Telecommun Syst (2015) 58:179–191
DOI 10.1007/s11235-014-9901-8

A cross-layer QoS-aware optimization protocol for guaranteed data streaming over wireless body area networks

Nedal Ababneh · Nicholas Timmons · Jim Morrison

Published online: 30 December 2014

Abstract In this paper, we study the problem of routing, bandwidth and flow assignment in wireless body area networks (BANs). We present an adaptive joint routing and bandwidth allocation protocol for traffic streaming in BAN. Our solution considers BAN for real-time data streaming applications, where the real-time nature of data streams is of critical importance for providing a useful and efficient sensorial feedback for the user while system lifetime should be maximized. Thus, bandwidth and energy efficiency of the communication between energy constrained sensor nodes must be carefully optimized. The proposed solution takes into account nodes' residual energy during the establishment of the routing paths and adaptively allocates bandwidth to the nodes in the network. We also formulate the joint routing tree construction and bandwidth allocation problem as Mixed Integer Linear Program that maximizes the network utility while satisfying the QoS requirements. We compare the resulting performance of our protocol with the optimal solution, and show that it closes a considerable portion of the gap from the theoretical optimal solution.

Keywords Wireless body area networks · Optimal solution · Routing · Data streaming · Bandwidth allocation

N. Ababneh (✉)
Department of Computer Engineering, College of Information Technology, University of Bahrain, Zallaq, Bahrain
e-mail: nababneh@uob.edu.bh

N. Timmons · J. Morrison
WiSAR Lab, School of Engineering, Letterkenny Institute of Technology, Port Road, Letterkenny, Ireland
e-mail: nick@wisar.org

J. Morrison
e-mail: jim@wisar.org

1 Introduction

A wireless body area network (WBAN, widely simplifies as BAN) consists of in-body and on-body sensor nodes that continuously monitor a patient's vital information for diagnosis and prescription [1]. Each sensor capable of sampling, processing, and communicating one or more vital signs, such as electrocardiogram (ECG), heart rate, blood pressure, blood oxygen levels, activity, etc. or environmental parameters, such as location, temperature, light, etc. Typically, these sensors are placed strategically on the human body as tiny patches or hidden in users' clothes allowing ubiquitous health monitoring in their native environment for extended periods of time without recharging or battery replacement. BAN is particularly suitable for post-operative care in hospitals and for treatment of chronically ill or aged patients at home. In such application, sensors continuously monitor human's physiological activities and actions, such as health status and motion pattern, which may occur in a more periodic manner, and may result in the applications' data streams exhibiting relatively stable rates [2].

The selection of network architecture for BANs is an important one because it significantly affects the overall system performance in many ways, such as power consumption, ability to handle different traffic loads, robustness against node failure and choices of MAC protocol. Network architecture refers to the logical organization or arrangement of communication devices in the network. In terms of investigating the architecture, there has been some research about the most optimal network architectures in BAN. Star architecture, where the nodes are connected to a central coordinator (i.e., the sink) in star manner, is typically assumed for BANs given the relatively small area over which the nodes are deployed. If the communication can be done in a single hop, then a simple star topology can be used, but multiple hops may be required to extend network connectivity due to constraints such as sensor power or location or factors such as multi-path fading or shadowing. Authors in [3] showed that star solution has inadequacies especially for BANs operating in dynamic, time-varying environments. One of the alternatives to star is a multi-hop architecture, where nodes have a choice to transmit data to the sink or to other nodes in the network. This can provide robustness to adverse environmental conditions. Recently, several studies [4] have confirmed that from an energy consumption point of view, a multi-hop architecture is beneficial, and thus, it is adopted in this paper.

It is well-known that energy is an extremely critical resource for battery-powered BANs, thus making energy-efficient protocol design a key challenging problem [5]. Most of the existing energy-efficient routing protocols always choose a static optimal path (minimum energy path) to the sink to merely minimize energy consumption, which readily causes an unbalanced distribution of residual energy among sensor nodes since the energy at the nodes on the optimal path is quickly depleted. Such imbalance of energy consumption is definitely undesirable for the long-term health of the network as it reduces the network lifetime and causes network partition [6]. If the sensor nodes consume their energy more evenly, the connectivity between them and the sink can be maintained for a longer time, thus postponing the network partition. This more graceful degradation of the network connectivity extends its operable lifetime.

In general, all data in the BAN is generated at the nodes and continuously fed as a stream to the sink. Most of the traffic is uplink, while a limited amount of control traffic will go in the other direction, which is neglected in this work. For these applications, it is essential to be able to reliably collect physiological readings from humans via sensor nodes. Such networks could benefit from quality of service (QoS) mechanisms that support prioritized data streams, especially when the channel is impaired by interference or fading [7]. For example, heart activity readings (e.g., ECG data) are often considered more important than body temperature readings, and hence can be assigned a higher priority in the system. QoS support is needed to ensure reliable data collection for high priority data streams and to dynamically re-allocate bandwidth as conditions change, especially when the effective channel bandwidth is reduced. In the standard system, when the effective throughput is scarce, data rate for all the nodes drops equally and, thus, the *utility* (the term utility is defined later in Sect. 3) of the whole monitoring system drops significantly. In this case, to guarantee the QoS we need to re-allocate resources from lower priority streams to higher priority streams. Adaptive QoS resource allocation is thus needed to provide bandwidth guarantees, which are essential for reliable data collection in BANs.

In this paper, we formulate the routing and bandwidth allocation problem as a linear optimization problem and provide a solution to it. We also present efficient heuristics to solve the problem and compare their relative performance. We propose an Adaptive Routing and Bandwidth Allocation protocol, termed ARBA[1], that improves bandwidth utilization and routing in BAN while balancing out energy consumption throughout the network ensures longer network lifetime. Our solution is based on the insight that, in seeking to maximize the network lifetime and throughput, the problem of assigning the highest possible data rate to source nodes and that of finding the best and efficient forwarding tree from sources to the sink are not independent, the solution of each has a profound impact on the outcome of the other. Therefore, the two problems should ideally be solved jointly. In ARBA, we define a new metric, called network *utility*, which takes both throughput and residual energy into account and includes parameters that can be used to emphasize one over the other to obtain a tradeoff between energy and throughput. This approach enables the service provider to plan and tune a BAN so that available capacity are used wisely to satisfy the QoS requirements. To maximize network throughput, we estimate the utility for different data streams and put some low utility streams offline and thereby maintain high utility of the monitoring system where utility of high priority streams is particularly improved. We select the best possible data rate for each node with efficient link utilization such that the network utility is maximized without violating the QoS constraints. ARBA constructs routing paths in terms of depth and residual energy. The goal of this basic approach is to force packets to move towards the sink through high energy nodes so as to protect the nodes with relatively low residual energy.

In short, following are the major research contributions of this paper, which makes ARBA distinct from existing work:

- We formally define the joint routing and bandwidth assignment optimization problem and present a mixed integer linear program (MILP) formulation for it that maximizes the system utility. Obviously, optimal results are only possible small-scale network instances. Therefore, it is merely used for benchmarking purposes.
- We propose a novel cross-layer solution for the problem. Our solution consists of: (1) A fully distributed routing algorithm that takes into account nodes' residual energy during tree (i.e., route) construction process, and (2) An effective bandwidth and rate assignment algorithm that takes into account QoS requirements. The network and physical layers QoS constraints are translated into lower/upper bound on node data rate, an upper bound on

[1] A preliminary version of this protocol was presented in [8], while here we present more design details and performance analysis of the protocol.

uplink physical capacity and an upper bound on channel capacity at each relay node in the tree.
- We evaluate the proposed protocol experimentally on a wide range of practical scenarios, showing that it consistently yields to a promising results.

