

An Enhanced Approach on Task Allocation in Wireless Sensor Actor Networks

Mona Sharifnejad, Sareh Beheshti, Mansoureh Ghiasabadi Farahani and Mohsen Sharifi

Abstract – Tasks allocation refers to a group of tasks assigned to actor nodes in wireless sensor actor networks (WSANs). In these networks, sensor nodes gather information about environment while actor nodes make appropriate actions on the environment based on sensory information they receive from sensors. There are many task allocation algorithms in WSANs; one of them is Max-min algorithm. In this paper, a unique modification of Max-min algorithm is proposed. Improved Max-min is based on the expected execution time instead of complete time as a selection basis. Experimental results showed better total makespan, total completion time smaller than original Max-min, concurrency execution of tasks using available actors and load balance in small WSANs.

Keywords – Wireless Sensor Actor Networks, Task Allocation, Max-Min Algorithm, Makespan, Energy-Aware, Time-Aware.

I. INTRODUCTION

Technology advances in wireless communication in recent years are making complex and diverse applications of sensor network possible. Wireless Sensor Actor Networks (WSANs) can be applied to from monitoring application such as fire detection, traffic monitoring and wildlife habitat monitoring, to computation-based application like target detection and tracking [11].

Sensor nodes gather information about environment while actor nodes make appropriate actions on the environment based on sensory information they receive from sensors [3].

There have been many algorithms used to schedule tasks on their actors. Some of well-known examples of such algorithms intended to be applied in WSANs are Max-min, Min-min, OLB, etc. [11].

OLB schedules the tasks in arbitrary order without considering the execution times or the completion times of tasks [2]. This simple approach can result in schedules with high makespan [4], making OLB inappropriate for WSANs that are critically constrained by time. In contrast, min-min and max-min consider the execution times of tasks when assigning tasks to actors [9].

One of the features of the Max-min strategy is that chooses large tasks to be executed firstly, which in turn small task delays for long time. On the other hand, Min-min is perfect in executing smaller tasks then large ones that is the reverse of Max-min [11].

The Max-min algorithm selects the task with the maximum completion time and assigns it to the resource on which achieve minimum execution time. It is clear the Max-min seems better choice whenever the number of small tasks is much more than large ones. But in other cases, early executing large tasks leads for increasing in total completion time of submitted tasks so Min-min is

better choice and visa-verse [2]-[11].

The reminder of this paper is organized as follows. Section II presents Related Works. Section III describes Task Allocation Algorithm. Section IV presents Proposed Approach. Section V evaluates proposed approach. Section VI concludes the paper.

II. RELATED WORKS

There are many parameters can be mentioned as factor of scheduling problem to be considered such as load balancing, system makespan, , energies of actors, and so on. Task scheduling algorithm is a decision making process about assigning and finding the best match between tasks and actors with the purpose of reducing response time and energy consumption.

Scheduling algorithms can be categorized according to many polices as immediate and batch scheduling, preemptive and non-preemptive scheduling, static and dynamic scheduling, etc.

Five immediate mode heuristics are minimum completion time, minimum execution time, switching algorithm, k-percent best, and opportunistic load balancing. Batch mode heuristics include Min-min heuristic, Max-min heuristic, and Suffrage heuristic [5].

H. Park et al. [11] have presented an energy-efficient task assignment and distributed task migration algorithm for sensor networks. The node that has low energy level has responsibility to migrate to healthier node before it dies. They calculate cost of transferring tasks to those nodes by requesting cost value to incoming and outgoing tasks of the originator task.

In [7]-[8]-[9] is presented a two level task allocation mechanism. It first breaks end-to end periodic tasks into real-time jobs, and then uses appropriate algorithms for sensing tasks and acting tasks. It proposes a model for WSANs using graph transformation systems. Using this formalism it analyzes the correctness of algorithms. This algorithms guarantee that the tasks complete their activities before their deadlines expire.

In [10] is taken both energy awareness and reduction of actor tasks' times to completion in WSANs into account and is proposed a two-phase task allocation technique based on Queuing theory. The main goal in both proposed algorithms is to reduce task completion time, while load balancing is also considered in the second algorithm.

