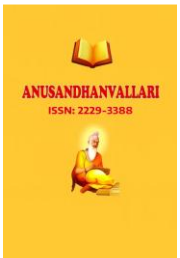

Anusandhanvallari
Vol 2023, No.1
July 2023
ISSN 2229-3388

Self-Adaptive Greedy Buffer Allocation and Scheduling (Sgbas) for Energy-Efficient Body Sensor Networks

Adhip Kishan Bopaiah[1], Dr. Yash Pal Singh [2]

Department of School of Electronics & Communication

[1,2] Sabarmati University University in Gujarat

Abstract: In order to minimize energy usage and reduce end-to-end delay in Body Sensor Networks (BSNs), this study introduces a method called Self-Adaptive Greedy Buffer Allocation and Scheduling (SGBAS). SGBAS reduces the number of transmissions and node energy consumption by adaptively resizing buffers and scheduling packet transmissions to minimize redundant data forwarding. We test the suggested approach in NS2 with different sensor fluxes and deployment densities. With up to 39% less energy consumption and major reductions in latency, the simulation findings show that standard Greedy Buffer Allocation (GBA) is significantly outperformed. The results show that SGBAS is a good way to extend the life of BSNs without lowering their data delivery performance.

Keywords: Body Sensor Networks, Sensed Data Processing, Burst Communications, Adaptive Buffer Allocation

1.INTRODUCTION

A greater understanding of the greenhouse effect has resulted from the ever-increasing number of computing devices, which in turn has increased research into power-conscious computing. It is also driven by the necessity to meet the operational requirements of battery-powered sensor networks in the long run **(Kanoun, et al 2021).** When it comes to computing for mobile and wearable healthcare monitoring systems, BSNs hold a lot of promise. Miniature sensor nodes allow these platforms to monitor vital signs and environmental conditions, take appropriate action, and notify users of impending danger. By bringing healthcare out of the clinic and into the house, BSNs may remotely monitor and test patients. It becomes more important to improve communication with lower power and buffer consumption as the network increases, despite the tiny size and low cost of these sensors. The motivation for this endeavor came from issues with power-aware computers and sensor networks that rely on batteries.

The number of tiny nodes in a sensor network might range from hundreds to thousands. The standard components of a sensor node include a pair of sensors, memory banks, a radio, and a microcontroller running a minimal operating system. It is possible for the sensor node to quantify, collect, send/receive, and route data packets. These implanted sensor nodes have the ability to self-organize into a perceptive network, which allows for round-the-clock healthcare monitoring, habitat monitoring, and distinctive emergency reactions. These sensor nodes need to run in harsh conditions for months or perhaps years without any help from humans or repairs. It may be impossible to recharge or replace batteries under some situations. Consequently, the capacity of body sensor networks to identify uncommon occurrences or consistently track evolving events is dictated by their lifespan. Research by **Behera et al. (2022)** suggests the Self-Adaptive Greedy buffer allocation and scheduling algorithm (SGBAS) as a means to minimize energy consumption and make better use of buffers. The technique prevents sensors from transmitting duplicate data to the BSN and adaptively changes the length of the buffer. To measure the risk of sensing continuity, we utilize an average sensing interval, which is the time it takes for a sensor node to send or receive a sensed message packet. By keeping an eye on these sensing intervals, we can roughly calculate the amount of buffer usage when sending or receiving signals from the sensor node. In order to lower the communication cost, each sensor module has a massive buffer that may be adjusted to its optimal size. According to **Zagrouba and Kardi (2021) and Nakas et al. (2020),** real-time sensing is not necessary for most medical monitoring applications. In order to optimize burst transmission energy, it is necessary to remove immediate data-processing deadlines. In signal processing, input from nearby sensor nodes is combined by forwarding modules in a directed acyclic dependency graph. Simulations show that our method reduces energy consumption by 78% without adding overhead and shortens the time it takes to reach sink nodes.