The remainder of this paper is organized as follows. Section 2 briefly provides the related work in this research area. Section 3 presents a MILP formulation of the problem. Section 4 presents the ARBA protocol, which is then evaluated in Sect. 5. Finally, Sect. 6 concludes the paper and introduces our future work.

2 Related work

Numerous studies on energy efficient routing and resource allocation in wireless sensor and mesh networks have been proposed in the literature [9–12]. These studies mainly focus on how to allocate limited energy, radio bandwidth, and other resources to maximize the value of each node's contribution to the network. However, these approaches cannot be applied into the BANs without modification. Therefore, several protocols have been proposed recently in the area of routing in BANs [1], mostly validated either using theoretical analysis, testbed implementation or simulation, and involve MAC or power control mechanisms. These protocols deal with limitations and unique characteristics of BANs, such as communication range or irregular traffic patterns. However, research on routing protocols for BANs is still at its infancy and can be categorized into:

- *Cluster based routing* This type of protocols uses clustering to reduce the number of direct transmissions to the remote base station. Hybrid Indirect Transmissions (HIT) [13] is a data gathering protocol based on LEACH [14], it combines clustering with forming chains. In doing so, the energy efficiency and as a consequence network lifetime is improved, specifically in sparse BAN. Reliability, however, is not considered. Another problem with HIT is the conflicting interaction between its communication routes and those desirable for the application. As a result, HIT requires more communication energy in a dense network.
- *Temperature routing* [15–17] When considering wireless transmission around and in the body, important issues are radiation absorption and heating effects on the human body. In this approach, they try to route data away from high temperature areas due to focusing data communications, defined as hot spots. Temperature routing protocols can be sub-divided further into two categories: (a) adjusting radio transmission power and traffic control algorithms, and (b) balancing the communication over the sensor nodes, which is used by thermal aware routing algorithm, least temperature routing (LTR) and adaptive LTR. Temperature routing protocols may suffer from short network lifetime, low delivery ratio, low reliability, inefficient energy usage, increased routing overhead and high end-to-end delay, which are problematic for BAN.
- *Cross-layer protocols* Cross-layer design is a way to improve the efficiency of and interaction between the protocols in the network by combining two or more layers from the protocol stack. Researchers have developed cross-layer techniques for leveraging postural mobility for optimizing energy usage by designing collision-free medium access and routing protocols in [18,19]. This allows nodes to save energy by sleeping when they are neither transmitting nor receiving.

An important issue in BAN, from an application point of view, is offering QoS. A virtual MAC is developed in BodyQoS [7] to make it radio-agnostic, allowing the protocol to schedule wireless resources and to provide adaptive resource scheduling using aggregator nodes. When the effective bandwidth of the wireless channel degrades due to RF interference or RF fading, the bandwidth is adaptively scheduled to meet the necessary QoS requirements. However, integration into existing MAC and especially routing protocols is not optimal [1]. When tackling the specific QoS requirements of BANs, integration with all layers should be strong. Given the difficult environment, where the path loss is high and as a result bandwidth is low, QoS protocols should expect and use all information about the network that is available.

3 Optimal solution

In this section, we present our optimal solution for the joint routing tree construction and rate allocation problem. For simplicity, we consider a single-radio, multi-channel BAN network, where potentially interfering wireless links should operate on different channels, enabling multiple parallel transmissions. The problem therefore consists of increasing the number of accepted high priority nodes (streams) in the routing tree at highest possible data rate while balancing out energy consumption to extend network operational lifetime.

3.1 The problem definition

The proposed BAN network is modeled as undirected graph $G = (V, E)$, called a connectivity graph, where V is the set of vertices representing the nodes in the network, and is composed of a group of sensor nodes denoted as V_n and one or more sinks denoted as V_s. E is the set of edges that represents the communication network topology, edge$(v_i, v_j) \in E$ iff v_i, v_j are within each other's communication range. Hence,

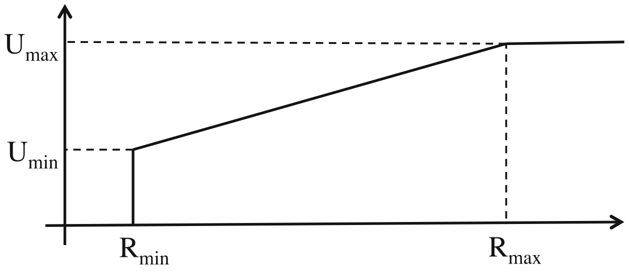


Fig. 1 Source utility function

nodes form a multi-hop ad-hoc network among themselves to relay traffic to the sink(s). Also, each edge (v_i, v_j) has a physical capacity L_{ij}, which represents the maximum amount of traffic that could pass through this particular link. At any given time, a node may either transmit or listen to a single wireless channel, with channel capacity C.

The resulting routing tree (final tree) $T = (V', E')$ is a subgraph of G, where E' represents the communication links in the final tree, and V' is the set of nodes and sinks included in the final tree. In order to evaluate the relative importance of nodes and the benefits gained when accepted in the network, we propose a *utility* function (represented below by the objective function). The utility of streaming data from node v_i is denoted by U_i. This utility depends on the minimum acceptable data rate W_{min} by node v_i, and the maximum possible data rate of a node W_{max}. It also depends on P_i, which represents the priority associated to each node (i.e., either high or low) based on the equipped sensor type or sensorial data, while r_i is the current rate (i.e., ratio to W_{max}) of data generated at node v_i. The utility of streaming data decreases with decreasing received data rate and available energy level, and the utility becomes insignificant beyond a certain value (i.e., R_{min} in this case). In cases where the data rate and residual energy level for a given node falls below the acceptable level (i.e., R_{min} and E_{min}), we propose to stop live data streaming and put the node offline. Figure 1 shows the utility function evolution with rate of a node $v_i \in V_n$, v_i prefers to send data streams higher than R_{min} and up to rate of R_{max}, and gives no preference to higher rates.