H. Momeni.[6] have proposed a distributed effective real time auction protocol. Sensors send their readings of events to their nearest actors using the location service. The task assignment problem is for each actor receiving sensory information to find a cost effective set of actors to perform a set of tasks in response to the event. In this protocol is assumed that the frequency of event

occurrences is not high and it initiates an auction whenever it wants to decide to which actors to assign a new task.

M. Sharif et al. [9] have presented a new time- and energy-aware starvation-free algorithm called Scate for assigning tasks to actors while satisfying the scalability and distribution requirements of WSNs with semi-automated architecture. The proposed algorithm allows concurrent executions of any mix of small and large tasks and yet prevents probable starvation of tasks.

III. TASK ALLOCATION ALGORITHMS

Many task allocation algorithms for distributed systems have been presented so far with the aim of minimizing the total completion time of tasks. So there are some parameters that try to reduce the total task completion time and energy consuming of the system, such as deadlines of tasks, distances of actors to the event location, remaining energies of actors, and a balanced load on all actors.

In WSN is assumed m actors A_j ($j = 1, \dots, m$) that should perform n tasks T_i ($i = 1, \dots, n$). Tasks are independent, non-preemptive, and not decomposable. Actors can search the whole network without any restriction on routing hops. Actors are idle at first. The total time taken by the actor A_j with no load at the time of assignment to execute a task T_i is called the execution time of task T_i on actor A_j and it is denoted by E_{ij} . The time taken by actor A_j to finish the execution of task T_i is called the completion time and it is denoted by C_{ij} . The completion time of a task T_i on an actor A_j is greater or at best equal to its execution time on the same actor, i.e., $C_{ij} \geq E_{ij}$. The expected time that actor A_j can perform a task after finishing the execution of all its previously assigned tasks is called the availability time of actor A_j and it is denoted by a_j . In fact, C_{ij} denotes the sum of the availability time (a_j) of actor A_j , and the execution time (E_{ij}) of task T_i , on A_j i.e., $C_{ij} = (E_{ij} + a_j)$ [9].

Makespan is defined as a measure of the throughput of the heterogeneous computing system and does not measure the quality of service imparted to an individual task [5].

According to the assignment policy of min-min that is based on allocating small tasks to fast actors. It estimates the execution and completion times of all unscheduled tasks on each actor first. It then repeatedly selects a task with the lowest completion time and assigns it to an actor with the least execution time until all tasks are assigned [9].

Although max-min and min-min have similar structures, they differ in their selections of actors to assign them tasks. In max-min, once an actor that can provide the earliest completion time for a task is determined, the task T_i with the maximum earliest completion time is chosen and mapped to the corresponding actor.

In the Max-min algorithm, shown in Fig 1, a_j represents the ready time of A_j to execute a task, while C_{ij} and E_{ij} represent the expected completion time and Execution time respectively. As shown, task T_k with maximum expected completion time is chosen to be assigned for

corresponding A_j that gives minimum execution time [5].

1. for all submitted tasks in meta-task; T_i
2. for all actors; A_j
3. $C_{ij} = E_{ij} + a_j$
4. While meta-task is not empty
5. Find task T_k consumes **maximum completion time**.
6. Assign T_k to the actors A_j which gives **minimum execution time**.
7. Remove T_k from meta-tasks set
8. Update a_j for selected A_j

Fig.1. The Max-Min Algorithm [5]

IV. PROPOSED APPROACH

Large tasks have highest priority rather than smaller tasks but Min-min algorithm is based on allocating small tasks. For example, if there is only one long task, Max-min will execute many short tasks concurrently with the long task. The resulting makespan might just be determined by the execution time of the long task in this case [5].

Generally, load balancing is a desirable property in distributed systems especially in small-scale distributed systems requiring short makespan. However, load balancing can lead to large makespan in large-scale distributed systems.[8] original Max-min algorithm may lose some of its major advantages as load balance between available actors in small-scale WSNs and small total completion time for all submitted tasks in large-scale ones because of meta-tasks can contain tasks have relatively different completion time and execution time, the makespan is not determined by one of submitted tasks. It would be similar to the Min-min makespan. So the Max-min algorithm is not suitable for this case.

