EFFICIENT TASK ALLOCATION STRATEGIES FOR WSNs

C.Cara Evangeline
Department of CSE
NIT Andhra Pradesh, India
Email:caraevangeline10@gmail.com

C.Kushala
Department of CSE
NIT Andhra Pradesh, India
Email:kushi134@gmail.com

BKSP Kumar Raju Alluri
Department of CSE
NIT Andhra Pradesh, India
Email:pavan0712@gmail.com

***Abstract*—Tasks for wireless sensor networks have severe time and energy constraints. Performance of a sensor network like energy consumption and lifetime is largely affected based on the task allocated to the nodes in a network. This paper presents an efficient task allocation framework for sensor networks. It is generally known that multiple sensor nodes collaborating with each other to perform tasks are an important means of energy balancing. Therefore in this paper, we propose two task allocation strategies which when compared with current related works is much more realistic and efficient for complex task processing in terms of shortening execution time.**

***Keywords*—Task Allocation, Resource based sequence task allocation(RBSTA), Quantum based task allocation(QBTA), Energy, Queue length.**

I. INTRODUCTION

In recent years, WSNs have been widely used in many applications including military, industrial, household, medical, marine and other fields, especially in natural disaster monitoring, early warning, rescuing and other emergency situations. This paper presents higher real-time measures such as energy and queue length for task allocation.

Task allocation is essential to allocate the workload of each task to proper nodes in an efficient manner. Multiple measures including task execution time, energy consumption, queue length and resources required by tasks are considered as a whole to achieve the best overall performance.

A. Problem Definition

Many task allocation methods have been proposed for WSNs. Most of them only concentrate on reducing energy consumption of sensor nodes. However, queue length is not taken into consideration which is an important factor for allocation of tasks.

Instead of allocating a task based on energy alone, we take into consideration resources such as CPU cycles, memory (buffer size) and energy. Based on these resources, the proposed approach constructs an allocation table using which the task assignment is completed.

B. Our Approach

It is difficult for a single sensor node to independently accomplish a complex task. Therefore, assigning a complex task to appropriate sensor nodes has become a pressing issue in WSNs. In this paper, we resolve the problem of assigning complex tasks using the proposed algorithm QBTA (Quantum Based Task Allocation) which efficiently allocates the task to multiple nodes based on energy, queue length and time quantum. Figure 2 shows QBTA flowchart.

It is generally known that sensor nodes in WSNs are small with limited processing and computing resources. An algorithm RBSTA (Resource Based Sequence for Task Allocation) is also proposed to efficiently allocate a task to a node in WSN. The flowchart of RBSTA algorithm is shown in Figure 1.

Section II contains Related work and comparison with our work. Section III consists of our proposed algorithms (QBTA and RBSTA) along with the other comparison algorithms and graphs. Tables and the Tools used are discussed in Section IV and a summary of our paper is presented in Section V.

II. RELATED WORK

In recent years, numerous studies have been conducted for task allocation and scheduling. For example, Giannecchini et al[1] proposed an online task scheduling mechanism called CoRAl[1] to dynamically reconfigure a sensor network whenever a new intruder is detected. Given the amount of available resources and the task set, CoRAl determines the sampling frequency for all the tasks so that the frequencies of the tasks on each sensor are optimized subject to the previously evaluated upper-bound execution frequencies. CoRAl does not address mapping tasks to sensor nodes.

In, an energy-constrained task mapping and scheduling EcoMapS[3] is proposed. It incorporates channel modeling, concurrent task mapping, communication and computation scheduling, and sensor failure handling algorithm. First, a channel model is presented for singlehop WSNs[3]. Then, based on this channel model, communication and computation are jointly scheduled in the initialization phase. Furthermore, a quick recovery algorithm is executed in the quick recovery phase in case of sensor node failures. However, the guarantee of execution deadline is not provided by EcoMapS, hence it is not suitable for real-time applications. It has not considered queue length parameter which is a required real time factor.

Shehory[9], proposed an efficient distributed algorithms with low ratio bounds and with low computational complexities. These properties are proven theoretically and supported by simulations. They are based on both the algorithmic aspects of combinatorics and approximation algorithms for NP-hard problems. First, an approach to agent coalition was proposed where each agent must be a member of only one coalition. Next, this approach is extended for overlapping coalitions. In this paper, we propose approaches for the same with improved reality.

