Journal of Artificial Intelligence and Data Mining (JAIDM), Vol. 10, No. 3, 2022, 303-310.

Shahrood University of Technology

Journal of Artificial Intelligence and Data Mining (JAIDM)
Journal homepage: http://jad.shahroodut.ac.ir

Research paper

Energy-Efficient Timing Assignment of Tasks to Actors in WSANs

Mohammad Reza Okhovvat[1], Mohammad Taghi Kheirabadi[1*], Ali Nodehi[1] and Morteza Okhovvat[2]

1. Department of Computer Engineering, Gorgan Branch, Islamic Azad University, Gorgan, Iran.
2. Health Management and Social Development Research Center, Golestan University of Medical Sciences, Gorgan, Iran.

Article Info

Article History:
Received 06 April 2022
Revised 28 May 2022
Accepted 08 June 2022

DOI: 10.22044/jadm.2022.11783.2325

Keywords:
Energy Consumption, Make-span, Task Assignment, Wireless Sensor Actor Networks.

**Corresponding author: m.kheirabadi@gorganiau.ac.ir (M.T. Kheirabadi).*

Abstract

Minimizing make-span and maximizing remaining energy is usually of chief importance in the applications of wireless sensor actor networks (WSANs). The current task assignment approaches are typically concerned with one of the timing or energy constraints. These approaches do not consider the types and various features of tasks may need to perform, and thus may not be applicable to some types of real applications such as search and rescue missions. To this end, an optimized and type aware task assignment approach called type aware task assignment (TATA) is proposed that considers the energy consumption as well as the make-span. TATA is an optimized task assignment approach and aware of the distribution necessities of WSANs with a hybrid architecture. TATA comprises two protocols, namely a Make-span Calculation (MaSC) protocol and an Energy Consumption Calculation (ECal) Protocol. Through considering both time and energy, TATA makes a trade-off between minimizing make-span and maximizing the residual energies of actors. A series of extensive simulation results on the typical scenarios show a shorter make-span and larger remaining energy in comparison to when one of the three related approaches, namely, stochastic task assignment (STA), opportunistic load balancing (OLB), and task assignment algorithm based on the quasi-Newton interior point (TA-QNIP) is applied.

1. Introduction

A collection of sensor nodes and actor nodes that communicate wirelessly forms a special type of network called Wireless Sensor Actor Networks (WSANs) [1, 2], wherein the sensors collect the environmental data, and the actors act in response to the sensory data. The principal parts of WSANs can be set differently based on the demands and desideratum of applications and the available technologies. This paper studies the hybrid architecture of WSANs [2, 3], wherein the sensors convey the sensing information to the actors. The actors investigate information, and possibly will refer to the sink(s) before doing any action. This means that the actors may take decisions and do action without interfering the sink or might notify the sink and postpone for which are created and assigned by the sink. Hence, we deal with two type of tasks, local tasks and global tasks. “Local tasks” are the routine and simple tasks that are usually determined without interfering the sink. The sensed data of events are collected and processed by the sensors and then the related tasks (local tasks) are defined and dispatched to the appropriate actors.

Global tasks are the more complex tasks defined based on the gotten sensory data by the sink, and then they are assigned by the sink to the proper actors to be done. The sink receives the sensory information of the global tasks, and then it determines and assigns the global tasks to proper actor(s). Hence, finding the best possible task assignment to run on the available actors is an interesting influencing make-span and remaining energies of actors.

This paper considers WSANs with hybrid architecture (Figure 1), wherein the sensory data is passed on to the sink to define the required actions (global tasks) to be carried out by the actors or sent to actors directly to decide and carry out appropriate tasks (local tasks). WSANs are typically used in critical applications in which the actors must react quickly; in that delays may result in a disaster [4, 5]. In addition, current restrictions such as energy constraints and dynamic features of environments have made this problem very challenging [6, 7]. Therefore, various task allocation approaches for ubiquitous systems have been presented so far in order to reduce the network make-span [8-11] but these approaches have usually neglected the energy consumption in the network.

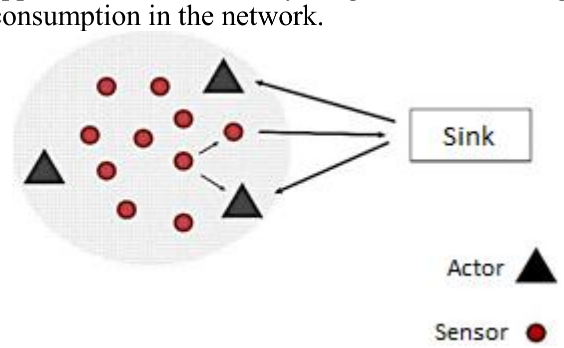

Figure 1. Typical WSAN with hybrid architecture.

