1.5em 0pt

Beam-ACO Approach For Data Collection in Heterogeneous Wireless Sensor Networks

Rijul Singla, Rakesh Banoth

***Abstract*—Heterogeneous sensors follow many different data-generating distributions in heterogeneous wireless sensor networks(HWSN), a compact process. Mobile sinks (MSs) are commonly employed to gather data from wireless sensors. Current approaches often assume synchronized data generation. All sensors produce data simultaneously, with consideration given to delay-tolerant and sensitive data within certain delay limits while disregarding the actual generation times of the data. This article shifts its focus to unsynchronized data collection for HWSNs, where sensors can generate data independently, mirroring the conditions of real monitoring applications by modeling the problem as a traveling salesman problem with time windows(TSPTW). This paper proposes a hybrid approach, merging ant colony optimization(ACO) with beam search to minimize travel time and address the task of reducing travel costs and energy constraints.**

***Index Terms*—Heterogeneous wireless sensor networks (HWSN), Beam-ACO (Beam search - ant colony optimization), Mobile sinks (MS), Travelling salesman problem with timing windows (TSPTW).**

I. Introduction

OVER recent years, wireless sensors have seen increased deployment in remote areas for surveying purposes like environmental, habitat, and weather tracking, as well as for applications in agriculture, military operations, and research [1]. Due to the absence of data stations or internet connectivity in these wild and sparsely populated regions, mobile sinks (MSs) — comprising robots, mobile vehicles (MVs), uncrewed aerial vehicles (UAVs), and similar platforms — are extensively employed to gather data from sensors, IoT devices, or disconnected sensor networks [2] [3] [4] [5]. Each MS can periodically visit designated sensors, gather data, and then relay it to nearby base stations. However, optimizing the navigation of (MSs) to ensure efficient data collection is a critical yet complex challenge, given their limited energy resources.

In conventional wireless sensor networks (WSNs), numerous data collection schemes utilizing mobile nodes are predicated on synchronized data generation, assuming all sensors produce data concurrently. This approach focuses on data delay limits, overlooking the actual generation times. Consequently, various MS-assisted strategies have emerged to facilitate effective path planning for MSs while adhering to the data delay constraints of sensors, especially for managing delay-tolerant and delay-sensitive data.

Compared to conventional WSNs comprising homogeneous sensors, heterogeneous WSNs (HWSNs) present a notably challenging network environment. Here, heterogeneous sensors may possess diverse computing capabilities, communication protocols, power supplies, and sensor types. Early research efforts on HWSNs often focus on security functions [6], coverage [7], cost considerations [8], energy conservation, and similar aspects.

Recently, a growing focus has been on data fusion schemes to mitigate challenges such as prolonged delays and outdated data in HWSNs. These schemes often involve carefully crafted data-gathering paths based on clustering strategies tailored to accommodate different sensor groups that follow distinct data-generating distributions. To some degree, such clustering provides a variable and adaptable approach to data gathering in HWSNs, which are marked by distinct heterogeneity.

For unsynchronized data collection, it's imperative that the MS visits sensors within their designated data gathering windows. We aim to chart the MS path to minimize overall energy consumption while meeting data lifetime constraints.

The Travelling Salesman Problem with Time Windows (TSPTW) holds significance in logistics, serving as a vital tool for modeling routing and task scheduling. In routing scenarios, it addresses the challenge of efficiently visiting multiple customers within specified time windows, starting and ending at a central depot. Different time windows model the heterogeneity of the network due to different sensors.

This research's primary focus and contributions are outlined as we reassess the premise of synchronized data in HWSNs and discover that existing data-gathering methods could result in unanticipated performance decline in heterogeneous sensors where data generation is not synchronized.

In Section II, we present the related works pertaining to the issue of data collection in HWSNs. Section III delineates the system model and problem formulation. The proposed Bram-ACO approach is detailed in Section IV. Section V contains the experimental results, including a comparison with baseline methods. Finally, Section VI serves as the conclusion, also offering insights into future avenues of exploration.

II. Related Work

Considering the critical role of energy in sustaining the network over time, the significance of energy-conserving designs persists in data-gathering strategies for HWSNs. Lin et al. [9] devised a data-gathering procedure in HWSNs assisted by mobile sinks (MSs) to extend the network's lifespan. They dynamically choose data collection points in each phase to balance longevity and quality. Additionally, they dynamically assign each sensor and its parent to form a tree topology, reducing energy consumption among data nodes.