From the above-mentioned related works, we can find that existing task allocation strategies always takes only one influencing factor such as energy consumption or network load or network lifetime into account for task allocation. However, other contributing factors such as queue length, resources cannot be neglected. Therefore, a novel distributed and energy efficient task allocation strategy is proposed.

III. PROPOSED APPROACH

A. Allocation of single task based on Energy and Queue Length

Considering energy factor, we allocate a task to the node with highest energy. When only energy is considered, the task gets allocated to a set of nodes.

When both energy and queue length are considered, the task gets allocated to different or same set of nodes based on queue length. Figure 1 and 2 shows the difference.

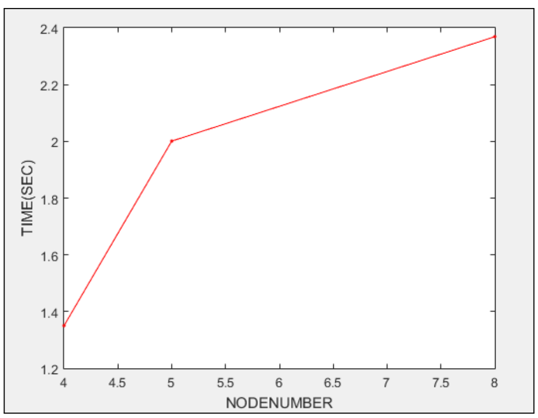

Fig. 1: Task allocation based on energy

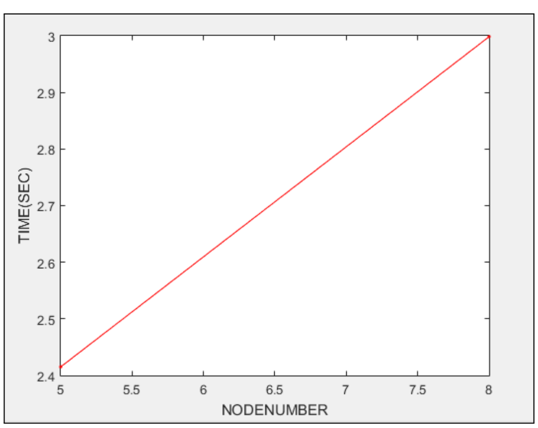

Fig. 2: Task allocation based on energy and Queue Length

The mapping of single task to multiple nodes is achieved through the following steps:

i. Until the task gets completed the workload is shared between nodes

ii. If the node with highest energy has full buffer size then the task is given to next highest energy node.

This process repeats until the assigned task gets completed.

B. Allocation of multiple tasks based on Energy and Queue length

Two different methodologies were proposed.

1) Allocation based on First come first served(FCFS): Initially, when all new tasks arrives, they are prioritized based on their execution time (ie.Shortest task is given highest priority). The task which has highest priority is given to highest energy node. Similarly, other tasks are mapped based on their priorities. Each node has n tasks in its queue. The tasks are performed in the order in which they are placed in the queue.

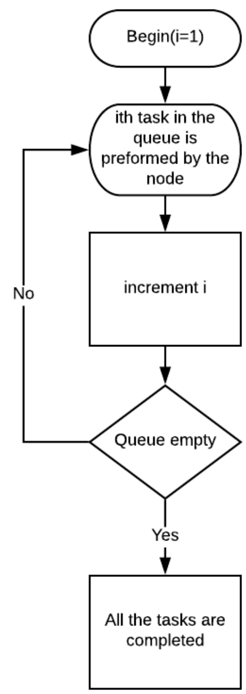

Fig. 3: FCFS Flow Chart

Figure 3 describes the order in which each task in a queue is performed by a node.

2) Allocation based on time Quantum(QBTA): Multiple tasks arrives and the tasks are prioritized based on their execution time. Now, highest priority task is performed by the highest energy node. Similarly, all the newly arriving tasks are allotted to each node based on task priority. Every node contains n tasks in its queue. Based on time quantum(q), a task is performed for q seconds and then added to the end of the queue. This is done till all tasks in the queue are completed. The procedure is followed for all newly arriving tasks.

