

GRID-BASED PRIORITY ROUTING PROTOCOL FOR UWSNs

Faiza Al-Salti, N. Alzeidi, Khaled Day, Bassel Arafeh and Abderezak Touzene

Department of Computer Science, Sultan Qaboos University, Oman

ABSTRACT

In this paper, we devise and evaluate a new Grid-Based Priority Routing (GBPR) protocol for Underwater Wireless Sensor Networks (UWSNs). GBPR utilizes a 3D logical grid view of the monitored area to deliver data packets to sink nodes. Particularly, data packets are forwarded on a cell-by-cell-basis using elected sensor nodes called cell-heads. The unique feature in GBPR is the classification of the neighboring cells in different priority levels according to their distances to the sink node. Cells closer to the sink are given higher priority to be selected as the next hop. This mechanism helps in reducing the number of hops; thus, reducing the energy consumption and end-to-end delay, and increasing the reliability. The protocol is evaluated and compared against EMGGR and EEF protocols available in the literature. Simulation results show that GBPR outperforms the other two protocols in terms of energy efficiency, average delay and packet delivery ratio.

KEYWORDS

Underwater wireless sensor networks, UWSNs, routing protocol, location-based, grid, cell-head election & void-handling algorithm.

1. INTRODUCTION

Underwater wireless sensor networks (UWSNs) consist of a set of sensor nodes deployed in the underwater environment. These nodes are capable of sensing, detecting, tracking and reporting data about the monitored environment to specific nodes called sinks, which are located at the water surface. They can measure a variety of features such as temperature, salinity and pressure; thus, enable several applications such as environmental monitoring, undersea exploration, disaster prevention, and assisted navigation [1]. However, UWSNs experience a number of challenges induced by the nature of the environment and the used underwater acoustic communication. Particularly, the speed of acoustic signals underwater is five orders of magnitude slower than the speed of radio signals in terrestrial networks. In addition, underwater channels are severely weakened due to the multi-path and fading effects. Moreover, underwater sensor nodes are equipped with batteries of limited energy, and they are difficult to be replaced or recharged. Besides, the available bandwidth is limited and inversely proportional to the transmission distance. Furthermore, the three-dimensional (3D) deployment and the dynamic environment due to the passive movement of sensors with water currents cause extra challenges in the development of new schemes (e.g. routing and localization) for such UWSNs.

Several solutions have been proposed to solve different issues in UWSNs such as node's deployment, node's mobility, MAC, routing, and localization. Nodes' deployment, for example, is an important task in UWSNs because several network services like routing protocols, localization schemes and network topology control are built on top of it [2]. Generally, the objectives of node deployment schemes in UWSNs are achieving high network connectivity, high coverage, less number of nodes, low energy consumption and high delivery ratio [2]. G. Han et al. [3] have surveyed and classified deployment strategies in UWSNs. The classification is based on three categories namely; static deployment, self-adjustment and movement-assisted deployment schemes.

Several MAC protocols have been proposed for UWSN such as those found in [4], [5], [6], [7], and [8]. According to [9], MAC protocols for UWSNs can be classified into three categories: contention-based, contention-free and hybrid MAC protocols.

Because sensor nodes might be deployed at farther distances from the sink nodes, multi-hop routing is necessary to manage the communications between sensors and sink nodes with efficient use of the available resources (i.e. bandwidth, power and memory storage). General objectives of the proposed routing protocols are high energy-efficiency, low average delay and high packet delivery ratio. Though, these protocols face some issues like high energy-consumption in dense networks, low packet delivery ratio in sparse networks and generally high delay. Therefore, in this paper, we present a Grid-Based Priority Routing (GBPR) protocol for UWSNs that aims to mitigate the earlier mentioned challenges, and to achieve high energy-conservation, low average end-to-end delay and high packet delivery ratio under different network conditions.

GBPR utilizes a 3D grid view to deliver data packets from cell to cell via special nodes called cell-heads that are elected using an election algorithm. The election algorithm, as described in section 3, uses short control packets in order to reduce the network overhead. The main feature of the proposed protocol that differentiates it from the existing grid-based protocols is the classification of the neighboring cells into different priority levels according to their distances to the sink node. This classification facilitates the selection of the packets' forwarders. Basically, in each hop, neighboring cells that are closer to the destination cell than the current cell are favored to forward data packets.

The rest of this article is organized as follows. In section 2, different classifications of existing routing protocols in UWSNs along with some examples are discussed. Section 3 describes the proposed routing protocol. Evaluation and comparison results of the protocol are provided in section 4. Finally, the paper is concluded in section 5.

2. RELATED WORK

Several routing protocols have been proposed specifically for UWSNs. These protocols can be classified based on different criteria. For example, VBF [10], HH-VBF [11], VBVA [12], LE-VBF [13] and DFR [14] are typical location and flooding based routing protocols that aim to direct packets' propagation and reduce the number of forwarders as compared to uncontrolled flooding protocols. Thus, they reduce the number of collisions and improve the energy consumption.

Depth-based routing protocols rely mainly on depth information for selecting candidate forwarders. Depth information can be determined using inexpensive depth sensors [15]. Example of such protocols is the EEF [16] routing protocol, which is developed to achieve energy efficiency. Depth information along with residual energy and the distances to the sink node and to the previous forwarder are used to calculate the fitness value. This value is used to determine the best eligible forwarders. Each possible candidate should hold the packet for a certain time based on its depth, distances and energy before forwarding it. If the period expires without overhearing the transmission of that packet, it broadcasts the packet. Despite its simplicity, it faces void problem in highly sparse networks and severe collisions between packets in dense networks.

Grid-based geographic routing protocols for UWSNs have been proposed in [17], [18] and [19]. They all aim to improve the energy efficiency of the networks. Mainly, MGGR [17] and EMGGR [18] are based on gateways (i.e. elected sensor nodes) to forward data packets on a cell-by-cell basis. They route packets over disjoint paths constructed based on the grid view rather than the position of the nodes in order to reduce the need for paths' maintenance due to topology changes. However, low packet delivery ratio is achieved and high average delay is incurred in sparse networks.

The GFGD and GGFGD [19] are two other grid-based protocols for UWSNs. They incorporate a duty cycle (i.e. some nodes are scheduled to sleep for some time) mechanism, which can balance and save energy. In addition, they use path delay, path loss and remaining energy for relay selection. However, the protocols assume that the channel link is symmetric, which is not practical in underwater since acoustic channels in underwater are known to be asymmetric [10][20].

Although several routing protocols have been proposed for UWSNs, they are facing general issues: (i) high energy consumption in dense networks due to the overhead incurred by broadcasting a massive number of packets, (ii) low packet delivery ratio in sparse networks, (iii) high end-to-end delay, and (iv) low performance with node mobility. Our proposed solution in this paper strives to solve these issues as will be described and proved in the following sections.

3. THE PROPOSED ROUTING PROTOCOL

GBPR is a location-based routing protocol that advances a routed packet, in each hop, towards the sink nodes at the surface level. It is based on viewing the network as a 3D logical grid and the forwarding is performed in a cell-by-cell manner. Additionally, a specific set of sensor nodes, called cell-heads, are the only nodes eligible for data forwarding at any point in time. Since applications may employ more than one link to receive data packets, the closest link to the current forwarder is selected as a destination target. The proposed protocol also adapts a void handling technique when a void cell (i.e. a cell with no elected node) is encountered. The proposed mechanisms are expected to achieve high energy-efficiency, low average delay and high delivery ratio.

3.1 SYSTEM MODEL

3.1.1 ASSUMPTIONS

In this paper, we consider a 3D underwater wireless sensor network in which sensor nodes are distributed in a 3D area and the sink nodes are positioned at the surface level (see Figure 1). The following is assumed:

- The network consists of a set $N = N_s \cup N_n$ of sink and sensor nodes, where N_s is the set of sink nodes and N_n is the set of sensor nodes. All nodes including the sinks are of equal acoustic communication range R . The sink nodes in N_s are assumed to be stationary. In addition, they use acoustic channels to communicate with the underwater sensor nodes and radio channels to communicate with each other and with terrestrial stations.
- All data packets are forwarded to the sink nodes.
- Because of the rapid speed of radio propagation in the air compared to acoustic propagation under water, data packets received by any sink are assumed to be received by other sinks in a negligible time.
- The set N_n of the sensor nodes are assumed to have similar capabilities (e.g. transmission range, storage space and initial energy level).
- Each sensor node i is assumed to be able to obtain its physical location information in the Cartesian coordinates (X_i, Y_i, Z_i) from an existing localization service. It is also assumed to know the location of the sink nodes.

