Enhanced Reactive Routing (E2R) for Wireless Sensor Networks

Murtale Priya Prakash
Department of Computer Science & Engineering
A.P.S College of Engineering,Bangalore,Karnataka,India

Prof Bhagyashree R
Assistant Professor, A.P.S.College of Engineering,Bangalore,Karnataka,India

Abstract—To Provide efficient and reliable communication under loss/ fading channels is one of the major challenge in wireless sensor networks (WSNs), especially in industrial WSNs (IWSNs) with dynamic and harsh environments. In this work, The proposed system present the Enhanced Reactive Routing (E2R) to increase the re-silience to link dynamics for WSNs/IWSNs. E2R is designed to en-hance existing reactive routing protocols to provide reliable and en-ergy-efficient packet delivery against the unreliable wireless links by utilizing the local path diversity. Proposed system introduce a bi-ased backoff scheme during the route-discovery phase to find a ro-bust guide path, with more cooperative forwarding opportunities. Along this guide path, data packets are greedily pro-gressed toward the destination through nodes' cooperation without utilizing the location information. Through extensive simulations, we demonstrate that compared to other protocols, E2R remark-ably improves the packet delivery ratio, while maintaining high energy efficiency and low delivery latency.

Index Terms—Industrial wireless sensor networks (IWSNs), op-portunistic routing, reliable forwarding, unreliable wireless links, R3E & E2R are used interchangeably in this paper.

I. INTRODUCTION

WIRELESS sensor networks (WSNs) are replacing the traditional wired industrial communication systems since the industrial wireless sensor networks (IWSNs) offer several advantages including easy and fast installation and low-cost maintenance [1]. IWSN applications, such as factory automation, industrial process monitoring and control, and plant monitoring, require reliability and timeliness in for-warding messages among nodes [2]. However, the traditional routing protocols, such as AODV [3], AOMDV [4], and DSR [5], may fi nd their limitations in industrial installations due to the harsh environmental conditions, interference issues, and other constraints [6].

In IWSNs, transmission failures can result in missing or de-laying of process or control data, and missing the process or con-trol deadline is normally intolerable for industrial applications, as it may cause chaos in industrial automation or possibly ter-minate the automation, ultimately resulting in economic losses [7]. The sensed data should be reliably and timely transmitted to the sink node, and the programming or retasking data for sensor node operation, command, and query should be reliably deliv-ered to the target nodes [8]. It is also required that these networks can operate for years without replacing the device batteries [9]. Therefore, the reliability, timeliness, and energy efficiency of data forwarding are crucial to ensure proper functioning of an IWSN. However, one of the major technical challenges for re-alization of IWSNs is to provide reliable and efficient commu-nication in dynamic and harsh environments [1], [10]. This is because, in harsh industrial environments, sensor nodes may be subject to radio frequency (RF) interference, highly caustic or corrosive environments, high humidity levels, vibrations, dirt and dust, or other conditions that challenge network perfor-mance [8].Since the varying wireless channel conditions and sensor node failures may cause network topology and connectivity changes over time, to forward a packet reliably at each hop, it may need multiple retransmissions. This results in undesirable delay as well as additional energy consumption. Opportunistic routing (OR) [11]–[16] has been proposed as an effective cross-layering technique to combat fading channels, thus improving the robustness and energy efficiency in wireless networks. The idea of opportunistic routing is to take advantage of the broadcast nature of wireless communication, involving multiple neighbors of the sender into local forwarding. Since the wireless medium is shared, each node can overhear data packets sent by its neighbors. In the network layer, a set of forwarding candidates are specified in the data packet and these nodes will follow the assigned priorities to relay the packet. Essentially, only one node is chosen as the actual forwarder at the MAC layer in an *a posteriori* manner.

Reactive routing protocols [3], [4], [17] are designed to re-duce the bandwidth and storage cost consumed in table driven protocols. These protocols apply the on-demand procedures to dynamically build the route between a source and a destination. Routes are generally created and maintained by two different phases namely route discovery and route maintenance. Route discovery usually occurs on -demand by fl ooding an RREQ (RouteRequest) through the network, i.e., when a node has data to send, it broadcasts an RREQ. When a route is found, the destination returns an RREP (RouteReply), which contains the route information (either the hop-by- hop information [3] or complete addresses from the source to the destination [5]) traversed by the RREQ.

In this work, we propose a Reliable Reactive Routing En-hancement (E2R) to increase the resilience to link dynamics for WSNs/IWSNs. Our design inherits the advantages of op-portunistic routing, thus achieving shorter end -to-end delivery delay, higher energy efficiency, and reliability. E2R is designed to augment existing reactive routing protocols to combat the channel variation by utilizing the local path diversity in the link layer. As a new addition to the cooperative forwarding design space in WSNs/IWSNs, our major contributions are as follows.

- We consider the effect of route discovery on the cooper-ative forwarding performance and combine the solutions to reliable route discovery and efficient cooperative for-warding problems. A robust virtual path that can provide more cooperative forwarding opportunities is found with a low overhead in the route discovery phase, which not only implements the forward path setup in reactive routing, but also facilitates cooperative forwarding along the discov-ered path.
- We propose a simple yet effective cooperative forwarding scheme. Along the discovered virtual path, data packets can be greedily forwarded toward the destination through nodes' cooperation without utilizing location information.
- Through comprehensive performance comparisons, we demonstrate the effectiveness and feasibility of E2R design, which is compatible with most existing reactive routing protocols in WSNs/IWSNs.

The remainder of this work is organized as follows. Section II describes the network model and motivation. Section III elab-orates the design of E2R. Section IV provides the simulation results followed by the related work in Section V. Finally, Section VI concludes the paper.

