Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi
Dokuz Eylul University Faculty of Engineering Journal of Science and Engineering
Elektronik/Online ISSN: 2547-958X

RESEARCH ARTICLE / ARAŞTIRMA MAKALESI

Online Planning for Data Collection in Multi-Robot Systems

Çok-Robotlu Sistemlerde Veri Toplama için Çevrimiçi Planlama

Deniz Özsoyeller

Yaşar University, Faculty of Engineering, Department of Software Engineering, İzmir, TÜRKİYE
Corresponding Author / Sorumlu Yazar : deniz.ozsoyeller@yasar.edu.tr

Abstract

Wireless sensors networks have been used for data collection in various civil and military applications. We consider a system where a group of mobile robots and a set of stationary wireless sensor nodes are sparsely deployed in a large unbounded area. In such scenarios, all sensor nodes may not be connected via a communication network. Furthermore, no pair of sensor nodes may be within the transmission range of each other. Therefore, many relay nodes are needed to guarantee the connectivity of the network. However, this approach will affect the lifetime of the system due to the energy consumption by data transmission. In this paper, we study the problem of data collection from the deployed sensors utilizing the robots. The robots do not know the locations of each other and the sensor nodes. Moreover, the sensor nodes do not know the locations of each other and the robots. We propose an online algorithm in which the robot explores the area to find the sensor nodes and collect their stored data. Depending on whether the number of robots is known by the robots in advance or not, we investigate and compare two cases of the problem. In simulations, we empirically evaluate the performance of our algorithm and show that it quantifies as a function of the environment size, the number of robots, and the communication range when both the number of robots and the number of sensors are not known in advance.

Keywords: *Data Collection, Online Planning, Multi-Robot Systems*

Öz

Kablosuz sensör ağları, çeşitli sivil ve askeri uygulamalarda veri toplamak için kullanılmıştır. Bir grup hareketli robotun ve bir küme hareketsiz kablosuz sensör düğümünün sınırsız geniş bir alanda aralıklı olarak konuşlandırıldığını düşünüyoruz. Bu gibi senaryolarda, tüm sensör düğümleri bir iletişim ağı ile bağlı olmayabilir. Buna ek olarak, birbirinin iletişim ağı içinde olan herhangi bir sensör düğümü çifti bulunmayabilir. Bu nedenle, ağ bağlantısının sağlanması için birçok aktarma düğümüne ihtiyaç vardır. Fakat, bu yaklaşım, veri iletiminden kaynaklanan enerji tüketiminden dolayı sistemin ömrünü etkiler. Bu makalede, robotlardan faydalanarak konuşlandırılmış olan sensör düğümlerinden veri toplama problemini çalışıyoruz. Robotlar, birbirlerinin ve sensör düğümlerin konumunu bilmezler. Dahası, sensör düğümleri de birbirlerinin ve robotların konumunu bilmezler. Robotların sensör düğümleri bulmak için alanda keşif yaptığı ve düğümlerdeki veriyi topladığı çevrimiçi bir algoritma öneriyoruz. Robotların, robot sayısını önceden bilip bilmediğine bağlı olarak problemin iki durumunu inceliyor ve karşılaştırıyoruz. Simülasyonlarla, algoritmamızın performansını deneysel olarak değerlendiriyor ve robotun, sensör düğümü ve robot sayısını önceden bilmediği durumda performansın alan boyutu, robot sayısı ve iletişim alanının bir fonksiyonu olarak ölçeklendiğini gösteriyoruz.

Anahtar Kelimeler: *Veri Toplama, Çevrimiçi Planlama, Çok-Robotlu Sistemler*

1. Introduction

Wireless Sensor Networks (WSNs) consist of physically small and battery-powered sensing devices with wireless communication and computation capabilities [1]. They have been used for data collection in various civil and military applications such as habitat monitoring and surveillance. One of the key challenges that affects the lifetime of WSNs is the limited energy of the sensors which is mostly consumed by data transmission.

When data collection process is completed, the sensor nodes need to form a connected network to relay the collected data via multi-hop routing to the data sink (depot station). An important issue in multi-hop communication is that it causes the sensors that are close to the data sink to deplete faster than the other sensors, since the packets destined for the sink have to be forwarded to these nodes. Moreover, in some cases, maintaining system-wide connectivity can only be possible using relay nodes. For example, when the sensor nodes are sparsely distributed across a large environment, there can be isolated nodes which cannot transmit data to any other node. In such scenarios, many relay nodes are needed to guarantee the connectivity of the network, since the transmission ranges of the sensor nodes are typically short. Another approach for data collection is using robots. In this approach, the robot moves to a point within the transmission range of the sensor node to download its sensed data. There are several advantages of utilizing robots as data mules over forming a dense static network. Since the robots visit the sensors to collect data, it is not necessary to deploy relay nodes to achieve network connectivity. This reduces the communication load between the sensors which in turn reduces the energy consumption and improves the lifetime of the system. Moreover, the proximity of the robot to the sensor when downloading data decreases the data loss rate, thus the number of retransmissions.