3.1.2 CONSTRUCTION OF THE LOGICAL GRID

The geographic region is divided into 3D logical grids as shown in Figure 2. Cells are viewed as cubes of equal volume d^3 , where d is the length of the cell side. The number of cells along i -axis (where i is one of the three dimensions x, y, z) is known as K_i (e.g. K_x is the number of cells

along the x-axis), which can vary from one axis to another. The physical location of the grid origin, which is assumed at the top left of the surface, is denoted by (X_0, Y_0, Z_0) .

The node uses its location information, the location of the grid origin (X_0, Y_0, Z_0) and the value of d to determine the XYZ coordinates of its cell (x_i, y_i, z_i) as given in (1). Note that the absolute physical location of a node i is referred to by the upper case (X_i, Y_i, Z_i) , while the grid coordinates of the cell in which it is hosted are denoted by the lower case (x_i, y_i, z_i) . Every sensor node has a unique ID (NID) and a cell ID (CID). CID is a number that identifies the cell that the sensor node belongs to, which can be determined from the XYZ coordinate of the cell (2).

$$(x_i, y_i, z_i)_i = \left(\left\lfloor \frac{X_i - X_0}{d} \right\rfloor, \left\lfloor \frac{Y_i - Y_0}{d} \right\rfloor, \left\lfloor \frac{Z_i - Z_0}{d} \right\rfloor \right) \quad (1)$$

$$CID_i = x_i + K_x * (y_i + K_y * z_i) \quad (2)$$

The transmission range R of the sensors is used to compute the value of cell side, such that a node in a cell can communicate directly with any node in its 32 neighboring cells (i.e. cells that have a common vertex, edge or face with it; or cells that have a common face with those having a common face with the original cell) as shown in Figure 3. This can be achieved when the communicating nodes are located at the farthest diagonally apart corners of the neighboring cubes. As illustrated in Figure 4, d should be selected to satisfy $R = 2d\sqrt{3}$.

The 32 neighboring cells are: $(x-1, y-1, z-1)$, $(x-1, y-1, z)$, $(x-1, y-1, z+1)$, $(x-1, y, z-1)$, $(x-1, y, z)$, $(x-1, y, z+1)$, $(x-1, y+1, z-1)$, $(x-1, y+1, z)$, $(x-1, y+1, z+1)$, $(x, y-1, z-1)$, $(x, y-1, z)$, $(x, y-1, z+1)$, $(x, y, z-1)$, $(x, y, z+1)$, $(x, y+1, z-1)$, $(x, y+1, z)$, $(x, y+1, z+1)$, $(x+1, y-1, z-1)$, $(x+1, y-1, z)$, $(x+1, y-1, z+1)$, $(x+1, y, z-1)$, $(x+1, y, z)$, $(x+1, y, z+1)$, $(x+1, y+1, z-1)$, $(x+1, y+1, z)$, $(x+1, y+1, z+1)$, $(x-2, y, z)$, $(x+2, y, z)$, $(x, y-2, z)$, $(x, y+2, z)$, $(x, y, z-2)$, $(x, y, z+2)$.

It is worth noting that a cell can communicate partially with other cells apart from these 32 neighbors; however, for the sake of simplicity, these are not considered as neighbors. Algorithm 1 is used to check whether a given cell is one of the 32 neighboring cells or not.

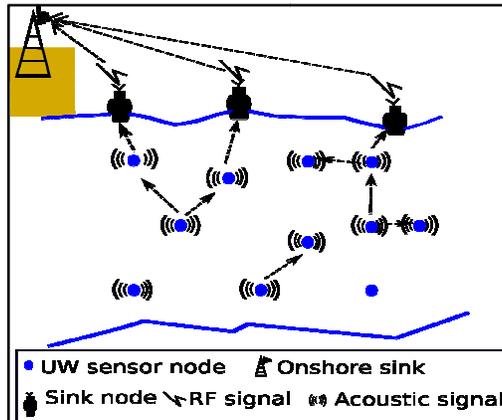


Figure 1: 3D UWSN architecture

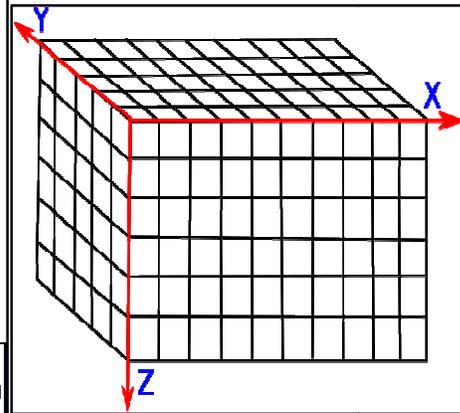


Figure 2: A view of 3D logical grid

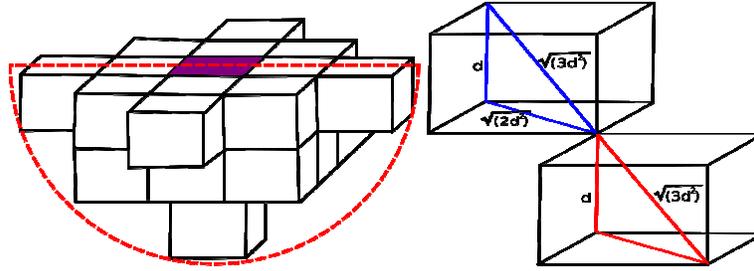


Figure 3: A cross section of the 32 neighboring cells Figure 4: Finding the value of the cell side

```

Procedure: isNeighbor (Cx, Cy, Cz, Nx, Ny, Nz)
//for checking if cell (Nx, Ny, Nz) is a neighbor of current cell (Cx, Cy, Cz)

x=|Cx- Nx|
y= |Cy- Ny|
z= |Cz- Nz|
if((x=0) & (y=0) & (z=0))
return false; //local cell
else if (((x≤1) & (y≤1) & (z≤1)) Or ((x=2) & (y=0) & (z=0)) Or ((x=0) &
(y=2) & (z=0)) Or ((x=0) & (y=0) & (z=2)))
return true; //neighbor cell
else
return false; //neither local nor neighbor
    
```

Algorithm 1: Procedure for checking a neighboring cell

3.2 NOTATIONS

We use the notations defined in Table 1 to describe the proposed GBPR protocol.

Table 1: Notations used in the protocol’s description

Symbol	Definition
D	The length of the cell side
R	Node’s transmission range
K_x, K_y, K_z	Number of cells along the x-axis, y-axis and z-axis respectively
<i>Cell-head</i>	A node elected to forward data packets from the corresponding cell
<i>Head-table</i>	A table used to store the cell-heads in the neighboring cells
NID	Unique node identifier
CID	Unique cell identifier
E	The minimum amount of energy to continue serving as a cell-head
<i>Head-packet</i>	A packet sent by a node upon electing itself as a cell-head
<i>Update-packet</i>	A periodic packet sent by a cell-head to inform neighbors about its existence
<i>Retire-packet</i>	A packet sent by a cell-head when its remaining energy falls below the threshold ϵ
<i>Exit-packet</i>	A packet sent by a cell head upon roaming out of its cell
<i>Update-timer</i>	A timer set by each cell-head to periodically inform neighbors about its existence by sending Update-packet
<i>Election-timer</i>	A timer set by non-cell head nodes to start an election process
<i>Local cell</i>	The cell in which the node resides in
<i>Neighbor cell</i>	A cell defined according to Algorithm 1 and illustrated in Figure 3

3.3 DETERMINING THE NEAREST SINK CELL

Recall that there might be more than one sink node deployed for receiving data packets. In that case, each node should construct its head-table relative to the nearest sink. A sensor node determines its nearest sink based on the Euclidean distances between the cell containing the sensor node and the cells containing the sink nodes.