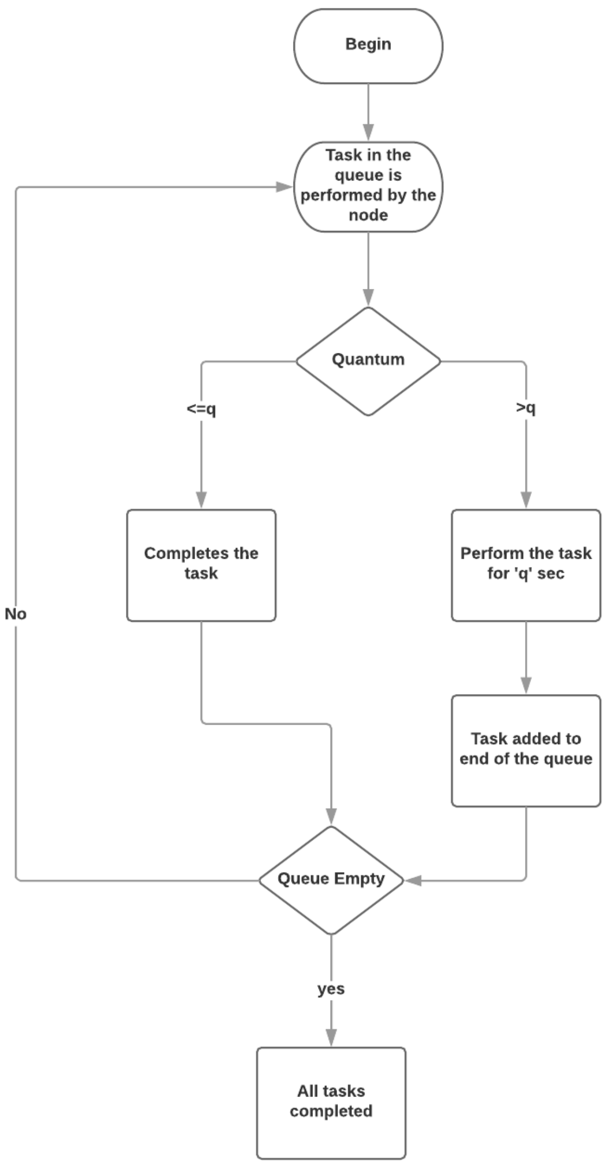

Fig. 4: QBTA flow chart

Figure 4 describes the order in which each task in a queue is performed by a node. QBTA procedure is described in Algorithm 1. In FCFS Algorithm, newly arriving task will be placed at the end of a queue which increases task completion time whereas in QBTA algorithm every task in a queue is performed for q seconds and placed at the end of the queue which decreases the task completion time. The difference in the efficiency of FCFS and QBTA is shown in Figure 5.

Algorithm 1: QBTA(Quantum Based Task Allocation)

Data: Energy, Queue length of each node and all the t tasks burst time bt
Result: Time for completion of all the tasks

i=1 ;
while $i<=t$ **do**
  Find the ith maximum energy node and allocate the task which requires ith least time ;
  // Every node has previously allocated tasks and some nodes have newly arrived tasks
end
i=1 ;
q=2 //Time Quantum ;
while $i<=t$ **do**
  Select the node which got the newly allocated task
  while *Until completion of all tasks in the Queue* **do**
    if *task has* $bt<=q$ **then**
      task will leave after completion ;
      node will perform next task in its queue ;
    else
      task is performed by the node for q seconds and then placed at the end of its queue ;
      Now,node performs the next task in the queue ;
    end
  end
  $ct[i] \leftarrow completion\_time$;
end
return ct[] ;