DOI: https://doi.org/10.21205/deufmd.2025278016

Geliş Tarihi / Received: 04.09.2024
Kabul Tarihi / Accepted: 29.10.2024

The problem of data collection from the sensor nodes (DCP) was studied both using a single robot [2-8] and multiple robots [9-11]. Bhadauria *et al.* [3] presented a constant factor approximation algorithm for DCP for the case where the single robot and the stationary sensors are deployed on a plane. Tsilomitrou and Tzes [10] studied the DCP with multi-robots and stationary sensors formulating the problem as a variation of Traveling Salesman Subset-tour Problem taking various constraints (e.g. pairwise distance between the sensors, visiting time, data download time) into account. When both robots and sensor nodes are mobile, Das *et al.* [9] showed that minimizing the number robots needed to complete data collection within a certain time is NP-Complete. Yedidsion *et al.* [4] focused on finding a tree connecting the sensors and the deployment of a single data mule such that the total distance traveled by the mule is minimized in case of any sensor failure. The study proposed approximation algorithms to solve this problem in Unit Disc Graph (UDG) topology. In UDG, two nodes with the same circular transmission range can communicate only if the distance between them is within this range.

Unmanned Aerial Vehicle (UAV) is another mobile entity that have been used for data collection from the sensor nodes [12-20]. An energy efficient UAV-assisted data aggregation algorithm based on the clustering of deployed sensors was proposed in [14]. The sensors are equipped with a GPS module thus know their own locations, and the sink knows the locations of all sensors in advance. Gul and Erkmen [15] considered that each clustered sensor network also includes robots. The proposed approach selects a cluster head robot (CH) which assigns the data collection task to the other robots within its cluster. Then, an unmanned aerial vehicle (UAV) with limited battery capacity collects data from the CH robots. For the case where the UAV does not have a limited battery capacity, Luo *et al.* [16] focused on optimizing the flight trajectory of the UAV while ensuring that a certain amount of data is collected from each sensor. Given that the data collection areas are disjoint, an approximation algorithm was presented to solve this problem. Wu *et al.* [17] proposed a data aggregation protocol that focused on balancing the energy consumption in the system and reducing the latency for data delivery to the sink by applying a genetic algorithm. Chen *et al.* [19] proposed a (1-1/*e*)-approximation algorithm for one-to-many data collection problem in WSNs where the UAV collects data from multiple sensors simultaneously. To collect data from ground sensor nodes using a UAV, two Reinforcement Learning (RL) approaches are combined in [20] to determine the UAV's trajectory in an environment with obstacles also the order of visiting the sensor nodes .

In addition to their usage on land, sensor nodes are also deployed under the water for a wide range of marine applications such as natural disaster prevention (e.g. earthquake and tsunami), pollution and environmental monitoring [22-23]. For long-distance communication, the commonly used technology under the water is acoustic communication. However, due to the external interference and strong signal attenuation, acoustic channels are constrained by limited bandwidth. Therefore, to reduce the signal propagation distance, thus improve the data reliability and the energy efficiency of the sensors, the autonomous underwater vehicles (AUV) are utilized to collect data from underwater wireless sensor networks (UWSN) [24-26].

In this paper, we study the problem of online planning of multiple mobile robots to collect data from the stationary sensors. The previous studies focused on planning paths for the robots assuming that the locations of the sensor nodes, the robots, and the sink node are known in advance. In contrast, in our work, we do not make any of these assumptions. The robots do not know the initial locations of each other, the sensor nodes, and the sink(s). Moreover, the sensors do not know the initial locations of each other, the robots, and the data sinks except the one where its data should be uploaded. Our contributions are as follows.

1. We propose an online algorithm which allows multi-robots to collect the stored data from all deployed sensors without a priori knowledge about the locations of each other and the sensors.
2. We consider two cases of this problem depending on whether the number of robots is known by each robot in advance or not, also present how to adapt our algorithm according to these cases.
3. We empirically evaluate the performance of our algorithm through simulations varying the key parameters of interest including the number of sensors, the number of robots, the communication range, and the environment size.

The rest of this paper is organized as follows. We formulate the DCP in Section 2 and present our algorithm including its example execution in Section 3. We report the simulation results in Section 4. The concluding remarks are presented in Section 5.

2. Problem Formulation

We consider a system that includes m identical stationary wireless sensors and n identical robots that have wireless communication capabilities and act as data mules. The robots and sensors are sparsely deployed in a large unbounded area. The sensors do not know the locations of each other and the robots. Moreover, the robots do not know the locations of each other and the sensors. There can be more than one data sink for the robot to later offload the gathered data. Each sensor knows only the location of the data sink where its data should be offloaded, but not the location of the other sinks. The robots do not have a priori knowledge about the number and the locations of data sinks. Due to the lack of prior knowledge, we propose an online algorithm which is executed by each robot autonomously.

The i-th robot and the j-th sensor in the environment are denoted by r_i and s_j, respectively, where $i, j \in Z^+$. Each sensor has a unique integer identifier and can sense and transmit data within the distance R. The robot can communicate with the sensor thus exchange messages with it, if the distance between them is within R. We make the following assumptions: (1) Same amount of data is collected from each sensor, (2) The time to collect the stored data from each sensor is the same, (3) The robot stops during data collection.