3.2 Mixed integer linear program formulation

Let z_i be a 0–1 integer variable for each node $v_i \in V_n$, such that $z_i = 1$ if the node v_i is accepted as a traffic source in the resulting tree $v_i \in V'_n$. Let r_i be a positive real variable for each $v_i \in V_n$, representing the effective data rate of v_i such that $r_i = 0$ if v_i is not included in the resulting routing tree (i.e., $z_i = 0$). Let x_{ij} be a 0–1 integer parameter for each edge $(v_i, v_j) \in E$, $x_{ij} = 1$ if the edge is included the resulting tree (i.e., edge $(v_i, v_j) \in E'$). Furthermore, let y_{ij} be a positive integer variable for each edge $(v_i, v_j) \in E'$, showing the amount of data transmitted from node v_i to node v_j (i.e., uplink effective data rate), the receiver could be a sink node. The MILP for the rate (r) and bandwidth (y) allocation problem can thus be stated as follows:

Objective function

$$max \sum_{i\in V_n}(z_i \times P_i \times U_{min}$$
$$+ \sum_{\forall j\in V:(i,j)\in E} (e_i - E_{min}) \times x_{ij} \times U^e_{step} \times L^{cost}_{Tx}$$
$$+ z_i \times P_i \times (r_i - R_{min}) \times U^r_{step})$$

The first term of the utility function is the minimum utility for each node in the network, the second and third terms denote the utility evolution with energy and rate, respectively. Multiplying the second term by the coefficient U^e_{step} and the third term by the coefficient U^r_{step} ensures utility evolution with energy and rate. The coefficients U^e_{step} and U^r_{step} can be used to emphasize one over the other to obtain a tradeoff between throughput and residual energy. Also, multiplying the first and the third term by z_i guarantees the consideration of the included vertices nodes in the resulting tree only. For two communicating nodes, the energy consumption of their communication grows at least quadratically with their distance [20]. Hence, to abandon long-distance communication, we assign a negative value L^{cost}_{Tx} (reflects the transmission cost over the link) for each link that is proportional to the distance. We multiply the second term by L^{cost}_{Tx} to ensure that each node $v_i \in V_n$ selects a parent node v_j with least distance to v_i. Each accepted node $v_i \in V'_n$ is assigned rate $r_i \geq R_{min}$. The coefficient U_{min} is the minimum utility of each accepted node v_i, and must be set to any positive value greater than zero (i.e., $U_{min} = 1$ in this work).

Constraints

$$y_{ij} \leq L_{ij} \times x_{ij}, \forall i \in V_n, \forall j \in V : (i, j) \in E \tag{1}$$

$$\sum_{\forall j\in V_n:(j,i)\in E} y_{ji} \leq C_i, \forall i \in V \tag{2}$$

$$\sum_{\forall j\in V:(i,j)\in E} y_{ij} - \sum_{\forall j\in V:(j,i)\in E} y_{ji} = r_i \times W_{max}, \forall i \in V_n \tag{3}$$

$$z_i \geq r_i, \forall i \in V_n \tag{4}$$

$$r_i \geq R_{min} \times z_i, \forall i \in V_n \tag{5}$$

$$E_i \geq E_{min} \times x_{ij}, \forall i \in V_n, \forall j \in V : (i, j) \in \tag{6}$$

Constraint (1) ensures that the uplink effective rate of each included edge in the resulting tree is bounded by the maximum physical link capacity. Constraint (2) provides an upper bound (i.e., the cell capacity) on the relay load constraint, it ensures that the incoming flow is always less than cell capacity C_i. Constraint (3) is for flow conservation. It implies that the difference between the outgoing traffic and the incoming traffic at node v_i is the volume of traffic generated by node v_i itself. Since all data flows are originated from nodes and do not return to the nodes, it will not lead to cycles in our solution. All data flows will eventually reach the sink. Constraint (4) ensures that node data rate is assigned to accepted nodes in the final routing tree only, i.e., nodes not included in the resulting tree have rate equal to zero. Constraint (5) ensures that each accepted node v_i in the resulting tree has to be assigned rate $r_i \geq R_{min}$, this is to satisfy the QoS requirements. Finally, Constraint (6) ensures that available energy E_i for each accepted node v_i in the resulting tree has to be $\geq E_{min}$, as E_{min} is the residual energy value where a node is still able to send/receive messages properly.

Figure 2a–f shows the effect of different number of high priority nodes on the streaming data quality coming from the accepted high priority nodes for different sink placement, while the sensory data is the actual information transferred across the wireless links. We classified the data quality based on the allocated data rate into three levels: poor (128–256 kbps), good (257–384 kbps) and excellent (385–512 kbps), the rest of system parameters are discussed in Sect. 5. To study the influence of number of high priority nodes on the data quality of the accepted high priority nodes we use the following metrics to construct the routing tree:

Energy We use node's residual energy as a selection metric. The sum of the residual energy of all nodes that reside in the path of a node, say node *a,* towards the sink is termed the *path-energy*, $P_{energy}(a)$. For each node to choose its parent (selecting node's parent is equivalent to build routing paths), it chooses the neighbor which renders the largest path-energy among all other potential parents (i.e., neighbor nodes). In fact, the path-energy metric is the sum of the residual energy metric of the node's predecessors. For example, assume *sink-c-b-a* is a path in the routing tree, and $P_{energy}(a)$ is the sum of node's residual energy through the path from node a whose parent node is b to the sink. Thus, $P_{energy}(a)$ is calculated as follows:

$$P_{energy}(a) = energy(a) + energy(b) + energy(c) \tag{7}$$

We suppose all the nodes enter the network one by one. When a node, say node a, enters, all its neighbors are eligible to be a's parent. In order to improve the network performance, node a selects a parent node, say node b, with largest value of $P_{energy}(b)$. The parent node is then selected such that:

$$Prnt(a) = max_{i \in neighbor(a)} P_{energy}(i) \tag{8}$$

Degree We define for each node, a *degree* function which shows the total number of neighboring nodes. The sum of the degree functions of all nodes that reside in the path of a node, say node *a,* is termed the *path-degree*, $P_{degree}(a)$. For each node to choose its parent, it chooses the neighbor which renders the smallest path-degree among all other neighbor nodes, as described above in the first algorithm (i.e., *Energy*).

Hop-Count Using the hop-count metric enables rapid convergence, however, simply constructing a routing tree by shortest path (SPT) is not sufficient to maximize network throughput as the link on the shortest path between two nodes may have bad quality [21]. However, results presented in Fig. 2 confirm that using shortest path routing is not always harmful and may perform as good as other routing algorithms in some scenarios.

