CORB: Context-Aware Opportunistic Resource-Based Routing for Stationary Wireless Sensor Networks

Mina Elias
Dept. of Engineering Mathematics and Physics
Cairo University
Giza, Egypt 12613
mina.y.elias@ieee.org

Ahmed Khattab, Khaled M. F. Elsayed
Dept. of Electronics and Communications Engineering
Cairo University
Giza, Egypt 12613
{akhattab, khaled}@ieee.org

Abstract— **In this paper, we present the Context-Aware Opportunistic Resource-Based Routing Protocol (CORB) for intermittently-connected stationary wireless sensor networks. This protocol targets domains such as ambiance control in smart buildings. Unlike existing context-aware routing approaches which consider node mobility as the main source of disconnection, CORB only considers the nodes' limited resources and context as the reasons for intermittent connection. Performance evaluation results show that the proposed CORB protocol does not only outperform existing context-aware routing protocols but also significantly outperforms the IP Ripple Routing Protocol (RPL) designed specifically for low power and lossy networks. In grid topologies (sink-centered topologies), the delivery ratio of CORB under heavy loads is up to 404% and 296% (143% and 46.6%) of the original context-aware routing protocol and IPv6 RPL, respectively.**

Keywords—WSN; probabilistic routing; context-aware; RPL

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have been widely considered as one of the most important enabling technologies for the future Internet of Things (IoT). WSNs have several unique characteristics such as the high node density, high unreliability and severe energy, computation, and storage constraints of the sensor nodes that distinguish WSNs from other communications networks. In this paper, we tackle the problem of how to achieve reliable communication via context-aware routing in intermittently-connected WSNs. We limit ourselves to stationary WSNs developed for IoT applications such as ambiance control in smart buildings, structural health monitoring, historical site health monitoring, and precision agriculture, etc. In such static networks, nodes are fixed in position and follow a multi-hop star topology layout in which the source leaf nodes collect data and forward it to a centralized sink node. Unlike mobile WSN networks, wherein node mobility is the fundamental cause of intermittent connection, the main cause of failures in stationary WSNs is the context of the WSN node (such as the weak processing capability, the heterogeneity in the node capabilities, and the dynamicity of the nodes joining/leaving the network [1]).

Reliable and context-aware routing for mobile WSNs has received significant research interest [2]-[7]. However, such routing techniques are not directly applicable to stationary WSNs. Examples include the traditional fault-tolerant routing techniques such as flooding techniques, epidemic routing, and gossiping [3] which require massive sharing of a message in order to ensure a reasonable successful delivery ratio of the message to its intended receiver. Such simple techniques are not suitable for IoT applications due to the huge protocol overhead. In order to reduce the amount of overhead, Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) [4], Context-Aware Routing (CAR) [5], and Sensor Context-Aware Routing (SCAR) [6] semi-epidemic routing techniques combine the node context with prediction techniques to selectively determine the replication pattern. While such techniques do not suffer overhead problems, they mainly target mobile networks in which node mobility is the main source of intermittent-connectivity (i.e., node context is mainly influenced by its mobility).

We present the Context-Aware Opportunistic Resource-Based Routing Protocol (CORB) optimized for stationary WSNs. Unlike existing context-aware approaches such as CAR/SCAR, CORB considers: (1) the strength of the connections of a node with its surrounding neighbors, (2) the rank of the nodes with respect to the sink to direct the messages to the sink node in the least number of transmissions in order to avoid unnecessary/potential loops, and (3) the available buffer space in the node as an important metrics to define the node's context in addition to the residual battery level. Our simulation results show significant performance gains in terms of the delivery ratio and the network life-time with respect to both SCAR as well as the standard IP Ripple Routing Protocol (RPL) [8] designed for low power and lossy networks irrespective of the considered topology. For instance, the delivery ratio of CORB is 404% and 296% over SCAR and RPL, respectively, in grid topologies, and 143% and 46.6%, respectively, in sink-centered topologies under heavy load conditions.