3. MDC: Multi-Robot Online Data Collection Algorithm

In this section, we introduce our multi-robot online data collection algorithm called MDC. The algorithm consists of two main phases: *phase-1*: exploration phase and *phase-2*: information exchange phase. The pseudocode of MDC is presented in Algorithm 1.

In phase-1, the robot explores the environment to find the sensors at unknown locations. The motion planning strategy of the robot in this phase is as follows (refer to Algorithm 1, lines 4–10). We divide the execution of the strategy in rounds that are indexed by $k \geq 0$. The robot draws concentric circles to explore the environment and find the sensors. Each circle is centered at the robot's initial location. In the first round, when $k = 0$, the robot moves R distance north, and then follows a circle of radius R. In the next rounds, when $k > 0$, the robot moves $2R$ distance north, and then follows a circle of radius $(2k+1)R$. We use this motion planning approach also in our previous work [19].

In phase-2, the robot exchanges information with the sensor that it finds to (1) learn if it is the first visitor of the sensor to decide whether it should collect the required data from the sensor, (2) relay the information of the set of sensors that it has found also learned that the others robots have found, (3) learn about the set of found sensors relayed to the sensor by its previous visitor robots.

When the robot finds a sensor s, it sends the set of the sensors that it has found to s. The sensor s accumulates the set sent by each robot that visits it and sends this gathered information to each new visitor robot. Through this information exchange between the robot and sensor, the robot learns whether it is the first visitor of the sensor or not, also determines whether to terminate the algorithm or not. Only the first robot that finds the sensor collects the required data from the sensor (refer to Algorithm 1, lines 16-19).

S_i denotes the set of all sensors that robot r_i has found during exploration. F_i denotes the set of sensors that r_i is first to find during exploration, that is, r_i finds the sensors in F_i before all the other robots.

The time when a robot meets a sensor is denoted by t which is represented as decreasing order. Thus, t denotes the current meeting, while t-1 denotes the previous meeting, and so on. The sensor information that robot r_i accumulates from each sensor in S_i until t is denoted by $A_i(t-1)$. Let V_j be the list of the robots that have visited sensor s_j. The information that sensor s_j accumulates from each robot in V_j until t is denoted by $S_j^*(t-1)$. When a robot $r_i \in V_j$ finds s_j at time t, it sends $A_i(t-1)$ to s_j and meanwhile receives $S_j^*(t-1)$. When a meeting between sensor s_j and robot r_i occurs, s_j and r_i update the sensor information that they keep as follows:

$$S_j^*(t) = S_j^*(t-1) \cup A_i(t-1) \text{ and}$$

$$A_i(t) = A_i(t-1) \cup S_j^*(t-1),$$

where $S_j^*(t-1) \subseteq \bigcup_{i \in 1,\dots|V_x|} S_{V_x[i]}(t-i)$ and $S_i \subseteq F_i$.

Consider the following example scenario for phase-2 with two robots (r_1 and r_2) and five sensors ($s_1, \cdots, s_5$). Suppose that the first sensor that r_1 finds is s_1. When r_1 and s_1 exchange information, r_1 sends $A_1 = \emptyset$ to s_1 and in return receives $S_1^* = \emptyset$ from s_1 which also implies that r_1 is the first robot that finds s_1. As a result of this phase, r_1 has $F_1 = \{s_1\}$ and $A_1 = \{s_1\}$, while s_1 has $S_1^* = \{s_1\}$ and $V_1 = \{r_1\}$. Next, r_1 finds s_2 which has already been found by r_2. Before, r_2 finds s_2, it has also found s_4 and s_5. Therefore, when r_1 finds s_2, $S_2^* = \{s_2, s_4, s_5\}$. As a result of the information exchange between r_1 and s_2, r_1 has $F_1 = \{s_1\}$ and $A_1 = \{s_1, s_2, s_4, s_5\}$, while s_2 has $S_2^* = \{s_1, s_2, s_4, s_5\}$ and $V_2 = \{r_1, r_2\}$. Next, r_1 finds s_3. Let r_1 be the first robot to find s_3. After r_1 and s_3 exchange information, r_1 has $F_1 = \{s_1, s_3\}$ and $A_1 = \{s_1, s_2, s_3, s_4, s_5\}$, while s_3 has $V_3 = \{r_1\}$ and $S_3^* = \{s_1, s_2, s_3, s_4, s_5\}$.

We consider two cases of the problem depending on whether *case-1*: both the number of sensors and number of robots are known by the robots in advance or *case-2*: the number of robots is unknown, but the number of sensors is known by the robots in advance. In case-1, we assume that $n|m$, that is, m/n is an integer. The approaches for the solutions of the cases differ from each other in the termination part of the algorithm. In case-1, the algorithm terminates when the robot collects data from m/n sensors such that the robot should be the first to find each of these sensors, which occurs when $|F_i| = m/n$ (refer to Algorithm 1, comments between lines 23–24). Whereas, in case-2, the algorithm terminates when the robot ensures that each sensor in the environment is found by a robot, which occurs when $|A_i| = m$ (refer to Algorithm 1, line 24).