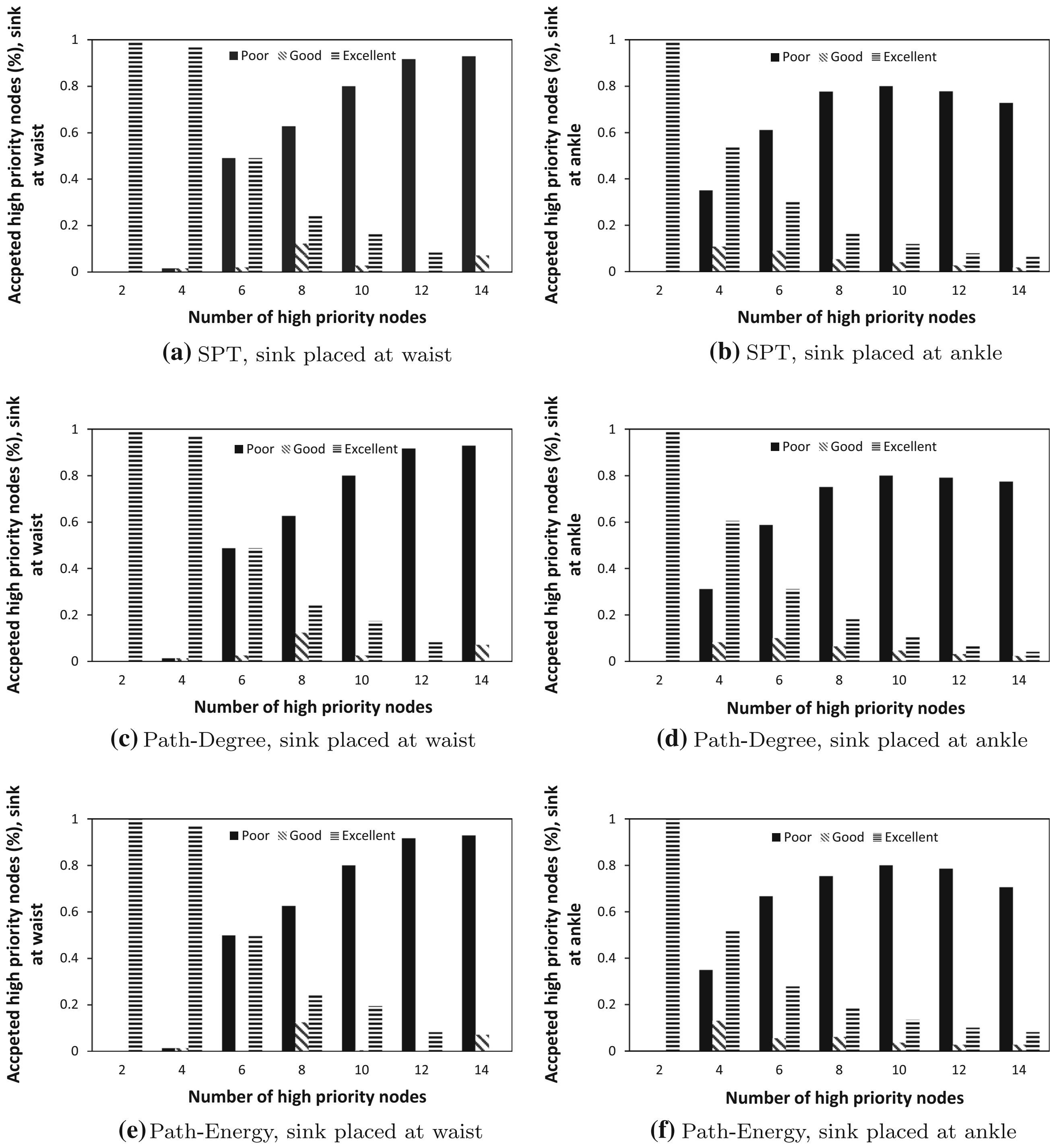
From the figures, we can see that the percentage of high priority nodes accepted at better quality, and thus higher utility, decreases with the the number of high priority nodes in all solutions. However, the three algorithms consistently perform better when the sink is placed at waist. This is because more routes are available in the network with possibly better capacity, and the rate assignment model (regardless the adopted routing algorithm) is able to select the best capacity path (advantageous route) and thus the best node data rates are allocated to high priority nodes. On the other hand, the performance is degraded when the sink is placed at ankle, this is because of the presence of traffic congestion. This is especially true for nodes closer to the sink as they are continuously used to forward the whole network traffic towards the sink, where relay capacity of each node is upper bounded by channel capacity. Thus, for small number of high priority nodes in the network, all high priority nodes will be accepted at highest possible data rate. On the other hand, when the number of high priority nodes is increased, more high priority nodes will be accepted at lower data rate.

4 Adaptive routing and bandwidth allocation protocol

In this section, we describe the design and implementation issues of the proposed protocol in depth. Our solution consists of four phases described as follows:

4.1 Topology discovery

One of the essential aspects of any BAN is the topology discovery. In our network scenario, the nodes use a simple technique to discover neighboring nodes as well as to detect transient channel variations and topology changes (e.g., node or link failure, injection of new nodes, etc.) similar to the one proposed in [22]. During some predetermined intervals, the nodes exchange HELLO messages to detect nodes in the surrounding neighborhood. In order to keep a constant record of neighboring activity, each node in the network will form a registry of neighbors (i.e., neighbor table). This registry will hold only the required information for forming, maintaining, and breaking connections.



(a) SPT, sink placed at waist

(b) SPT, sink placed at ankle

(c) Path-Degree, sink placed at waist

(d) Path-Degree, sink placed at ankle

(e) Path-Energy, sink placed at waist

(f) Path-Energy, sink placed at ankle

Fig. 2 Received data quality of accepted high priority nodes as a function of number of high priority (HP) nodes

4.2 Energy-aware routing tree construction

To achieve high network performance in BAN, the route construction within the network is a crucial task. To maximize network operable lifetime and throughput capacity, we use nodes' residual energy as a selection metric, if two nodes have the same residual energy; the node with shorter path to the sink will then be selected as a parent, details of the selection procedure are discussed above in Sect. 3.2.

4.3 Rate and bandwidth allocation

Our solution consists of computing the number of high priority streams (flows) in the tree and then to assign a bandwidth

Fig. 3 Pseudo-code of the ensuring consistent link capacity step

```
▷ Ensure consistent link capacity along the path
for each level $h = 1$ to $TreeDepth - 1$ do
    for each node $i$ such that $H(i) = h$ do
        if $UplinkCap(i) > UplinkCap(prnt(i))$ then
            $UplinkCap(i) = UplinkCap(prnt(i))$
```

Fig. 4 Pseudo-code of the bandwidth assignment step

```
▷ Assign uplink capacity proportionally to Flow count
for each level $h = TreeDepth$ to 1 do
    for each node $i$ such that $H(i) = h$ and $F_p > 0$ do
        $UplinkCap(i) = \sum_{j \in Child(i)} UplinkCap(j)$
        $UplinkCap(i) = UplinkCap(i) + (AvailableBandwidth/F_p)$
        ▷ Check physical link capacity constraints
        if $UplinkCap(i) > L_{i,prnt(i)}$ then
            $UplinkCap(i) = L_{i,prnt(i)}$
        ▷ If allocated bandwidth does not satisfy QoS requirements,
        put node $i$ offline
        if $UplinkCap(i) < W_{min}$ then
            $UplinkCap(i) = UplinkCap(i) - r_i$
            $F_p = F_p - 1$, $r_i = 0$
```

(capacity) to each branch that is proportional to this quantity. We then check whether this amount of bandwidth is large enough to accommodate the data stream (i.e., $> W_{min}$). If this is not the case, the flow counts are adjusted accordingly and the capacity assigned to this branch is released. It means that the flow(s) that pass through this branch have to be put offline in this critical situation where there is not enough available bandwidth. The capacity of this branch is redistributed to the neighboring branches. Similarly, these capacity allocation must also be compared to the physical capacity of each branch (i.e. L_{ij}). When the assigned amount of capacity is too large to be accommodated by the branch, it is then adjusted and the remaining capacity is assigned to the neighboring branches. This procedure is repeated iteratively from the leaf nodes towards the sink in a leaves-to-sink manner. When several priorities are available (we consider two priority levels in this work i.e., high and low), the algorithm starts with the highest level of priority. The remaining bandwidth is then shared by the lower level of priority by a new instance of the algorithm and so on. The proposed algorithm runs independently at each sink in a centralized manner. The three steps of the algorithm are then described as follows:

Initialization step The algorithm starts with initialization of the variables. More precisely, the number of flows with priority p in the tree and denoted by F_p. The parent node of a node i is denoted by $prnt(i)$. Consequently, the set $Child(i)$ represents the set of nodes which are directly connected to node i. $H(i)$ denotes the hierarchical level of node i in the tree. (e.g., $H(sink) = 0$, $H(i) = 1$ for a node i that is directly connected to the sink, etc). The capacity (bandwidth) allocated to node i is denoted by $UplinkCap(i)$. It corresponds to the bit rate available for this node at the uplink (how much traffic it can relay including its own generated data). This value is bounded by the cell capacity C at each node as well as to the physical link capacity $L_{i,prnt(i)}$. The $UplinkCap(i)$ is dynamically updated to ensure the accuracy of the algorithm based on the available capacity upstream and the number of high priority flows. Once initialized, these $UplinkCap$ values should be coherent. In fact, the available $UplinkCap$ of a node can be greater than that of the nodes in the path to the sink. Figure 3 demonstrates the procedure used to ensure a node is assigned $UplinkCap$ not greater than the $UplinkCap$ of the nodes in its path towards the sink.