II. NETWORK MODEL AND MOTIVATION

A. Network Model

We consider a dense multihop static WSN deployed in the sensing field. Since opportunistic routing is normally effective for wireless networks with higher node densities (e.g., more than ten neighbors per node) [18], we assume that each node has plenty of neighbors. When a node has packets to send to the destination, it launches the on-demand route discovery to find a route if there is not a recent route to a destination.

We assume that the MAC layer provides the link quality esti-mation service. There has been a lot of existing work on how to measure wireless link quality in an efficient and accurate manner. We can use a representative link estimation method, such as those in [19] and [20]. In [21], through a set of real ex-periments, the authors reported that the size of the packets has a direct relationship with the packet reception ratio (PRR) in wireless networks. Short control messages, such as RREQ and RREP, have higher PRRs than data packets.

Each node periodically sends HELLO messages to keep track of its neighborhood information. The HELLO message con-tains the IDs (addresses) of a node's one- hop neighbors and the PRRs of the corresponding links. After the HELLO message exchange, essentially, each node maintains the two-hop neigh-borhood information.

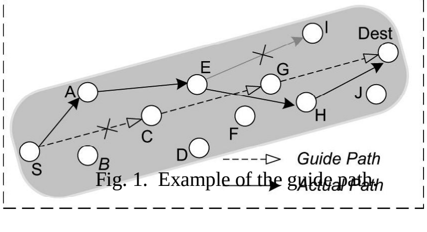


Fig. 1. Example of the guide path.

B. Motivation

Before providing the detailed design, we first illustrate the motivation behind E2R design. The idea of opportunistic routing is to utilize the path diversity for cooperative caching, that is, in each hop, neighboring nodes that hold the copies of a data packet serve as caches, thus the downstream node could retrieve the packet from any of them [22]. The rationale is that, the path with higher spatial diversity (more potential helper nodes) may possibly provide more reliable and efficient packet delivery against the unreliable links. With this observation, we aim to find such a reliable virtual path to guide the packets to be progressed toward the destination. We call this virtual path a *guide path*, in which the nodes are named as *guide nodes*. As shown in Fig.1, [S-> C -> G -> Dest], is a guide path and nodes C and G are said to be guide nodes. The guide path points out the general direction toward the destination, and the routing decision is made a *posteriori*, i.e., the actual forwarders are chosen based on the packet reception results at each hop.

III. MAIN DESIGN

Here, we first present the E2R functional architecture overview, followed by a detailed description of E2R design, which is compatible with most existing reactive routing proto-cols in WSNs/IWSNs.

A. Architecture Overview

Fig. 2 illustrates an overview of the functional architecture of E2R, which is a middle-ware design across the MAC and the network layers to increase the resilience to link dynamics for WSNs/IWSNs. The E2R enhancement layer consists of three main modules, the reliable route discovery module, the poten-tial forwarder selection and prioritization module, and the for-warding decision module. The helper node and potential for-warder are interchangeable in this work.

The reliable route discovery module finds and maintains the route information for each node. During the route discovery phase, each node involved in the cooperative forwarding

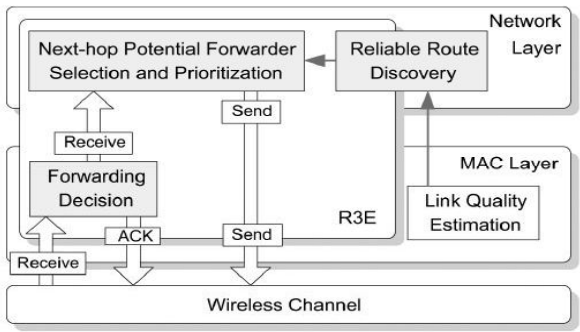


Fig 2. Functional architecture Overview of E2R

process stores the downstream neighborhood information, that is to say, when a node serves as a forwarder, it already knows the next- hop forwarding candidates along the discovered path. The other two modules are responsible for the runtime forwarding phase. When a node successfully receives a data packet, the forwarding decision module checks whether it is one of the intended receivers. If yes, this node will cache the incoming packet and start a backoff timer to return an ACK message, where the timer value is related with its ranking in the intended receiver list (called forwarding candidate list). If there is no other forwarder candidate with higher priority transmitting an ACK before its backoff timer expires, it will broadcast an ACK and deliver the packet to the upper layer, i.e., trigger a receiving event in the network layer. Then, the potential forwarder selection and prioritization module attaches the ordered forwarder list in the data packet header for the next hop. Finally, the outgoing packet will be submitted to the MAC layer and forwarded towards the destination.

B. Reliable Guide Path Discovery

1) RouteRequest (RREQ) Propagation: If a node has data packets to send to a destination, it initiates a route discovery by flooding an RREQ message. When a node receives a non-dupli-cate RREQ, it stores the upstream node id and RREQ's sequence number for reverse route learning. Instead of rebroadcasting the RREQ immediately in existing reactive routing protocols, we introduce a biased backoff scheme at the current RREQ for-warding node. The aim of this operation is to intentionally am-plify the differences of RREQ's traversing delays along dif-ferent paths. This operation enables the RREQ to travel faster along the preferred path according to a certain defined metric.

