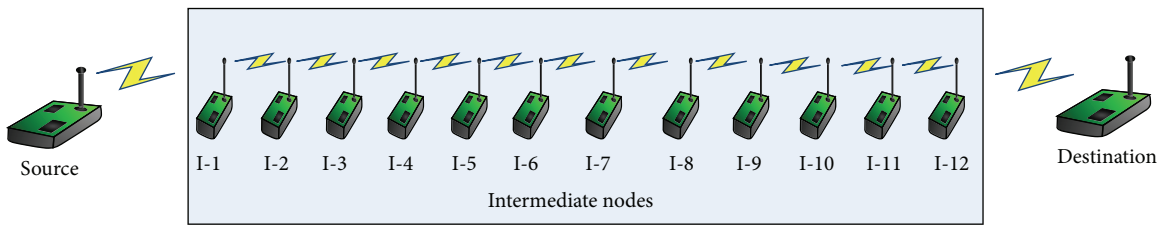


FIGURE 9: Network topology 2.

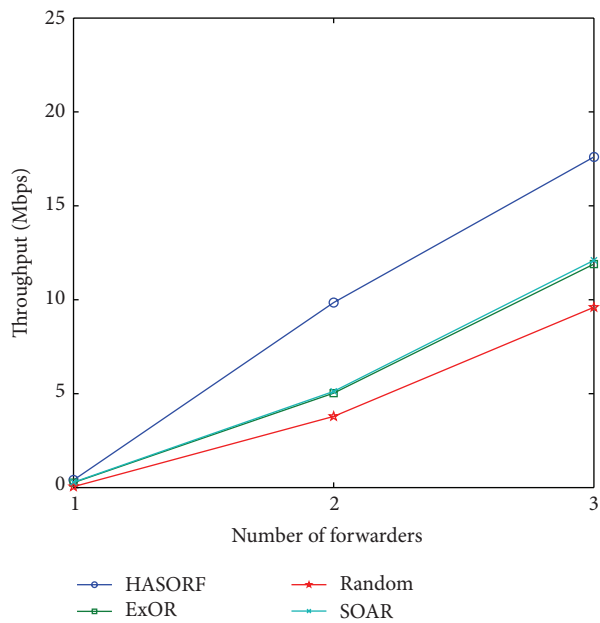


FIGURE 10: Number of forwarders and throughput for network topology 2.

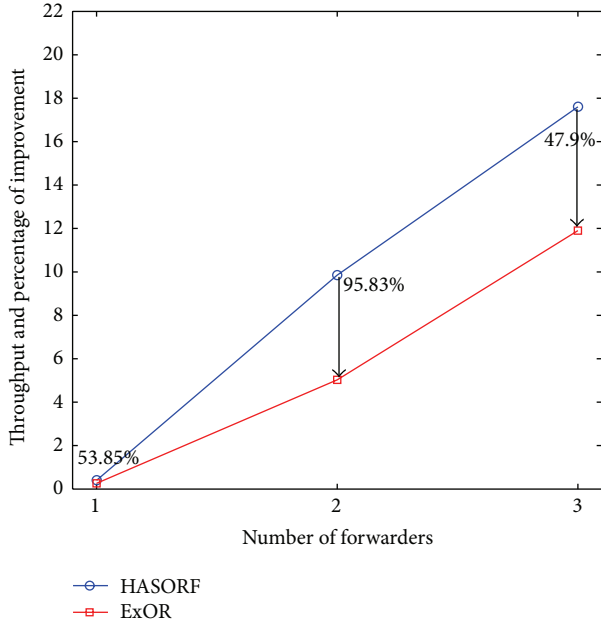


FIGURE 11: Percentage throughput improvement of HASORF and ExOR schemes for network topology 2.