Rate and bandwidth allocation step Next, the main capacity allocation procedure, illustrated in Fig. 4, is executed (once for each priority level, starting with nodes with high priority value of p). Once this procedure is completed, the data rate of the nodes of priority p can be then determined. Starting from leaf nodes, a leaf node i will be assigned rate $r_i = UplinkCap(i)$, such that the QoS requirements are satisfied. The node will update its incoming and outgoing traffic ($inTraffic$ and $outTraffic$ variables, respectively) to allow accurate rate allocation for upstream nodes. For leaf node i, $inTraffic = 0$ and $outTraffic = r_i$. Nodes in the upper level will then compute their $inTraffic$, and their rates will be calculated as follows: $r_j = UplinkCap(j) - inTraffic(j)$. The rate assignment process will recursively continue level by level until the sink is reached.

Allocation improvement step After the nodes' data rates are assigned according to the previous steps, we check for any extra available bandwidth at each node. For instance, a leaf node might be allocated an $UplinkCap > W_{max}$, in this case $UplinkCap - W_{max}$ extra bandwidth is available. In our algorithm, such extra bandwidth will be allocated to nodes without violating QoS constraints. Checking for extra bandwidth process commence in a leaves-to-sink manner, where unallocated bandwidth at node i is computed as $UplinkCap(i) - (r_i + inTraffic(i))$ and will be moved to the sink to be allocated to other nodes where possible. This procedure will be repeated recursively until there is no extra bandwidth available in the network or the given constraints (QoS) are not permitting any further bandwidth allocation.

4.4 Load balancing routing and energy usage

ARBA employs a load balancing strategy so that all nodes remain up and running together for as long as possible. The proposed algorithm is run repeatedly in a series of rounds. To ensure fairness, the group of selected nodes is rotated periodically to ensure energy-balanced operations. A node switches state from time to time between being *Accepted* as traffic source and being *non-Accepted*. While *Accepted*, the node continuously streams sensorial data and, possibly, forwards other nodes' data up towards the sink. After a node remains in *Accepted* state for time T_{round}, it changes its state to *discovery* state to give a chance to other nodes in the neighborhood to be accepted by the sink. Recall that, nodes are ranked according to their importance (type of equipped sensors and sensorial data) and remaining energy levels. When the accepted node changes its state to *discovery*, it is more likely that it has less remaining energy in its battery than its neighbors because presumably the neighbor nodes were in the *power-saving* (non-accepted) mode for time T_{round}. Consequently, the node that was accepted is less likely to remain in its current state after the *discovery* phase. Also, when a node detects topology change (e.g., link or node failure) it sends an EMERGENCY message to the sink to react and adaptively re-allocate available bandwidth to the nodes. It is worth mentioning that another reason why we chose to run ARBA in rounds is to react to topology changes (e.g., node or link failure, adding new nodes, etc.) and dynamic network conditions. Figure 5 illustrates an example of 200 s application cycle with ARBA.

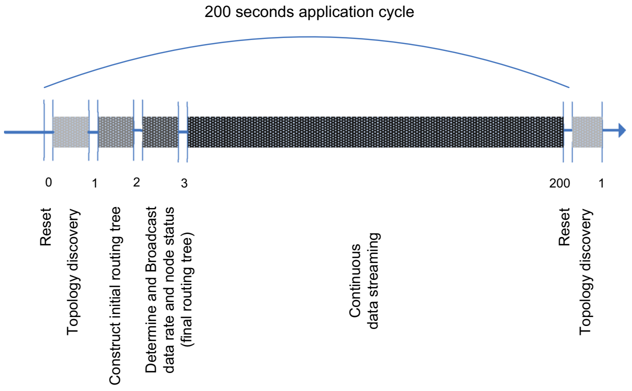

200 seconds application cycle
0
1
2
3
200
1
Reset
Topology discovery
Construct initial routing tree
Determine and Broadcast data rate and node status (final routing tree)
Continuous data streaming
Reset
Topology discovery

Fig. 5 The different phases of the network cycle with ARBA

5 Performance evaluation

For the evaluation, we have conducted an extensive set of experiments using a VC++ coded simulator. We studied the performance of the the proposed protocol, ARBA, by comparisons with the optimal solution. We simulated 16 node network illustrated in Fig. 6a, where high priority nodes are randomly selected in each run. To be accurate, we solved the integer linear program presented in (3) by using AMPL and CPLEX. The wireless channel capacity is $C = 2$ Mbps, and link capacity $L = 1$ Mbps. The minimum acceptable data rate generated by each node R_{min} is 128 Kbps and R_{max} is fixed at 512 kbps. The minimum utility of each accepted node $U_{min} = 1$ and $U^e_{step} = U^r_{step} = 1/4$ (i.e., R_{min}/R_{max}). Our goal of implementing ARBA protocol is to minimize the total energy spent in the network to communicate the information gathered by sensor nodes to the information-processing center (i.e., the sink), while achieving higher network utilization. Unless otherwise stated, all the following investigations adopt these values. For all simulation results in this paper, each point in the plots is averaged over 10 runs, and the simulation lasts for 5,000 s. Varying number of high priority nodes (data streams) and sink location (node #0 and node #15 in Fig. (6)), by mean of simulations, we are interested in the following issues:

1. How many traffic sources are accepted in the network and how does that affect the network utility?
2. How is the distribution of energy dissipation of nodes and how is this affected by routing tree size?
3. How the sink placement affects the network energy behavior and network performance? One of the commonly made assumptions in BAN literature is that the sink node is located at the waist [1,2,23]. To investigate how the sink placement affects the network performance, we repositioned the sink to the ankle.

To get initial answers to these questions, we focused our simulations on the tree construction and rate allocation process, which are mainly responsible for network utility and energy consumption. We first evaluate the performance of ARBA using both algorithmic, ARBA-ALG, and optimal, ARBA-OPT, rate assignment followed by performance comparison of ARBA to the full optimal solution (i.e., joint routing tree construction and rate assignment).