3.4 CLASSIFICATION OF THE NEIGHBORING CELLS

Remember that a node in a cell i can have up to 32 neighboring cells; yet, these cells differ in their distances to the sink cell (the nearest sink). In other words, some of the cells are closer to the sink than the cell i and some are farther from the sink. However, to achieve low average end-to-end delay and to save the overall energy, cells that are closer to the sink cell are given higher priority to be selected for packet relay. Thus, neighboring cells are ordered in different priority levels according to their distances to the sink cell. To explain these levels, assume that the source cell i , the sink cell s , the neighboring cell to be classified n have the coordinates (x_i, y_i, z_i) , $(x_s, y_s, 0)$ and (x_n, y_n, z_n) , respectively. Let $\delta m_{jk} = |m_j - m_k|$ (where m is either x , y or z) be the distance along the m -axis between cell j and cell k .

- Vertical positive level (group G1): This set $G1$ of neighboring cells includes those cells that are closer to the sink cell along the z dimension, and closer or equal to the sink cell along each of the x and y -dimensions. In other words, if the δz_{ns} is less than δz_{is} , and δx_{ns} and δy_{ns} is less than or equal to δx_{is} and δy_{is} , respectively, then the neighbor cell is classified as a vertical positive neighbor cell and placed in this group $G1$.
- Horizontal positive level (group G2): This set $G2$ of neighboring cells includes those cells that are closer to the sink cell along the x dimension or y dimension, and the remaining are equal to that of the source cell. In other words, if one or both of δx_{ns} and δy_{ns} is, respectively less than δx_{is} and δy_{is} , and the remaining of δx_{ns} , δy_{ns} and δz_{ns} are equal to the corresponding δx_{is} , δy_{is} and δz_{is} , then the neighbor cell falls in this group $G2$.
- Positive and negative level (group G3): This set of cells includes those cells that are closer to the sink cells along one or more dimensions. However, it is farther to the sink than the current cells along the other dimensions. That is if any of δx_{ns} , δy_{ns} and δz_{ns} is less than δx_{is} , δy_{is} and δz_{is} , respectively and the remaining are greater than the corresponding δx_{is} , δy_{is} and δz_{is} , then the neighbor cell falls in this group $G3$.
- Negative level (group G4): This set of cells includes those cells that are farther from the sink cells along one or more dimensions while the other dimensions remain equal to that of the current cell. In other words, if any of δx_{ns} , δy_{ns} and δz_{ns} is greater than δx_{is} , δy_{is} and δz_{is} , respectively, and the remaining are equal to the corresponding δx_{is} , δy_{is} and δz_{is} , then the neighbor cell is placed in this group $G4$.

Algorithm 2 further defines these levels. The head-table of each cell-head is divided into four disjoint sub-tables according to the above priority levels. Figure 5 shows a general structure of the head-table. It is worth mentioning that the classification of neighboring cells into the priority levels takes place while constructing the head-tables. Particularly, when a cell-head receives a head-packet or an update-packet (see section 3.5) from one of the neighboring cells, it checks the priority level of that cell and accordingly stores the information in the relevant sub-table.

When a node enters a new cell, the priority levels of the neighboring cells might change. In addition, some of the cells may no longer be neighbor cells and need to be removed from the head-table. Furthermore, new cells are becoming neighboring cells for that node. Thus, when the node moves out from its cell, it updates its head-table based on the new cell it enters.

Figure 6 illustrates these priority levels for the 2D grid. The same idea can be extended to the 3D grid. To further illustrate this, assume that the source node is in the cell $S(2,2)$ and the sink node is in the cell $D(5,6)$. Figure 6 shows 12 neighboring cells for S . Considering the neighboring cell with xy-coordinates $(3,2)$, it is clear that the cell is closer to the sink cell than S along the x-dimension. However, the y-dimension is same as that for S . Thus the cell-head in S puts this cell in the second priority group ($G2$). Another example is the cell with xy-coordinate $(1,3)$. This cell is closer to the sink in the x-dimension, but farther from the sink than S in the y-dimension. Therefore, this cell is placed in the third priority group ($G3$). The priority levels of the 12 neighboring cells are shown inside each cell into the three priority levels. Note that there is no cell in $G1$ in the figure since the source and sink cells are in the same z-level.

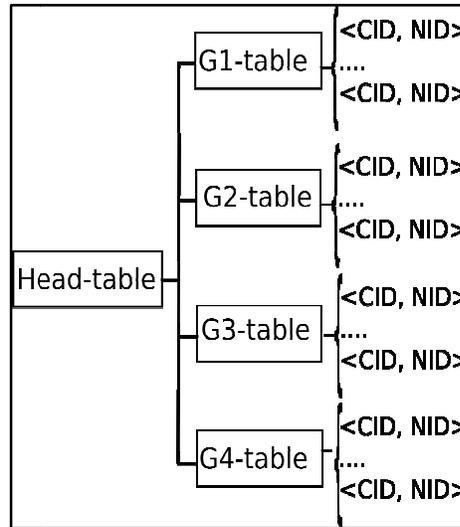


Figure 5: The structure of the head-table

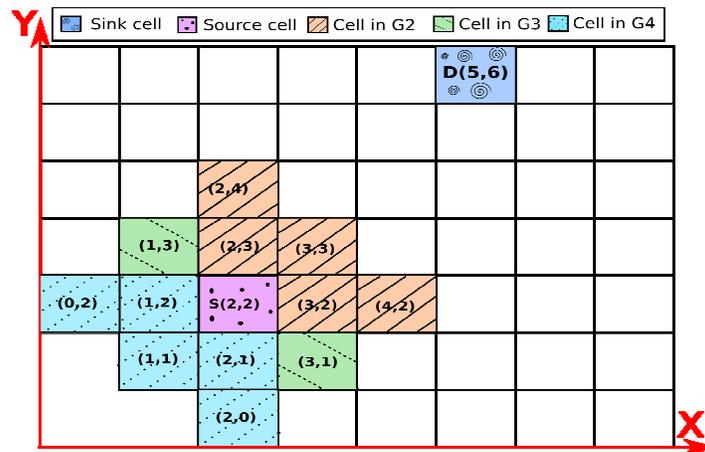


Figure 6: Demonstration of the priority levels in a 2D grid

```

Procedure: classifyCell(nx, ny, nz, ix, iy, iz, sx, sy, sz)
// (nx,ny,nz) => xyz-coordinates of the neighbor cell
// (sx,sy,sz) => xyz-coordinates of the sink cell
// (ix,iy,iz) => xyz-coordinates of the current cell

 $\delta x_{is} = ix - sx$ 
 $\delta y_{is} = iy - sy$ 
 $\delta z_{is} = iz - sz$ 
 $\delta x_{ns} = nx - sx$ 
 $\delta y_{ns} = ny - sy$ 
 $\delta z_{ns} = nz - sz$ 
if ( $\delta x_{ns} \leq \delta x_{is}$ ) && ( $\delta y_{ns} \leq \delta y_{is}$ ) && ( $\delta z_{ns} < \delta z_{is}$ )
    return G1; //vertical positive level
else if ( $\delta x_{ns} \leq \delta x_{is}$ ) && ( $\delta y_{ns} \leq \delta y_{is}$ ) && ( $\delta z_{ns} == \delta z_{is}$ )
    return G2; //horizontal positive level
else if ( $\delta x_{ns} \geq \delta x_{is}$ ) && ( $\delta y_{ns} \geq \delta y_{is}$ ) && ( $\delta z_{ns} \geq \delta z_{is}$ )
    return G4; //negative level
else
    return G3; //positive and negative level

```

Algorithm 2: Procedure for classifying a neighboring cell

3.5 CELL-HEAD ELECTION

We present in this section a cell-head election algorithm for electing cell-heads, which will be responsible for forwarding data packets. Using cell-heads has the objective of reducing the number of data packets propagated in the network. Particularly, each cell-head is responsible for forwarding packets from its cell to a neighboring cell. In order to reduce the overhead that can be incurred by the election algorithm, the proposed election algorithm uses only few and small control packets. The adopted algorithm is based mainly on the residual energy of the nodes. Basically, in a given cell each node with residual energy above a predefined threshold ε can compete for being the head of the cell. Each node, including non-heads, records the cell-heads in the local and neighboring cells in a table called *head-table*, as defined earlier. The election algorithm consists of two phases; an initialization phase and a maintenance phase as described below.