If we can assign large tasks to be executed by slower actors then waiting time of short tasks minimize, because small tasks execute concurrently on fastest actor to finish large number of tasks during complete execute at least one large task on slower actor.

In order to reach above idea, we proposed a new approach to improve Max-min algorithm efficiency. Proposed approach allows executing tasks on actors concurrently.

Our proposed approach for each task in Metatask includes following phases:

1. Expected execution time and completion time of each task on actors is computed.
2. Task with max execution time is selected and is assigned to be executed by actor with minimum completion time.
3. Ready time of actor is updated and expected complete time is updated.

Enhanced Max-min algorithm improved load balance of available actors and provide concurrent execution of submitted tasks with higher probability rather than original Max-min algorithm.

V. EVALUATION

To show the efficiency of our approach, It is compared with max-min algorithm in terms of total completion time of tasks and load balance of actors. we assume a two-dimensional square field, 15 m×15 m, containing 100 sensor nodes with a 1 m transmission range and 10 actor nodes. The tasks to be executed by actors are independent.

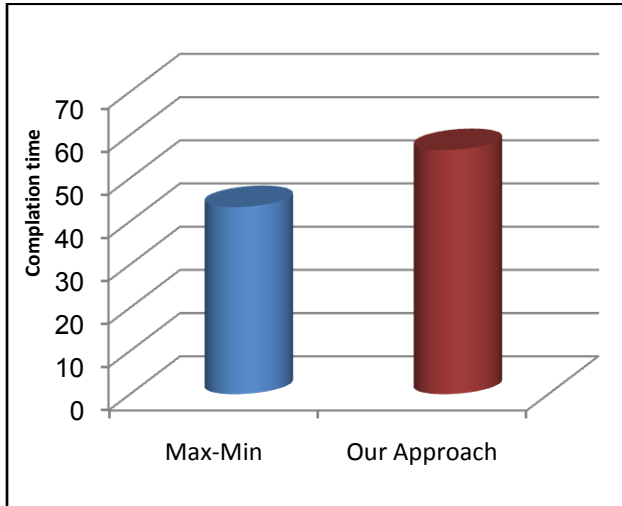


Fig2: Comparison of completion time

Figure 2 compares the completion time of the algorithms. Completion time of proposed approach has improved rather than original Max-min algorithm.

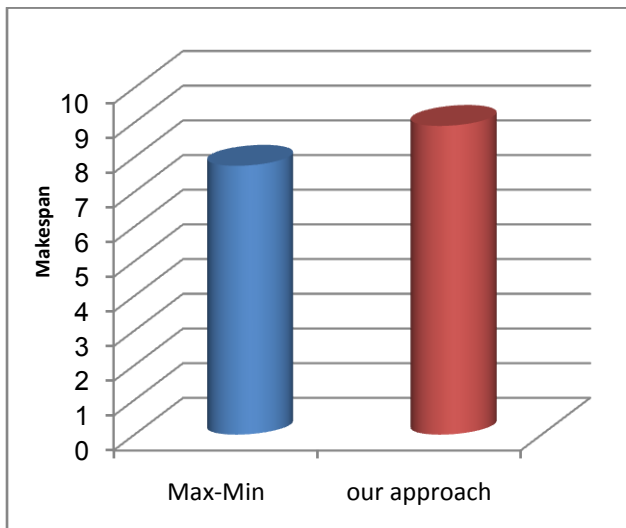


Fig.3. Comparison of makespan

Figure 3 shows the makespan of the algorithms. Our approach results shorter makespan compared with original Max-min algorithm and lead to load balance.

our evaluation shows that the total completion time in proposed approach is same or smaller than original Max-min. Also, Its scheduling presents concurrency execution of tasks using available actors and load balance in WSANs.

VI. CONCLUSION

In this paper, we proposed improved Max-min algorithm in wireless sensor actor networks. It is based on the expected execution time instead of complete time as a selection basis. It allows concurrent executions of any mix of small and large tasks and yet prevents probable starvation of tasks, but Max-min algorithm can't execute tasks concurrently, makespan become large.