3.1. Example Execution of the Algorithm

In Figure 1, we present the example execution of algorithm MDC for case-1. n = 3 robots, $r_1, \cdots, r_3$, and m = 6 sensors, $s_1, \cdots, s_6$, are distributed uniformly at random in a circular area of radius A = 100 with center x = 500 and y = 500. The initial configuration of the robots and sensors is shown in Figure 1:A. We assume that the robot moves at unit speed, thus time and distance are equivalent. The communication range is R = 10. The time for data download from each sensor is fixed to 50 and represented in terms of distance traveled units. The trajectories of robots r_1, r_2, and r_3 are shown in Figure1:B-C, D-G, and H-I, respectively. Table I presents the elements of A_i and F_i for robot i, and S_j^* for the found sensor j after the message exchange phase is finished. Last three columns of the table provide the total time spent, the distance traveled, and the number of rounds executed by the robot until it finds the sensor. Note that total time data in Table-1 is given in distance traveled units and is equal to the sum of the distance traveled by the robot and the download time from the sensors found by the robot, until the robot finds the current sensor. We observe that r_1 is the first visitor of the sensors s_2 and s_3, and r_3 is the first visitor of the sensors s1 and s_5. The first sensor that r_2 finds is s_1. Since s_1 has already been found by r_3, robot r_2 does not collect data from s_1 thus the total time stays the same as the total distance traveled. Robot r_2 continues the exploration until its $|F_2|$ = 2, which occurs when it finds the sensors s_4 and s_6.

Algorithm 1: MDC: Multi-Robot Data Collection Algorithm

// n:the number of robots
// m:the number of sensors
// r_i:i-th robot
// s_j:j-th sensor
1
2 $k = 1$ // the current round
3 **while true do**
4   **if** $k == 1$ **then**
5   Move R forward
6   Follow a circle of radius of R
7   **else if** $k > 1$ **then**
8   Move $2R$ forward
9   Follow a circle of radius of $2R(k+1)$
10   **end**
11   **while** *executing round* k **do**
12   **if** *a sensor* s_j *is found* **then**
13       $\mathcal{A}_i = \bigcup_{j \in \mathcal{S}_i} \mathcal{S}_j^*$
14       $\mathcal{S}_i = \mathcal{S}_i \cup \{s_j\}$
15   Get $\mathcal{S}_j^*$ from s_j
16   **if** s_j *has not been visited by a robot except* r_i **then**
17   Collect data from s_j
18         $\mathcal{F}_i = \mathcal{F}_i \cup \{s_j\}$
19   **end**
20   Send $\mathcal{A}_i$ to s_j
21       $\mathcal{A}_i = \mathcal{A}_i \cup \mathcal{S}_j^*$
      // s_j performs $S_j^* = S_j^* \cup \mathcal{A}_i$ operation when it receives $\mathcal{A}_i$ from r_i.
22   **end**
23   **end**
  // For Case-1, Line 24 is:
  // 'If $|\mathcal{F}_i| == m/n$'.
24   **if** $|\mathcal{A}_i| == m$ **then**
25   **break**
26   **else**
27     $k = k + 1$
28   **end**
29 **end**