3.5.1. INITIALIZATION PHASE

- After deployment, sensor nodes in each cell select a random time to initiate an election process. Each node broadcasts a *head-packet* $\langle type, CID, NID \rangle$, where *type* indicates the type of the packet (*head-packet* in this case), *CID* is the cell ID that the node belongs to, and *NID* is the ID of the node sending the packet. In addition, the node records itself as the head of that cell, and sets an *update-timer* to send an *update-packet* upon timer expiration.
- Each node, upon receiving a *head-packet* from a local or a neighboring cell, stores the head information in its *head-table*. Furthermore, local nodes stop competing for being the heads for the current period. However, they set *election-timers* to compete for cell-head in the next period. If more than one head-packet is received from the same cell (due, for example, to two nodes initiating the election process at nearly the same time), then the node corresponding to the last received head-packet is assumed the cell-head.

3.5.2. MAINTENANCE PHASE

- When the *update-timer* of a cell-head expires, it checks its remaining energy. If it is above the threshold ε , the node broadcasts an *update-packet* $\langle type, CID, NID \rangle$, and continues serving as a cell-head for the next period. Local and neighboring nodes update their head-tables accordingly. Otherwise, if its energy is below the threshold, it broadcasts a *retire-packet* $\langle type, CID, NID \rangle$ to inform local nodes to start a new election.
- Local and neighboring nodes upon receiving a *retire-packet* remove the information related to that node from their head-tables. In addition, local nodes with energy levels above the threshold ε start a new election process as given in the initialization phase.
- When a head node exits from its cell, it broadcasts an *exit-packet* $\langle type, CID, NID \rangle$. Upon receiving that packet, local and neighboring nodes remove the information related to the source node of that packet from their head-tables. Also, local nodes start a new election process as described in the initialization phase.
- When an *election-timer* of a node expires, the node sets itself as a cell-head and broadcasts a *head-packet*. Local and neighboring nodes when receiving that packet set that node as the cell-head. In addition, the head in the local cell, if any, cancels its *update-timer*.

3.6 RELAY SELECTION AND PACKET FORWARDING

Each data packet consists of *current NID*, *next NID*, *crossed-cells* and *payload*. *Current NID* is the ID of the current forwarder. *Next NID* is the ID of the forwarder in the next hop, *crossed-cells* is a sequence of CIDs of the cells that the packet already crossed, and the *payload* is the information to be delivered to the sink nodes. Figure 7 demonstrates a general format of the data packet.

When a node has a data packet to be forwarded, it looks for a cell-head in the first priority group in a round robin fashion. It selects a cell from that group given that it has a cell-head. Once the cell-head is selected, the node updates the packet header by setting *current NID* to its ID and *next NID* to the ID of the selected node, and appending its CID to the end of the *crossed-cell*. Then, it sends the packet to the selected node. This forwarding process continues until the packet is delivered to the sink node. If the first group is empty (i.e. does not contain any cell-head), it searches for a cell-head in the second group in a similar way (i.e. round robin). If there is no cell-head in the second group, it looks for a cell-head in the third group and then in the fourth group. Once the cell-head is selected, the node checks that the selected node is not hosted in a cell contained in the *crossed-cell* of the packet. If it is the case, then the node updates the packet header by setting *current NID* to its ID and *next NID* to the ID of the selected node and appending its CID to the end of the *crossed-cell*. Then, it sends the packet to the selected node. Otherwise, the node looks for another node to be selected. This check is used to ensure a loop-free forwarding. The process continues until the packet is delivered to the sink node.

If at any hop, the forwarder could not find any cell-head in all its neighboring cells (i.e. throughout the paper, this node is called a void node and its cell as a void cell), it sends a negative acknowledgment to the previous forwarder. The previous forwarder, in turn, looks for another cell (other than that cell) containing a node to forward the packet to it, in the same way, explained earlier.

Current NID	Next NID	crossed-cells	payload
----------------	-------------	---------------	---------

Figure 7: Format of the data packet

4. SIMULATION RESULTS

4.1 SIMULATION SETTINGS

In this section, we evaluate the performance of our proposed routing protocol against EEF and EMGGR routing protocols (summarized in section 2) via Aqua-Sim [21] simulator. Aqua-Sim is a simulation package developed specifically for UWSNs, and it is based on NS2 simulator. Sensor nodes are deployed in a 3D area of size $(3 \times 3 \times 3)$ km³. Sink nodes are located at the surface level and are assumed to be stationary. The *broadcastMac* protocol [10] is used in all scenarios. The idea of this MAC protocol is as follows: when a node has a packet to be transmitted, it senses the channel and broadcasts the packet if the channel is idle; otherwise, it backs off. If the number of back-off times exceeds a specified limit, the packet will be dropped. The back-off limit used in all simulations in this research is four, which is the default value used in the simulator. Upon receiving the packet, the receiver does not need to acknowledge the reception of the packet.

The capabilities of the sensor nodes are set as follows. The transmission range is set to 1.5 km. The transmission, reception and idle powers are set to 8.0 W, 0.80 W and 0.008 W, respectively. The frequency is set to 35.695 kHz, and 17.80 kbps is used for the bit rate. The bit error rate is set to $10e^{-9}$. This setup is similar to the specification of the real underwater acoustic sensor modem *UWM2000H*[22].

The simulation type adopted in this evaluation is the terminating state where each run lasts for 2000 seconds. Results from the first 150 seconds and the last 150 seconds are discarded to minimize the warm-up effect. In each run, 10 sensor nodes are selected randomly from the set of deployed sensors to inject exponentially distributed traffic into the network. Sensor nodes mobility, if not otherwise specified, is assumed to follow the 2D random walk [23][24] mobility model which is one of the most widely used mobility model [23]. Table 2 summarizes other system parameters.

Table 2: Additional system parameters

Parameter	Value
Number of nodes	200, 300, 400
Number of sinks	1, 2, 3
Initial energy	300 J
Data packet size	150 Byte
Energy threshold	20J
Alpha for EMGGR	0.5

4.2 INVESTIGATED METRICS

The performance of the proposed routing protocol is studied and compared with the performance of existing solutions by investigating the following performance metrics [25][18]:

- Packet Delivery Ratio (PDR): PDR is the ratio of the number of distinct data packets delivered successfully to the sink nodes to the total number of data packets generated at the source nodes.
- Average End-To-End Delay: The average time taken by a data packet to arrive to the destination. It is computed from the time the packet is generated until it reaches the destination. Only data packets that were successfully delivered to the sink nodes are counted.
- Energy Consumption: The total energy consumed by all nodes during the simulation. It includes the power consumed in the transmission, reception and idle modes.

- It is worth noting that the mentioned metrics are investigated under the effect of varying the number of sensor nodes, traffic load and node's mobility as follows:
- Number Of Nodes: The effect of the number of sensor nodes is investigated and assessed. Basically, 200, 300 and 400 stationary nodes are deployed in the investigated environment. The traffic injection rate is set to 0.07 packets/s.
- Traffic Load: The impact of traffic load is assessed by varying the packet injection rate from 0.05 to 0.11 packets/s. In this set of experiments, 300 static nodes are deployed in the environment.
- Node's Mobility: According to [26], underwater objects can move at speed 3-6 km/h. Therefore, the effect of nodes' mobility is evaluated by varying the maximum speed of nodes from 0 to 1.5 m/s. The minimum speed is set to 0 m/s. The number of nodes used in this set of experiments is 300 and the traffic injection rate is set to 0.07 packets/s.

4.3 SIMULATION RESULTS

A large number of experiments have been conducted to evaluate the performance of the proposed protocol. In the first subsection, the impacts of the number of sensor nodes, traffic rate and speed of nodes of the three protocols are assessed under random deployment. In this set of experiments, a single sink is used, and it is deployed at the middle of the top surface level. The second subsection evaluates the protocol using multiple sinks. 95% confidence intervals were calculated for all collected results. The average of 20 batch runs along with error bars are presented in the figures.