Osamy et al. [10] established energy-conscious disjoint supreme sets, serving as data-gathering nodes per phase to enhance the network's lifespan. Their methodology employs an energy-sensitive algorithm rooted in swarm intelligence to create these disjoint dominating sets. Additionally, it determines data gathering paths to optimize data gathering efficiency while minimizing energy usage.

Kumar et al. [11] formulated an improved energy-efficient clustering strategy aimed at boosting the network's lifespan and curbing energy usage in HWSNs. Their approach takes into account multiple parameters, including the selection of cluster heads, the initial energy levels of sinks, the count of active/inactive nodes, and the remaining energy reserves.

Palanisamy et al. [12] introduced a framework for scheduling multisensor data synchronization of HWSNs to enhance data gathering efficiency at the sink in HWSNs. Their approach involves developing in-network gathering and sensor data routing based on dynamic routing rules, ensuring dependable data transmission.

Sun et al. [13] investigated the challenge of rapid change detection in unidentified HWSNs. For instance, they addressed scenarios where an event alters the data-generating distribution within the network at an undetermined moment. They developed a blended CuSum algorithm and demonstrated its optimality according to Lorden's criterion.

Yang et al. [14] uses a GAT network to model the network. They then employ a twin PPO and Deep Q-Learning model to estimate rewards and find the optimal path.

III. PROBLEM FORMULATION

A. System model

This system comprises wireless sensors, one MS depot, and one data station. The data station, serving as the endpoint for data collection, could be a nearby base station, Wi-Fi access points, satellite stations, or similar facilities.

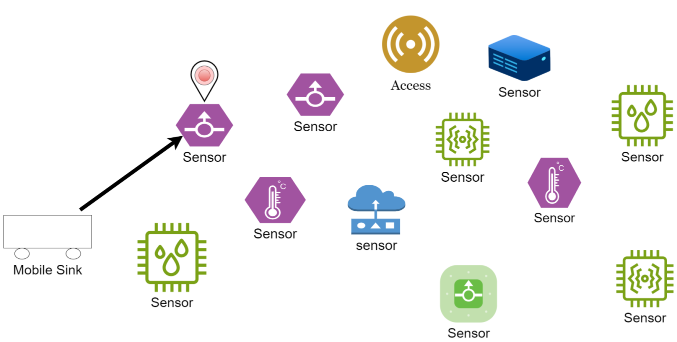

Fig. 1: Heterogeneous Wireless Sensor Networks

1) Assumption of Sensors: In an area with sparsely distributed wireless sensors, each sensor can only establish a connection with a single mobile sink at any given time. These sensors generate sensory data and initiate collection tasks with designated time windows. Each sensor ($i \in I$) determines the collection time window for its data as $[e_i,\ l_i]$, where e_i indicates the time of data generation, and l_i is based on the maximum delay limit of the data. The data size for each sensor is given ϕ_i.

Considering the gathered previous sensor data, the predictability of the sensor's data generation cycle forms the foundation for MS trajectory planning.

2) Assumption of MS: Sensor location points, data initialization settings, control messages, and other modifications and updates are received by MSs linked to base station links that go to backend servers.

We will use v m/s and qm J/m, respectively, to represent the MS's moving speed and energy consumption throughout the experiment. Only a single sensor data node is accessed at a time by each MS in order to collect data. This study makes the assumption that the MS can gather data from sensors rather than delving into the particulars of MS types.

3) Assumption of MS Depot: In the designated area, a single depot positioned at location point L serves as the initial point of the MS path. The depot is capable of replenishing MS energy at any given moment.

4) The TSP with time windows(TSPTW): The paper assumes that $G = (N, A)$ is an undirected complete graph. Nodes indicating the depot and the (node 0) and the n customers are given by N = {0, 1, ..., n}. The set of edges connecting these sensor nodes is represented by and $A = N \times N$ where the conjunction of i^{th} and $(i+1)^{th}$ nodes represents an edge. A solution to the TSPTW is a tour that visits each node once, initially and finally, at the MS depot, represented by node 0. This means a tour is defined by $P = (p_0 = 0, p_1, \ldots, p_n, p_{n+1} = 0)$. Here the sequence $(p_1, \ldots, p_k, \ldots, p_n)$ signifies a permutation of the nodes in N excluding 0, where p_k represents the index of the customer at the kth position of the tour. The inclusion of $p_0 = 0$ and $p_{n+1} = 0$ indicates that each tour starts and ends at the central depot (node 0).