2. LITERATURE REVIEW AND RESEARCH GAP

Optimizing routing and data management for improved energy efficiency and reduced transmission delays has been the focus of recent research in Body Sensor Networks (BSNs). An extensive review of energy-aware system design for autonomous wireless sensor nodes was given by **Kanoun et al. (2021),** who stressed the significance of adaptive techniques in extending operational lifetime. Adaptive scheduling and buffer optimization were highlighted as critical components in the energy-efficient routing protocols examined by **Behera et al. (2022)** that improved performance. There are a number of protocols that deal with energy saving, but the most of them ignore the complementary advantages of scheduling and buffer management in BSNs in favor of routing. There has been little research into reducing redundant transmissions and maintaining low end-to-end delay through the use of intelligent scheduling in conjunction with self-adaptive buffer allocation, despite improvements in routing efficiency. The suggested SGBAS algorithm fills this void by dynamically adjusting buffer sizes and transmission scheduling to achieve a balance between energy consumption and timely data delivery.

3.METHODOLOGY

The sensor nodes in our system (A, B, C, D) have different capacities and are all powered by a battery. They each have a buffer, processing, and routing module **(Yalçın, & Erdem, 2022).** Size of the buffer, which stores sensed data, depends on the data type and the type of sensor node it is connected to. Data in the buffer is processed by the processing module, which determines which data to broadcast at a moment **(Vellela, & Balamanigandan, 2023).** Nodes can contain a variety of sensors, including but not limited to temperature, pressure, gyro, accelerometer, and more. Various areas of the body are equipped with sensor nodes. Physical activity, internal body changes, health concerns, and additional processing can affect the sensed data acquired by each sensor node.

Data that has already undergone preprocessing, segmentation, feature extraction, and classification is sent into our node. We then transfer this processed data to the buffer. Using the information stored in the buffer, each sensor node determines its own classification..

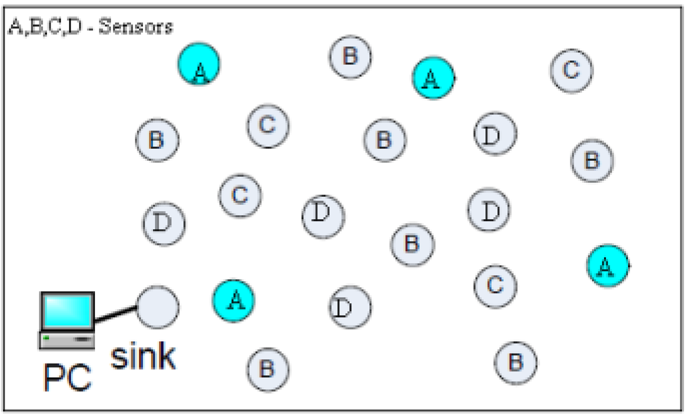

Figure 1: Sample Scenario

A central node makes a final decision using the buffer's all-local data. Because nodes only have so much RAM, the amount of data that buffers can hold is limited. Reading, segmentation, feature extraction, and classification all use distinct hardware processing units, which in turn generate data blocks of varying sizes. There are two kinds of buffers that are kept by our communication model for every sensor node. To facilitate in-node processing, a buffer known as an intra-node buffer is used. The subsequent processing unit on the same node will use the data stored in this buffer **(Wang, et al 2012).** In addition, an inter-node buffer is a buffer that is assigned to a processing unit as a result of dependencies between nodes. Data for subsequent dependent units is provided by these buffers. The amount of activities that each connection would store determines the size of the inter-node buffer. A buffer is assigned to the source unit and another buffer to the destination unit by an inter-node link with an action count of $X_{ij}{}^{(1j)}$ Data for both links can be stored in a single buffer when a source unit has multiple outgoing edges. Adaptively, the size of this buffer is set according to the maximum data that needs to be communicated among the links. The sizes of the inter-node buffer are subsequently determined by the values of $X_{ij}{}^{SS}$ and $X_{ij}{}^{SF}$. However, the link's maximum necessary size is determined dynamically and adapts itself to handle the data provided by this unit. There is congestion when the wait time is long. Instead of using the actual wait time, which might be affected by sudden changes in demand, we utilize the average wait time **(Kulkarni, & Wang, 2005).** Out of these queues, packets are either ignored or processed independently. In the buffer, each packet that arrives will be queued. If not, the type of sensed data should dictate the likelihood of dropping a packet. All packets, regardless of their source sub-node, have the same priority. The one down below is a packet scheduling technique that determines