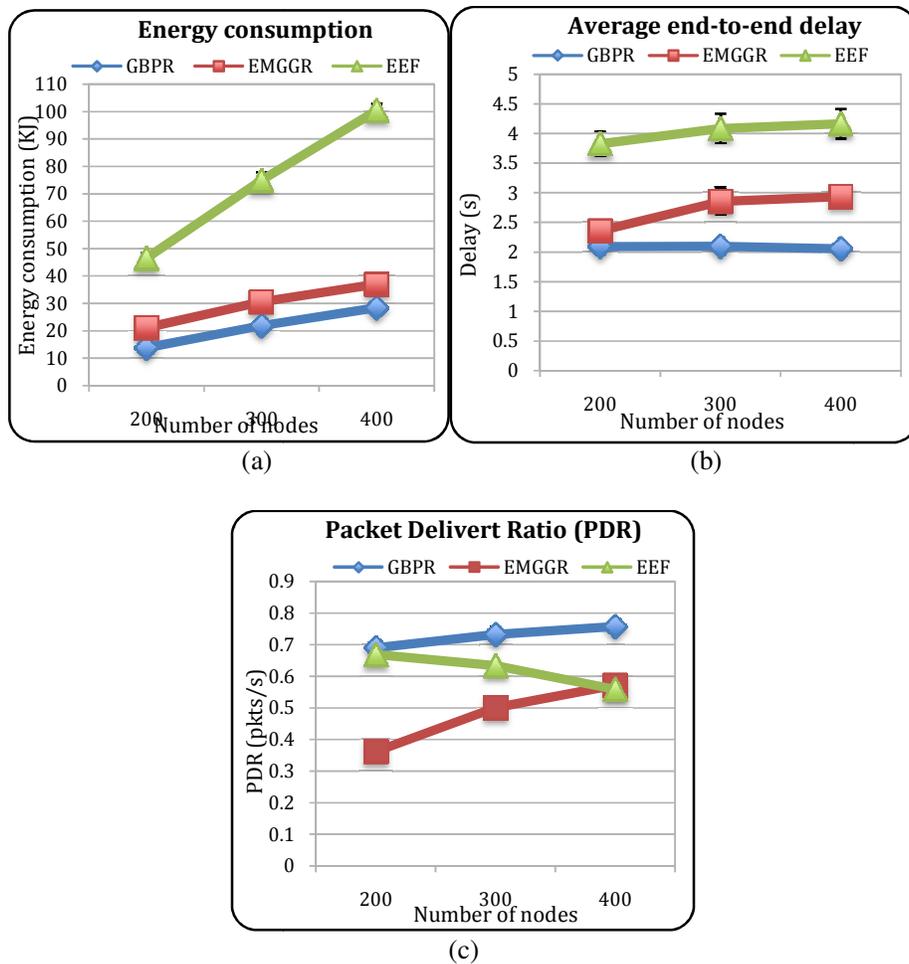
4.3.1 GBPR VERSES EEF AND EMGGR

Impact Of The Number Of Sensor Nodes:

Figure 8.a shows the impact of the number of nodes on the total energy consumed by the nodes. The figure reveals that the three protocols show an increase in the total consumed energy as the number of nodes increases. This can be justified by the increase in the number of packets propagated in the network and the number of nodes participating in the transmission and reception of these packets. However, the increase is very sharp in the EEF protocol since each forwarder broadcasts the packet to all its neighbors, and those with higher fitness than the previous hop are possible eligible forwarders. Although, in GBPR and EMGGR, data packets are forwarded hop-by-hop through cell-heads only, all nodes in the range may receive the packets due to the shared transmission media. The energy consumption in EMGGR is a little bit higher than GBPR due to the long paths a high number of control packets. Specifically, when the number of nodes is 300, GBPR is better than EEF and EMGGR in saving energy by 70% and 28% respectively.

Figure 8.b demonstrates the effect of the number of sensor nodes on the average end-to-end delay. As revealed in the figure, the average delay of both EMGGR and EEF exhibits a similar trend as the number of nodes increases. EEF shows the highest end-to-end delay. This is due to the fact each node holds the packet for a certain time based on its depth, residual energy, and its distance from the previous forwarder. In addition, since each relay broadcasts the packet to all its neighbors, there will be a high workload and congestion in the channel, which increases the queuing time and results in a high end-to-end delay. In contrast, GBPR and EMGGR do not employ any waiting time because only a single forwarder is selected in each hop. In GBPR, the increase in the number of nodes minimizes the number of voids; hence, increasing the probability of finding forwarders that give positive progress towards the sink nodes. As a result, packets are propagated in less number of hops. This is the justification for the decrease in the average end-to-end delay in GBPR. With 300 nodes, GBPR is better in the average end-to-end delay than EEF and EMGGR by 48% and 26%, respectively.

Figure 8.c illustrates the packet delivery ratio (PDR) as a function of the number of nodes. By increasing the number of nodes, both GBPR and EMGGR protocols provide an improvement in the PDR. The improvement can be justified by the mechanisms adopted by the protocols that try to (i) reduce the number of packets propagated in the network by selecting a single forwarder in each hop, and (ii) distribute the load among the nodes by alternating the selection of the forwarders. These indeed reduce the number of collisions. Moreover, the probability of encountering void cells (i.e. cells with no cell-heads) decreases by increasing the number of nodes; hence, packets have better chances to be forwarded successfully to their next hops on shorter paths. On the contrary, as the number of nodes increases in EEF, less delivery ratio is achieved due to the increase in the number of candidate forwarders (i.e. nodes with high fitness than the previous forwarders) as the number of nodes increases. Hence, packets collide with each other and fail to be delivered successfully to the destinations. It is worth mentioning that GBPR outperforms both EEF and EMMGR overall used number of nodes. For example, when the number of nodes is set to 300, GBPR is, better than EEF and EMGGR by 13% and 31%, respectively.



(a) Energy consumption (b) Average end-to-end delay (c) PDR

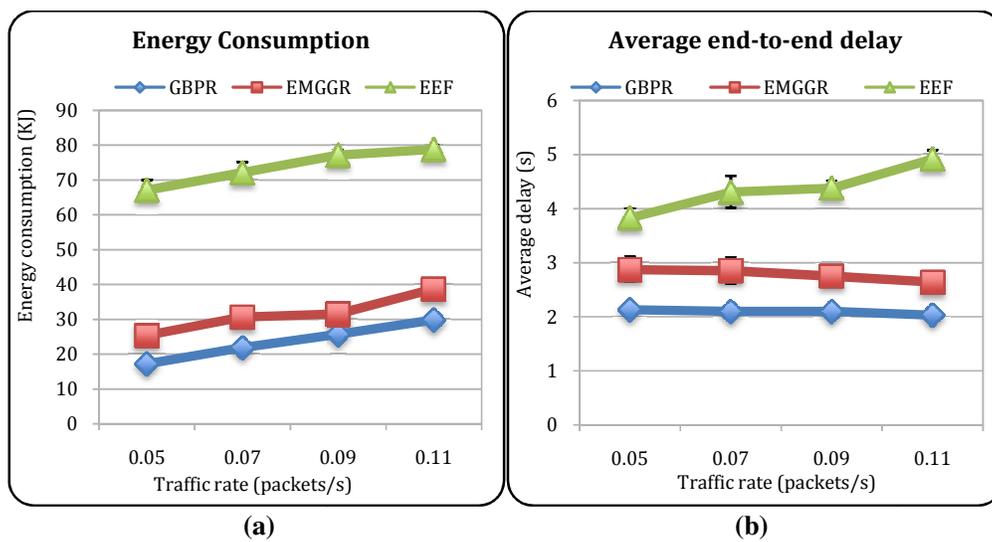
Figure 8: Impact of the number of sensor nodes on

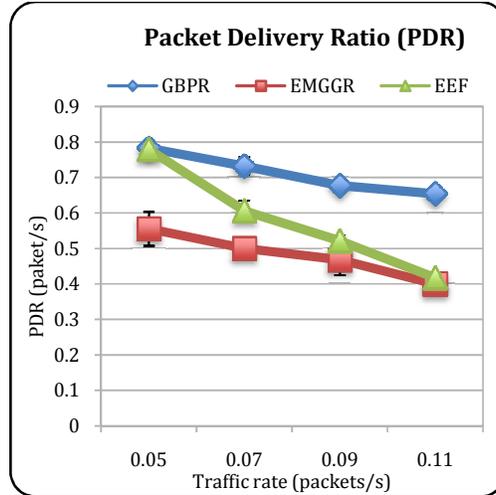
Impact Of Traffic Rate:

As revealed in Figure 9.a, the energy consumption is positively proportional to the injection rate of the traffic overall evaluated protocols. However, GBPR is the most energy efficient among the three protocols. The key causes of the energy efficiency of GBPR are the lower number of control packets employed by GBPR, the selection of a single forwarding candidate in each hop, and the selection of the closest feasible neighboring cell to the destination cell. The difference in the energy consumption between GBPR and EEF is significant overall used traffic rates. However, EMGGR incurs only a small increase in the energy consumption compared to GBPR. This is expected since both of them select a single candidate forwarder in each hop, however, EMGGR delivers the packets on longer propagation paths. For example, with 0.09 packets/s injection rate, GBPR is better in saving energy by 66% and 18% than EEF and EMGGR, respectively.