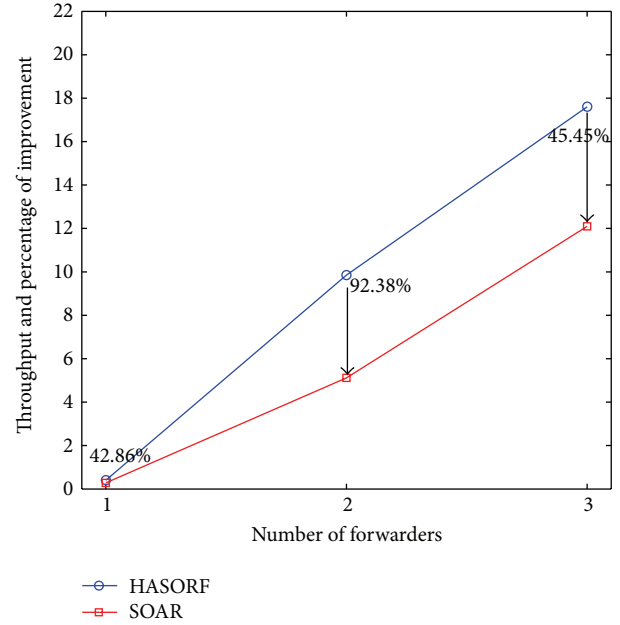


FIGURE 13: Percentage throughput improvement of HASORF and SOAR schemes for network topology 2.

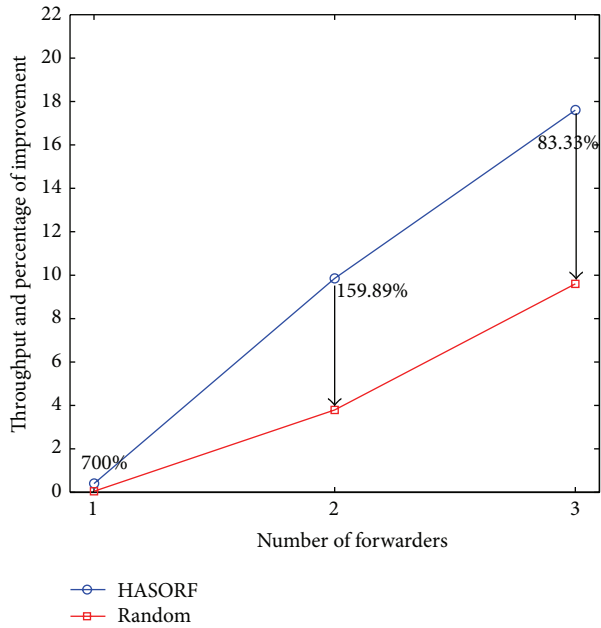


FIGURE 12: Percentage throughput improvement of HASORF and random schemes for network topology 2.

in the given time but also saves power while reducing the number of retransmissions. Recalling that power is one of the most critical resources in sensor networks, we conclude that HASORF improves overall system performance.

Figure 14 represents the bioinspired wireless network scenario. In this scenario, all sensor nodes are equipped with a global positioning system device. Hence, the distance can be calculated using geometric coordinates, and positions of

all the sensors can be found [30]. The source can select the forwarders according to the HASORF scheme and deliver data to the destinations with potentially the best forwarders among the intermediate nodes. In case of the heart beat monitoring device, the delivery of sensor data reliably and in a shorter time is very crucial. The HASORF serves this purpose well. Because one of the main goals of bioinspired systems is to deliver packets over the shortest path in less time, HASORF can be a good candidate for bioinspired networks.

## 6. Conclusion

After the introduction of ExOR, many researchers proposed various opportunistic routing schemes. It is well known that the performance of OR schemes heavily depends on the selection of appropriate forwarders from among the intermediate nodes. However, the impact on throughput is not clearly understood yet. In this paper, we analyzed OR schemes in WSNs while focusing on forwarder selection and intuited that a node midway between the previous and next forwarders is a good choice. Inspired by this intuition, we proposed a heuristic forwarder selection scheme, called HASORF, for chain networks. The simulation results show that our proposed scheme attains high throughput. We compared our scheme with the previous schemes, such as ExOR, random routing, and SOAR, and found that the proposed scheme outperforms all the competitors in terms of throughput.