The remainder of the paper is organized as follows. In Section II, we discuss the preliminaries of context-aware routing. Then we present the proposed CORB protocol in section III. In section IV, we present an exhaustive set of simulations demonstrating its performance gains. We conclude the paper in Section V.

II. PRELIMINARIES AND MOTIVATION

In this paper, we adopt the probabilistic approach developed for the CAR/SCAR context-aware routing protocols for mobile WSN to find the best carrier for a message out of the node's neighbors who change frequently due to node mobility. The best next carrier of a message is one of the node's neighbors that has the highest probability of delivering the message to the sink. Each node calculates its own probability of forwarding a message to the sink(s). Then neighboring nodes exchange their probabilities (also referred to as context) with each other. Hence, each node builds a table that contains all of the surrounding nodes s_j ordered according to their delivery probability/context value $U(s_j)$. A node locally chooses the best carrier of a message according to the highest probability value in this routing table.

The total utility function (or the probability of reaching the sink) is the weighted sum of the utilities of the individual measures, i.e., $U(s_i) = \sum_k w_k \, U_k(s_i)$, where, w_k is the weight that reflects the significance of each utility which takes a value in [0, 1]. To send the message, the node selects the best neighbor that has the best trade-off between the different considered utilities defined through an arbitrary function *f( )*. More specifically, node *i* chooses the neighbor *j* in its neighbor set $\aleph_i$ that has the maximum utility:

$$arg \max_{j \in \aleph_i} \left\{ f\left(U(s_j)\right) = f\left(\sum_k w_k U_k(s_j)\right) \right\} \quad (1)$$

CAR/SCAR define the node's context via three measures: (1) its colocation with the sink(s), (2) its degree of connectivity, and (3) its remaining battery capacity. The respective utility functions are $U_{coloc}(s_i)$, $U_{cdc}(s_i)$ and $U_{bat}(s_i)$ that are used in conjunction with the weights $w_{coloc}(s_i)$, $w_{cdc}(s_i)$ and $w_{bat}(s_i)$ to formulate the SCAR context as

$$U(s_i) = w_{coloc} U_{coloc}(s_i) + w_{cdc} U_{cdc}(s_i) + w_{bat} U_{bat}(s_i) \quad (2)$$

Note that $U_{bat}(s_i)$ is equal to 1 when the battery level is the full battery capacity, and it decreases until it reaches 0 when the battery is completely drained. Meanwhile, $U_{coloc}(s_i)$ takes only two values: 1 when the neighbor is collocated with a sink and 0 otherwise. $U_{cdc}(s_i)$ reflects the normalized change in the number of neighbors $N_i = |\aleph_i|$ over the last period [$t-1$, t] due to node mobility according to the following relationship [6]:

$$U_{cdc}(s_i) = \frac{\left|N_{i_{t-1}} \cup N_{i_t}\right| - \left|N_{i_{t-1}} \cap N_{i_t}\right|}{\left|N_{i_{t-1}} \cup N_{i_t}\right|} \quad (3)$$

The $U_{coloc}(s_i)$ and $U_{cdc}(s_i)$ utilities are inapplicable in stationary networks, and hence, new utilities are needed.

A. *Why SCAR utilities are not suitable for static networks?*

The colocation utility $U_{coloc}(s_i)$ in SCAR indicates the connectivity of the nodes with the sink. This utility takes binary values: either 0 when the node is not within range of the sink or 1 when the node has a direct connection with the sink. In mobile WSNs, $U_{coloc}(s_i)$ is changing over time for a given node as it approaches/moves away from the sink. Hence, such a binary measure can be used. In contrast, nodes in a static WSN have fixed relationships with the sink node which does not change with time. A new measure is needed to reflect the relative location of such static nodes with respect to the sink node. The new measure should help the node to select a neighbor that is one step closer to the sink than itself to forward its message. Hence, unnecessary packet transmissions are avoided and loops are prevented.