Application Layer		
Self-Adaptive buffer	QoS Optimization	Burst Communication
EEBC Routing		
IEEE 802.11		
Physical Layer		

Figure 2: System Architecture

Self-Adaptive Greedy buffer allocation and scheduling algorithm (SGBAS):
Input: Sensed local Data, Data from neighbor nodes.
Output: Queue in buffer and forward schedule.
Variables: Queue for local packets, Queue for transit packets, minimum threshold, maximum threshold, current buffer lengths, incoming packet.

Figure3 : algorithm (SGBAS):

4. PERFORMANCE EVALUATION AND RESULTS

Here we see how the suggested architecture affects power consumption by displaying the mean remaining power of all sensor nodes after 50 seconds of simulation with varying flows (e.g., from Sensors A, B, C, and D) and periodic transmissions. At the outset, every sensor node has 100 joules. Processing (including adaptive resizing and scheduling) consumes 0.142 watts/sec, packet reception 0.022 watts/sec, and transmission 0.016 watts/sec. Table 1 below shows the different simulation parameters that were employed. To test the suggested method, we run a number of tests using the Body Sensor Network in the NS2 simulator.

Table 1: Simulation Parameters

Parameter	Value
Transmission range	2.5 m
Initial energy	100 joules
Idle power	0.0012 watt
Sleeping power	0.0001 watt
Transmitting power	16.0 mW
Receiving power	22.0 mW
Processing power	142.0 mW
Bandwidth	1 mbps
Area	15 m ∗ 15 m
Carrier sense threshold	3.22 m
Rx threshold	1.565 m

Due to periodic monitoring by several types of sensors, SGBAS consumes energy rapidly as time grows. The energy consumption of BSN nodes is recorded during a single flow and then assessed with additional flows of periodic sensing. Energy consumption is sufficiently stable, according to the results. The same is assessed as the number of sensor nodes increases. Regardless of the density of sensors deployed, the intermediary nodes rarely vary since the sensed data is sent to the sink using the shortest hop-path policy. Without a buffering strategy, data from each sensor must be transmitted to the neighboring sensor whenever a sensing event occurs in order for the data to reach its destination, which might be the sink. During the 50 seconds when the system is operational, each node sends out 231 transmissions. For two setups, Table 2 compares greedy buffer allocations to SGBAS. In order to assess the time complexity of SGBAS, the end-to-end delay is examined. The results show that the protocol is faster than greedy algorithms..

Table 2 Average Energy Consumption between SGBAS and GBA

No. of Sensors	Average Energy Consumption(J)	
	GBA	SGBAS
20	23.3225	14.1632
40	25.7552	16.2539
60	24.8826	17.8586
80	24.6554	19.8586
100	24.6228	21.8586

Anusandhanvallari
Vol 2023, No.1
July 2023
ISSN 2229-3388

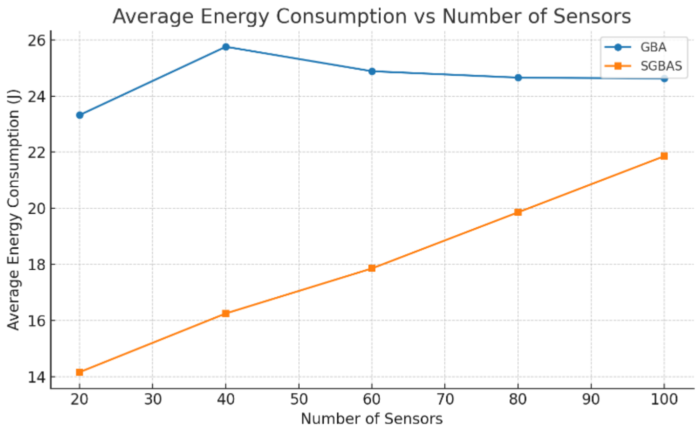

Figure 3: Energy Consumption

Table 3: End-To-End Delay Comparison For Experiment1

No. of Sensors	Average End-to-End Delay	
	GBA	SGBAS
20	12.1084	6.85872
40	12.5853	5.86157
60	12.0417	5.32473
80	13.0941	5.96680
100	13.3403	5.87954