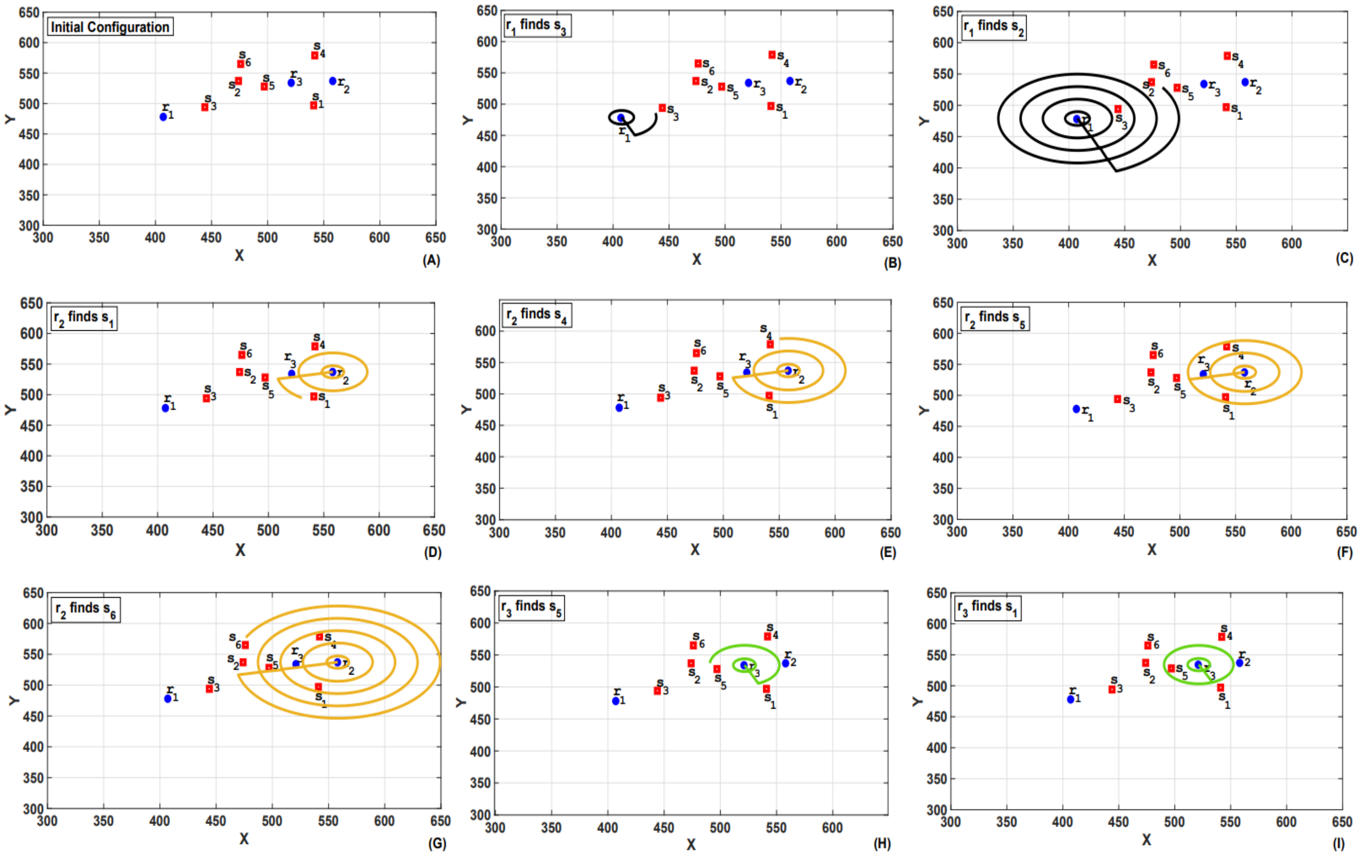

Figure 1. Example execution of Algorithm MDC. The robot is depicted with a circle and the sensor is depicted with a square.

Table 1. Output data for the example execution of Algorithm MDC which is shown in Figure 1.

r_i	s_i	F_i	A_i	S_j^*	Total Time	Total Distance	Round
r_1	s_3	$F_1 = \{3\}$	$A_1 = \{3\}$	$S_3^* = \{3\}$	139	139	1
r_3	s_5	$F_3 = \{5\}$	$A_3 = \{5\}$	$S_5^* = \{5\}$	223	223	1
r_3	s_1	$F_3 = \{1, 5\}$	$A_3 = \{1, 5\}$	$S_1^* = \{1, 5\}$	335	285	2
r_2	s_1	$F_2 = \emptyset$	$A_2 = \{1, 5\}$	$S_1^* = \{1, 5\}$	344	344	2
r_2	s_4	$F_2 = \{4\}$	$A_2 = \{1, 4, 5\}$	$S_4^* = \{1, 4, 5\}$	538	538	2
r_2	s_5	$F_2 = \{4\}$	$A_2 = \{1, 4, 5\}$	$S_5^* = \{1, 4, 5\}$	670	620	3
r_1	s_2	$F_1 = \{2, 3\}$	$A_1 = \{2, 3\}$	$S_2^* = \{2, 3\}$	1309	1259	4
r_2	s_6	$F_2 = \{4, 6\}$	$A_2 = \{1, 4, 5, 6\}$	$S_6^* = \{1, 4, 5, 6\}$	1655	1605	4

4. Simulations

We evaluate the performance of our algorithm MDC through simulations by implementing a multi-threaded system. Each thread represents a robot and runs MDC. The simulation results are shown in Figure 2:A-O. The robots and sensors are distributed uniformly at random in an unbounded circular area of radius A.

We investigate the distance traveled by the robot, the total time spent by the robot, the number of rounds executed by the robot, the number of sensors found in a round by the robot, the number of sensors found by the robot, and the number of sensors that the robot is first to find by varying the key parameters of interest including the number of robots (n), the number of sensors (m), the communication range (R), and the environment size (represented with its radius A). We vary n between 2 and 32, m between 8 and 64, A between 100 and 200, and R between 5 and 50. For each combination of the tested parameter pair, we take the average of the performance over 100 trials. The maximum of the evaluated data among the robots is used for each trial.

Figure 2:A-J shows the performance of algorithm MDC for case-2. For fixed n and R, Figure 2:A shows the distance traveled by the robot with respect to the change in m and A. We observe that the distance traveled stays constant over increasing m and increases as A increases. For fixed n and A, Figure 2:B shows the distance traveled by the robot with respect to the change in m and R. The robot travels less distance and complete the data collection task sooner as R increases. When m and R are fixed, Figure 2:C shows that increasing the number of robots improves the performance of the algorithm. The total number of rounds that the robot executes is presented in Figure 2:D with respect to the change in m and R, and in Figure 2:E with respect to the change in n and A. These results are proportional to the distance traveled results shown in Figure 2:B-C, since as the robot executes more rounds, its traveled distance also increases.