Let v_i and v_j denote the last-hop node and current forwarding node of an RREQ, respectively. Let $N(i)$ denote the set of v_i 's one-hop neighbors, and $CN(i,j)$ denote the common neighbor set between v_i and v_j . We define a helper v_k between v_i and v_j as the common neighbor of v_i and v_j , satisfying $P_{ik}>P_{ij}$ and $P_{kj}>P_{ij}$, where P_{ij} is the PRR between v_i and v_j . For cooperative routing, there exists an implicit constraint, that is, the nodes in the helper set should be able to hear from each other with a reasonably high probability. Let $H(i,j)$ denote the set of helpers between v_i and v_j . In other words, $H(i,j)$ is the common neighbor set between v_i and v_j on the premise that

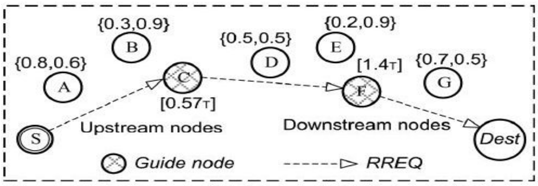


Fig. 3. Example illustrating the biased backoff scheme for RREQ propagation during the route discovery phase. The RREQ that travels along the path $S \to C \to F$ arrives at the $Dest$ first.

any two nodes in H(i,j) can over hear each other, and

$$\forall v_k \in H(i,j), P_{ik} > P_{ij}, P_{kj} > P_{ij}. H(i,j) \subseteq \{N(i) \cap N(j)\}.$$

Let T*ij* denotes the backoff delay at the current forwarding node V*j* which recives a RREQ from Vi. Tij is calculated as defined below

$$t_{ij} = \frac{HopCount}{\sum_k P_{ik}P_{kj}+1} \cdot \tau, \quad v_k \in H(i,j) \qquad (1)$$

Where *T i*s a time slot unit,The HopCount is RREQ's hop distance from the source node thus far. The rationale is that, the neighbor with more forwarding candidates, better link qualities, as well as shorter hop-count will have a shorter backoff delay to rebroadcast th

Fig. 3 illustrates the biased backoff schen delay by assuming itself as a guide node, ar example, nodes A, B, and C receive an RRE considers itself as a guide node and S as the A and B are helper nodes. Then, it can calcul When C's backoff timer first expires, the RR forward the RREQ. Similarly, node F forwa path with more potential helpers. Upon recei back to the source along the reverse route. Ir shall only reply to the first received RREQ received RREQ.

Algorithm 1 How a node v_j handles the RREQ node v_i

Procedure: void *RecvRREQ (Packet *p)*
if Non-duplicate RREQ **then**
    if v_j is the destination node **then**
        Send out RREP;
    else

Algorithm 1 How a node v_j handles the RREQ received from node v_i

Procedure: void *RecvRREQ (Packet *p)*
if Non-duplicate RREQ **then**
    if v_j is the destination node **then**
        Send out RREP;
    else

```
    CN(i, j) = N(i) ∩ N(j);
    //get common neighbor set CN(i, j), v_k ∈ CN(i, j);
    Sort CN(i, j) descendingly ordered by P_ik P_kj;
    H(i, j) = {cn_1}, CN(i, j) = CN(i, j) − {cn_1};
      // cn_1 is always the first item of CN(i, j);
      While CN(i, j) ≠ ∅ do
        if CheckConnectivity(H(i, j), cn_1) then
           // cn_1 is within the transmission range of any
           node in H(i, j);
           H(i, j) = H(i, j) ∪ {cn_1};
        end
        CN(i, j) = CN(i, j) − cn_1;
      end
    Calculate t_ij and call Backoff(t_ij, p);
    //schedule a timer whose value is t_ij, then call
    forwardRREQ(p) when the timer expires;
    end
else
    Drop p;
end
```

2)RouteReply (RREP) Propagation: When a node receives an RREP, it checks if it is the selected next-hop (the upstream guide node) of the RREP. If that is the case, the node realizes that it is on the guide path to the source, thus it marks itself as a guide node. Then, the node records its upstream guide node ID for this RREP and forwards it. In this way, the RREP is propagated by each guide node until it reaches the source via the reverse route of the corresponding RREQ. Finally, this process finds a guide path from the source to the destination.

Algorithm 2: How a node v_j handles the RREP received from its downstream guide node v_i

---

```
Procedure:void RecvRREP (Packet *p)
if Non-duplicate RREP then
   if v_j == v_{i-1} then
      //v_j is the selected next-hop & guide node v_{i-1};
      Mark myself as a guide node;
      Record v_i and H(i − 1, i);
      Get RREP's next-hop node id v_{i-2};
      Attach v_i, H(i − 1, i), v_{i-1} and H(i − 2, i − 1)
        to RREP;
      /* v_{i-2} is v_{i-1} 's upstream guide node; the helper set
        is ordered descendingly by the PRR toward the
        downstream guide node;*/
        Call forwardRREP p;
   else if v_j ∈ H(i − 1, i) then
        // v_j is a helper in H(i − 1, i);
        Record v_{i+1}, H(i, i + 1), v_i and H(i − 1, i);
        Drop p;
   else
        Drop p;
   end
else
   Drop p;
end
```

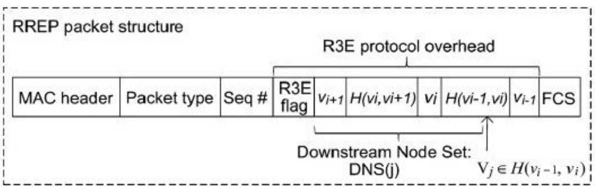


Fig. 4. RREP packet structure. Suppose guide node (v_i) sends out an RREP to the upstream guide node v_{i-1}, and node $v_j \in H(i-1, i)$ overhears this message.

In our design, the RREP message has twofold functions. It not only implements the forward path setup, i.e., marking guide nodes along the reverse route, but also noti fies the potential helpers to facilitate cooperative forwarding. Specifically, two sets of helpers and their relay priority assignments are included in the RREP. Suppose v_{i-1}, v_i, and v_{i+1} are three adjacent guide nodes,the upstream link helper set H(i-1 ,j)and downstream linjk helper set H(I,i+1) together with the PRR's towards the corresponding downstreams guide nodes are piggybacked to the RREP when node v_i forwards it. Due to the broad-cast nature of wireless communication, all of the helper nodes in H(i-1,i) are expected to overhear this RREP.When the guide node Vi-1 recives the RREP from node Vi; it records its downstream guide node Vi and H (i-1,i) . When the upstream link helper in H(i-1,i) recives the RREP,they record Vi+1, H(i, i+1),Vi and H(i-1,i) which will be used in data forwarding phase. Alogarithm 2 describes how the nodes handle the RREP recived from downstream guide node.