This paper presents an energy-efficient timing task assignment approach for WSANs. We have named our approach as TATA, which stands for type aware task assignment. In TATA, the local tasks are determined and assigned to actors without involving the sink but the sink determines the global tasks, and then it assigns them to appropriate actors. TATA achieves its superiority by considering both make-span and residual energies of actors in choosing actors to perform tasks. In Section 5, the applicability of TATA to small- and large-scale networks through extensive experiments is shown.

The rest of the paper is organized as what follows. Section 2 discusses the outstanding relevant works. Section 3 presents our assumptions. Section 4 presents our proposed approach, TATA. In section 5 reports the simulation results, and Section 6 concludes the paper.

2. Related works

Considering the WSANs' restrictions such as energy limitations, capability constraints, and dynamicity, the general-purpose task assignment approaches are typically inapplicable to WSANs. There are, however, various task assignment approaches for WSANs with various purposes such as reducing delays, enhancing remaining energy, and reducing response time of the network.

Byun and So [12] have proposed a scheduling approach for WSANs that tries to meet the delay requirements while increase the average remaining energy. Their work is based on an epidemic-inspired algorithm for data dissemination. They predict behavior of the system based on converge time through mathematical analysis. It is asserted the approach, extends the lifetime of network, and decreases the overall energy consumption of nodes in WSANs but make-span is not considered by their approach.

Okhovvat *et al*. [13] have proposed an analytical task assignment approach to reduce tasks completion time in WSANs. In this work, the appropriate dispatching rates of tasks are calculated. They also presented a formal model based on generalized stochastic Petri net (GSPN). According to the reported results, the total completion time of tasks is minimized but energy consumption is neglected by their approach.

Kong *et al*. [14] have presented a task assignment strategy to increase balance of workloads on the resources and to minimize the task execution time in multi-robot networks. Their strategy includes two steps. In the first step, finding the proper combination of robots and tasks is done based on the particle swarm optimization (PSO) algorithm. In the second step, the execution order of tasks is sorted using a greedy algorithm, and then the overall cost of tasks execution is calculated. This procedure is repeated until the optimal task assignment solution is found. Although the proposed approach considered execution time of tasks, it overlooked energy consumption of resources.

Huang *et al*. [15] have presented a task assignment mechanism for heterogeneous multi-robot systems based on the auction theory. They categorized the capabilities of robots using distributed auction algorithm. In this algorithm, both tasks and robots are modeled, and considering the features of tasks, the distance between robots and tasks, and capabilities of robots tasks are mapped to the robots. However, the proposed mechanism focuses on the possibility of performing tasks by heterogeneous robots but it considers neither energy nor time, explicitly.

Wang *et al*. [16] have proposed an algorithm called link quality matrix (LQM) for real-time resource retrieval in adhoc networks. This algorithm is based on the auction theory, and considers real-time requirements of multi-robot (multi-actor) systems. It tries to decrease global communication and redundant computation but energy consumption of nodes is ignored.

M. Younis *et al.* [17] have proposed a task assignment technique in wireless sensor networks (WSNs). They have simplified the scheduling of tasks on CHs by taking into account the computation time of collected data lags in cycle(s). Their algorithm tries to minimize the network lifetime but the implementation time of tasks in assigning tasks is ignored, and thus the make-span would be increased.

Nevertheless, little study has been done on the optimal task assignment in WSANs with multiple objectives such as reducing make-span and enhancing remaining energies of actors. Hence, this paper presents an energy-efficient and time-aware task assignment approach for WSANs that addresses the aforementioned issues.

3. Assumptions and System Model

In this section, the assumptions and the system model are described. In order to present the proposed task assignment approach, firstly our assumptions and then models are presented.

3.1. Assumptions

We assumed a typical WSAN with a hybrid architecture including a sink, sensors, and actors wherein n tasks T_i *(i=1,…, n)* should be done by m actors A_j *(j=1,…, m)*. The sensors are collecting data from the surroundings, and define local tasks or transmitting them to the sink to define global tasks. A local task assigns to an actor directly based on an assignment approach but if the sensory information received by the sink, the sink node determines the proper global tasks and then assigns tasks to appropriate actors. The actors are idle at first, and they can search the entire network without any restriction on routing hops. It is assumed that an actor can run only one task at once, and the entire network is monitored by the sink.

We have constrained the objectives of our task assignment approach to reduce both make-span and energy consumption. In order to achieve this goal, the information of the capability of actors such as its speed and its current task load are considered. The tasks are independent, and their generations follows a Poisson distribution.

3.2. Actor model

There are a set of m actors A_j *(j=1,…, m)* that perform their assigned tasks. We used the *M/M/1* queuing system [18, 19] to model each actor. The arrival rates of global tasks and local tasks at the actor A_i are λ_i and λ'_i, respectively, but the tasks are done with μ_i rate.