The optimized number of forwarders for a given network is still an open research issue. This research problem is NP-hard and still there is no foundation for analytic thinking, but it is a really interesting problem. We intend to dig into this compulsive and strenuous problem in the future.



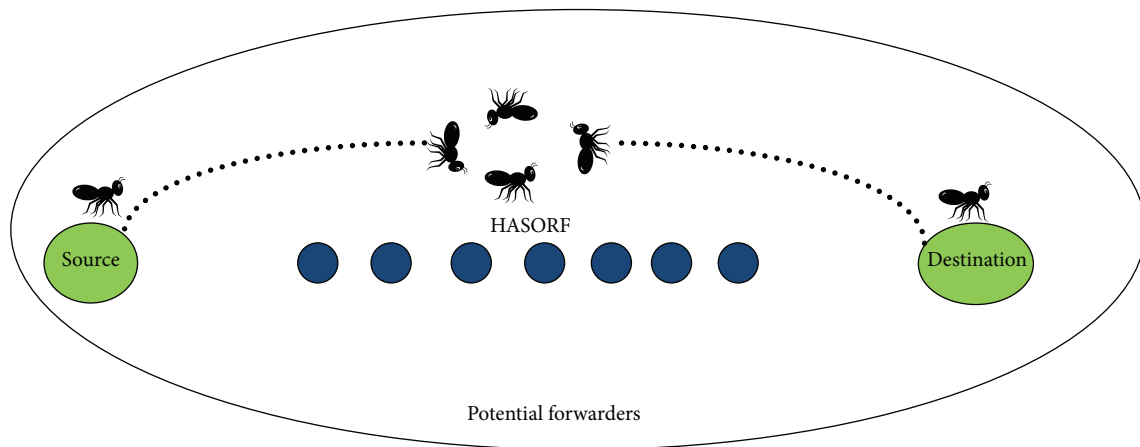


FIGURE 14: HASORF bioinspired WSN scenario.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (NRF-2012R1A1B4000536).