Likewise, the degree of connectivity utility $U_{cdc}(s_i)$ in SCAR is designed to explicitly reflect the degree of mobility of the node in mobile WSNs. By definition, $U_{cdc}(s_i)$ reaches its maximum value when the node is attached to a brand new set of nodes and leaves all the previously attached nodes. Meanwhile, the value of $U_{cdc}(s_i)$ is at its minimum when the node keeps the connections with all its last state nodes and does not connect with any new nodes. This measure was designed to indicate the node's ability to reach the sink based on this hypothesis: a highly mobile host is a good carrier as it meets many hosts. However, $U_{cdc}(s_i)$ does not impact the utility function in static networks as nodes do not move (i.e., $N_{i_{t-1}} = N_{i_t}$ for all values of t). Hence, this measure does not apply to static networks.

We conclude that only the available battery capacity reflected through $U_{bat}(s_i)$ of the original SCAR can be extended to static network, and new metrics of the node's utility are needed to cope up with the stationary nature of nodes.

III. Context-Aware Opportunistic Resource-based Routing Protocol (CORB)

In this section, we propose the Context-Aware Opportunistic Resource-Based Routing Protocol (CORB) protocol and present its key design choices. Unlike existing probabilistic context-aware approaches which target mobile WSN, CORB is specifically designed for static sensor networks by presenting a new set of utility functions that suite static WSNs.

A. *Proposed CORB Utilities*

We develop new measures of the total utility function that matches the nature of static WSNs. The design goal is to introduce new measures that allow a node in the static WSN to precisely select the best carrier instead of the original SCAR functions – which only fit mobile WSNs.

The proposed CORB utility function is computed by considering the node's (1) battery level, (2) rank (the order of the node with respect to the sink), (3) connectivity strength with the surrounding nodes and (4) its buffer space. The respective utilities of these measures are $U_{bat}(s_i)$, $U_{rank}(s_i)$, $U_{link}(s_i)$ and $U_{buf}(s_i)$. Hence, the total utility is calculated as follows:

$$U(s_i) = w_{bat} U_{bat}(s_i) + w_{rank} U_{rank}(s_i) + w_{link} U_{link}(s_i) + w_{buf} U_{buf}(s_i) \quad (4)$$

where $\sum_k w_k = 1$ and $U_k(s_i) \epsilon [0,1]$.

Each node maintains a routing table containing the surrounding nodes' information. Unlike SCAR routing table which has two elements per neighbor $\{NodeId, DeliveryProbability\}$, each entry in the CORB routing table consists of four elements per neighbor s_j: $\{NodeId, DeliveryProbability, NodeRank, LinkStrength\}$, where $NodeId$ is the node identifier, $DeliveryProbability$ is the probability of node s_j to deliver the message to the sink, $NodeRank$ is the rank of node s_j relative to the sink, and $LinkStrength$ indicates the connectivity of node s_i with the neighbor s_j. We next explain CORB proposed utilities.

1) *Rank Utility –* $U_{rank}(s_i)$

This utility function takes into consideration the relative location of the node with respect to the sink. Such a utility measure improves the delivery probability of the packets and alleviates potential loops and unnecessary message exchanges since each node decides to forward its messages to the next node that brings the message one step closer towards the sink without looping.

Each node calculates the rank as follows. The sink broadcasts its rank with respect to itself (which is equal to 0) in the beacon messages. The neighbors that receive this value consider their new rank as the received value incremented by one. The neighbors then broadcast their new rank values to their neighbors, and the cycle goes on until all nodes in the network compute their rank. The node's rank is calculated locally as

$$NodeRank(s_i) = min\{NodeRank(s_j)\} + 1 \quad \forall j \in \aleph_i \quad (5)$$

where j represents an index of a node's neighbor that is currently in its routing table. Hence the rank utility function is defined as

$$U_{rank}(s_i) = \frac{1}{NodeRank(s_i)}, NodeRank(s_i) \neq 0 \quad (6)$$

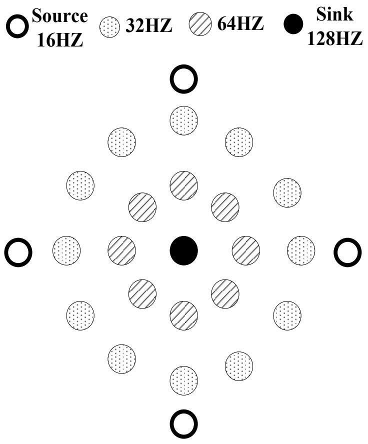

Fig. 1. Sink-centered circular topology in which the black node is the sink with 128 Hz radio duty-cycle, the dotted and dashed nodes are intermediate nodes that generate traffic with 64 Hz and 32 Hz duty cycles, respectively, and the white nodes are the main data sources with 16 Hz duty-cycle.