We consider an assignment function $T_i \rightarrow A_j$. The related assignment vector noted by $X_{i,j}$, where:

$$X_{i,j} = \begin{cases} 1 & \text{if } T_i \text{ is assigned to } A_j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

3.3. Energy model

An energy model is proposed to calculate the energy consumption of an actor A_j as follows:

$$E_j^{Consume} \leftarrow \alpha_j \times Time_j \quad (2)$$

In (2), $E_j^{Consume}$ denotes the required energy to carry out all tasks assigned to the actor A_j. The average rate of energy consumption by the actor A_j per unit of time is shown by α_j, and the period of time that the actor A_j passes to run its assigned tasks is shown by $Time_j$. Thus the remaining energy of the actor A_j, E_j^{Rem}, can be calculated by (3), wherein EA_j denotes the current energy of the actor A_j.

$$E_j^{Rem} \leftarrow EA_j - E_j^{Consume} \quad (3)$$

3.4. Network model

A typical WSAN with a hybrid architecture containing a sink, actors, and sensors is considered. The sensors gather the environmental information, and the actors are responsible to execute the tasks. The actors and sensors are spread uniformly, and the number of sensor nodes is bigger than the number of actor nodes. We used the queuing theory to model and analyse the task assignment problem. Figure 2 shows the queuing model of such a network.

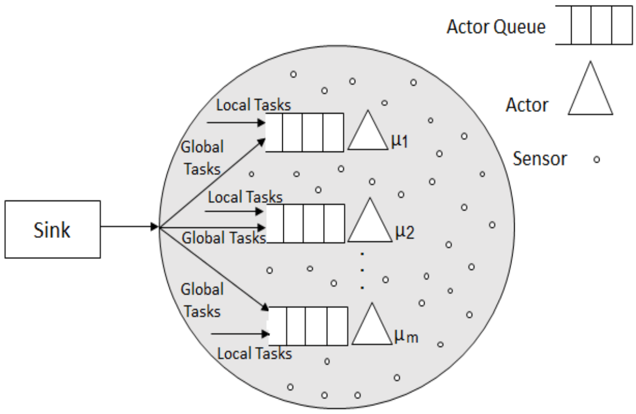

Figure 2. Model of WSAN based on queueing theory.

The network make-span is defined as the finish time of entire tasks in the network. The make-span can be calculated by (4), wherein $Time_j$

denotes the expected finish time of tasks in an actor A_j, and m is the number of actors. Hereafter, the words make-span and Make-span$_{\text{assignment}}$ are used interchangeably.

$$Makespan = \max\{Time_j\}; 1 \le j \le m \tag{4}$$

4. Proposed Approach

According to the policy of TATA in assigning tasks to actors that is optimization of the make-span and energy consumption, a fitness function is defined to find the best assignment rates of tasks to the actors. The pseudo-code of TATA is as follows:

Pseudo-code of TATA

Input: Information of each available actor A_j(*e.g.* μ_j, E_j, α_j), importance of time and energy in the application (W_1, W_2), and the entrance rates of tasks to the sink (λ_T)

Output: Assignment of tasks to actors

1. MaSC ( );
2. ECal ( );
3. For j=1 to m do
4. {Compute the global arrival rates λ_j resulted from:

$$Min\left[\begin{array}{l} W_1 \cdot \left(\dfrac{Makespan_{allocation} - Makespan_{\min}}{Makespan_{\max} - Makespan_{\min}}\right) \\ + \; W_2 \cdot \left(\dfrac{E^{Consume}_{allocation} - E^{Consume}_{min}}{E^{Consume}_{max} - E^{Consume}_{min}}\right)\end{array}\right];$$

//determining arrival rates that minimize the FitnessFunction

5. Assign global and local tasks to the related A_j with the rates of λ_j, λ'_j, respectively;}}

Equation (5) shows the fitness function. TATA tries to determine the best dispatching rate of tasks to actors in such a way that lead to the lowest value of the fitness function.

$$FitnessFunction = W_1 \cdot \left(\frac{Makespan_{allocation} - Makespan_{\min}}{Makespan_{\max} - Makespan_{\min}}\right) + W_2 \cdot \left(\frac{E^{Consume}_{allocation} - E^{Consume}_{min}}{E^{Consume}_{max} - E^{Consume}_{min}}\right) \tag{5}$$

Here, Make-span$_{\min}$ and Make-span$_{\max}$ are the minimum make-span and maximum make-span, respectively, while $E^{Consume}_{Min}$ and $E^{Consume}_{Max}$ are the minimum and maximum values of total energy consumption of actors, respectively. The fitness function has two parts. The first part computes the make-span, while the second part calculates the energy consumption of the actors. To be valid aggregating the first part to the second part, each part should be normalized. W_1 and W_2 are the fitness values, and are set based on the trade-off requirement of the application. TATA uses two protocols called MaSC and ECal in order to determine the first half and the second half of the Fitness Function, respectively, and tries to determine the assignment rates of tasks to the actors that minimize the Fitness Function. These methods are presented in Sections 4.1 and 4.2.

