

Article

A Matching Game-Based Data Collection Algorithm with Mobile Collectors

Chun Zhang ^{1,*} and Shumin Fei ²

¹ School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

² School of Automation, South and East University, Nanjing 210018, China; smfei@seu.edu.cn

* Correspondence: zhc1088@njupt.edu.cn

Received: 6 February 2020; Accepted: 1 March 2020; Published: 4 March 2020



Abstract: Data collection is one of the key technologies in wireless sensor networks. Due to the limited battery resources of sensors, mobile collectors are introduced to collect data instead of multi-hop data relay. However, how to decrease the data delay based on the cooperation of mobile collectors is a main problem. To solve this problem, a matching game-based data collection algorithm is proposed. First, some high-level cluster heads are elected. Second, by introducing a matching game model, the data collection problem is modeled as a one to one matching problem. Then, according to the preferences of mobile collectors and cluster heads, the benefit matrices are established. Based on the proposed matching algorithm, each mobile collector selects a cluster head to collect the data packets. Performance analysis proves that the matching result is stable, optimal, and unique. Simulation results show that the proposed algorithm is superior to other existing approach in terms of the reduction in data delay.

Keywords: wireless sensor networks; data collection; mobile collectors; matching game; data delay

1. Introduction

The wireless sensor network consists of a large number of sensor nodes. The sensor nodes monitor the environment, process the data packets and then send the data packets to other sensor nodes or the sink. With the development of computer network technology and sensor technology, wireless sensor networks are more and more widely used in various fields [1–3]. Whether the wireless sensor network is applied in which field, it is data-centric, that is, obtaining information as an important goal. The sink makes decisions based on the information obtained from the sensor nodes. Therefore, data collection is a key technology of a wireless sensor network.

Multi-hop transmission is commonly used in wireless sensor networks. Sensor nodes are divided into several clusters. Each cluster elects a cluster head. Ordinary nodes send information to the cluster head they are attached to. Cluster heads merge the information and send it to the sink. LEACH [4] algorithm is a classic cluster algorithm. First, at the beginning of each period, based on the number of cluster heads needed in the network and the number of times each node has been elected as the cluster head, some nodes are elected as cluster heads. Then other nodes decide which cluster heads they are attached to based on the signal strength received. A hybrid, energy-efficient, distributed clustering algorithm (HEED) [5], a stable election algorithm (SEP) [6] and a hierarchical agglomerative clustering algorithm (DHAC) [7] have improved the LEACH algorithm. In recent years, the clustering method and the method of cluster head selection are optimized, and many cluster algorithm have been proposed. However, using multi-hop transmission method, nodes consume a lot of energy in data transmission. Additionally, in many environments, the battery cannot be replaced after the nodes' power is exhausted.

References [8–11] have introduced mobile sink to collect data. In traditional methods, the sink is static, and waits to receive data packets generated by sensor nodes. In order to reduce the energy consumption for sensor nodes sending data to the sink, the mobile sink is used in some scenarios. The mobile sink can move to the vicinity of the nodes to collect data within the monitored range. Due to the constant movement of the sink, the nodes need to broadcast their own information and track the sink constantly, which also consumes a lot of energy. Therefore, many literatures propose the idea of using mobile collectors to collect data packets.

Reference [12] has proposed a data collection method based on the genetic algorithm. First, the nodes within the network are divided into several clusters. Then the overlaps of the communication ranges are found. Mobile collectors move to these locations to collect data packets, and an algorithm is proposed to calculate the paths for them. The defect of the algorithm is that the data packets are sent to the sink after all the data packets have been collected, so the data delay is long. When there are multiple mobile collectors in the network, the delay can be reduced compared to only one mobile collector. The goal of reference [13] is to plan the paths of multiple mobile collectors and minimize the data delay. There is a probability that a region is visited by different mobile collectors, which results in the waste of resources. The reference [14] has divided the data collection problem into three problems: task scheduling, path calculation, and speed control. The greedy algorithm is introduced to assign tasks to mobile collectors. The mobile collectors need to visit each node, therefore the data delay is long. The goal of reference [15] is to select the optimal collection point. First a node sets itself as the collection point. Then it sends information to neighbor nodes within its multi-hop range, including its identification, the number of neighbor nodes and the distance from the sink. When it finds a node closer to the sink and having more neighbor nodes than it, it marks the node as a collection point. This algorithm is suitable for the environment where nodes are evenly distributed. When nodes are randomly distributed, it is difficult to get the optimal solution. The problem solved by reference [16] is how to plan a path for each mobile collector. This problem is defined as an optimization problem and solved by a linear programming approach. Reference [17] considers a scenario where an unmanned aerial vehicle collects data from a set of sensors on a straight line. Reference [18] proposes a fault tolerant algorithm for data aggregation to plan the itinerary for a mobile agent and another alternative itinerary in case of sensor nodes failure.