## References

- [1] F. Dressler and O. B. Akan, "A survey on bio-inspired networking," *Computer Networks*, vol. 54, no. 6, pp. 881–900, 2010.
- [2] S. S. Iyengar, H.-C. Wu, N. Balakrishnan, and S. Y. Chang, "Biologically inspired cooperative routing for wireless mobile sensor networks," *IEEE Systems Journal*, vol. 1, no. 1, pp. 29–37, 2007.
- [3] K. Martinez, R. Ong, J. K. Hart, and J. Stefanov, "Glacsweb: a sensor web for glaciers," in *Proceedings of the IEEE 1st European Workshop on Wireless Sensor Networks (EWSN '04)*, Berlin, Germany, January 2004.
- [4] F. Michahelles, P. Matter, A. Schmidt, and B. Schiele, "Applying wearable sensors to avalanche rescue," *Computers and Graphics*, vol. 27, no. 6, pp. 839–847, 2003.
- [5] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 54–61, 2004.
- [6] W. Marshall, C. Roadknight, I. Wokoma, and L. Sacks, *Self-Organizing Sensor Networks*, UbiNet, London, UK, 2003.
- [7] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.-S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet," in *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 96–107, San Jose, Calif, USA, October 2002.
- [8] Z. Butler, P. Corke, R. Peterson, and D. Rus, "Networked cows: virtual fences for controlling cows," in *Proceedings of the Workshop on Applications of Mobile Embedded Systems (WAMES '04)*, Boston, Mass, USA, June 2004.
- [9] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pp. 88–97, Atlanta, Ga, USA, September 2002.
- [10] R. Beckwith, D. Teibel, and P. Bowen, "Pervasive computing and proactive agriculture," in *Proceedings of the 2nd International Conference of Pervasive Computing*, Vienna, Austria, April 2004.
- [11] F. Michahelles, P. Matter, A. Schmidt, and B. Schiele, "Applying wearable sensors to avalanche rescue," *Computers & Graphics*, vol. 27, no. 6, pp. 839–847, 2003.
- [12] Neuron, 2012, <http://www.motorola.com/content.jsp?globalObjectId=290>.
- [13] Holter Systems, Med-electronics Inc, 2014, <http://med-electronics.com/>.
- [14] Cardio Micro Sensor, 2014, <http://www.cardiomems.com>.
- [15] D. S. J. de Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom '03)*, pp. 134–146, 2003.
- [16] S. Biswas and R. Morris, "Opportunistic routing in multi-hop wireless networks," in *Proceedings of the 2nd Workshop on Hot Topics in Networks (HotNets '03)*, Cambridge, Mass, USA, November 2003.
- [17] S. Biswas and R. Morris, "ExOR: opportunistic multi-hop routing for wireless networks," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '05)*, pp. 133–144, ACM, 2005.
- [18] E. Rozner, J. Seshadri, Y. A. Mehta, and L. Qiu, "SOAR: simple opportunistic adaptive routing protocol for wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 12, pp. 1622–1635, 2009.
- [19] T. Li, D. Leith, and L. Qiu, "Opportunistic routing for interactive traffic in wireless networks," in *Proceedings of the 30th International Conference on Distributed Computing Systems (ICDCS '10)*, pp. 458–467, IEEE, Genoa, Italy, June 2010.
- [20] Z. Wang, Y. Chen, and C. Li, "CORMAN: a novel cooperative opportunistic routing scheme in mobile Ad Hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 2, pp. 289–296, 2012.

- [21] E. Rozner, J. Seshadri, Y. Mehta, and L. Qiu, "Simple opportunistic adaptive routing protocol for wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 12, pp. 1622–1635, 2009.
- [22] S. Chachulski, M. Jennings, and S. Katti, "Trading structure for randomness in wireless opportunistic routing," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '07)*, vol. 37, pp. 169–180, 2007.
- [23] Y. C. Cheng, J. Bellardo, P. Benko et al., "Jigsaw: solving the puzzle of enterprise 802.11 analysis," in *Proceedings of the ACM SIGCOMM Conference*, 2006.
- [24] N. Celandroni, F. Davoli, E. Ferro, and A. Gotta, "Long-lived TCP connections via satellite: cross-layer bandwidth allocation, pricing, and adaptive control," *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 1019–1030, 2006.
- [25] Y. Yuan, H. Yang, S. Wong, S. Lu, and W. Arbaugh, "ROMER: resilient opportunistic mesh routing for wireless mesh networks," in *Proceedings of the 1st IEEE Workshop on Wireless Mesh Networks (WiMesh'05)*, 2005.
- [26] K. L. Judd, "The law of large numbers with a continuum of IID random variables," *Journal of Economic Theory*, vol. 35, no. 1, pp. 19–25, 1985.
- [27] <http://www.isi.edu/nsnam/ns/>.
- [28] J. Cai, X. Song, J. Wang, and M. Gu, "Reliability analysis for chain topology wireless sensor networks with multiple-sending transmission scheme," *EURASIP Journal on Wireless Communications and Networking*, vol. 2014, article 156, 13 pages, 2014.
- [29] J. Y. Yoo and J. Kim, "Maximum end-to-end throughput of chain-topology wireless multi-hop networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 4282–4286, March 2007.
- [30] G. Guido, A. Vitale, V. Astarita, F. Saccomanno, V. P. Giofré, and V. Gallelli, "Estimation of safety performance measures from smartphone sensors," *Procedia-Social and Behavioral Journal*, vol. 54, pp. 1095–1103, 2012.