Since E2R is an enhancement layer over existing reactive routing protocols, a bit (the E2R flag) is used to indicate that the E2R function is enabled. There is not much protocol overhead for the RREQ message, where only the hopcount is included. E2R incurs a certain protocol overhead in RREP, i.e., a sequence of node IDs are piggybacked to the RREP, as shown in Fig. 4. However, the overhead will be eventually compensated by performance gain during the data transmission phase. Suppose the guide node Vi sends out an RREP to the upstream guide node Vi-1, and node Vj(Vj € H(i-1,i)) overhears this message.We define the downstream node set of , denoted by DNS(j) , as the sequential nodes that rank ahead of in the piggybacked node list of RREP. As seen in Fig. 4,N(j)∩ DNS(j) is the potential downstream helper set of Vj

Fig. 5 is an example illustrating the RREP propagation corresponding to Fig. 3. Suppose F is the current RREP forwarding guide node, C is F's upstream guide node, the upstream link helper set H(C,F) and downstream link helper set H(F,Dest)will be attached in the RREP. When the guide node F forwards the RREP to its upstream guide node C, for example, helper D overhears this RREP and records the piggybacked information {Dest,G ,F,E,D} . D's one-hop neighbor set N(D)is {B,C,E,F,G} . Therefore, its potential forwarding candidates when forwarding data packets toward the Dest would be N(D) ∩ {Dest,G,F,E,D} = {G,F,E}

Since the wireless links are unreliable, especially in harsh IWSNs, we also consider the possibilities of RREQ and RREP transmission failures. E2R is fault-tolerant to the failure of RREQ, since there exist multiple paths between the source and the destination. However, the transmission reliability of RREP

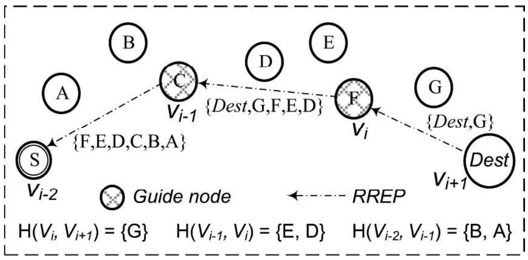


Fig. 5. Example illustrating theRREP propagation to implement cooperative forwarding, in which v_{i-2} , v_{i-1} , v_i and v_{i-1} are adjacent guide nodes in sequence.

is desirable to be guaranteed, since the RREP returned by the destination node may collide with RREQs in the network, which will be shown in Section IV-C. In addition, if an RREP is lost, the source node probably needs to launch another route discovery process again, which will result in a long routing discovery delay.

C. Cooperative Forwarding

The cooperative forwarding procedure in E2R is described as follows. The source node broadcasts a data packet, which in-cludes the list of forwarding candidates (helper nodes and the downstream guide node) and their priorities. Those candidates follow the assigned priorities to relay the packet. Each candi-date, if having received the data packet correctly, will start a timer whose value depends on its priority. The higher the priority, the shorter is the timer value. The candidate whose timer expires will reply with an ACK to notify the sender, as well as to suppress other contenders. Then, it rebroadcasts the data packet toward its downstream link. If no forwarding candidate has successfully received the packet, the sender will retransmit the packet if the retransmission mechanism is enabled. We de-note $t(k)$ as the backoff timer value of the kth candidate. Since the lower priority forwarding candidate needs to wait and con-firm that no higher priority candidate has relayed the packet be-fore it takes the forwarding task, $t(k)$ is an increasing function of k. Suppose the link layer protocol is based on CSMA/CA MAC, $t(k)$ can be defined as

$$_{\mathrm{om}}t(k) = (T_{\mathrm{SIFS}} + T_{\mathrm{ACK}}) \cdot k \qquad (2)$$

where T_{SIFS} is the value of *Short Inter Frame Space*, T_{ACK} is the transmission delay for sending an ACK.
We denote $T_d(k)$ as the one-hop medium delay [23] of the kth candidate, which is the time interval between the sender broadcasting a data packet and the kth candidate claiming that it has received the packet. We define this as

$$T_{\mathrm{omd}}(k) = T_{\mathrm{DIFS}} + T_{\mathrm{DATA}} + (T_{\mathrm{SIFS}} + T_{\mathrm{ACK}}) \cdot k \qquad (3)$$

where T_{DIFS} is the value of *Distributed Interframe Space*

(DIFS), T_{DATA} is the transmission delay of data packet, and the signal propagation delay is ignored. For sender v_i, given n sequential forwarding candidates(j1>j2>……… >jn),we have expected one hop media delay Eomd(i) as

$$E_{\mathrm{omd}}(i) = \sum_{k=1}^{n} \left\{ T_{\mathrm{omd}}(k) P_{ij_k} \prod_{m=0}^{k-1} \overline{P}_{ij_m} \right\} + T_{\mathrm{omd}}(n) \prod_{m=1}^{n} \overline{P}_{ij_m} \qquad (4)$$

where $\overline{P}_{ij_m} = 1 - P_{ij_m}$ and $\overline{P}_{ij_0} = 1$.

minimum number of transmissions, we can apply the basic greedy for-warding rule in geographic routing [24] as the relay priority rule in E2R, i.e., data packets are greedily forwarded to the neighbor geographically closest to the destination. In this way, E2R ob-viates the necessity of utilizing location information, while en-abling data packets to be greedily forwarded toward the des-tination with the help of the robust guide path. Note that the relay priority rule can also adopt other variant metrics, e.g., the one-hop throughput metric [23] to achieve the best path throughput.