4.1. Make-span calculation protocol

In order to calculate the make-span of the network, a protocol called MaSC is proposed. As mentioned in Section 3.2, the *M/M/1* queuing system is used to model an actor. The tasks are arrived to the actor A_j with ($\boldsymbol{\lambda'_j} + \boldsymbol{\lambda_j}$) rate, and they are run with the rate of μ_i. Total rate of global tasks (λ_T) and total rate of local tasks ($\boldsymbol{\lambda'_T}$) can be computed by (6), in which m denotes the total number of actors.

$$\lambda_T = \sum_{j=1}^{m} \lambda_j, \quad \lambda'_T = \sum_{j=1}^{m} \lambda'_j \tag{6}$$

In order to have a steady state analysis of the CTMC, we write the flow equations as shown by (7), wherein P_i indicates the steady state probability of existing tasks in state *i*.

$$\begin{aligned} &(\lambda_j + \lambda'_j)P_0 + \mu_j P_2 = \mu_j P_1 + (\lambda_j + \lambda'_j)P_1 \\ &(\lambda_j + \lambda'_j)P_1 + \mu_j P_3 = \mu_j P_2 + (\lambda_j + \lambda'_j)P_2 \\ &. \\ &. \\ &. \end{aligned} \tag{7}$$

Considering the fact that total probability is equal to 1, (8) computes P_o.

$$P_0 = \frac{1}{\sum_{n=0}^{k} \left(\dfrac{\lambda_j + \lambda'_j}{\mu_j}\right)^n} \tag{8}$$

Lemma 1. Since each P_n is a function of P_0, every P_n is more than zero if and only if P_0 is more than zero. Considering lemma 1, (9) is resulted wherein α is a positive constant. This relation indicates P_0, and hence all P_n are bigger than zero.

$$\left[\sum_{j=0}^{n} \left(\frac{\lambda_j + \lambda'_j}{\mu_j}\right)^n\right] \prec \alpha; \quad \forall \lambda'_j, \lambda_j, \mu'_j, \mu_j \tag{9}$$

Therefore, (10) can compute P_n for states of actor A_j:

$$P_n = P_0 \times \left(\frac{\lambda_j + \lambda'_j}{\mu_j}\right)^n \tag{10}$$

Analyzing the steady state case of the CTMC gotten from the *M/M/1* queues, the finish time of the actors, $Time_j$, is calculated by (11).

$$Time_j = \frac{1}{\mu_j - \left(\lambda_j + \lambda_j'\right)} \quad (11)$$

To compute the maximum make-span ($Make\text{-}span_{max}$), we find the maximum assignment rate (λ_{MT}) to the actor (A_{MT}) with the lowest service rate result in $Time_{MT}$. $Time_{MT}$ denotes the time that A_{MT} needs to finish its all allocated tasks, and hence (12) is derived:

$$Makespan_{\max} \leftarrow Time_{MT} \quad (12)$$

Considering (4) and (11), Make-span$_{min}$ can be calculated by (13):

$$Makespan_{\min} = Min\left\{\sum_{j=1}^{m}\left(\frac{1}{\mu_j - \left(\lambda_j + \lambda_j'\right)}\right)\right\} \quad (13)$$

Having $Make\text{-}span_{max}$, $Make\text{-}span_{min}$, and $Make\text{-}span_{assignment}$, the first half of *Fitness Fuction* can be calculated. Figure 2 illustrates the pseudo code of MaSC.

Pseudo-code of MaSC

Input: Information of each available actor A_j (*e.g.* μ_j, E_j, α_j), importance of time and energy in the application (W_1, W_2), and the entrance rate of tasks to the sink (λ_T)

Output: $Makespan_{min}$, $Makespan_{max}$, $Makespan_{assignment}$

1. {**For** all actor A_j do
2. Find the local arrival rates λ_j';
3. **For** j=1 to m do // m is the number of actors
4. $Makespan_{min} \leftarrow$ Min $\{\sum_{j=1}^{m}(\frac{1}{\mu_j - (\lambda_j + \lambda'_j)})\}$
5. Calculate $Time_{MT}$ using (11) and λ_{MT};
6. $Makespan_{max} \leftarrow Time_{MT}$;
7. **For** all actor A_j do
8. { Compute $Time_j$;
9. $Makespan_{assignment} \leftarrow$ Max ($Time_j$);
10. **Return** $Makespan_{min}$, $Makespan_{max}$, $Makespan_{assignment}$; }

4.2. Energy consumption calculation protocol

The Energy Consumption Calculation (ECal) method calculates the average energy consumption of actor *j* ($E_j^{Consume}$) participating in the task assignment. ECal aims to determine the second half part of the *Fitness Function* derived from (5). As described in Section 3.3, $E_j^{Consume}$ can be calculated by (2), and hence, the energy consumption of actors to perform allocated tasks ($E_{allocation}^{Consume}$) would be equal to the total energy consumption of the actors. The pseudo-code of ECal is as follows:

Pseudo-code of ECal

Input: Information of each available actor A_j(*e.g.* μ_j, E_j, α_j)