In this paper, the wireless sensor network is distributed in the monitored area. The sensor nodes are event-driven. The problem to be solved in this paper is, how multiple mobile collectors collaborate to collect event data packets generated by sensor nodes with the minimum data delay.

Game theory studies the decision-making behavior when decision-makers interact with each other. Game theory researches how individuals make decisions and how they are influenced by other players. Therefore, each participant's decision includes the decisions of others. In general, game theory attempts to open the black box of individual decision-making.

The innovations of this paper include the following aspects.

- (1) The matching game model in game theory is introduced. The problem of cooperative data collection by multiple mobile collectors is modeled as the matching problem between mobile collectors and high-level cluster heads.
- (2) According to the preferences of mobile collectors and high-level cluster heads, their benefit ranking matrices are built. Then, a one-to-one matching algorithm is proposed.
- (3) Theoretical analysis proves that the matching result between mobile collectors and high-level cluster heads is stable, optimal and unique.

2. Data Collection Method Based on Matching Game Theory

2.1. Preliminaries

The following assumptions are made for our considered network environment in this paper.

- (1) Sensor nodes are distributed in the area of interest randomly, and all the sensors are static.

- (2) The positions of sensor nodes are obtained by GPS or existing positioning algorithms. All the sensor nodes have homogeneous capabilities, such as the sensing range, communication range, fusion power and ability of localization.
- (3) The sensor nodes are event-driven. They generate event packets when they detect events of interest to the sink. Then the event packets are sent to the sink by mobile collectors. The sink is static.
- (4) The network topology is connected.

Some definitions are given as follows.

Mobile collector: A mobile collector is a power unit which can move freely and carries radio frequency transceivers, which is expressed by g_i , where i is the identification of the mobile collector.

Visit: For any sensor node S_n , if a mobile collector communicates with it and receives data packets from it, it is called that S_n is visited by the mobile collector.

High-level cluster head: In the three-layer sensor network, sensor nodes are separated to play different roles, such as cluster heads and ordinary nodes. First, all nodes are grouped into low-level clusters and the low-level cluster heads are elected by a cluster algorithm. Then, the low-level cluster heads are organized into high-level clusters and high-level cluster heads are elected. A high-level cluster head is expressed by v_j , where j is the identification of the high-level cluster head.

Request sequence set: During a collection cycle, the data packets generated by ordinary sensor nodes are relayed to high-level cluster heads through low-level cluster head. When a high-level cluster head has received these data packets, it will send a request message to the sink. The request message received by the sink during a collection cycle constitutes a set named RG, which is expressed as Equation (1), where $RG_n(t_n, \varepsilon_n, (x_n, y_n))$ is the request information sent by the high-level cluster head named v_n , t_n is the time when the data packets received by v_n were generated, ε_n is the priority of these data packets, x_n and y_n are the abscissa and ordinate of v_n , respectively. There are two cases. In one case, the generation times and priorities of the data packets received by v_n are the same. Then the generation time and priority are the values of t_n and ε_n . In another case, the generation times and priorities of the data packets received by v_n are different. Then t_n is defined as the minimum of these generation times, and ε_n is defined as the maximum of these priorities.

$$RG = \{RG_a(t_a, \varepsilon_a, (x_a, y_a)), RG_b(t_b, \varepsilon_b, (x_b, y_b)), \dots, RG_k(t_k, \varepsilon_k, (x_k, y_k))\} \quad (1)$$

2.2. System Model

A wireless sensor network consists of many sensor nodes deployed in a monitored region. Each sensor node has a communication range r_c , which is a circle with the position of the node as the center and r_c as the radius. Each sensor node consists of four components: sensing unit, microcontroller unit, radio unit, and a battery. The topology of the sensor network is represented by the graph $\Gamma = (K, Z)$, where K represents the set of sensor nodes and Z represents the set of wireless links. For any two different nodes $S_1, S_2 \in K$, we say the wireless link $(S_1, S_2) \in Z$ if $d(S_1, S_2) \leq r_c$; otherwise $(S_1, S_2) \notin Z$, where $d(S_1, S_2)$ is the Euclidean distance between S_1 and S_2 . In order to form a topology for a wireless sensor network, the medium access control (MAC) protocol proposed in reference [19], named self-organizing medium access control for sensor networks (SMACS), is used in this paper. By SMACS [19], the sensor nodes can discover their neighbors and establish transmission/reception schedules for communicating with them. After the topology of the wireless sensor network is formed, mobile collectors can communicate with sensor nodes. Since the sensor nodes use time division multiple access (TDMA) technology, some time slots are reserved for mobile collectors to send invitations. When a mobile collector needs to communicate with a node, it moves to the communication range of this node, sends an invitation to the node and waits for a reply from the node. If the node is unwilling to establish a connection with the collector, it sends a rejection message to the mobile collector. If the node accepts the invitation from the mobile collector, it sends a reply to the collector. Then the collector

is registered at the node, and the node allocates some time slots for the communication between them. When the communication between them ends, the collector informs the node to release the connection.