From the realistic link conditions in wireless networks, a potential forwarder with a higher PRR toward the downstream guide node possibly has a shorter distance from that guide node, as longer distances normally result in lower received signal strength and thus increased probability of packet loss [25]. Therefore, the relay priority rule is as follows. When a
guide node Vi-1 transmits the data packet, the downstream guide node Vi has the highest priority; and the helper nodes
in are ordered descendingly according to their PRRs toward . Suppose the downstream guide node Vi fails to receive the packet, while a helper Vj in H(i-1,i) receives the packet and takes the forwarding task. The forwarding candidates of Vj are given by N(j) ∩ DNS(j) . More specifically, the forwarding candidate set of Vj is composed of three parts: 1) the helper nodes who have higher priorities than Vj in H(i-1,i) ; 2) the downstream guide node ; and 3)N(j) ∩(H(i,i+1) U {Vi+1}) . To achieve the minimum number of transmissions, the relay priorities are ordered as: Priority of 3) > Priority of 2) > Priority of 1).

Revisiting the examples in Figs. 3 and 5, we show the helper nodes and their priorities at each hop in the data forwarding phase in Fig. 6(a). The helper nodes and their priorities at the first hop are CBA . Suppose guide node C fails to receive the packet correctly, while helper B successfully receives the packet, as shown in Fig. 6(b). B takes the forwarding task in-stead of C. Then B updates its helper set as DC and forwards the data packet to its downstream potential forwarders. It can be seen that E2R is resilient to the wireless link dynamics.

IV. PERFORMANCE EVALUATION

Here, we present performance evaluation results. We imple-ment E2R as an extension to AODV [3], named AODV-E2R, using the ns-2 simulator [26], and compare the performance with three baseline routing protocols.

All the results have been averaged over 100 runs, and the related standard deviations are provided as error bars.

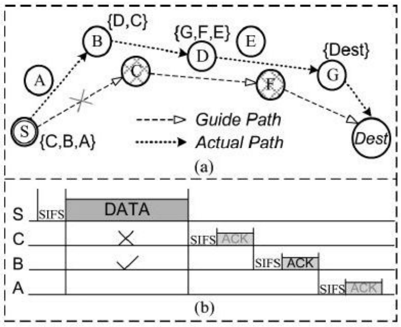


Fig. 6. (a) Example illustrating the forwarding candidates and their priorities at each hop. (b) Corresponding cooperative forwarding procedure at the first hop in the MAC layer.

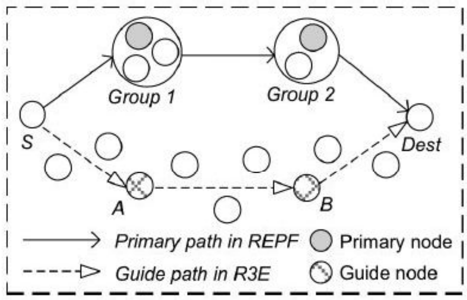


Fig. 7. Example illustrating the primary path in REPF [27]. Only the nodes which can connect the two-hop away primary forwarding nodes are considered as helper nodes. Thus, it restricts the cooperative forwarding to a very limited scope.

A. Comparison Baselines

We first introduce the comparison baselines.We choose three routing protocols, described here.

• **AODV-ETX**[3]: The route discovery phase finds a least-ETX (*expected transmission count*) path from a source to a destination. In AODV-ETX, the link layer retransmission is enabled, i.e., at most three retransmissions at each hop are sanctioned. Note that the other routing protocols do not adopt the retransmission mechanism.

• **REPF** [27]: REPF (Reliable and Efficient Packet Forwarding) protocol is designed to improve the AODV routing performance by utilizing local path diversity. The route discovery phase finds an efficient primary path (composed of a set of primary forwarding nodes) in terms of the accumulated path ETX, and alternative paths which have similar cost. However, REPF restricts the helper nodes to a very limited scope, i.e., only the nodes which can connect the two-hop away primary forwarding nodes are considered as helper nodes, as shown in Fig. 7. As a result, it does not fully utilize the forwarding opportunities provided by available neighboring nodes in evenly distributed networks.

• **GOR**[24]: In order to show that E2R enables data packets to be greedily progressed toward the destination, we also report the evaluation results of the Geographic Opportunistic Routing (GOR). In our simulation, both E2R and GOR follow the same relay priority rule, i.e., minimizing the number of end-to-end data transmissions. We implement GOR as follows: all of the one-hop neighbors that are nearer from the destination than the current forwarding node and can hear from each other are selected as helper nodes, and the nodes closer to the destination are given higher relay priorities. Since the network is densely deployed, the routing recovery mechanism bypassing "holes" is not considered in the simulations.

B. Simulation Setup

We define the node density as the number of nodes deployed in a 200 m 200 m square area. One hundred randomly connected topologies for each node density setting are generated using the *setdest* tool in ns-2, ranging from 100 to 200. The node transmission range is set to 50 m. The destination node is positioned at bottom left (0 m, 0 m), and the source node is positioned at top right (200 m, 200 m). In AODV-E2R, the system parameter is set to 0.005s. In order to highlight the potential collision problem between the returning RREP and RREQs, RREP is not acknowledged by the guide nodes at each hop in our simulation.

We use the Nakagami distribution defined as

$$f(x, m, \Omega) = \frac{m^m x^{m-1}}{\Gamma(m)\Omega^m} \exp\left(-\frac{mx}{\Omega}\right) \qquad (5)$$

to describe the power of a received signal, where is the Gamma function, denotes the Nakagami fading parameter, and is the average received power. We set in our simulation. Assuming *TwoRayGround* signal propagation, can be expressed in

$$\Omega(d) = \frac{P_t G_t G_r h_t^2 h_r^2}{d^n L} \qquad (6)$$

By using (5) and (6), we can derive the PRR at a certain distance [18]. We choose four main evaluation metrics.

• *Packet delivery ratio*: the ratio of the number of packets received by the destination to the total number of packets sent by the source.

• *End-to-end delay*: the time taken for a packet to be transmitted from the source node to the destination node.