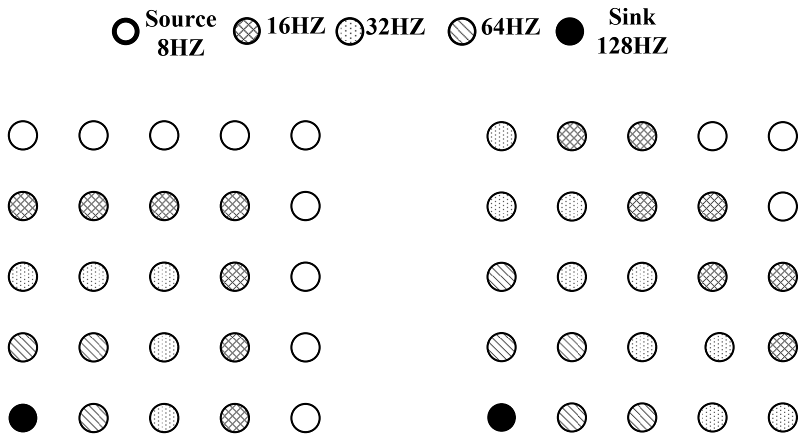

Fig. 2. The first grid topology in which the black node is the sink with 128 Hz radio duty-cycle, the dashed, dotted and double dashed nodes are intermediate nodes that generate traffic with 64 Hz , 32 Hz and 16 Hz duty cycles, respectively, and the white nodes are the sources with 8 Hz duty-cycle.

Fig. 3. The second grid topology in which the black node is the sink with 128 Hz radio duty-cycle, the dashed, dotted and double dashed nodes are intermediate nodes that generate traffic with 64 Hz , 32Hz and 16 Hz duty cycles, respectively, and the white nodes are the sources with 8 Hz duty-cycle.

2) *Link strength Utility* – $U_{link}(s_i)$

We introduce this measure to reflect the node's connectivity with the surrounding nodes that implicitly reflects the node's capability of forwarding the message. We define the link strength of the node as

$$U_{link}(s_i) = \frac{\sum_{j=1}^{N_i} LinkStrength(s_j)}{N_i}, U_{link}(s_i) \in [0,1] \quad (7)$$

where N is the number of the surrounding nodes and $LinkStrength(s_j)$ represents the strength of the connection between the node s_i and its neighbor node s_j. The $LinkStrength(s_j)$ value increases if s_i is regularly connected with s_j (i.e., s_i periodically receives beacons from s_j), and it decreases when s_i occasionally or non-periodically receives beacons from s_j due to MAC issues or interference. We linearly update $LinkStrength(s_j)$ for each node in the routing table. For instance, within a certain period of time, if node i receives a beacon message from node j, $LinkStrength(s_j)$ increment by 1, otherwise, it is decremented by 1. We limit $LinkStrength(s_j)$ to values in [0,5] then normalize it by 5 before using it in (7) to compute $U_{link}(s_i)$.

3) *Buffer Space Utility* – $U_{buffer}(s_i)$

Unlike the original SCAR, we consider the node's buffer free space as one of the node's resources besides the battery. We define a utility function $U_{buffer}(s_i)$ that acts as a measure of the average free space left at the node's message buffer. This measure improves the CORB performance as it decreases buffer overflow. Our proposed buffer utility function is:

$$U_{buffer}(s_i) = \frac{BufferSize - AverageUsedSpace}{BufferSize} \quad (8)$$

where $BufferSize$ and $AverageUsedSpace$ are the total buffer size and the three-point-moving-average of the used buffer space, respectively.