The problem of data collection can be solved by the following steps.

- (1) The wireless sensor network is grouped into three-layer clusters by a clustering algorithm.
- (2) During a collection cycle, the request sequence set RG is sent to each collector by the sink. Based on the matching game theory, each collector selects a high-level cluster head.
- (3) Each collector moves to the communication range of a high-level cluster head and visits it. Then the collector sends the data packets which it has received to the sink.

The step 1 can be solved by a clustering algorithm. Many literatures have proposed clustering algorithms, therefore the clustering algorithm is not the focus of this paper. For example, the self-organization clustering algorithm (SOC) proposed in reference [20] can be used in this paper. Architecture of the wireless sensor network is given in Figure 1. In step 3, the collectors communicate with high-level cluster heads and the sink, and relay data packets from high-level cluster heads to the sink. According to the above analysis, steps 1 and 3 are relatively easy to implement, so the focus of this paper is step 2.

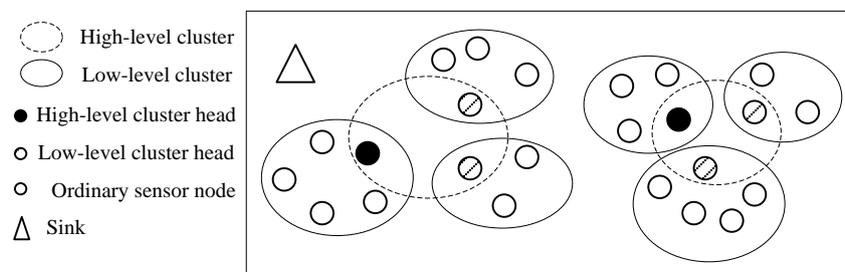


Figure 1. Architecture of the wireless sensor network.

A system that applies a one-to-one matching game model must satisfy at least two important assumptions.

- (1) From the beginning of the game, participants must belong to two disjoint sets. For example, the sets of the two sides are S and T , respectively, and satisfy Equation (2).

$$S \cap T = \emptyset \quad (2)$$

- (2) A match can only be formed after the agreement of both sides.

A number of mobile collectors are randomly scattered in the monitored area. During each collection cycle, each collector selects a high-level cluster head to visit by a matching-game based algorithm. This process is accomplished by the communication between collectors and high-level cluster heads. Let $\mathbf{G} = \{g_1, g_2, \dots, g_n\}$ and $\mathbf{V} = \{v_1, v_2, \dots, v_m\}$ denote the set of collectors and the set of high-level cluster heads which have received data packets from low-level clusters, respectively. Thus, their relationship satisfies Equation (3). The benefit matrices of the two sets \mathbf{G} and \mathbf{V} are represented by $\mathbf{C}_{M \times N}$ and $\mathbf{H}_{M \times N}$, respectively, where N and M represent the number of elements in \mathbf{G} and \mathbf{V} , respectively. \mathbf{G} and \mathbf{V} make decisions based on $\mathbf{H}_{M \times N}$ and $\mathbf{C}_{M \times N}$.

$$\mathbf{G} \cap \mathbf{V} = \emptyset \quad (3)$$

Suppose the symbol $>$ represents a strict preference relationship. For \mathbf{G} , x, y, z represent three options. If \mathbf{G} prefers x to y , it can be expressed as $x > y$. For \mathbf{G} , the following three conclusions hold.

- (1) The expression $x > y$ holds or the expression $y > x$ holds. Thus $>$ is complete.

- (2) If the expressions $x > y$ and $y > z$ hold, the expression $x > z$ holds. Thus $>$ is transferable.
 (3) $>$ is complete and transferable, therefore $>$ is rational.

Similarly, the preference of \mathbf{V} is rational. Therefore, \mathbf{G} and \mathbf{V} are two disjoint, rational sets. Furthermore, the problem studied in this paper can be modeled as a one-to-one matching problem between collectors and high-level cluster heads which have received data packets from low-level clusters.