5.1 ARBA performance comparison with optimal rate assignment

5.1.1 Load distribution and energy savings

Network lifetime is a crucial metric of BAN, but it can be a crude measure of actual energy consumption because node

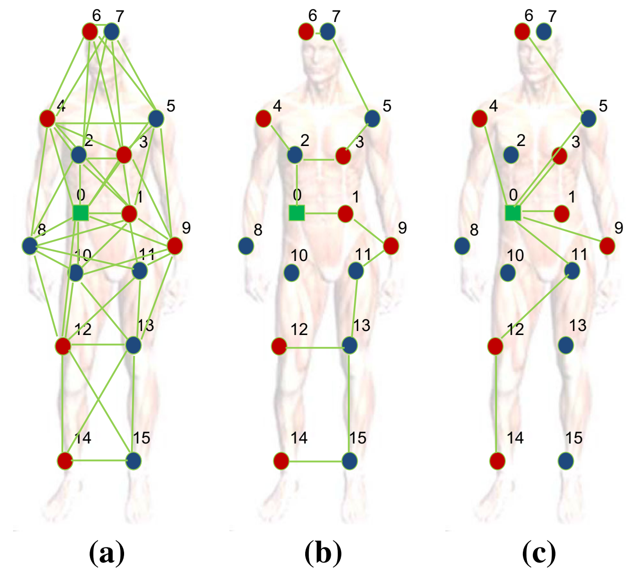


Fig. 6 Example of BAN routing tree formation based on ARBA: **a** original topology, **b** optimal, and **c** ARBA. A *red filled circle* represents a high priority node, a *blue filled circle* represents a regular node and a *green filled square* represents the sink. *Solid line* indicates communication link, and each node is sending a constant bit rate (CBR) stream to the sink with radios capable of transmitting up to 0.5 Mbps (Color figure online)

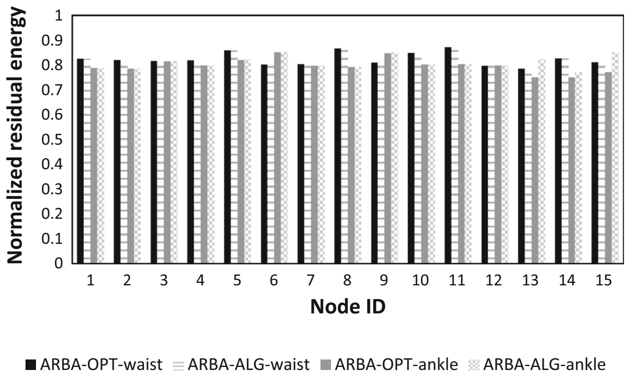


Fig. 7 Residual energy distribution of nodes after 2000 s simulation of ARBA: The nodes are initially equipped with 2 J battery energy

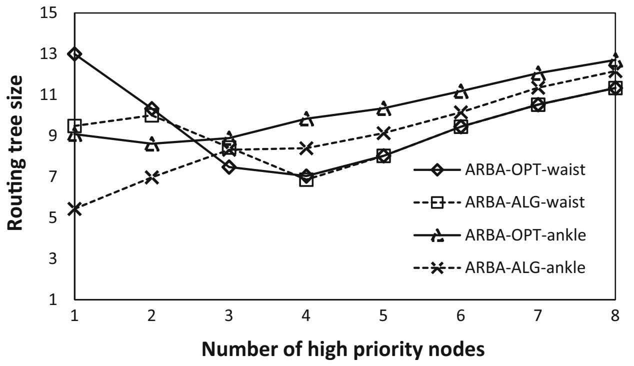


Fig. 8 Routing tree size as a function of number of high priority nodes

life is binary, so *n* about to die nodes are considered good as *n* fresh nodes. In order to quantify how much energy ARBA saves, we instead plot the residual energy per node. In order to observe how well ARBA promotes load balancing and thus saves energy among the nodes compared to the optimal solution, we ran a simulation for 2,000 s. At the outset, each node had 2 J battery energy. For simplicity, we only account for the radio receiving and transmitting energy. Figure 7 shows relative residual energy at the end of simulation across nodes of ARBA and the optimal solution. It is obvious that ARBA performs well in terms of balancing out energy consumption in the network, which yields to longer network lifetime. As expected, the sink location at the waist is more energy efficient than the ankle location. This is mainly due to the reduced average hops between source nodes and the sink resulting in lower forwarding energy cost. This effect is more pronounced for the ankle location. This confirms that depth has a negative impact on network energy distribution as the sink will have smaller node degree and network congestion may occur in this case. It is interesting to note that when sink is placed at waist, nodes have more residual energy, this is because the routing tree size (i.e., number of nodes that forms the routing backbone) is smaller as depicted in Fig. 8.

5.1.2 Routing tree size

Figure 8 shows the impact of the number of high priority nodes on the total size of the routing trees (i.e., total number of nodes in the resulting routing backbone) that each solution obtains. We note that all curves depict a larger routing tree for larger number of high priority nodes, which suggests that when increasing number of high priority nodes in the network the proposed protocol tries to accept as many nodes in the network as possible and thus increases the routing tree size. Tree size influences the total consumed energy and thus network operable lifetime. The larger the tree the more energy consumed in the network. It is apparent that, when the sink is located at the ankle both solutions attain larger routing trees, this is because far away nodes need to have more relay nodes to communicate their data to the sink.

5.1.3 Accepted traffic sources

Figure 9 depicts the number of accepted nodes in each solution for different number of high priority nodes. As can be seen from the graphs, the proposed scheme reacts smoothly and consistently as the number of high priority nodes increases. It is clear that, as the number of high priority nodes increases, ARBA scheme experiences an increase in the number of accepted nodes. In other words, as we increase the number high priority nodes deployed in the network, the

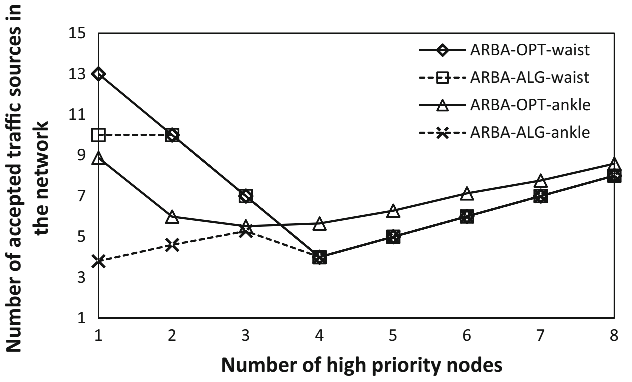


Fig. 9 Number of accepted traffic sources as a function of number of high priority nodes

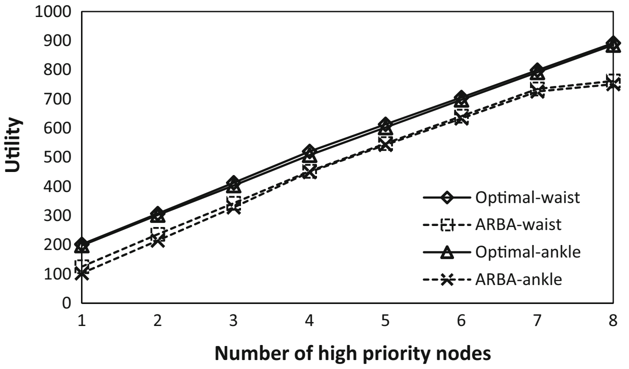


Fig. 10 Performance comparison of network utility as a function of number of high priority nodes

proposed solutions will try to accept as many of them as possible with best data rate as discussed above. When there is less than 4 high priority nodes deployed in the network, ARBA is able to accept them at the highest data rate and, furthermore, accept an additional number of low priority nodes at lesser data rate.