Figure 9.b illustrates the average delay as a function of the packet injection rate. With the used traffic rates, GBPR is stable against the increase in the amount of traffic. This is because GBPR always favors nodes that advance the packets toward the sink node. In addition, the assumed number of nodes (300 nodes) seems to be sufficient to minimize the number of void cells; thus packets are propagated in the network in fewer hops. This also explains the reduction in the average delay in EMGGR. However, in EEF increasing the traffic rate induces extra delay due to the increased number of packets propagated in the network, which leads to collisions and high propagation time. With 0.09 packets/s injection rate, GBPR outperforms EEF (EMGGR) by 52% (23%).

Figure 9.c shows the PDR as a function of traffic generation rate. Clearly, the PDR is inversely proportional to the packet injection rate, as expected. This happens because increasing the amount of traffic increases the number of packets propagated in the network; consequently, the nodes contend for the channel to deliver their packets. This increase in the congestion increases the chance of packets' collisions, which reduces the successfully received packets at the sink node. Nevertheless, GBPR outperforms EEF and EMGGR in delivering data packets over all used traffic rate. In addition, the decrease in the PDR in EEF is very sharp compared to the GBPR protocol, and this proves the benefits of using a single forwarder (i.e. especially in networks with high traffic load) of each packet to avoid collisions between packets. For instance, when the injection traffic rate is 0.09 packets/s, GBPR gives better PDR than EMGGR and EEF by 31% and 23%, respectively. Furthermore, the results shade some lights on the superiority of our protocol under high traffic load conditions, and this makes it suitable for applications that require frequent data transmissions from the deployed sensors such as monitoring applications.





(c)

(a) Energy consumption (b) Average end-to-end delay (c) PDR

Figure 9: Impact of traffic rate on

Impact of Nodes' Mobility:

Figure 10.a depicts the impact of nodes' mobility on the energy consumption. While EEF shows a little increase in the energy consumption, both GBPR and EMGGR consume a little less energy with the increase in the nodes' maximum speed. The reason is that with the increase in the nodes' speed, the number of void cells in GBPR and EMGGR increases; hence, fewer data packets are propagated to their destinations. This decrease in the number of propagated packets reduces the amount of consumed energy. In EEF, since the packets are broadcasted to all the nodes in the range, new nodes can enter the range with the increase in the nodes' speed, which slightly increases the energy consumption. Generally, GBPR is the most energy efficient compared to the other two protocols. Specifically, when the maximum speed of nodes is set to 1m/s, GBPR is able to save energy up to 72% and 20% compared to the EEF and EMGGR, respectively.

Figure 10.b reveals the average end-to-end delay as a function of the nodes' mobility. Generally, there is an inverse proportionality between the speed of nodes and the average delay. This can be justified as follows. In GBPR and EMGGR, increasing the speed of nodes increases the number of voids; hence, elections are performed frequently, and the number of dropped packets increases. In other words, only a few packets get delivered to the destinations, and fewer congestions and queuing delay are incurred. In EEF, the increase in the movement of the nodes reduces the congestions in the channel since nodes change their positions; thus, the less queuing delay is incurred. However, as observed in the previous experiments (Figure 8.b and Figure 9.b), GBPR provides the best performance compared to EEF and EMGGR protocols in terms of the average delay. It is worth mentioning that when the maximum speed of nodes is set to 1 m/s, GBPR outperforms EEF and EMGGR by 50% and 28%, respectively.

Figure 10.c demonstrates the impact of nodes' mobility on the PDR. As the maximum speed of nodes increases, the three protocols show a decrease in the number of successfully received packets. EMGGR and GBPR depend on cell-heads for forwarding the packets, and the movement of the nodes results in moving out of their cells, which results in initiating elections frequently. This increases the number of control packets propagated in the network as well as increasing the number of void cells. However, the decrease is sharper in EMGGR due to the longer propagation paths; thus, the probability of the packets being dropped increases. In EEF, each forwarder

broadcasts the packets to all its neighbors; hence, the probability of all neighbors being deviated from the range is less. Generally, GBPR outperforms EEF and EMGGR in PDR overall used speeds. For instance, when the maximum speed of nodes is set to 1 m/s, GBPR is superior to deliver packets to the destinations than EEF and EMGGR by 15% and 49%, respectively.