For a given tour P, the leaving time from customer p_k can be evaluated using the formula $Dep_{p_k} = \max(Arr_{p_k}, c_{p_k})$, where the arrival time at customer p_k is represented by $Arr_{p_k} = Dep_{p_{k-1}} + c(a_{p_{k-1}}, a_{p_k})$.

As previously mentioned, this paper focuses on minimizing the travel cost, which entails reducing the cost associated with traversing the edges along the tour.

The TSPTW is formally defined as the minimization of the objective function $f(P) = \sum_{k=0}^{n} E(a_{p_k, p_{k+1}})$, with the added constraint of minimizing $Vio(P) = \sum_{k=0}^{n+1} \omega(p_k)$, where $\omega(p_k) = 1$ if $Arr_{p_k} \leq l_{p_k}$, and $\omega(p_k) = 0$ otherwise, with $Arr_{p_{k+1}} = \max(Arr_{p_k}, e_{p_k}) + t(a_{p_k, p_{k+1}})$.

The term $Vio(P)$ represents the count of time window constraints breached by tour P, which must be minimized.

B. Energy Model

1) Assumption of MS: The primary energy consumption of the MS occurs during its trajectory. Assuming the MS maintains a constant speed in calm surroundings, its total energy expenditure is computed as

$$E^u(\psi(t)) = \left((F_f + F_w) \cdot V_{\max} \cdot \frac{1}{\eta} \cdot t\right) \qquad (1)$$

The resistance from rolling, denoted by $F_f = mg \cdot f$, arises due to the MS's weight m multiplied by the rolling resistance coefficient f. Windward resistance, expressed as $F_w = \frac{1}{2} C_D A \rho V^2$, accounts for the opposing force exerted by the wind. Here, C_D represents the coefficient of wind resistance, A signifies the windward area, while ρ denotes air density and V indicates velocity. Moreover, the mechanical efficiency is represented by η.

C. Energy of transmission

The energy consumed in transmission is given by:

$$E_{\text{trans}} = \sum_{i=1}^{n} P_{\text{trans}} c_{\text{trans},i} \phi_i \tag{2}$$

where P_{trans} is the power of transmission, ϕ_i is data size and $c_{\text{trans},i}$ is the channel capacity.

IV. Proposed Solution

The paper uses an implementation of Ant Colony Optimization combined with Beam search to find the solution for the problem of data collection in heterogeneous wireless sensor networks(HWSNs). Ant colony optimization (ACO) is a problem-solving technique that constructs solutions based on probabilities. In each iteration of the algorithm, several solutions are built independently. However, Beam-ACO uses a probabilistic beam search method instead. This approach creates multiple parallel and interdependent solutions during each iteration.

During each stage of construction, beam search retains a select count of the most promising partial solutions upon which to continue expanding. The pheromone model :

$$p_i(j) = \frac{\tau_{ij} \times \eta_{ij}}{\sum_{k \in N(P)} \tau_{ik} \cdot \eta_{ik}}, \quad \text{if } j \in N(P). \tag{3}$$

is usually used as a probability function to go from node i to node j. The paper uses a slightly different function to compare different paths. For a given path calculate the sum of all ranks. Instead of using η_{ij}, we use the new ranksum function v as the greedy function.

$$v(p) = \sum_{i=1}^{l} \text{rank}(i) \tag{4}$$

where l is the length of the path. The probability function

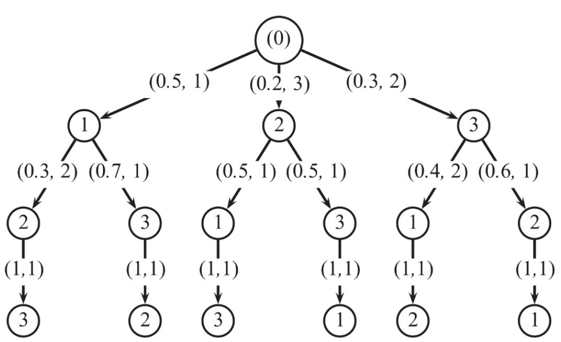

Fig. 2: Ranks of paths

thus becomes

$$p_i(j) = \frac{\tau_{ij} \times v_{ij}^{-1}}{\sum_{k \in N(P)} \tau_{ik} \cdot v_{ij}^{-1}}, \quad \text{if } j \in N(P). \tag{5}$$