In Figure 2:F, we observe that the number of sensors that the robot is first to find increases as m increases and decreases as R increases. Figure 2:C shows that as the data collection task is distributed among more robots, the maximum of the distance traveled among the robots decreases. This further implies that the robot has yet explored a smaller part of the area. Therefore, the number of sensors that robot i is first to find and the total number of sensors that robot i finds to achieve $|A_i| = m$ decreases as n increases. This outcome is depicted in Figure:G and Figure:I, respectively. From Figure 2:H, we obtain that when m and n are fixed, the number of sensors found by the robot is not affected by the change in A. For fixed R and n, Figure 2:J shows that the number of sensors found in a round by the robot increases as m increases, but decreases as A increases since the pairwise distance between the sensors increases.

In Figure 2:K-O, we compare the proposed approaches for case-1 and case-2 and show the effect on the performance of the algorithm. Recall that in case-2, unlike case-1, the robots do not know n in advance and the number of sensors that each robot collects data from is not neccessarily balanced. To terminate MDC, in case-1, robot i must satisfy $|F_i| = m/n$, whereas in case-2, robot i must satisfy $|A_i| = m$. Figure 2:K-L shows the distance traveled by the robot with respect to the change in m for fixed n and R, while Figure 2:M-N shows the distance traveled by the robot with respect to the change in n for fixed m and R. The results show that MDC performs better in case-2 comparing to case-1. The total time which is shown in Figure 2:O is the sum of the total distance traveled by the robot and the total download time from the sensors that the robot is first to find. The download time is fixed and assigned to 50.

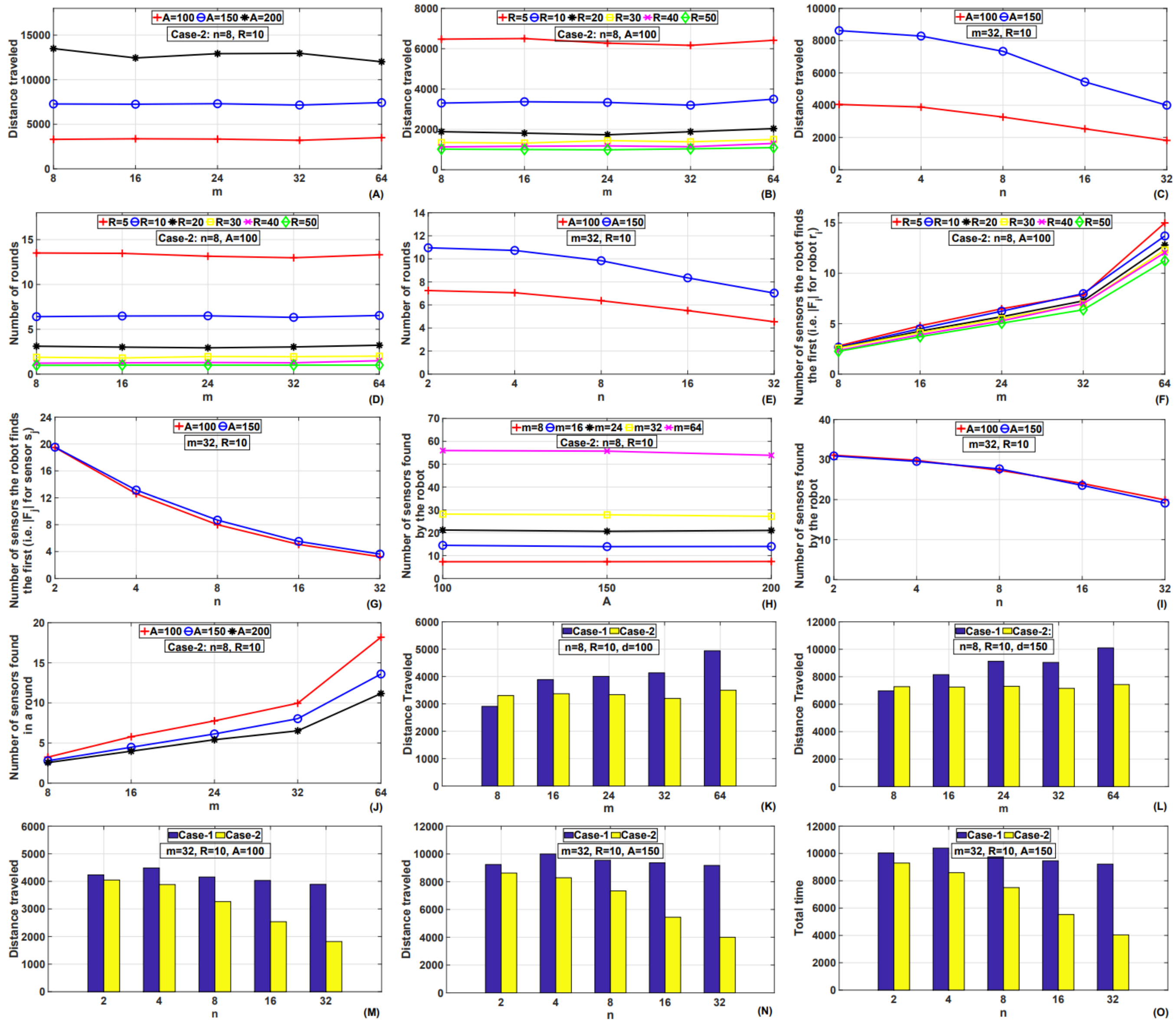

Figure 2. Simulation results that show the performance of Algorithm MDC.