IV. Performance Evaluation

In this section, we evaluate the performance of the proposed CORB compared to the original SCAR as well as RPL: the de-facto standard for routing in IPv6 based WSN networks [8]. RPL builds its forwarding mechanism based on a directed graph formed through an objective function that considers the node's rank in the decision making process.

A. Simulation Setup

Our work targets ambiance control in smart buildings. Such a stationary WSN application typically follows a multi-hop star topology in which the different nodes spread all over the building collect the ambiance information and direct it to a centralized sink node that is responsible for ambiance control. We consider three representative topologies that are commonly encountered in stationary WSN applications such as smart building control: sink-centered circular topology shown in Fig. 1, and two rectangular grid topologies in which the sink is at the corner as shown in Fig. 2 and Fig. 3. Furthermore, we evaluate the performance of the proposed scheme in random topologies. In each topology we compare the performance of the proposed CORB protocol against both the original SCAR and the IPv6 RPL.

Each of the considered topologies is composed of 25 nodes. We use ContikiMAC [11] as a common radio duty cycling protocol for evaluating the three routing protocols. Nodes with the same shape have the same duty-cycle, and the duty cycle frequency is halved as we move further from the sink. We arbitrarily set the buffer space to 8 units for all nodes in the network. We use COOJA [13], a cross-level WSN simulator for the Contiki Operating System, for our simulation experiments. The radio range for each node is 50 meters. In the topologies depicted in Fig. 1 and Fig. 2, and the random topologies we generate the main traffic (the x-axis of all the figures below) by the outer (white) nodes, while the intermediate nodes generate $1/8$ of the main traffic (i.e. if the white nodes generates 8 messages/second then the intermediate nodes generate 1 message/second). Unlike the grid topology in Fig. 2 which has only 3 highly active nodes near the sink node, the grid topology in

Fig. 3 has more highly active nodes near the sink, and hence, more options to choose from to forward a message. To better assess the performance in such a grid topology we increase the offered traffic such that all the intermediate nodes generate the same traffic as the outer nodes. We assign equal weights to the different CORB utilities defined in (4) as the case with SCAR.

Our performance metrics are: 1) the delivery ratio which is the ratio between the number of received messages and the number of sent messages; and 2) the time to first failure (TTFF) which is the time at which the first node completely drains its battery.

B. Sink-Centered Circular Topology

Fig. 4 shows the significant gain in the delivery ratio that CORB achieves in static WSNs (the delivery ratio approaches 100%). Moreover, the delivery ratio outperforms SCAR and RPL for all data rates. At the high data rate (8 messages/sec), the gain in the delivery ratio of CORB is 143% and 46.6% over SCAR and RPL, respectively. At lower data rate, the average CORB gain is 40.75% and 14% over SCAR and RPL, respectively. Fig. 5 shows that the CORB always outperforms RPL and is either better or slightly inferior to SCAR in terms of the TTFF.

C. Grid Topologies

Next, we evaluate CORB in the grid topologies shown in Fig. 2 and Fig. 3. For the first grid topology shown in Fig. 2, Fig. 6 shows that CORB maintains its performance advantage despite the deterioration in the delivery ratio of both CORB and RPL compared to the sink-centered circular topology. The gain in the delivery ratio of CORB with respect to RPL increased to be up to 296% at data rate of 8 messages/second. This gain demonstrates the ability of CORB to preserve a high delivery ratio even in such a bottlenecked topology with only three highly active forwarding neighbors of the sink (the three dashes node) given a high offered load (as there is nine white nodes). CORB's superior performance is attributed to: (1) The buffer utility function, U_{buffer}, that allows the nodes to instantaneously choose one of the three bottlenecked nodes that surround the sink based on their remaining buffer space. Consequently, fewer packets will be dropped as compared to RPL. (2) CORB's U_{rank} that enables the intermediate nodes to direct the messages towards the sink by choosing a next carrier that brings the message one step closer towards the sink without looping. This contrasts with SCAR's U_{coloc} and U_{cdc} which fail to direct messages effectively to the sink as the number of intermediate nodes increase and lead to routing loops as discussed in section III. The delivery ratio gain CORB achieves with respect to SCAR is 404%. Fig. 7 indicates that the TTFF of SCAR is slightly higher than that of CORB. This is misleading as the three nodes near the sink rarely received messages to deliver to the sink because of routing loops. Such nodes – which are typically the first to be battery depleted in CORB and RPL – have longer lifetimes in SCAR at the expense of a very low delivery ratio.