2 shows an example of how ranksum function calculates the rank of a path. Each tuple on a node represents the probability of going to that node and the its rank among other nodes. The total rank of a path is calculated by summing up all the rank of its nodes in its path. Finally, the path with the lowest ranksum is chosen.
The ranks are calculated by and compared on the product of τ_{ij} and η_{ij}

$$n_{ij} = \lambda c \cdot \frac{c_{\text{max}} - c_i}{c_{\text{max}} - c_{\text{min}}} + \lambda l \cdot \frac{l_{\text{max}} - l_j}{l_{\text{max}} - l_{\text{min}}} + \lambda e \cdot \frac{l_{\text{max}} - l_i}{l_{\text{max}} - l_{\text{min}}} + \lambda p \cdot \frac{\phi_{\text{max}} - \phi_i}{\phi_{\text{max}} - \phi_{\text{min}}} \tag{6}$$

Below there is an implementation of the 3 algorithms used in the solution. In one iteration of algorithm 1, we obtain a single path. Run through this algorithm multiple times with the primary goal of reducing the number of violations (Mobile Sink arriving at the node outside its time window) and the secondary goal of reducing path length. The ChooseFrom function takes in a list of paths and outputs a single path with probabilities given by 3.

Algorithm 1 Beam Search with ACO

Input: η, τ
$B = \{0\}$
Assign values to λ_c, λ_l, λ_e, λ_p
for $t \leftarrow 1$ to n **do**
  $c = \text{GetNextPaths}(B)$
  for $k \leftarrow 1$ to $\min(|c|, \mu \cdot \text{k}_{bw})$ **do**
    $p = \text{ChooseBest}(c, \eta, \tau)$
    Remove path from c
    $B_{t+1} = B_t \cup p$
  end for
end for
Output: b[0]

To compare 2 paths, a lex function is used. If the number of violations of the first path is 2 or less than the number of the second, the first one is deemed to be a lexicographically smaller path. In case of conflict, travel cost is used as the tie-breaker. The pheromone values are updated following the principles of the ACO algorithm after every successful convergence. The convergence factor is calculated as :

$$p_i(j) = \frac{\tau_{ij} \times \eta_j}{\sum_{k \in N(P)} T_{ik} \cdot \eta_{ik}}, \quad \text{if } j \in N(P). \tag{7}$$

The GetNextPaths function takes in a list of paths as an input. It exhaustively adds the next unvisited node to all the paths, making the algorithm exhaustive and complete. It then returns a sorted list where the probabilities sort all the paths.

$$cf = 2\left(\frac{\sum_{T_{ij} \in \emptyset} \max\{\tau^{\max} - \tau_{ij}, \tau_{ij} - \tau^{\min}\}}{\tau^{\max} - \tau^{\min}} - 0.5\right) \tag{8}$$

Algorithm 2 GetPBS

Input: η, τ
$cf = 0.5$
$prb, pbf =$ NULL
$\tau_{ij} = 0.5$ for all i, j
for $t \leftarrow 1$ to 10000 **do**
    $pib =$ pbs(η, τ)
    if lex(pib) < lex(prb) **then**
        $prb = pib$
    end if
    if lex(pib) < lex(pbf) **then**
        $pbf = pib$
    end if
    $cf =$ ComputeCF(τ)
    if isend = true and $cf > 0.99$ **then**
        $\tau_{ij} = 0.5$ for all i, j
        isend = false
    else
        if isend = false **then**
            isend = true
        end if
        UpdatePheronome()
    end if
end for
Output: pbf

Algorithm 3 GetNextPaths

Input: List of paths B
for $i \leftarrow 1$ to $|B|$ **do**
    for $j \leftarrow 1$ to n **do**
        if j not in $B[i]$ **then**
            Add j to $B[i]$
        end if
    end for
end for
Sort the paths based on the maximal product of η_{ij} and τ_{ij}

V. EXPERIMENTS

A sample of 16 coordinates is distributed in a 100 X 100 grid with different time windows and with different data sizes, as shown in fig 1. The path that the MS follows by utilizing our approach is displayed in the fig. 3. The results were compared with the naive greedy implementation and with the MSPP algorithm proposed in [6].