5. Conclusion

We studied the problem of data collection from a set of stationary wireless sensor nodes utilizing multi-robots. The robots and sensors are deployed in an unbounded large area and unaware of each other. We proposed an online algorithm which plans the actions of the robot without requiring the locations of the sensors and the other robots as input. Moreover, we compare two cases of the problem depending on whether the robots know the number of robots in advance or not. For these cases, we showed how to adapt our algorithm. The performance of the proposed algorithm was empirically evaluated also compared for the mentioned two cases through simulations varying the key parameters of interest including the number of sensor nodes, the number of robots, the communication range, and the environment size. Through the reported simulations results, we showed that the approach used in case-2, where the robot terminates the strategy after ensuring that each sensor node is found by at least one robot, performs better than the approach used in case-1, where the robot terminates the strategy as soon as it finds a certain number of sensor nodes. Moreover, we validated that the performance of our algorithm for case-2 quantifies as a function of the environment size, the number of robots, and the communication range.

The proposed algorithm can be extended to UAVs considering that the UAV flies at a fixed altitude and has enough battery capacity to complete its task, and no obstacles exist at the altitude that it flies. However, for long running or persistent tasks, having unlimited operational time for the UAV is not a reasonable assumption. Therefore, our future work includes replanning the path of the UAV during exploration taking the visits to the recharging station(s) into account. In future, we also intend to implement the extended algorithm on an actual robotic system and carry out real-world experiments.

Another challenging future work is to consider the same problem for a heterogeneous multi-robot system where the robots have varying capabilities (e.g. communication range, velocity) and are located in environments with obstacles. The main difference between the proposed solution and the solution to be designed for the environments with obstacles would be the motion planning strategy that the robot executes for exploring the environment. Taking uncertainties into account while planning the path of the robot is another interesting future work. The proposed algorithm can be extended considering the robot failures or new robots that are added to the system during the mission.

Ethics committee approval and conflict of interest statement

This article does not require ethics committee approval. This article has no conflicts of interest with any individual or institution.

References

[1] Yick, J., Mukherjee, B., Ghosal, D., 2008. Wireless sensor network survey. Computer Networks, Vol.52(12), pp.2292–2330. DOI: 10.1016/j.comnet.2008.04.002.

[2] Hu, Y., Zhang, F., Tian, T., Ma, D., Shi, Z., 2022. Shortest path planning of a data mule in wireless sensor networks. Wireless Networks, Vol.28(3), pp.1129–1145. DOI: 10.1007/s11276-022-02891-4.

[3] Bhadauria, D., Tekdas, O., Isler, V., 2011. Robotic data mules for collecting data over sparse sensor fields. Journal of Field Robotics, Vol.28(3), pp.388–404. DOI: 10.1002/rob.20384.

[4] Yedidsion, H., Ashur, S., Banik, A., Carmi, P., Katz, M.J., Segal, M., 2020. Sensor network topology design and analysis for efficient data gathering by a mobile mule. Algorithmica, Vol.82, pp.2784–2808. DOI: 10.1007/s00453-020-00704-8.

[5] Ma, M., Yang, Y., Zhao, M., 2013. Tour planning for mobile data gathering mechanisms in wireless sensor networks. IEEE Transactions on Vehicular Technology, Vol.62(4), pp.1472–1483. DOI: 10.1109/TVT.2012.2229309.

[6] Bhadauria, D., Isler, V., 2009. Data gathering tours for mobile robots. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.3868–3873. DOI: 10.1109/IROS.2009.5354343.

[7] Chang, J.-Y., Jeng, J.-T., Sheu, Y.-H., Jian, Z.-J., Chang, W.-Y., 2020. An efficient data collection path planning scheme for wireless sensor networks with mobile sinks. EURASIP Journal on Wireless Communications and Networking, Vol.2020(1), p.257. DOI: 10.1186/s13638-020-01873-4.

[8] Chen, T.-C., Chen, T.-S., Wu, P.-W., 2011. On data collection using mobile robot in wireless sensor networks. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, Vol.41(6), pp.1213–1224. DOI: 10.1109/TSMCA.2011.2157132.

[9] Das, A., Mazumder, A., Sen, A., Mitton, N., 2016. On mobile sensor data collection using data mules. In: 2016 International Conference on Computing, Networking and Communications (ICNC), pp.1–7. DOI: 10.1109/ICCNC.2016.7440562.

[10] Tsilomitrou, O., Tzes, A., 2022. Mobile data-mule optimal path planning for wireless sensor networks. Applied Sciences, Vol.12(1), p.247. DOI: 10.3390/app12010247.