In the second grid topology depicted in Fig. 3, all nodes generate the same amount of traffic. However, such a grid topology has more highly active nodes near the sink. Fig. 8 and Fig. 9 show that CORB outperforms RPL by 235% increase in the delivery ratio – with also a significantly better TTFF performance in this topology. Meanwhile, the delivery ratio of SCAR is still significantly low with higher TTFF.

D. Random Sink-Centered Topologies

Finally, we consider 10 random sink-centered topologies – each containing 24 nodes - to demonstrate the generality of CORB. As shown in Fig. 10 and Fig. 11. The delivery ratio of CORB is greater than RPL and SCAR by up to 58% and 200%, respectively, at high data rates. We conclude that CORB superior performance is not dependent on the network topology.

V. Conclusions and Future Directions

In this paper, we have presented the CORB routing protocol for stationary WSNs. Unlike existing context-aware routing protocols – designed for mobile WSNs, we have introduced new context measures that are appropriate for static WSNs. Simulation results have shown that CORB outperforms existing protocols such as SCAR and RPL in terms of delivery ratio and network lifetime by reducing the consumption of nodes' batteries. This is due to CORB ability to successfully provide effective measures that increase the awareness of the available resources at the neighboring nodes which results in significantly higher delivery ratios. We plan to do a formal optimization of the CORB weights for the different topologies. We also plan to implement and evaluate the performance of the proposed CORB in real test bed for smart building control to demonstrate its significant performance gains in real life settings.

Acknowledgements

This work is part of the ITEA2 Web of Objects Project funded in Egypt by the Egyptian Ministry of Communication and Information Technology.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787-2805, Oct. 2010.

[2] J.N. Al-Karaki, and A.E.Kamal,"Routing techniques in wireless sensor networks: A survey," IEEE Wireless Communications, vol.11, no.6, pp.6-28, Dec. 2004.

[3] R. J. D'Souza, and J. Jose, "Routing approaches in delay tolerant networks: A survey," International Journal of Computer Applications, vol. 1, no.17, pp.8-14, Feb. 2010.

[4] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," Mobile Computing and Communications Review, vol. 7, no. 3, July 2003.

[5] M. Musolesi, and C. Mascolo, "CAR: context-aware adaptive routing for delay-tolerant mobile networks," IEEE Mobile Computing, vol. 8, no. 2, pp.246-260, Feb.2009.

[6] C. Mascolo, and M. Musolesi, "Opportunistic Mobile Sensor Data Collection with SCAR," In Proc. of IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2007). Pisa, Italy, Oct. 2007.

[7] Y. Wang, H. Wu, and H. Dang. "Analytic study of delay/fault-tolerant mobile sensor networks (DFT-MSN's)," In Proc. of World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2009.

[8] T. Winter and P. Thubert, "RPL: IPv6 Routing Protocol for Low power and Lossy Networks, RFC 6550," IETF ROLL, Mar. 2012.

[9] A. Dunkels, B. Grönvall, and V. Thiemo, "Contiki a lightweight and flexible operating system for tiny networked sensors," in Proc. of IEEE Workshop on Embedded Networked Sensors, Tampa, FL, Nov. 2004.

[10] A. Dunkels, "Rime - a lightweight layered communication stack for sensor networks," in Proc. of the European Conference on Wireless Sensor Networks (EWSN), Delft, the Netherlands, Jan. 2007.

[11] A. Dunkels, "The ContikiMAC Radio Duty Cycling Protocol," Technical Report T2011:13, Swedish Institute of Computer Science, Dec. 2011.

[12] A. Dunkels, F. Osterlind, N. Tsiftes, and Z. He. "Software-based on-line energy estimation for sensor nodes," In Proceedings of the 4th workshop on Embedded networked sensors, pp. 28-32. ACM, 2007.