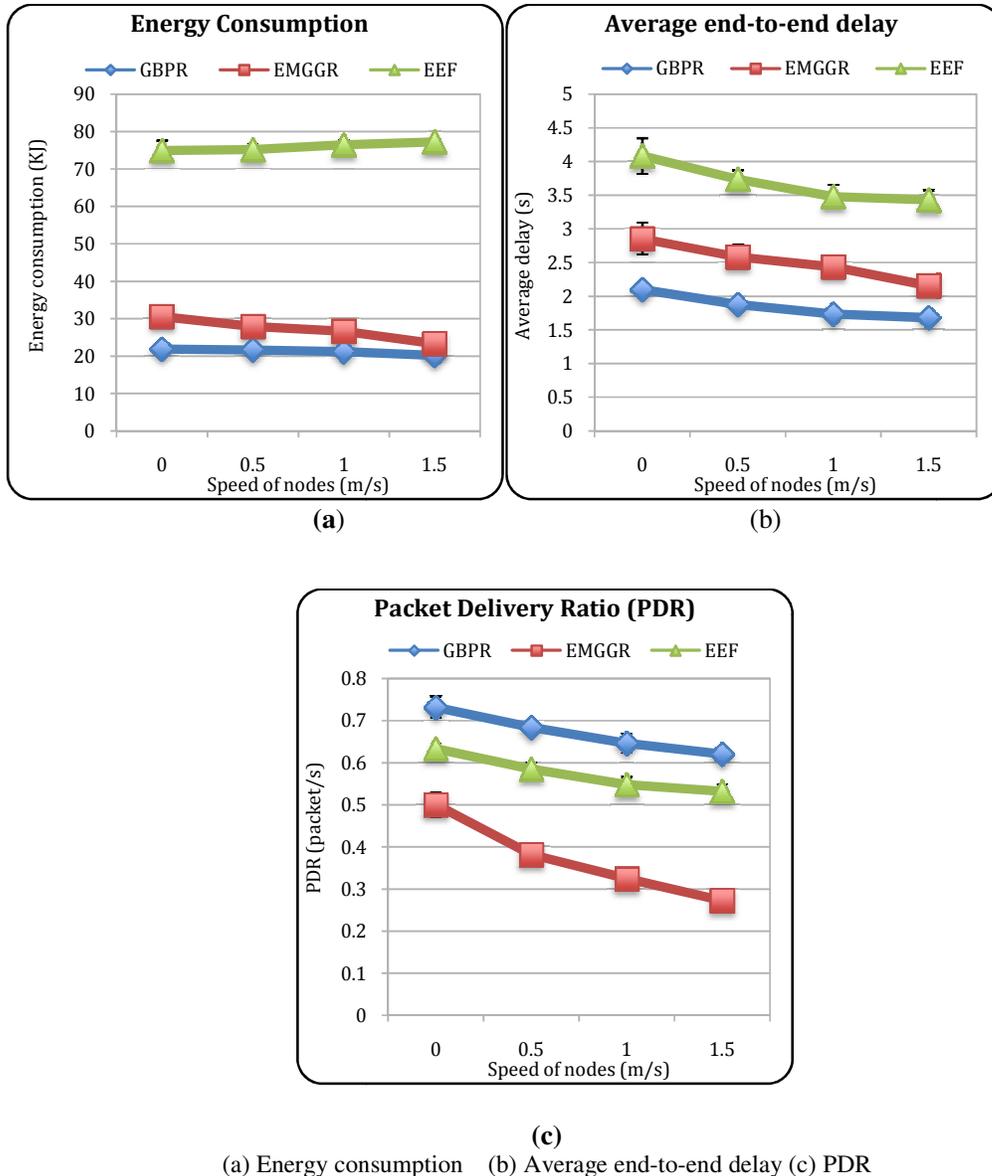


Figure 10: Impact of nodes' mobility on

4.3.2 GBPR WITH MULTIPLE SINKS:

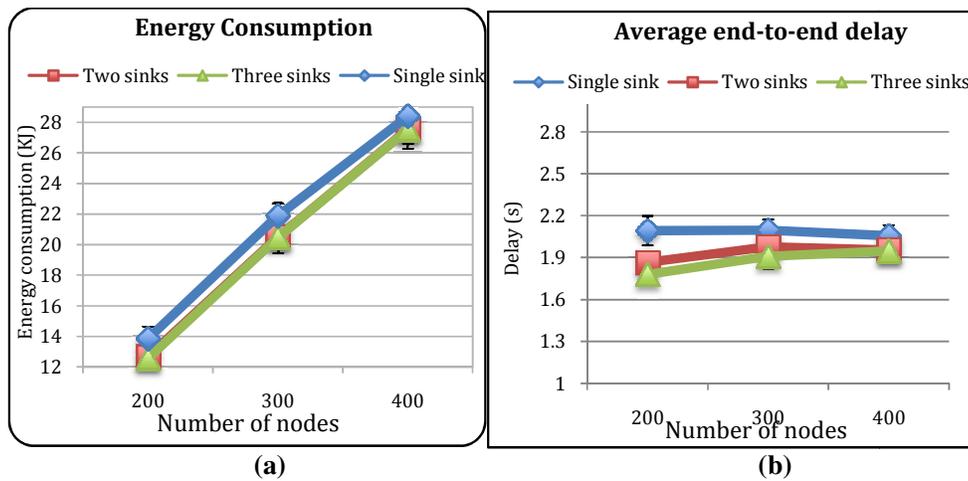
The aim of this set of simulations is to study the effects of using multiple sink nodes on the performance of the proposed protocol. Specifically, the protocol is evaluated under the use of one, two and three sinks. 200, 300 and 400 static nodes are deployed randomly in the environment.

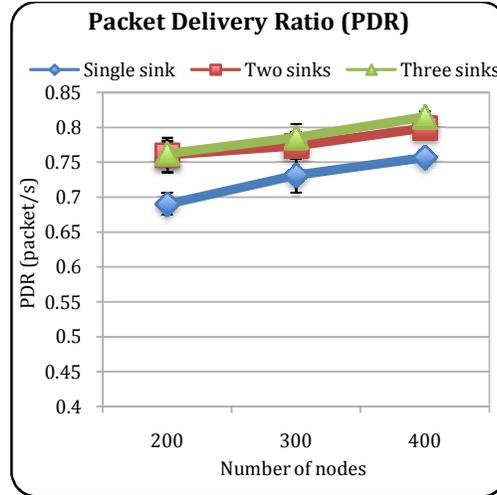
Figure 11.a illustrates the energy efficiency of the protocol under the use of a different number of sink nodes. Clearly, using multiple sinks provide better performance than using a single sink. The

cause of this is that using multiple sinks allow nodes to send their packets to the closer sink; hence, reduces the number of hops packets need to make. In addition, the load on the nodes near the sinks is distributed, and the collisions between packets are reduced. On average, using two sinks provides 5.3% better energy efficiency over scenarios with a single sink. However, the improvement is just 0.3% when using three sinks over the use of two sinks. The trend is logical, and the further increase in the number of sinks will not give much improvement in saving energy. The reason behind this is that the number of sinks to be used depends on the size of the monitored area and its expansion in the horizontal plane. Basically, the larger the area, the more sinks need to be used. Therefore, network developers need to link the number of sinks with the size of the monitored area in order to achieve better performance.

The average end-to-end delay experienced by data packets received by the sink nodes is demonstrated in Figure 11.b. As we can see from the figure, the scenario with three sinks outperforms the other two scenarios with a single and two sinks. The justification of this is that the packets are delivered successfully to the closer sink nodes via fewer hops, which decrease the propagation delay and back-off times. With 200 nodes, the improvement of using two sinks is approximately 10.9% compared to the scenario with a single sink. However, the delay of three sinks is 4.5% lower than that of two sinks. The gap in the differences reduces with the increase in the number of nodes due to the resulting decrease in the number of void cells.

The PDR as a function of the number of nodes for a different number of sinks is examined in Figure 11.c. The figure shows an increase in delivering data packets when multiple sinks are used compared to the scenario with a single sink. Specifically, with two sinks, the protocol succeeded in delivering 76% of the packets to the sink nodes even in scenarios of very sparse networks (i.e. less than 200 nodes). Nevertheless, there is a slight increase when using three sinks compared with the scenario of two sinks. In particular, the average improvement when using two sink nodes is 6.6% overall used number of nodes compared with a single sink setup. However, the improvement of scenarios of three sinks is just 1.3% over those with two sinks.





(c)
 (a) Energy consumption (b) Average end-to-end delay (c) PDR

Figure 11: Impact of varying the number of sinks on

5. CONCLUSION AND FUTURE WORKS

In this paper, we proposed a new GBPR routing protocol for UWSNs. It is a grid-based routing protocol in which the packet is forwarded on a cell-by-cell basis. The key feature of the GBPR protocol is the classification of the neighboring cells into priority levels according to their distances to the sink nodes. Basically, neighboring cells that are closer to the sink are given higher priority to be selected as the next hop. Hence, packets are propagated in less number of hops, and therefore high PDR, less energy and low end-to-end delay are achieved. Moreover, GBPR implements a simple election algorithm with short control packets to select nodes that are responsible for forwarding data packets to the sink nodes. In addition, to efficiently conserve the available limited resources (e.g. energy and bandwidth), data packets are forwarded to a single cell-head in each hop and cells with a positive advance toward the sink nodes are favored. The evaluation results show that the proposed protocol achieves the best performance in terms of energy consumption, average end-to-end delay and PDR than EEF and EMGGR over all experiments. Furthermore, multiple sinks scenarios provide better performance results over single sink counterparts. Therefore, the GBPR can suit a number of applications such as intruder detection, disaster prevention and early warning, and deep-water oil drilling.

For future direction, we aim to investigate how to adaptively order the selection of the next forwarder according to the channel conditions (e.g. amount of traffic) in order to further improve the performance of the protocol. Furthermore, we plan to investigate the performance of the enhanced protocol with other protocols available in the literature and under different network conditions. In addition, we aim to find the optimal number of sink nodes to be used as well as their optimal placement that will lead to better performance results. This is because sink nodes might be more powerful and more expensive than sensor nodes; therefore, finding the optimal number and optimal placement is a valuable study.

ACKNOWLEDGMENT

This work is supported by The Research Council (TRC) of the Sultanate of Oman under the research grant number RC/SCI/COMP/15/02.