5.2 ARBA performance comparison with optimal solution

In this section, we extend the rate assignment model presented in Sect. 3.2 to account for routing tree construction too. We redefine x_{ij} to be a 0–1 integer variable for each edge $(v_i, v_j) \in E, x_{ij} = 1$ if the edge is included the resulting tree (i.e., edge $(v_i, v_j) \in E'$). Thus, constraint (1) is re-written as follows:

$$x_{ij} \leq y_{ij} \leqslant L_{ij} \times x_{ij}, \forall i \in V_n, \forall j \in V : (i, j) \in E \tag{9}$$

And, additional constraints for the tree construction are added as follows:

$$\sum_{\forall j \in V:(i,j)\in E} x_{ij} = 0, \forall i \in V_s \tag{10}$$

$$\sum_{\forall j \in V:(i,j)\in E} y_{ij} = 0, \forall i \in V_s \tag{11}$$

$$\sum_{\forall j \in V:(i,j)\in E} x_{ij} \leq 1, \forall i \in V_n \tag{12}$$

Constraint (10) prevents a sink node from having a parent. Constraint (11) denotes that a sink has no uplink traffic within the network. Constraints (12) ensures that each node has exactly one parent.

5.2.1 Utility

To study the impact of the number of high priority nodes on the network performance we vary the number of high priority nodes from 1 to 8. The utility of a data transmission rate below R_{min} (i.e., 128 kbps) is considered to be insignificant, hence, the node is put offline. Utility for each offline node is assumed to be zero. We can clearly see in Fig. 10 that ARBA protocol experiences a linear utility increase with the increase of high priority nodes. This is because as more high priority nodes are deployed, ARBA will try to accommodate as many of them as possible at the best possible data rate, which results in a better utility. In Fig. 10, we note that the gap between the outcome of ARBA and the optimal solution tends to remain nearly constant, which suggests that the number of high priority nodes (i.e., the problem size) has little impact on the performance ratio between the two. It is also interesting to note that, sink location has no impact on the network performance in terms of utility.

5.2.2 Throughput

We also studied per node throughput when the sink is located at waist. The throughput of a node *a* is defined as the amount of data received from node *a* at the sink over time. Figure 11a, b shows each node contribution in overall system throughput and times a node is selected as high priority node (i.e., times selected over number of runs), respectively. It is interesting to note that nodes' throughput (rate) is proportional to number of times they are selected as high priority nodes. When a node is selected as high priority node both ARBA and optimal solution try to allocate the highest rate to it. In the comparison, to ensure fairness, we have evaluated both solutions under the same topologies. That is why we see the same values of the number of times a node is selected as high priority node for both solutions in Fig. 11b.

5.2.3 Load distribution and energy savings

Figure 12a, b shows relative residual energy at the end of simulation across nodes of ARBA and the optimal solution

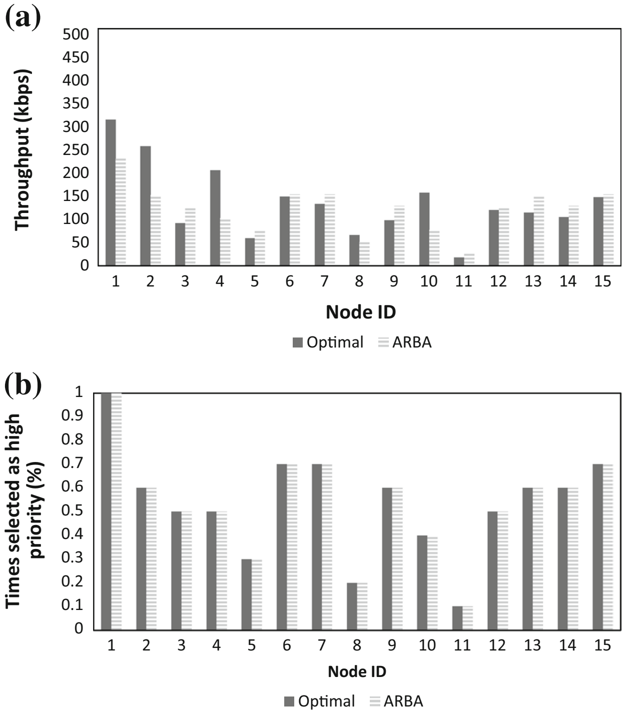


Fig. 11 Performance comparison of throughput when sink is placed at waist

when there is 5 and 8 high priority nodes, respectively. It is also important to note that ARBA outperforms the optimal solution in terms of residual energy when the sink is placed at waist, this is because the routing tree size is smaller in ARBA as the optimal solution (i.e., MILP) does not account for hop-count and a node may select a longer path to the sink as long as that path provides the highest possible data rate and best residual energy. The behavior explanation of ARBA appears in Fig. 7 is still valid here.

6 Conclusion and future work

This paper proposes an adaptive routing and bandwidth allocation protocol, termed ARBA, that performs real-time monitoring of complex conditions on streaming data from various body sensors within a BAN. ARBA calculates the utility of different streams, and based on the estimated utility, decides which stream(s) to put offline so that overall utility of the whole monitoring system is improved. BANs for healthcare put more emphasis on adaptation to changes in context and application requirement, thus, ARBA adopts an adaptive resource allocation policy to continuously meet QoS requirements. Furthermore, the proposed solution no longer considers the naive approach of only sending data to the direct neighbors of a node. Instead, the data is transported more intelligently and the burden of forwarding the data is more equally spread among the nodes. We studied the energy saving and how it is affected by factors such as sink placement and the number of high priority data streams. As expected, it was found that depth has a negative impact on energy consumption. It is shown in the paper that the ARBA is energy efficient for streaming communication while balancing energy consumption across the BAN guarantees longer lifetime.

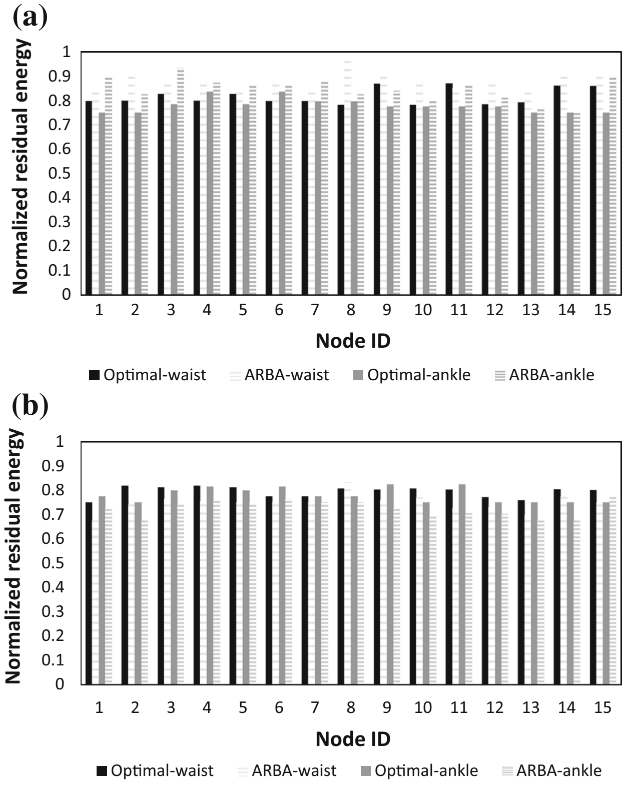


Fig. 12 Residual energy distribution of nodes after 2000 s simulation of ARBA, **a** 5 high priority nodes selected, and **b** 8 high priority nodes selected: The nodes are initially equipped with 2 J battery energy

The results presented in this paper indicate that the gains achievable under ideal channel conditions with no transmission errors. Future work will be necessary to characterize the performance under non-ideal channels. We are also interested in extending our optimization model to account for hop-count metric.