Output: $E_{Max}^{Consume}$, $E_{Min}^{Consume}$, and $E_{allocation}^{Consume}$

1. { $E_{allocation}^{Consume} \leftarrow 0$;
2. **For** j=1 to m do
3. { Calculate $Time_j$ based on Eq. (11)
4. $E_j^{Consume} \leftarrow \alpha_j \times Time_j$;
5. $E_{allocation}^{Consume} \leftarrow E_j^{Consume} + E_{allocation}^{Consume}$; }
6. MaEC( );
7. MiEC ( );
8. **Return** $E_{Max}^{Consume}$, $E_{Min}^{Consume}$, and $E_{allocation}^{Consume}$; }

ECal uses two functions called MaEC and MiEC to calculate maximum energy consumption ($E_{Max}^{Consume}$) and minimum energy consumption ($E_{Min}^{Consume}$), respectively. MaEC and MiEC are based on two lemmas, as follows:

Lemma 2. Total energy consumption of actors is maximum when the tasks are assigned to the most energy consuming actors. Similarly, total energy consumption of actors is minimum when the tasks are assigned to the least energy consuming actors.

Lemma 3. If the maximum task assignment rate is assigned to the most energy consuming actor, the maximum energy consumption of the actor is resulted.

Considering lemma 3, the maximum energy consumption can be calculated by considering the maximum task assignment rate to the actor with maximum α_j. The actor and its assignment rate are shown by A_{EMax} and λ_{EMax}, respectively. Having λ_{EMax}, $Time_{EMax}$ is calculated using (11), and then using (2), the energy consumption by A_{EMin} (E_{Min}) can be calculated. This process is repeated for the actor with the next highest α_j until λ_T is distributed to the actors. Accordingly, $E_{Max}^{Consume}$ can be computed by MaEC. Similarly, if the maximum task assignment rate is assigned to the least energy consuming actor, the maximum energy consumption of the actor is resulted.

The actor and its assignment rate are shown by A_{EMin} and λ_{EMin}, respectively. Having λ_{EMin}, $Time_{EMin}$ can be calculated using (11), and then using (2), E_{Min} can be figured. Consequently, $E_{Min}^{Consume}$ can be computed, which is equal to the total energy consumption of the least consumed actors. The pseudo-code of MaEC is as follows:

Pseudo-code of MaEC

Input: Context information of each actor A_j(*e.g.* $\mu_j, E_j, \alpha_j, \lambda_j'$), arrival rate of tasks to the sink (λ_T)

Output: $E_{Max}^{Consume}$ as the maximum energy consumption of actors

1. { $E_{max}^{Consume} \leftarrow 0$; $\lambda T \leftarrow \lambda_T$;
2. X ← { } // X is theset of actors with higher energy consumptions
3. **For** all actor A_j not in set X do
4. { Sort all actors in terms of α_j;
5. Determine maximum assignment rate (λ_{EMax}) for the actor (A_{EMax}) with <u>*biggest*</u> α
6. Calculate $Time_j$ using λ_{EMax} based on (11)
7. $E_{Max} \leftarrow \alpha_j \times Time_j$;
8. $E_{Max}^{Consume} \leftarrow E_{Max} + E_{Max}^{Consume}$;
9. **If** $(\lambda T - \lambda_{EMax} > 0)$
10. { $\lambda T \leftarrow \lambda T - \lambda_{EMax}$;
11. Add A_j to set X;
12. Repeat line 3 to 7; } }
13. **Return** $E_{Max}^{Consume}$;}

To compute $E_{Min}^{Consume}$, ECal uses the MiEC function, which is similar to MaEC but it differs from MaEC in that it assigns the maximum rates of tasks to the least energy consuming actors, i.e. in line (5) of Pseudo-code of MaEC, the word "*biggest*" should be changed to "*least*". Consequently, the terms $E_{Max}^{Consume}$, E_{Max}, λ_{EMax}, and A_{EMax}, should be changed to $E_{Min}^{Consume}$, E_{Min}, λ_{EMin}, and A_{EMin}, respectively. Finally, having $E_{assignment}^{Consume}$, $E_{Max}^{Consume}$, and $E_{Min}^{Consume}$, the second half of the *Fitness Function* can be computed.

5. Experimental Results

In order to evaluate the performance of TATA, it is compared with TA-QNIP [21], OLB [22], and stochastic task assignment (STA) [23] in terms of make-span, residual energies of actors, and network life time. Furthermore, to evaluate the role of scale on the efficiency of TATA, simulations are run in small and large scales with two different settings, wherein the actors are chosen from various groups with fast, $<\mu_m \le \mu_j \le \mu_{max}>$, medium, $<\mu_e \le \mu_j \le \mu_m>$, and slow service rates $<\mu_j \le \mu_e>$, wherein μ_j denotes the service rate of actor, and A_j, μ_e and μ_m show the minimum and maximum threshold of medium service rates, respectively.