• *Data transmission cost*: it is measured as the total number of data transmissions for an end-to-end delivery per packet.

• *Control message cost*: it is defined as the total number of control message transmissions (such as RTS, CTS and ACK) for sending a single packet to the destination.

C. Performance Overview

We evaluate the impact of network density on the performance of the different protocols. The evaluation results are

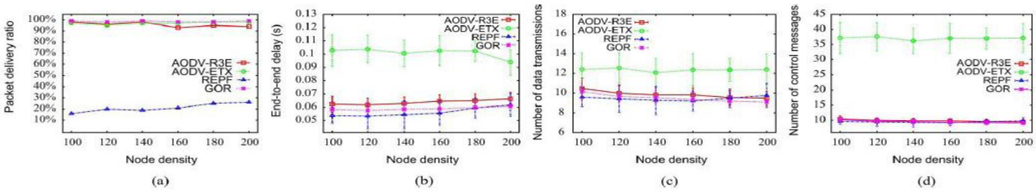


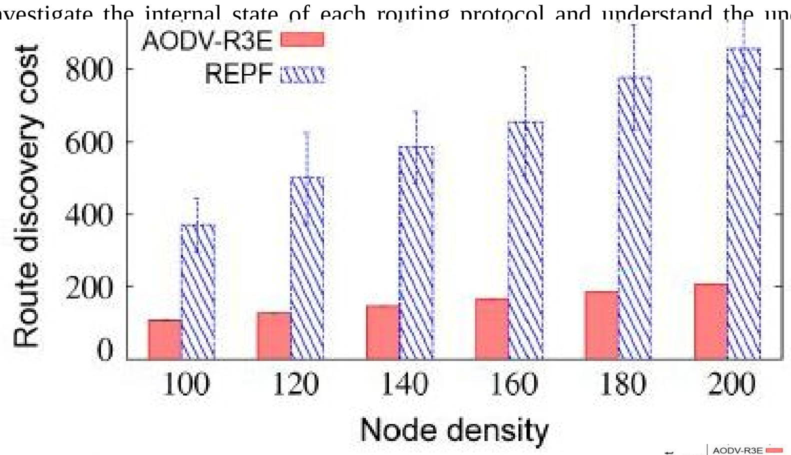
Fig. 8. Performance comparison, only the successful end-to-end transmissions are counted. (a) Packet delivery ratio. (b) End-to-end delay. (c) Data transmission cost. (d) Control message cost.

shown in Fig. 8, where the node density is varied from 100 to 200. Fig. 8(a) illustrates the end-to-end packet delivery ratio under different node densities. GOR achieves very high packet delivery ratio, since the node density is always high, and it takes full advantage of the local forwarding opportunities, involving maximum possible number of neighbors which are closer to the destination. For AODV-ETX, it already finds a more efficient path using the ETX metric than using the hop-count metric [3]. In addition, packets are forwarded using unicast transmission, and three retransmissions at each hop are allowed in the link layer in AODV-ETX. That is why it also achieves a high packet delivery ratio. We observe that AODV-E2R shows above 95% packet delivery ratio on average, which is comparable to that of AODV-ETX and GOR, without enabling the MAC layer retransmission mechanism.However, REPF only provides around 20% packet delivery ratio on average. The main reason is that REPF restricts the node cooperation to a very limited scope. With the increase of node density, the intuition is that the packet delivery ratio should also increase. However, it is surprising that the delivery ratio drops a little as the node density increases in AODV-E2R, as shown in Fig. 8(a). An examination of the simulation traces in MAC layer reveals that, as the network becomes more congested, the RREP returned by the destination node may collide with RREQs in the etwork. Therefore, it is better that each guide node acknowledges the reception of the RREP. Fig. 8(b) depicts the performance comparison on the end-to-end packet delivery delay. We know that the delay incurred by retransmission is much larger than the coordination delay introduced by the cooperative forwarding. Therefore, AODV-ETX requires longer time to finish the end-to-end transmission, because it requires several retransmissions to forward a packet to the destination. AODV-E2R shows very close performance to the GOR. Note that only the successful end-to-end transmissions are counted in Fig. 8. From Fig. 8(b), it seems that REPF yields the best performance. Actually, this is not true. Because it only shows the results for around 20% successful transmissions. REPF should yield worse results if the failed end-to-end transmissions are counted, which we

will discuss in Section IV-D. Overall, AODV-E2R can provide 50% improvement over REPF in terms of the delivery delay. Fig. 8(c) and (d) reports the changes of the data transmission cost and controlmessage cost, respectively, under different node
densities. From Fig. 8(c), it clearly shows that REPF requires two or three retransmissions to finish the transmission task. As expected, AODV-E2R only incurs approximately 5% higher transmission cost than the ideal baseline scheme GOR. However, GOR requires the location information. Since the traffic is transmitted using unicast in AODV-ETX, it incurs a higher control message overhead compared with other protocols, as shown in Fig. 8(d). Another interesting observation about AODV-E2R and GOR is that, in Fig. 8(b), as the node density increases, the end-to-end delay also slowly increases, whereas the number of data transmissions slightly decreases, as can be seen in Fig. 8(c). This is because of the relay priority rule applied in AODV-E2R and GOR. As the node density increases, more available forwarding candidates can be involved into the cooperative forwarding. However, the higher forwarding priorities will be assigned to neighbors with better advancements towards the destination, but with relatively lower PRRs. Consequently, it increases the one-hop medium time [23], which is defined as the interval between the sender broadcasting a data packet andthe forwarding candidate's first claim of receiving the packet.