[13] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt. "Cross-level sensor network simulation with Cooja," In Proc. of 31st IEEE Conference on Local Computer Networks, 2006.

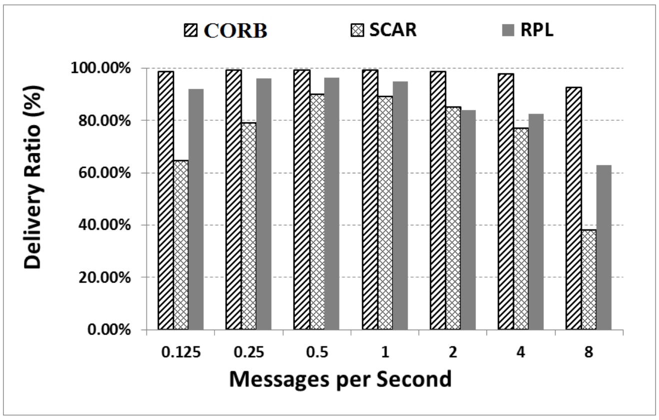

Fig. 4. Delivery ratio vs data rate for the sink-centric circular topology.

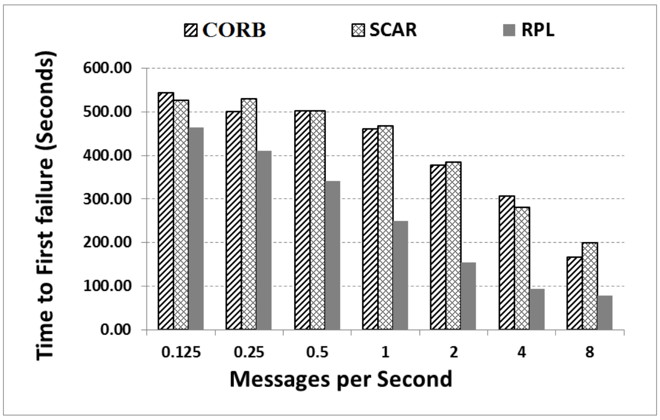

Fig. 5. TTFF vs data rate for the sink-centric circular topology.