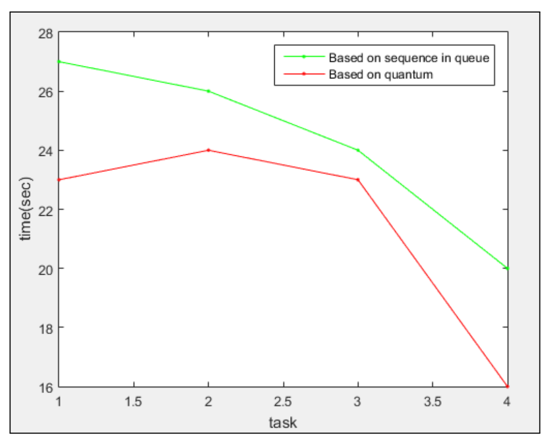

Fig. 5: Multiple task allocation

C. Allocation of task based on sequence table

In the previous proposed approach, tasks are allotted to a node based on energy and queue length. In RBSTA algorithm, before giving a newly arriving task to a node, the completion time of the tasks when given to all nodes individually is calculated and the node which performs the task in less time is the efficient node to perform the arrived task. Considering multiple measures (ie. resources such as CPU cycles, energy and queue length) here a sequence table is constructed based on the sequence obtained by RBSTA algorithm. Through the sequence, completion time of newly arriving task for each node is known. Similar to routing algorithm where packet is transferred to the next node which is at shortest distance from the source node, in RBSTA the task is given to the efficient node which performs the newly arrived task in less time.

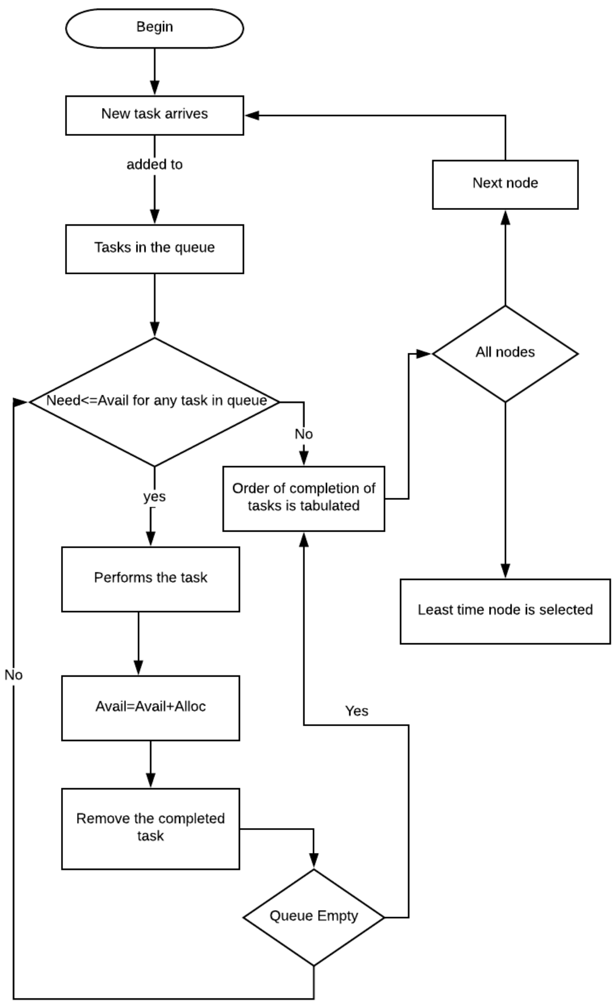

Fig. 6: RBSTA flow chart

Figure 6 describes the flow of RBSTA Algorithm. RBSTA procedure is described in Algorithm 2.

IV. EXPERIMENTAL RESULTS

The proposed algorithms were tested in simulation in a Matlab framework. MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment.

Table 1 shows the dependence of allocation of task on Queue length which the existing works failed to consider. Efficient way of allocating a task based on time quantum is

Algorithm 2: RBSTA(Resource Based Sequence for Task Allocation)

Data: Resources required for newly arriving task(max)
Result: Efficient node for performing a given task

i=1 ;
j=1 ;
while $i <= n$ **do**
  $max[][] \leftarrow maximum\_resources\_of\_tasks\_in\_queue()$;
  $alloc[][] \leftarrow allocated\_resources\_of\_tasks\_in\_queue()$;
  //Add the maximum resources of the newly arriving tasks to max[][] and append [0,0,0] to alloc[][] ;
  $avail[] \leftarrow free_resources_of_ith_node$;
  need[][]=max[][]-alloc[][] ;
  while *Until all tasks in the queue of ith node gets checked* **do**
    if *need[][]<= avail[]* **then**
      Complete the task ;
      avail[]=avail[]+alloc[][] ;
    else
      check for the next task ;
    end
  end
  $sequence \leftarrow order\_of\_tasks\_completion$;
end
Based on the sequence table, time for newly arriving task for each node is calculated ;
return min_ time_ node() ;

shown in Table 2. Table 3 is the sequence table from which we can get the efficient node to perform a task.