TABLE I: Experimental Parameters

Parameter	Value
Number of MSs	1
v, Speed of the MS	1 m/s
$P_{m\text{trans}}$, Transmission power from Mobile Sink	10 W
m, The mass of Mobile Sink	1000 Kg
f, coefficient of rolling friction	0.032
C_D, coefficient of drag	0.28
A, area of windward	1.87 m^2
ρ, Coefficient of correction	0.05
η, mechanical efficiency	0.8
τ_{max}	0.999
τ_{min}	0.001
k_{bw}	10
μ	1.5
ρ	0.1

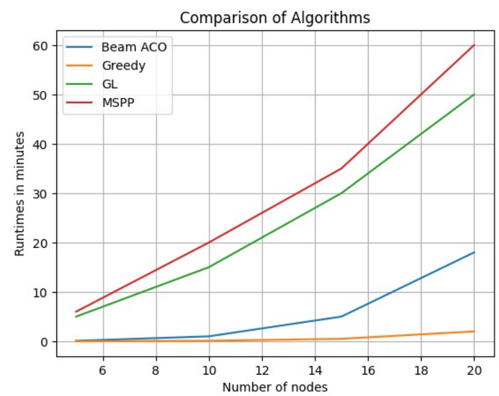

Fig. 3: Runtime Comparison

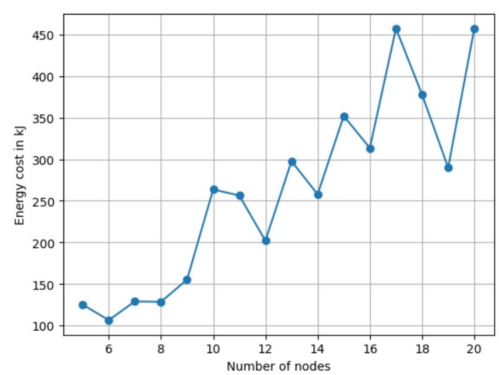

Fig. 4: Comparison with number of nodes

3 shows the running-time comparison with other algorithms used in the space. Beam-ACO has much lower running times than other implementations. shows the scalability of our algorithm. 4 shows the energy comparison with various number of randomly distributed sensors.

A. *Comparison with Baselines*

1) Greedy approach: Fig. 4. shows that our approach gives much lower energy consumption and much higher data coverage than the naive approach. The naive algorithm greedily selects the next best vertex, considering the constraints.

2) MSPP algorithm: The MSPP algorithm given in [14] uses a GAT network to model the system architecture. It then uses a twin PPO and Deep Q-Learning approach to search for the next best vertex based on a reward function. While it gives a higher data collection rate, our paper gives better energy consumption while not compromising much on the data collection.

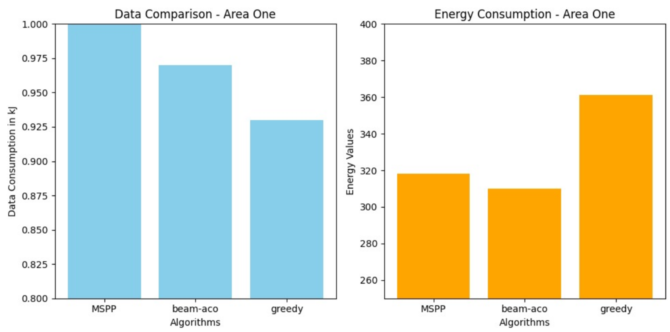

Fig. 5: Area One

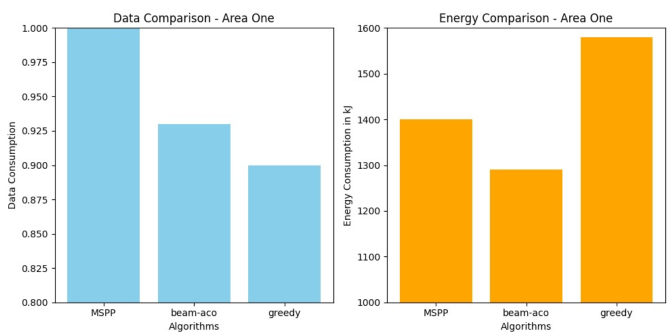

Fig. 6: Area Two

VI. Conclusion

Based on real-world datasets, this paper gives the conclusion that the application of Ant Colony Optimisation algorithm coupled with Beam Search is capable of finding the solution to the problem of data collection in wireless hetereogeneous sensor networks(HWSNs). This problem has widespread application in many real-world scenarios, including monitoring pollution levels and collecting wildlife data in remote places where only UAVs can be used to navigate data collection. Future works can further optimize data collection. The idea of multiple MSs optimally dividing the path to be visited can also be explored.