- In the Setting I (small scale), a 10 m × 10 m field is assumed including 100 sensors with 1 m transmission range and 4 actors. It is assumed that both local tasks and global tasks may exist in any time that should be executed by the actors. The primary energy of each actor is supposed to be the same as the others and equal to 25 J.
- In the Setting II (large scale), a 100 m × 100 m field is assumed including 1,000 sensors with 10 m transmission range and 10 actors. The primary energy of each actor is supposed to be the same as others and equal to 25 J.

Figures 3 and 4 show the results of simulations in terms of make-span.

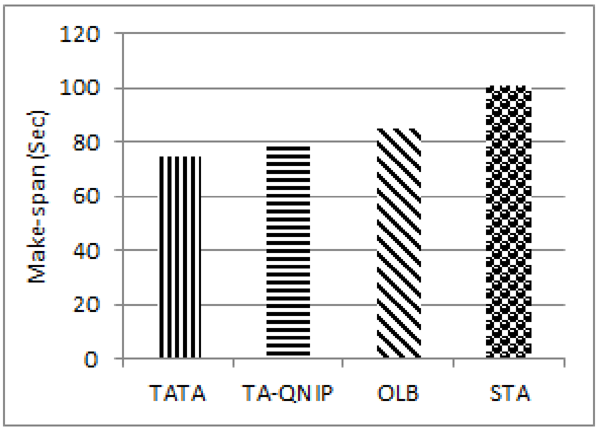

Figure 3. Make-span in setting I.

As shown in Figure 3, in the small scale settings, TATA reduced the make-span by nearly 6% in compare with TA-QNIP but TA-QNIP operated better than STA and OLB in reducing make-span. Since STA assigns tasks to the actors stochastically, it has the worst operation in terms of make-span.

In the large scale settings (Figure 4), TATA performed about 11% better than TA-QNIP but STA still shows the worst performance compared to the other three approaches. However, one of the strengths of TATA compared to TA-QNIP is the reduction of additional overhead. Since the number of tasks and overhead in large scale setting is greater than as they are in small scale setting, in large scale, a greater difference between TATA and TA-QNIP is observed.

In order to achieve a better evaluation, TATA also was compared with three mentioned approaches in terms of the residual energies of the actors. As Figure 5 shows, in small scale setting, TATA and TA-QNIP consume about the same amount of energy but quite less than the other two approaches.

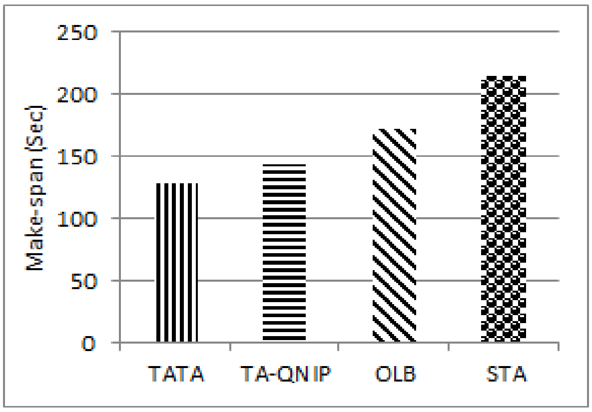

Figure 4. Make-span in setting II.

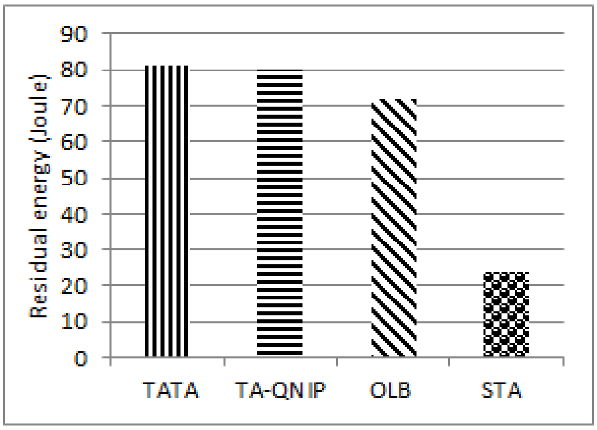

Figure 5. Residual energies of actors in setting I.

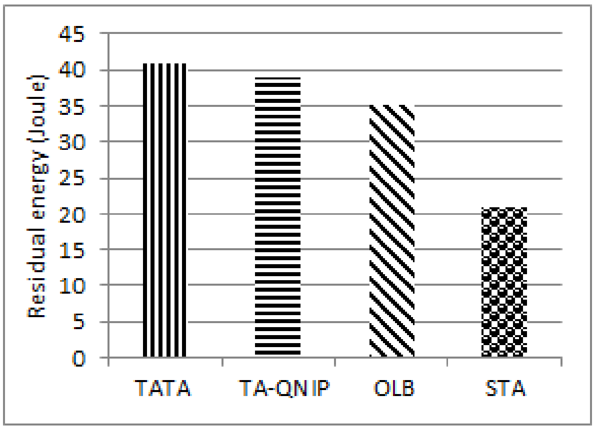

Figure 6. Residual energies of actors in setting II.