[11] Papatheodorou, S., Smyrnakis, M., Hamidou, T., Tzes, A., 2018. Path planning and task assignment for data retrieval from wireless sensor nodes relying on game-theoretic learning. In: 2018 5th International Conference on Control, Decision and Information Technologies (CoDIT), pp.1073–1078. DOI: 10.1109/CoDIT.2018.8394924.

[12] Nguyen, M.T., Nguyen, C.V., Do, H.T., Hua, H.T., Tran, T.A., Nguyen, A.D., Ala, G., Viola, F., 2021. UAV-assisted data collection in wireless sensor networks: A comprehensive survey. Electronics, Vol.10(21), p.2603. DOI: 10.3390/electronics10212603.

[13] Gul, O.M., Erkmen, A.M., Kantarci, B., 2022. UAV-driven sustainable and quality-aware data collection in robotic wireless sensor networks. IEEE Internet of Things Journal, Vol.9(24), pp.25150–25164. DOI: 10.1109/JIOT.2022.3195677.

[14] Wu, Q., Sun, P., Boukerche, A., 2019. Unmanned aerial vehicle-assisted energy-efficient data collection scheme for sustainable wireless sensor networks. Computer Networks, Vol.165, p.106927. DOI: 10.1016/j.comnet.2019.106927.

[15] Gul, O.M., Erkmen, A.M., 2023. Energy-aware UAV-driven data collection with priority in robotic wireless sensor network. IEEE Sensors Journal, Vol.23(15), pp.17667–17675. DOI: 10.1109/JSEN.2023.3286877.

[16] Luo, C., Chen, W., Li, D., Wang, Y., Du, H., Wu, L., Wu, W., 2021. Optimizing flight trajectory of UAV for efficient data collection in wireless sensor networks. Theoretical Computer Science, Vol.853, pp.25–42. DOI: 10.1016/j.tcs.2020.05.019.

[17] Wu, Q., Sun, P., Boukerche, A., 2018. An energy-efficient UAV-based data aggregation protocol in wireless sensor networks. In: Proceedings of the 8th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, pp.34–40. DOI: 10.1145/3272036.3272047.

[18] Alfattani, S., Jaafar, W., Yanikomeroglu, H., Yongacoglu, A., 2019. Multi-UAV data collection framework for wireless sensor networks. In: 2019 IEEE Global Communications Conference (GLOBECOM), pp.1–6. DOI: 10.1109/GLOBECOM38437.2019.9014306.

[19] Chen, M., Liang, W., Li, Y., 2020. Data collection maximization for UAV-enabled wireless sensor networks. In: 29th International Conference on Computer Communications and Networks (ICCCN), pp.1–9. DOI: 10.1109/ICCCN49398.2020.9209619.

[20] Bouhamed, O., Ghazzai, H., Besbes, H., Massoud, Y., 2020. A UAV-assisted data collection for wireless sensor networks: Autonomous navigation and scheduling. IEEE Access, Vol.8, pp.110446–110460. DOI: 10.1109/ACCESS.2020.3002538.

[21] Rezende, J.d.C.V., Silva, R.I.d., Souza, M.J.F., 2020. Gathering big data in wireless sensor networks by drone. Sensors, Vol.20(23), p.6954. DOI: 10.3390/s20236954.

[22] Chaudhary, M., Goyal, N., Benslimane, A., Awasthi, L.K., Alwadain, A., Singh, A., 2022. Underwater wireless sensor networks: enabling technologies for node deployment and data collection challenges. IEEE Internet of Things Journal, Vol.10(4), pp.3500–3524. DOI: 10.1109/JIOT.2022.3218766.

[23] Wei, X., Guo, H., Wang, X., Wang, X., Qiu, M., 2021. Reliable data collection techniques in underwater wireless sensor networks: A survey. IEEE Communications Surveys & Tutorials, Vol.24(1), pp.404–431. DOI: 10.1109/COMST.2021.3134955.

[24] Liu, Z., Meng, X., Liu, Y., Yang, Y., Wang, Y., 2021. AUV-aided hybrid data collection scheme based on value of information for internet of underwater things. IEEE Internet of Things Journal, Vol.9(9), pp.6944–6955. DOI: 10.1109/JIOT.2021.3115800.

[25] Khan, W., Hua, W., Anwar, M.S., Alharbi, A., Imran, M., Khan, J.A., 2022. An effective data-collection scheme with AUV path planning in underwater wireless sensor networks. Wireless Communications and Mobile Computing, Vol.2022(1), p.8154573. DOI: 10.1155/2022/8154573.

[26] Huang, M., Zhang, K., Zeng, Z., Wang, T., Liu, Y., 2020. An AUV-assisted data gathering scheme based on clustering and matrix completion for smart ocean. IEEE Internet of Things Journal, Vol.7(10), pp.9904–9918. DOI: 10.1109/JIOT.2020.2988035.

[27] Ozsoyeller, D., Isler, V., Beveridge, A., 2012. Symmetric rendezvous in planar environments with and without obstacles. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol.26, pp.2046–2052. DOI: 10.1609/aaai.v26i1.8385.