TABLE I: Single task shared between nodes

Nodenumber	Energy	Queue length	E	E and QL
1	0.4709	7	–	–
2	0.1948	4	–	–
3	0.2277	8	–	–
4	0.9234	15	yes	–
5	0.9049	6	yes	yes
6	0.1111	11	–	–
7	0.5949	10	–	–
8	0.7112	14	yes	yes
9	0.2967	13	–	–
10	0.5079	9	–	–

From table 1, it is known that when a node with highest energy has full buffer size, then the task is allocated to other nodes.

TABLE II: Comparision between FCFS and QBTA

Task number	Burst Time	FCFS CT	QBTA CT
1	2.9983	27	23
2	3.9999	26	24
3	5.0169	24	23
4	6.0162	20	16

From table 2, QBTA is giving a more efficient result which when compared to FCFS algorithm.

TABLE III: RBSTA Sequence Table

Node	QL	s1	s2	s3	s4	s5	s6	Time
1	6	T4	T2	T1	T3	T5	T6	15
2	6	T1	T2	T3	T4	T5	T6	11
3	2	–	–	–	–	–	–	–
4	6	T2	T3	T4	T1	T5	T6	16
5	5	T1	T2	T4	T3	T5	–	9

Similar to Routing table, a sequence table is constructed based on which task can be allocated to a node which completes it in less time. From Table 3 we can infer that task is allocated to node 5 since it performs the task in less time while compared to other nodes.

Hence, Table 1 shows the dependence of task allocation on Queue Length, from Table 2 we can infer that Quantum based task allocation is efficient and Table 3 gives the efficient node to which a task can be allocated.

V. Conclusion

In order to reduce average energy consumption of sensor nodes, shorten task execution time, this paper studied a complex task allocation problem and proposed two efficient task allocation strategies. Task allocation is implemented based on parameters like energy, queue length and the resources available. The nodes are efficiently made to complete complex tasks according to their energy and resources available. Simulation results and the tables show that the proposed algorithm is much more efficient than related works. Also, it is quite suitable for complex task processing.

References

[1] Giannecchini, Simone, Marco Caccamo, and Chi-Sheng Shih. ”Collaborative resource allocation in wireless sensor networks.” Real-Time Systems, 2004. ECRTS 2004. Proceedings. 16th Euromicro Conference on. IEEE, 2004.

[2] Gerkey, Brian P., and Maja J. Matari. ”A formal analysis and taxonomy of task allocation in multi-robot systems.” The International Journal of Robotics Research 23.9 (2004): 939-954.

[3] Tian, Yuan, Eylem Ekici, and Fusun Ozguner. ”Energy-constrained task mapping and scheduling in wireless sensor networks.” Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on. IEEE, 2005.

[4] Younis, Mohamed, Kemal Akkaya, and Anugeetha Kunjithapatham. ”Optimization of task allocation in a cluster-based sensor network.” Computers and Communication, 2003.(ISCC 2003). Proceedings. Eighth IEEE International Symposium on. IEEE, 2003.

[5] Botelho, Sylvia C., and Rachid Alami. ”M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement.” Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on. Vol. 2. IEEE, 1999.

[6] Edalat, Neda, Chen-Khong Tham, and Wendong Xiao. ”An auction-based strategy for distributed task allocation in wireless sensor networks.” Computer Communications 35.8 (2012): 916-928.

[7] Deng, Yueyue, et al. ”Task allocation and path planning for collaborative AUVs operating through an underwater acoustic network.” OCEANS 2010. IEEE, 2010.

[8] Ferri, Gabriele, et al. ”A market-based task allocation framework for autonomous underwater surveillance networks.” OCEANS 2017-Aberdeen. IEEE, 2017.

[9] Shehory, Onn, and Sarit Kraus. ”Methods for task allocation via agent coalition formation.” Artificial intelligence 101.1-2 (1998): 165-200.

[10] Kang, Qin-Ma, et al. ”Task allocation for maximizing reliability of distributed computing systems using honeybee mating optimization.” Journal of Systems and Software 83.11 (2010): 2165-2174.

[11] Yu, Yang, and Viktor K. Prasanna. ”Energy-balanced task allocation for collaborative processing in wireless sensor networks.” Mobile Networks and Applications 10, no. 1-2 (2005): 115-131.