In the large scale case, that workload is heavier in compare with the small scale setting, TATA results in the maximum residual energy of actors, with the TA-QNIP being the next in row. STA has the worst consumption rate and the least energy conservation. However, as shown in Figure 5 and Figure 6, in terms of residual energies of actors, TATA shows a better operation about 7% (in average) in comparison with TA-QNIP. All in all, it is concluded that although in small case shorter performance difference between the mentioned approaches is observed, a significant performance difference can be obtained in the large scale case. Since the assignment of tasks in STA is stochastically and without explicit consideration of time and energy, it yielded the weakest results in terms of energy preservation and Make-span.
Nevertheless, STA results in both scales are the worst, while TATA shows the best performance in terms of enhancing the residual energy and reducing make-span.

6. Conclusion

Assignment of tasks to the actors to minimize the network make-span without taking to consideration the energies of nodes is not enough because an actor node may run out of energy, leading to the death of that actor. Since maximizing the residual energy and minimizing the make-span are inconsistent objectives, a balance model should be applied to determine a fitness function. Applying the fitness function helps to figure out the most near optimal task assignment solutions. In this work, an energy-efficient timing task assignment approach called TATA was proposed to assign tasks to the actors in WSANs.
Simultaneously, reducing the make-span and enlarging the energy consumption of actors are the two objectives of TATA. In order to achieve this goal, two protocols called MaSC and ECal were proposed to calculate the network make-span and energy consumption of the actor nodes, respectively. The outcomes of extensive simulations in small scale and large scale networks revealed that TATA yields shorter make-span and higher residual energy in comparison to when one of the TA-QNIP, OLB, and STA approaches was applied.

References

[1] S.K. Mothku and R.R. Rout, “Markov decision process and network coding for reliable data transmission in wireless sensor and actor networks,” *Pervasive and Mobile Computing*, Vol. 56, pp. 29-44, 2019.

[2] M. Okhovvat and M.R. Kangavari, “TSLBS: A Time-Sensitive and Load Balanced Scheduling Approach to Wireless Sensor Actor Networks,” *Computer System Science and Engineering*, Vol. 34, No.1, pp. 13-21, 2019.

[3] A. Nayak and I. Stojmenovic, “Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scal able Coordination and Data Communication,” Hoboken, New Jersey: Wiley Press, 2010.

[4] R. Shams, P. Otero, M. Aamir and F. H. Khan, “Joint Algorithm for Multi-Hop Localization and Time Synchronization in Underwater Sensors Networks

Using Single Anchor," *IEEE Access*, Vol. 9, pp. 27945-27958, 2021.

[5] A. Khosravi, A. Alfi and A. Roshandel, "Delay-dependent stability for transparent bilateral teleoperation system: an LMI approach," *Journal of AI and Data Mining*, Vol. 1, No. 2, pp. 75-87, 2013.

[6] S.M. Dima, C. Antonopoulos and S. Koubias, "Resource Aware Sensor-to-Actor Assignment Framework for WSANs Based on Voronoi Cells Theory," *Journal of Sensors*, Vol. 2017, pp. 1-16, 2017.

[7] M. Sharifi and M. Okhovvat, "Scate: A Scalable Time and Energy Aware Actor Task Assignment Algorithm in Wireless Sensor and Actor Networks," *ETRI Journal*, Vol. 34, No. 3, pp. 330-340, 2012.

[8] M. Arnesano, B. Bueno, A. Pracucci, S. Magnagni, O. Casadei and G. M. Revel, "Sensors and control solutions for Smart-IoT façade modules," in *IEEE International Symposium on Measurements & Networking (M&N)*, pp. 1-6, 2019.

[9] U. Fischer-Hirchert, P. Kussmann, S. Hoppstock and M. Haupt, "Real-time Intelligent Tele-Care Assistance Systems," *Broadband Coverage in Germany; 12th ITG-Symposium*, pp. 1-4, 2018.

[10] C. Konstantopoulos, G. Pantziou, I.E. Venetis and D. Gavalas, "Efficient event handling in Wireless Sensor and Actor Networks: An on-line computation approach," *Journal of Network and Computer Applications*, Vol. 75, pp. 181-199, 2016.

[11] V. P. Modekurthy, A. Saifullah and S. Madria, "DistributedHART: A Distributed Real-Time Scheduling System for WirelessHART Networks," in *IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pp. 216-227, 2019.

[12] H. Byun and J. So, "Node Scheduling Control Inspired by Epidemic Theory for Data Dissemination in Wireless Sensor-Actuator Networks with Delay Constraints," *IEEE Transactions on Wireless Communications*, Vol. 15, No. 3, pp. 1794-1807, 2016.

[13] M.R. Okhovvat, M.T. Kheirabadi, A. Nodehi and M. Okhovvat, "Task Assignment Approach for Minimizing Make-Span in Wireless Sensor Actor Networks," *Computer Systems Science and Technology*, Vol. 39, No. 2, pp.165-178, 2021.