References

1. Ullah, S., Higgins, H., Braem, B., Latre, B., Blondia, C., Moerman, I., et al. (2010). A comprehensive survey of wireless body area networks. *Journal of Medical Systems*, *36*, 1–30.
2. Chen, M., Gonzalez, S., Vasilakos, A., Cao, H., & Leung, V. C. M. (2010). Body area networks: A survey. *Mobile Networks and Applications*, *15*, 853–865.

3. Natarajan, A., de Silva, B., Yap, K.-K., & Motani, M. (2009). To hop or not to hop: Network architecture for body sensor networks. In *Proceedings of IEEE SECON* (pp. 1–9).
4. Benoît, L., Günter, V., Luc, M., & Piet, D. (2004). Networking and propagation issues in body area networks. In *SCVT*.
5. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A survey on sensor networks. *IEEE Communications Magazine*, *40*(8), 102–114.
6. Ren, F., Zhang, J., He, T., Lin, C. & Ren, S. (2011). EBRP: Energy-balanced routing protocol for data gathering in wireless sensor networks. In *IEEE Transactions on Parallel and Distributed Systems, no. 99*.
7. Zhou, G., Lu, J., Wan, C.-Y., Yarvis, M.D., & Stankovic, J.A. (2008). Bodyqos: Adaptive and radio-agnostic qos for body sensor networks. In *Proceedings of IEEE INFOCOM*.
8. Ababneh, N., Timmons, N. & Morrison, J. (2012). An adaptive bandwidth allocation scheme for data streaming over body area networks. In *Proceedings of WWIC*.
9. Liu, J., & Zhang, D. (2009). Toward accurate and efficient available bandwidth measurement. *Telecommunication Systems*, *41*(3), 211–227.
10. Santos, D., de Sousa, A., Alvelos, F., Dzida, M., & Pióro, M. (2011). Optimization of link load balancing in multiple spanning tree routing networks. *Telecommunication Systems*, *48*(1–2), 109–124.
11. Masri, W., & Mammeri, Z. (2010). Mapping density to bandwidth in tree-based wireless sensor networks. *Telecommunication Systems*, *43*(1–2), 73–81.
12. Hu, C.-C., Kuo, Y.-L., Chiu, C.-Y., & Huang, Y.-M. (2010). Maximum bandwidth routing and maximum flow routing in wireless mesh networks. *Telecommunication Systems*, *44*(1–2), 125–134.
13. Moh, M., Culpepper, B. J., Dung, L., Moh, T.-S., Hamada, T. & Su C.-F. (2005). On data gathering protocols for in-body biomedical sensor networks. In *Proceedings of IEEE GLOBECOM*.
14. Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, *1*(4), 660–670.
15. Ren, H., & Meng, M.Q.-H. (2006). Rate control to reduce bioeffects in wireless biomedical sensor networks. In *Proceedings of MobiQuitous*.
16. Tang, Q., Tummala, N., Gupta, S. K. S., & Schwiebert, L. (2005). Communication scheduling to minimize thermal effects of implanted biosensor networks in homogeneous tissue. *IEEE Transactions of Biomedical Engineering*, *52*(7), 1285–1294.
17. Bag, A. & Bassiouni, M. A. (2006). Energy efficient thermal aware routing algorithms for embedded biomedical sensor networks. In *Proceedings of IEEE MASS*.
18. Braem, B., Latre, B., Moerman, I., Blondia, C., & Demeester, P. (2006). The wireless autonomous spanning tree protocol for multihop wireless body area networks. In *Proceedings of MobiQuitous*.
19. Latre, B., Braem, B., Moerman, I., Blondia, C., Reusens, E., Joseph, W. & Demeester P. (2007). A low-delay protocol for multihop wireless body area networks. In *Proceedings of MobiQuitous*.
20. Wattenhofer, R. & Zollinger, A. (2004) XTC: A practical topology control algorithm for ad-hoc networks. In *Proceedings of IPDPS*.
21. De Couto, D. S. J., Aguayo, D., Chambers, B. A. & Morris, R. (2002). Performance of multihop wireless networks: Shortest path is not enough. In *Proceedings of HotNets-I (Princeton, New Jersey)*.
22. Sohrabi, K., Gao, J., Ailawadhi, V., & Pottie, G. J. (2000). Protocols for self-organization of a wireless sensor network. *IEEE Personal Communications*, *7*(5), 16–27.
23. Latre, B., Braem, B., Moerman, I., Blondia, C., & Demeester, P. (2011). A survey on wireless body area networks. *Wireless Networks*, *17*, 1–18.

Nedal Ababneh (MIEEE) received his Ph.D. in 2009 from The University of Sydney, and Master degree in 2004 from The University of New South Wales, Australia, both in Computer Science and Engineering. In Sep 2012, he joined the department of Computer Engineering at The University of Bahrain as an Assistant Professor. Before that, he worked as Postdoctoral Research Fellow at WiSAR Lab, LyIT, Ireland for two years, wherein he conducted research and development work in the development of ultra low-power protocols for Wireless Sensor Networks solutions, with an emphasis on Body Area Networks. His current research interests lie broadly in the design, implementation, real deployment, and experimentation of wireless communications protocols. In Particular, he has an interest in Wireless Sensor Networks, Medical Body Area Networks, Communications for Intelligent Transport Systems and WiMAX Wireless Mesh Networks.

Nicholas Timmons (MIEEE) holds a Ph.D. (2011) in Wireless Body Area Networks from Queen's University Belfast, an M.Sc. (1994) in Satellite Engineering Design from the University of Surrey, and a B.E. (Hons) (1989) from the University of Ulster, Jordanstown. He is also a member of the ECIT Radio Communications research group at Queen's University Belfast. He joined LyIT in 1997 as a Lecturer and was a co-founder of the WiSAR Lab in 2009. Prior to that he had 10 years industrial experience in the area of RF design including roles as a Product Manager for Connaught Electronics leading the development of low power RF transceivers for the automobile industry and as a Senior Principal RF Engineer in Siemens working on Satellite Communications RF/microwave design. His current research interests are in the areas of Wireless Sensor Networks (WSNs), Medical Body Area Networks (MBAN), Sensors, Ultra Low Power Communication Protocols: MAC and Network/Routing protocols and Energy harvesting.

Jim Morrison (MIEEE) holds a Ph.D. (2006) in Digital Security from the University of Limerick, an M.Sc. (1992) in Industrial Applications of Computing from the Open University and a B.E. (1981) in Electronics and Electrical Engineering from Queens University of Belfast. He has been a member of the IEEE and its Communications Society and Computer Society for over 20 years. He joined LyIT in 1985 as a Lecturer and became Head of Department of Electronics and Mechanical engineering in 2004. He was a co-founder of the WiSAR Lab in 2009 and was a co-founder of the associated EpiCentre in 2004. His current research interests are in the areas of Wireless Sensor Networks (WSNs), Medical Body Area Networks (MBAN), Sensors, Ultra Low Power Communication Protocols and associated security issues.