Our experimental results showed better total makespan, total completion time smaller than original Max-min, concurrency execution of tasks using available actors and load balance in small WSANs.

REFERENCES

- [1] I. F. Akyildiz and I. H. Kasimoglu, "Wireless Sensor and Actor Networks: Research Challenges", *Ad Hoc Networks*, vol. 2, 2004, pp. 351-367
- [2] R. Armstrong, D. Hensgen, and T. Kidd, "The Relative Performance of Various Mapping Algorithms is Independent of Sizable Variances in Run-Time Predictions", *Proc. IEEE Int Workshop Heterogeneous Comput.*, 1998, pp. 79-87.
- [3] S. Beheshti, M.Ghiasabadi, M. Sharifnejad, M. Sharifi, "Survey on Task Allocation Algorithms in Wireless Sensor and Actor Networks", *International Journal of Electronics Communication and Computer Engineering* Volume 5, Issue 6, ISSN (Online): 2249-071X, ISSN (Print): 2278-4209, Nov. 2014.
- [4] T.D. Braun et al., "A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems", *Parallel Distrib. Comput.*, vol.61, 2001, pp. 810-837.
- [5] M. Maheswaran, S. Ali, H. J. Siegel, D. Hensgen and R. F. Freund, "Dynamic Mapping of a Class of Independent Tasks onto Heterogeneous Computing Systems", *Journal of Parallel and Distributed Computing* 59(1999), 107-121.
- [6] H. Momeni, M. Sharifi and M. Okhovvat, "Distributed Assignment of Real-Time Tasks in Wireless Sensor Actor Networks", *IEICE Electronics Express Journal*, ISSN: 1349-2543, Vol. 8, No. 7, pp. 429-435, April 2011.
- [7] H. Momeni, M. Sharifi and S. Sedighian, "A New Approach to Task Allocation in Wireless Sensor Actor Networks", *First International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)*, Indore, India (2009).
- [8] H. Momeni, V. Rafe, M. Sharifi and A. T. Rahmani, "A Graph Transformation-Based Approach to Task Allocation in Wireless Sensor Actor Networks", *21th IEEE International Conference on Software Engineering and Knowledge Engineering (SEKE09)*, USA(2009).12.
- [9] M. Sharifi and M. Okhovvat, "Scate: A Scalable Time and Energy Aware Actor Task Allocation Algorithm in Wireless Sensor and Actor Networks", *ISSN (Print): 1226-6463*, Vol. 34, No. 3, June 2012.
- [10] M. Okhovvat, M. Sharifi, and H. Momeni, "Task Allocation to Actors in Wireless Sensor Actor Networks: An Energy and Time Aware Technique", *Procedia Computer Science*, vol. 3, 2011, pp.484-490.
- [11] H. Park and M. B. Srivastava, "Energy-Efficient Task Assignment Framework for Wireless Sensor Networks", *Center for Embedded Networking Sensing (CENS)*, Technical Report (2003).
- [12] Sharifi, M., Momeni, H., Rafe, V. (2010). "Optimal real-time task allocation in wireless sensor actor networks", *Australian Journal of Basic and Applied Sciences*, Vol. 4, No. 8, pp. 3455-3466.



AUTHOR'S PROFILE

Mona Sharifnejad

Member of Faculty Department of Computer Engineering Arak Branch,
Islamic Azad University, Arak, Iran.
Email: m-sharifnejad@ iau-arak.ac.ir

Sareh Beheshti

Member of Faculty Department of Computer Engineering Arak Branch,
Islamic Azad University, Arak, Iran.
Email: s-beheshti @ iau-arak.ac.ir

Mansoureh Ghiasabadi Frahani

Member of Faculty Department of Computer Engineering Arak Branch,
Islamic Azad University, Arak, Iran.
Email: m-ghias @ iau-arak.ac.ir

Mohsen Sharifi

Member of Faculty Iran University of Science and Technology.
Email: msharifi@iust.ac.ir