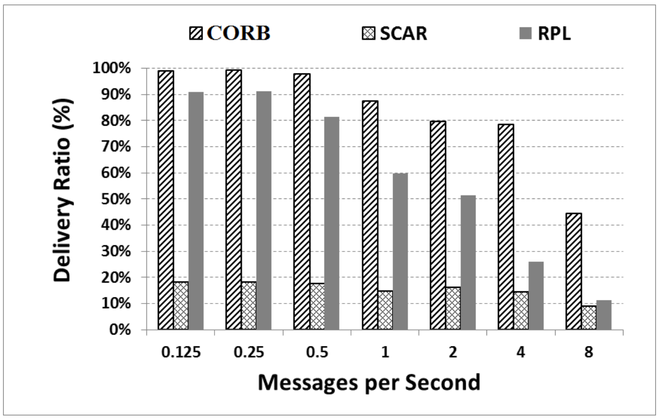

Fig. 6. Delivery ratio vs data rate for the grid topology in Fig. 2

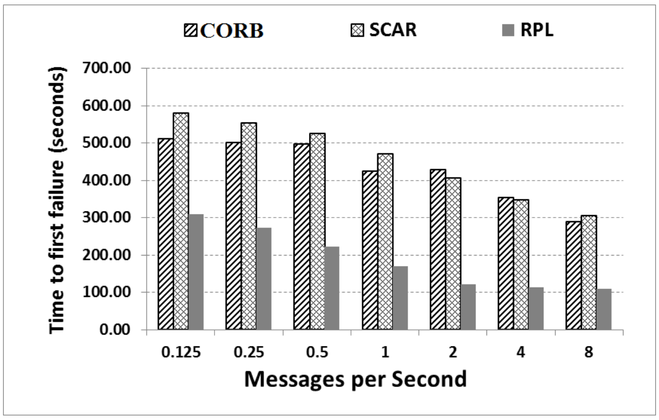

Fig. 7. TTFF vs data rate for the grid topology in Fig. 2

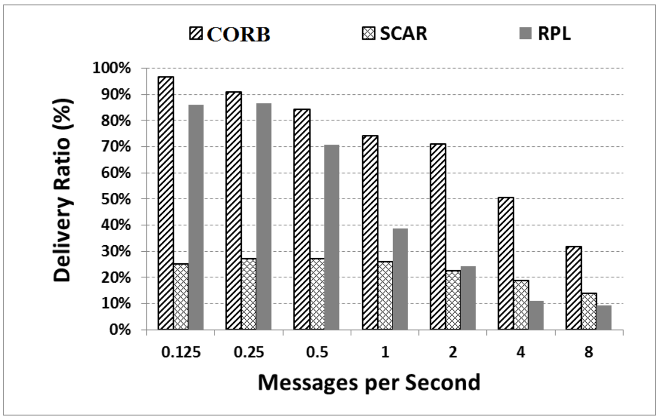

Fig. 8. Delivery ratio vs data rate for the gird topology in Fig. 3.

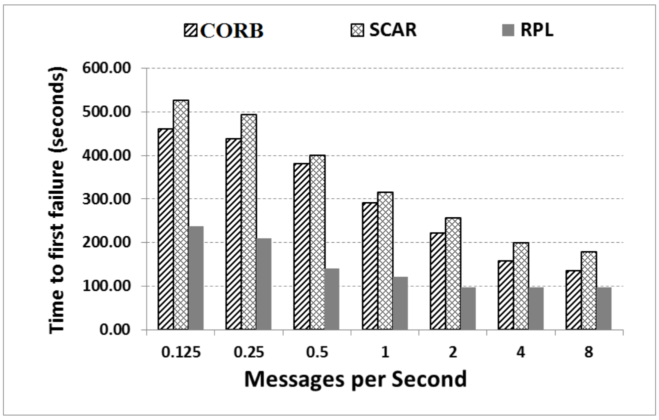

Fig. 9. TTFF vs data rate for the grid topology in Fig. 3**.**

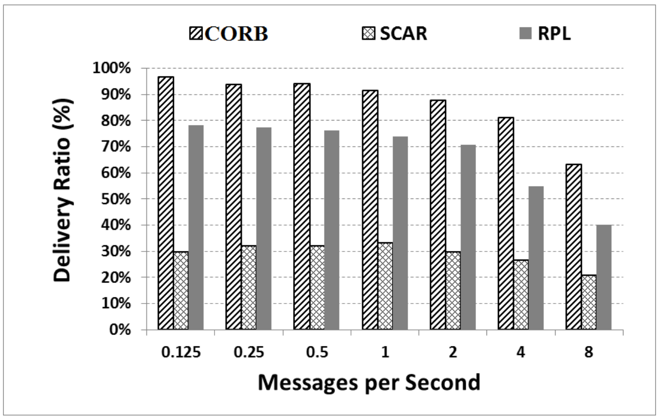

Fig. 10. Delivery ratio vs data rate for the sink-centered random topologies.

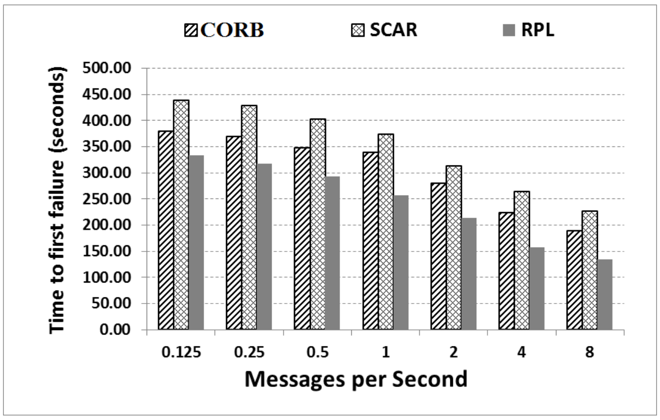

Fig. 11. TTFF vs data rate for the sink-centered random topologies.