REFERENCES

- [1] I. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257–279, May 2005.
- [2] G. Han, C. Zhang, L. Shu, and J. J. P. C. Rodrigues, "Impacts of Deployment Strategies on Localization Performance in Underwater Acoustic Sensor Networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1725–1733, March 2015.
- [3] G. Han, C. Zhang, L. Shu, N. Sun, and Q. Li, "A Survey on Deployment Algorithms in Underwater Acoustic Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 2013, p. 11, November 2013, Article ID 314049.
- [4] F. Bouabdallah and R. Boutaba, "A Distributed OFDMA Medium Access Control for Underwater Acoustic Sensors Networks," *Proceedings of the 2011 IEEE International Conference on Communications (ICC)*, 5-9 June 2011, pp. 1–5, Kyoto, Japan.
- [5] C. Lv, S. Wang, M. Tan, and L. Chen, "UA-MAC: An Underwater Acoustic Channel Access Method for Dense Mobile Underwater Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 2014, no. 2, pp. 1–10, February 2014, Article ID 374028.
- [6] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "Aloha-Based MAC Protocols with Collision Avoidance for Underwater Acoustic Networks," *Proceedings of the IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, 6-12 May 2007, pp. 2271–2275, Anchorage, AK, Alaska.
- [7] W. Lin and K. Chen, "MHM: A Multiple Handshaking MAC Protocol for Underwater Acoustic Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 12, no 5, pp. 1–12, May 2016.
- [8] D. Pompili, T. Melodia, and I. F. Akyildiz, "A CDMA-based Medium Access Control for UnderWater Acoustic Sensor Networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 4, pp. 1899–1909, April 2009.
- [9] K. Chen, M. Ma, E. Cheng, F. Yuan, and W. Su, "A Survey on MAC Protocols for Underwater Wireless Sensor Networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1433–1447, Third Quarter 2014.
- [10] P. Xie, J.-H. Cui, and L. Lao, "VBF: vector-based forwarding protocol for underwater sensor networks," *Proceedings of IFIP Networking'06*, 15-19 May 2006, pp. 1216–1221, Coimbra, Portugal.
- [11] N. Nicolaou, A. See, P. Xie, J.-H. Cui, and D. Maggiorini, "Improving the Robustness of Location-Based Routing for Underwater Sensor Networks," *Proceedings of OCEANS 2007 - Europe*, 18-21 June 2007, pp. 1–6, Aberdeen, Scotland.
- [12] Z. S. Peng Xie, Zhong Zhou, Zheng Peng, Jun-hong Cui, "Void Avoidance in Mobile Underwater Sensor Networks," *Proceedings of WUWNet'07*, September 2007.
- [13] X. Xiao, X. P. Ji, G. Yang, and Y. P. Cong, "LE-VBF: Lifetime-Extended Vector-Based Forwarding Routing," *Proceedings of 2012 International Conference on Computer Science and Service System*, 11-13 August 2012, pp. 1201–1203, Nanjing, China.
- [14] D. Shin, D. Hwang, and D. Kim, "DFR: an efficient directional flooding-based routing protocol in underwater sensor networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 17, pp. 1517–1527, 10 December 2012.
- [15] H. Yan, Z. J. Shi, and J.-H. Cui, "DBR: depth-based routing for underwater sensor networks," *Proceeding of the International Conference on Research in Networking*, Springer Berlin Heidelberg, 5-9 May 2008, pp. 72–86, Singapore.
- [16] M. Ashrafuddin, M. M. Islam, and M. Mamun-or-Rashid, "Energy Efficient Fitness Based Routing Protocol for Underwater Sensor Network," *International Journal of Intelligent Systems and Applications*, vol. 5, no. 6, pp. 61-69, May 2013.
- [17] F. Al-Salti, N. Alzeidi, B. Arafah, "A New Multipath Grid-Based Geographic Routing Protocol for Underwater Wireless Sensor Networks," *Proceedings of Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, pp.331,336, 13-15 October 2014, Shanghai, China.

- [18] F. Al Salti, N. Alzeidi, and B. R. Arafeh, "EMGGR: an energy-efficient multipath grid-based geographic routing protocol for underwater wireless sensor networks," *Wireless Networks*, February 2016.
- [19] J. Jiang, G. Han, H. Guo, L. Shu, and J. J. P. Rodrigues, "Geographic multipath routing based on geospatial division in duty-cycled underwater wireless sensor networks," *Journal of Networks and Computer Applications*, vol. 59, pp. 4–13, January 2016.
- [20] M. Tariq, M. ShafieAbdLatiff, M. Ayaz, Y. Coulibaly, and N. Al-Areqi, "Distance based Reliable and Energy Efficient (DREE) Routing Protocol for Underwater Acoustic Sensor Networks," *Journal of Networks*, vol. 10, no. 5, pp. 311–321, May 2015.
- [21] P. Xie, Z. Zhou, Z. Peng, H. Yan, T. Hu, J.-H. Cui, Z. Shi, Y. Fei, and S. Zhou, "Aqua-Sim: An NS-2 based simulator for underwater sensor networks," *Proceedings of MTS/IEEE Biloxi - Marine Technology for Our Future: Global and Local Challenges (OCEANS 2009)*, 26-29 October 2009, pp. 1–7, Biloxi, Mississippi.
- [22] "LinkQuest: Underwater acoustic modem models." [Online]. Available: <http://www.linkquest.com/html/models1.htm>. [Accessed: 08-Feb-2016].
- [23] V. Davies, "Evaluating Mobility Models Within An Ad Hoc Network," Master's thesis, Colorado School of Mines, 2000.
- [24] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, August 2002.
- [25] M. Xu, G. Liu, and H. Wu, "An Energy-Efficient Routing Algorithm for Underwater Wireless Sensor Networks Inspired by Ultrasonic Frogs," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 351520, 12 pages, 2014.
- [26] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," *IEEE Network*, vol. 20, no. 3, pp. 12–18, May 2006.

AUTHORS

MS. Faiza Al-Salti received her B.Sc. and M.Sc. Degrees in computer science from the Sultan Qaboos University (Oman) in 2012 and 2015, respectively. She is currently a Ph.D. candidate in the Sultan Qaboos University. Her research interest includes communication protocols, terrestrial and underwater wireless sensor networks.



Dr. Nasser Alzeidi received his PhD degree in Computer Science from the University of Glasgow (UK) in 2007. He is currently an assistant professor of computer science and the director of the Center for Information Systems at Sultan Qaboos University, Oman. His research interests include performance evaluation of communication systems, wireless networks, interconnection networks, System on Chip architectures and parallel and distributed computing. He is a member of the IEEE.



Prof. Khaled Day received his undergraduate degree in computer science from the University of Tunis in 1986. He was awarded in 1986 the 'President Habib Bourguiba Prize' for excellent academic performance upon completion of his undergraduate degree. He was then awarded from the Tunisian government a scholarship to pursue graduate studies in the USA starting 1987. He obtained the M.Sc. and Ph.D. degrees in computer science from the University of Minnesota (USA) in 1989 and 1992 respectively. Dr. Day has worked at the University of Bahrain in the period 1992-1996 as Assistant Professor. He then joined in 1996 Sultan Qaboos University as Assistant Professor. He was promoted to Associate Professor in 1999 and then to Professor in 2005. He served as the Head of the Department of Computer Science at Sultan Qaboos University during the periods 2000-2002 and 2006-2010. He was appointed in January 2013 as the Dean of Research of Sultan Qaboos University. His areas of research interest include interconnection networks, parallel algorithms, distributed systems and wireless networks. He has published over 100 research papers in international journals and conferences. He has received in 2000 the Abdul Hameed Shoman Prize for Young Arab Researchers. He is a senior member of IEEE.



Dr. Bassel R. Arafeh received the B.Sc. in Electrical Engineering from Cairo University, Egypt, in 1974; and the M.E. degree in Electrical Engineering and the Ph.D. degree in Computer Science from Texas A & M University, College Station, Texas in 1980 and 1986, respectively. Dr. Arafeh is currently an associate professor in the Department of Computer Science, Sultan Qaboos University, Muscat, Oman. His research interests include wireless ad hoc and sensor networks, and parallel and distributed computing systems. He is a member of the ACM, the IEEE and IEEE Computer Society.



Dr. Abderezak Touzene received the BS degree in computer science from University Algiers in 1987, MS degree in computer science from Paris-SudOrsay University (France) in 1988 and Phd degree in computer science from Institutpolytechnique de Grenoble (France) in 1992. He is an Associate. Prof. in Sultan Qaboos University, Oman. Dr. Touzene's research includes Interconnection Networks, Parallel and Distributed Systems, Network-On-chip, Wireless and Mobile Networks, Performance Evaluation, Cryptography. Dr. Touzene is a member of the IEEE Computer Society.