D. Insights

Here, we investigate the internal state of each routing protocol and understand the underlying reasons why



llustrates the
rks and 200-
s, for a fair
, the average
0538 s. From
10(a) and (b)
obvious that
e forwarding
ith only local
has the least
estricts the
ıvolves three

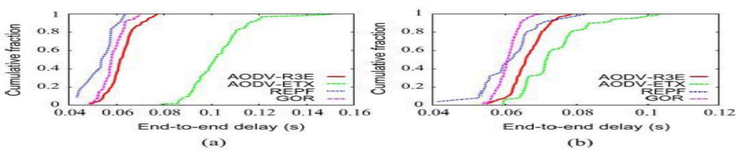


Fig. 9. CDF of end-to-end delay: (a) 100-node networks and (b) 200-node networks.

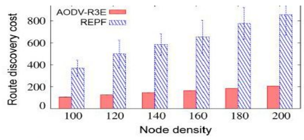


Fig. 12. Route discovery cost.

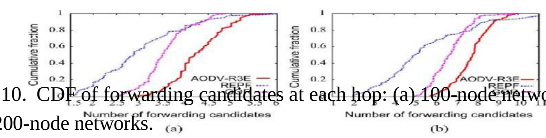


Fig. 10. CDF of forwarding candidates at each hop: (a) 100-node networks and (b) 200-node networks.

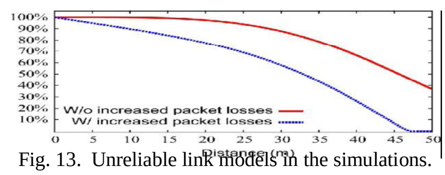


Fig. 13. Unreliable link models in the simulations.

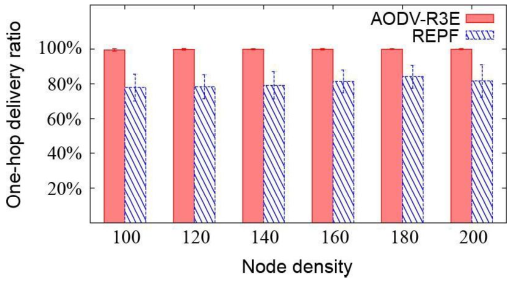


Fig. 11. One-hop delivery ratio.

forwarding candidates at each hop, whereas GOR and AODV-E2R, on average, have 3.6 and 4.3 forwarding candidates, respectively. In 200-node networks, REPF has 5.5 forwarding candidates, whereas GOR and AODV-E2R on average involve 6.7 and 7.5 forwarding candidates at each hop, respectively.

Although having more cooperative forwarding opportunities per hop than GOR, AODV-E2R still generates a little larger de-livery delay and transmission cost than that of GOR. The reason is that AODV-E2R prioritizes forwarding candidates based on only the PRRs toward the downstream guide node. In some cases, the potential forwarder with the highest PRR toward the downstream guide node is not necessarily the closest node to the destination node. Therefore, AODV-E2R does not always find the optimal order of forwarding candidates without utilizing the location information.

Fig. 11 plots the one-hop delivery ratio for AODV -E2R and REPF, which is defined as the probability that at least one forwarding candidate successfully receives the packet at each hop. From the figure, we can see that, as compared with
REPF, AODV-E2R yields a remarkable increase of 20% in one-hop delivery ratio. It reveals the reason why REPF only achieves a very low packet delivery ratio on average when the retransmission mechanism is disabled.

Fig. 12 compares the route discovery overhead between AODV - E2R and REPF, which measures the total number of routing discovery control message transmissions, e.g., RREQ and RREP. REPF incurs much higher communication overhead and increases the network load during the route discovery phase, since it allows duplicate propagating RREP packets in order to fi nd a better virtual path. We observe that the route discovery overhead in both AODV-E2R and AODV-ETX is
$O(n)$, where n is the number of nodes. However, the route discovery overhead in REPF increases exponentially as the network size increases.

E. Results With Increased Packet Losses

In an IWSN, the harsh industrial environments may result in very unreliable wireless links due to many factors [1], [8]. To evaluate the effect of increased packet losses, here, we use a simple linear link lossy model based on the distance between transmitter and receiver. In these simulations, a link with a dis-tance of 0 m has 0% probability of additional packet loss, and a link with distance of 50 m has 50% probability of additional loss, which is the same as the setting in [25]. The PRRs de-rived from the distance between transmitter and receiver with this modified channel model are shown in Fig. 13.

Fig. 14(a) shows the end-to-end packet delivery ratio as the node density increases. Compared with the results using the channel model without increased packet losses shown in Fig. 8(a), the packet delivery ratio for all protocols decreases. From this figure, we can see that AODV-E2R even outperforms

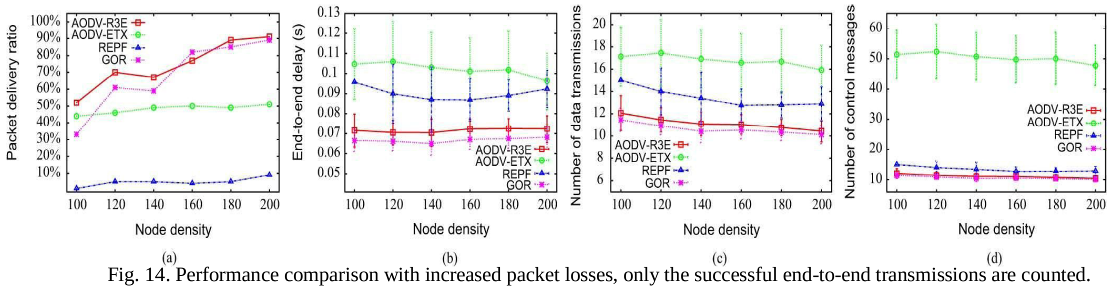


Fig. 14. Performance comparison with increased packet losses, only the successful end-to-end transmissions are counted. (a) Packet delivery ratio. (b) End-to-end delay. (c) Data transmission cost. (d) Control message cost.