References

[1] M. . Monwar, O. Semiari, and W. Saad, "Optimized path planning for inspection by unmanned aerial vehicles swarm with energy constraints," in Proc. IEEE Global Commun. Conf. (GLOBECOM), Dec. 2018, pp. 1–6.

[2] P. Wu, C. Sha, H. Huang, and H. Wang, "Delay-sensitive trajectory designing for UAV-enabled data collection in Internet of Things," in Proc. ACM Turing Celebration Conf.-China, May 2020, pp. 77–81.

[3] Z. Jia, M. Sheng, J. Li, D. Niyato, and Z. Han, "LEO-satellite-assisted UAV: Joint trajectory and data collection for Internet of Remote Things in 6G aerial access networks," IEEE Internet Things J., vol. 8, no. 12, pp. 9814–9826, Jun. 2021.

[4] X. Jiao, W. Lou, S. Guo, N. Wang, C. Chen, and K. Liu, "Hypergraphbased active minimum delay data aggregation scheduling in wirelesspowered IoT," IEEE Internet Things J., vol. 9, no. 11, pp. 8786–8799, Jun. 2022.

[5] X. Jiao et al., "Delay efficient scheduling algorithms for data aggregation in multi-channel asynchronous duty-cycled WSNs," IEEE Trans. Commun., vol. 67, no. 9, pp. 6179–6192, Sep. 2019.

[6] X . Du, Y. Xiao, M. Guizani, and H.-H. Chen, "An effective key management scheme for heterogeneous sensor networks," Ad Hoc Netw., vol. 5, no. 1, pp. 24–34, Jan. 2007.

[7] L. Lazos and R. Poovendran, "Stochastic coverage in heterogeneous sensor networks," ACM Trans. Sensor Netw. (TOSN), vol. 2, no. 3, pp. 325–358, Aug. 2006.

[8] V. Mhatre and C. Rosenberg, "Homogeneous vs heterogeneous clustered sensor networks: A comparative study," in Proc. IEEE Int. Conf. Commun., Jun. 2004, pp. 3646–3651.

[9] Z. Lin, H.-C. Keh, R. Wu, and D. S. Roy, "Joint data collection and fusion using mobile sink in heterogeneous wireless sensor networks," IEEE Sensors J., vol. 21, no. 2, pp. 2364–2376, Jan. 2021.

[10] . Osamy, A. Salim, A. M. Khedr, and A. A. El-Sawy, "IDCT: Intelligent data collection technique for IoT-enabled heterogeneous wireless sensor networks in smart environments," IEEE Sensors J., vol. 21, no. 18, pp. 21099–21112, Sep. 2021.

[11] N. Kumar, P. Rani, V. Kumar, S. V. Athawale, and D. Koundal, "THWSN: Enhanced energy-efficient clustering approach for three-tier heterogeneous wireless sensor networks," IEEE Sensors J., vol. 22, no. 20, pp. 20053–20062, Oct. 2022.

[12] T. Palanisamy, D. Alghazzawi, S. Bhatia, A. A. Malibari, P. Dadheech, and S. Sengan, "Improved energy based multi-sensor object detection in wireless sensor networks," Intell. Autom. Soft Comput., vol. 33, no. 1, pp. 227–244, 2022.

[13] Z. Sun, S. Zou, R. Zhang, and Q. Li, "Quickest change detection in anonymous heterogeneous sensor networks," IEEE Trans. Signal Process., vol. 70, pp. 1041–1055, 2022.

[14] M. Yang, N. Liu, Y. Feng, H. Gong, X. Wang and M. Liu, "Dynamic Mobile Sink Path Planning for Unsynchronized Data Collection in Heterogeneous Wireless Sensor Networks," in IEEE Sensors Journal, vol. 23, no. 17, pp. 20310-20320, 1 Sept.1, 2023, doi: 10.1109/JSEN.2023.3294232.