[14] X. Kong, Y. Gao, T. Wang, J. Liu and W. Xu, "Multi-robot Task Assignment Strategy based on Particle Swarm Optimization and Greedy Algorithm," in *IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, pp. 1643-1646, 2019.

[15] Y. Huang, Y. Zhang and H. Xiao, "Multi-robot system task assignment mechanism for smart factory," in *Conference (ITAIC)*, pp. 587-591, 2019.

[16] L. Wang, M. Liu and M. Q. Meng, "A Hierarchical Auction-Based Mechanism for Real-Time Resource Assignment in Cloud Robotic Systems," *IEEE Transactions on Cybernetics*, Vol. 47, No. 2, pp. 473-484, 2017.

[17] M. Younis, K. Akkaya and A. Kunjithapatham, "Optimization of Task Assignment in a Cluster-based Sensor Network," In *Proceeding of 8th IEEE International Symposium on Computers and Communication*, pp. 329-334, 2003.

[18] G. Bolch, S. Greiner, H. De Meer and KS. Trivedi, "Queuing Networks and Markov Chains," Wiley, 2nd Edition, USA, 2006.

[19] J. Kim, A. Dudin, S. Dudin and C. Kim, "Analysis of a semi-open queuing network with Markovian arrival process," *Performance Evaluation*, Vol. 120, pp. 1-19, 2018.

[20] P. L. Burke, "The Output of a Queuing System," *Operation Research*, Vol. 4, No. 6, pp. 699-704, 1956.

[21] J .Xue, Z. Wang, Y. Zhang and L. Wang, "Task Assignment Optimization Scheme Based on Queuing Theory for Mobile Edge Computing in 5G Heterogeneous Networks," *Mobile Information Systems*, Vol. 2020, pp. 1-12, 2020.

[22] 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*, Vol. 59, pp. 107-121, 1999.

[23] S. C. Sarin, B. Nagarajan and L. Liao, "Stochastic Scheduling: Expectation-Variance Analysis of a Schedule," Cambridge: Cambridge University Press, 2010.

انتساب زمانی و انرژی کارای وظایف به عملگرها در WSANs

محمدرضا اخوت[۱]، محمدتقی خیرآبادی[۱*]، علی نودهی[۱] و مرتضی اخوت[۲]

[۱] گروه کامپیوتر، واحد گرگان، دانشگاه آزاد اسلامی، گرگان، ایران.

[۲] گروه فناوری اطلاعات سلامت، مرکز تحقیقات مدیریت سلامت و توسعه اجتماعی، دانشکده پیراپزشکی، دانشگاه علوم پزشکی گلستان، گرگان، ایران.

ارسال ۲۰۲۲/۰۴/۰۶؛ بازنگری ۲۰۲۲/۰۵/۲۸؛ پذیرش ۲۰۲۲/۰۶/۰۸

چکیده:

کمینه‌سازی زمان پاسخ و بیشینه کردن انرژی باقی‌مانده در کاربردهای شبکه‌های حسگر عملگر بی‌سیم (WSAN) از اهمیت بالایی برخوردار است. راهکارهای فعلی انتساب وظایف معمولاً یکی از محدودیت‌های زمانی یا انرژی را لحاظ می‌کنند. این راهکارها انواع و ویژگی‌های مختلف مربوط به وظایف را در اجرای وظایف در نظر نمی‌گیرند و بنابراین ممکن است در برخی از کاربردهای واقعی نظیر مأموریت‌های جستجو و نجات قابل بکارگیری نباشند. بدین منظور، یک راهکار انتساب وظایف بهینه و آگاه از نوع (TATA) پیشنهاد شده است که مصرف انرژی و همچنین زمان پاسخ را در نظر می‌گیرد. TATA یک راهکار انتساب وظایف بهینه و آگاه از ضروریات توزیعی محیط شبکه‌های WSAN با معماری ترکیبی است و شامل دو پروتکل به نام‌های پروتکل محاسبه زمان پاسخ (MaSC) و پروتکل محاسبه مصرف انرژی (ECal) است. TATA با درنظر گرفتن زمان و انرژی، بین کمینه‌سازی زمان پاسخ و بیشینه کردن انرژی باقی‌مانده عملگرها تعادل برقرار می‌کند. نتایج حاصل از شبیه‌سازی‌های گسترده در سناریوهای نوعی نشان‌دهنده زمان پاسخ کمتر و انرژی باقی‌مانده بیشتر در مقایسه با حالتی است که یکی از سه راهکار مرتبط به نام‌های تخصیص وظایف تصادفی (STA)، متعادل‌سازی بار فرصت‌طلبانه (OLB)، و الگوریتم انتساب وظایف مبتنی بر نقطه داخلی گوسی-نیوتن (TA-QNIP) بکار گرفته شود.

کلمات کلیدی: مصرف انرژی، زمان پاسخ، انتساب وظیفه، شبکه‌های حسگر عملگر بی‌سیم.