GOR with increased packet losses and shows the highest av-erage packet delivery ratio among these protocols. It indicates that finding a reliable guide path is very important if the wire-less links are highly unreliable. We also observe that generally, as the node density increases, the packet delivery ratio for both AODV-E2R and GOR also increases. While the AODV-ETX only provides less than 50% delivery ratio, REPF shows less than 10% delivery ratio.

Fig. 14(b) plots the end - to-end delay among the four compared schemes. For the same reason discussed earlier, AODV-E2R has a little larger delay than GOR. When com-paring with the results shown in Fig. 8(b), the delivery delays of both AODV-E2R and GOR increase due to the poor link qualities.

Fig. 14(c) and (d) shows the data transmission cost and control message cost, respectively, under the modifi ed channel model. It is interesting that, for AODV-E2R and GOR, the performance on these two metrics does not change much compared with the results in Fig. 8(c) and (d). The reason for this is that these results only count the successful end-to-end transmissions, and there is no retransmission in AODV-E2R and GOR. Thus, the end-to-end hop-count does not increase for those successful transmissions. However, in the modified channel model with increased packet losses, the end-to-end delivery delay increases since the one -hop medium time is increased. This is because, from (4), the one-hop medium time increases as the link qualities become poor.

F. Performance Summary

In the previous sections, we see that AODV-E2R without uti-lizing the location information achieves a comparable perfor-mance to GOR. Compared with REPF, it signifi cantly improves the packet delivery ratio, and, compared with AODV-ETX, it provides much lower data transmission cost and control mes-sage cost. Remarkably, in the modified channel model with in-creased packet losses, AODV-E2R even performs better than GOR, since it forwards data packets along the path which pro-vides more cooperative forwarding opportunities.

VI. CONCLUSION

In this work, we presented E2R, which can augment most ex-isting reactive routing protocols in WSNs/IWSNs to provide re-liable and energy-efficient packet delivery against the unreliable wireless links. We introduced a biased backoff scheme in the route discovery phase to find a robust virtual path with low over-head. Without utilizing the location information, data packets can still be greedily progressed toward the destination along the virtual path. Therefore, E2R provides very close routing per-formance to the geographic opportunistic routing protocol. We extended AODV with E2R to demonstrate its effectiveness and feasibility. Simulation results showed that, as compared with other protocols, AODV -E2R can effectively improve robust-ness, end-to-end energy efficiency, and latency.

REFERENCES

[1] V. Gungor and G. Hancke, “Industrial wireless sensor networks: Chal-lenges, design principles, and technical approaches,” *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4258–4265, Oct. 2009.

[2] S. eun Yoo, P. K. Chong, D. Kim, Y. Doh, M.-L. Pham, E. Choi, and J. Huh, “Guaranteeing real-time services for industrial wireless sensor networks with IEEE 802.15.4,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3868–3876, Nov. 2010.

[3] C. Perkins and E. Royer, “Ad-hoc on-demand distance vector routing,” in *Proc. IEEE WMCSA*, 1999, pp. 90–100.

[4] M. Marina and S. Das, “On-demand multipath distance vector routing in ad hoc networks,” in *Proc. IEEE ICNP*, Nov. 2001, pp. 14–23.

[5] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” *Mobile Computing*, pp. 153–181, 1996.

[6] K. A. Agha, M.-H. Bertin, T. Dang, A. Guitton, P. Minet, T. Val, and J.-B. Viollet, “Which wireless technology for industrial wireless sensor networks? the development of Ocari technology,” *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4266–4278, Oct. 2009.

[7] K. Yu, M. Gidlund, J. Åkerberg, and M. Björkman, “Reliable RSS-based routing protocol for industrial wireless sensor networks,” in *Proc. IECON*, 2012, pp. 3231–3237.

[8] “U. D. of Energy, Industrial wireless technology for the 21st century,” Office of Energy and Renewable Energy Report, 2002.

[9] Y. Gu and T. He, “Dynamic switching-based data forwarding for low-duty-cycle wireless sensor networks,” *IEEE Trans. Mobile Comput.*, vol. 10, no. 12, pp. 1741–1754, Dec. 2011.

[10] V. Gungor, O. Akan, and I. Akyildiz, “A real-time and reliable trans-port protocol for wireless sensor and actor networks,” *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, pp. 359–370, Apr. 2008.

[11] S. Biswas and R. Morris, “ExOr: opportunistic multi-hop routing for wireless networks,” in *Proc. ACM SIGCOMM*, 2005, pp. 133–144.

[12] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, “XORs in the air: Practical wireless network coding,” *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 497–510, Jun. 2008.

[13] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” in *Proc. AGM SIG-COMM*, 2007, pp. 169–180.

[14] K. Zeng, W. Lou, J. Yang, and D. Brown, “On geographic collaborative forwarding in wireless ad hoc and sensor networks,” in *Proc. WASA*, 2007, pp.

[15] X. Mao, S. Tang, X. Xu, Li X.-Y, and H. Ma, “Energy-efficient op-portunistic routing in wireless sensor networks,” *IEEE Trans. Parallel Distrib.Syst.*, vol. 22, no. 11, pp. 1934–1942, Nov. 2011.

[16] R. Bruno and M. Nurchis, “Survey on diversity-based routing in wire-less mesh networks: Challenges and solutions,” *Computer Commun.*, vol. 33, no. 3, pp. 269–282, Feb. 2010.

[17] E. Royer and C.-K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *IEEE Pers. Commun.*, vol. 6, no. 2, pp. 46–55, 1999.

[18] R. Shah, S. Wietholter, A. Wolisz, and J. Rabaey, “When does op-portunistic routing make sense?,” in *Proc. IEEE PerCom Workshops*, 2005, pp. 350–356.

[19] K.-H. Kim and K. G. Shin, “On accurate measurement of link quality in multi-hop wireless mesh networks,” in *Proc. ACM MobiCom*, 2006, 38–49.

[20] R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis, “Four-bit wireless link estimation,” in *Proc. HotNets VI*, 2007, pp. 1–7.