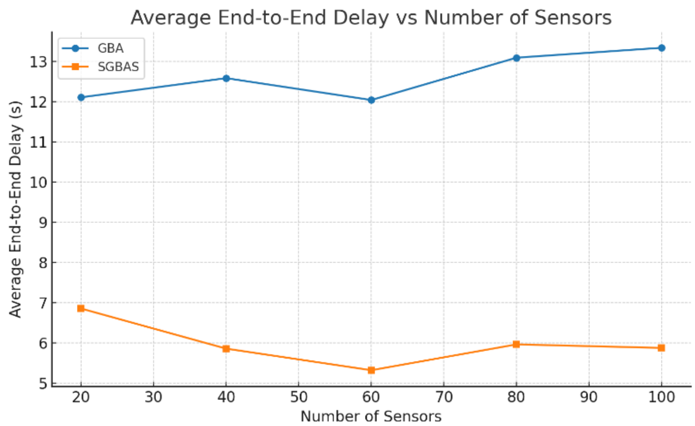

Figure 4: End-to-End Delay

5.CONCLUSION

In order to decrease the amount of transmissions over the BSN networks, we provide an algorithm called SGBAS, which stands for Self-Adaptive Greedy buffer allocation and scheduling. Successful packet handling and scheduling in queues are achieved by means of the method's self-adaptive buffer mechanism. As an added bonus, it reduces energy usage and guarantees the lightest data transmission latency possible from beginning to end. The suggested SGBAS scheduling method outperforms the state-of-the-art GBA and Multilevel Queue Scheduler in terms of energy consumption and average end-to-end delay, according to the experimental results. Reducing processing overhead and saving bandwidth are two potential areas for future growth. In addition, additional network factors should be considered when evaluating the suggested strategy. Additionally, we would use a physical test bed to verify the accuracy of the simulation.

REFERENCE

1. Kanoun, O., Bradai, S., Khriji, S., Bouattour, G., El Houssaini, D., Ben Ammar, M., ... & Viehweger, C. (2021). Energy-aware system design for autonomous wireless sensor nodes: A comprehensive review. *Sensors*, *21*(2), 548.
2. Behera, T. M., Samal, U. C., Mohapatra, S. K., Khan, M. S., Appasani, B., Bizon, N., & Thounthong, P. (2022). Energy-efficient routing protocols for wireless sensor networks: Architectures, strategies, and performance. *Electronics*, *11*(15), 2282.
3. Zagrouba, R., & Kardi, A. (2021). Comparative study of energy efficient routing techniques in wireless sensor networks. *Information*, *12*(1), 42.
4. Nakas, C., Kandris, D., & Visvardis, G. (2020). Energy efficient routing in wireless sensor networks: A comprehensive survey. *Algorithms*, *13*(3), 72.
5. Vellela, S. S., & Balamanigandan, R. (2023). An intelligent sleep-awake energy management system for wireless sensor network. *Peer-to-Peer Networking and Applications*, *16*(6), 2714-2731.
6. Yalçın, S., & Erdem, E. (2022). Performance analysis of burst traffic awareness based mobile sink routing technique for wireless sensor networks. *Gazi University Journal of Science*, *35*(2), 506-522.
7. Wang, A. P., Hahn, J., Roumi, M., & Chou, P. H. (2012). Buffer optimization and dispatching scheme for embedded systems with behavioral transparency. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, *17*(4), 1-26.
8. Kulkarni, S. S., & Wang, L. (2005, June). MNP: Multihop network reprogramming service for sensor networks. In *25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)* (pp. 7-16). IEEE.