Each mobile collector ranks the high-level cluster heads in the order of its preference. Suppose that the set of preferences of a mobile collector g_n on the set \mathbf{V} is represented by $\mathbf{P}(g_n)$. For example, Equation (4) shows that, g_n ranks v_1 first, v_5 second, v_8 the last. This indicates that g_n most wants to match v_1 , followed by v_5 , and finally v_8 , which can be expressed as $v_1 >_{g_n} v_5, v_5 >_{g_n} v_8$. Similarly, the set of preferences of a high-level cluster head v_n on the set \mathbf{G} is represented by $\mathbf{P}(v_n)$. Equation (5) shows that, v_n ranks g_1 first, g_3 second, g_{10} the last. The preference set of two sides is represented by \mathbf{P} . As shown in Equation (6), a two-sided matching market is expressed as $(\mathbf{G}, \mathbf{V}; \mathbf{P})$.

$$\mathbf{P}(g_n) = v_1, v_5, \dots, v_8 \quad (4)$$

$$\mathbf{P}(v_n) = g_1, g_3, \dots, g_{10} \quad (5)$$

$$\mathbf{P} = \{\mathbf{P}(g_1), \mathbf{P}(g_2), \dots, \mathbf{P}(g_n); \mathbf{P}(v_1), \mathbf{P}(v_2), \dots, \mathbf{P}(v_m)\} \quad (6)$$

Theorem 1. *The matching function $\mu : (\mathbf{G} \cup \mathbf{V}) \rightarrow (\mathbf{G} \cup \mathbf{V})$: If $\mu(g) \neq g$, then $\mu(g) \in \mathbf{V}$. If $\mu(v) \neq v$, then $\mu(v) \in \mathbf{G}$. $\mu(x)$ is called the matching object of x . The set of matched elements is represented by $\mu(\mathbf{G}, \mathbf{V})$.*

Theorem 2. *The kernel of a matching game is equal to the set of stable matches.*

When researching the matching game, one critical assumption is that the matching process is voluntary. An element can send a matching invitation to another element. It also can reject a matching invitation from another element. If a match process is not implemented, it is said to be blocked. This assumption determines the core problem of the matching process between \mathbf{G} and \mathbf{V} : the stability of the matching process.

If a match μ is upset by another match $\tilde{\mu}(g, v)$, $\tilde{\mu}$ is better than μ and μ is unstable. Therefore, if μ is not in the kernel, it must be blocked by another match $\tilde{\mu}$. For any $g \in \mathbf{G}$, if the expression $\tilde{\mu}(g) \in \mathbf{V}$ is valid, $\tilde{\mu}(g)$ is better than $\mu(g)$. Meanwhile, suppose the expression $\tilde{\mu}(g) = v$ is valid. Then v considers g to be better than $\mu(v)$. In brief, matches that are not in the kernel must be unstable, and unstable matches must not be in the kernel.

Therefore, the problem studied in this paper is equivalent to finding a stable matching set for the bilateral matching problem $(\mathbf{G}, \mathbf{V}; \mathbf{P})$.

2.3. Benefit Matrix

The stable matching set of $(\mathbf{G}, \mathbf{V}; \mathbf{P})$ is related to the benefit matrices of high-level cluster heads $\mathbf{H}_{M \times N}$ and mobile collectors $\mathbf{C}_{M \times N}$, which is expressed by Equations (7) and (8), where M is the number of high-level cluster head, and N is the number of collectors. The calculation method of $\mathbf{H}_{M \times N}$ and $\mathbf{C}_{M \times N}$ is introduced in this section.

$$\mathbf{H}_{M \times N} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1N} \\ h_{21} & h_{22} & \cdots & h_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ h_{M1} & h_{M2} & \cdots & h_{MN} \end{bmatrix} \quad (7)$$

$$\mathbf{C}_{M \times N} = \begin{bmatrix} c_{11} & c_{21} & \cdots & c_{M1} \\ c_{12} & c_{22} & \cdots & c_{M2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1N} & c_{2N} & \cdots & c_{MN} \end{bmatrix} \quad (8)$$

Reference [6] has proposed the energy consumption model of sensor nodes. When the transmitter sends L -bit information to the receiver with a distance of d , the energy consumed is expressed by Equation (9). Where δ_{fs} is the amplification coefficient, and δ_{mp} is the multipath fading coefficient. The radio dissipates χ_{elec} per bit to run the radio circuitry. We find that, the farther the sender and receiver are, the more energy they consume while communicating. In addition, each sensor node has a fixed communication range. When a collector wants to communicate with a cluster head, if it is not within the communication range of the cluster head, it needs to move to the range first. The farther it is from the cluster head, the longer it needs to move, and the greater the data delay. If it is within the communication range of the cluster head, it can communicate with the cluster head without moving. The closer they are, the less energy the cluster head consumes when communicating with it. Therefore, no matter from the perspective of energy consumption or data delay, the benefit of a high-level cluster head is mainly affected by the distance to the matched collector. If a high-level cluster head v_m selects a collector g_i , the benefit of v_m is represented by Equation (10), where d_{im} is the distance between g_i and v_m , and θ is the weight of the distance factor.

$$T_x(L, d) = \begin{cases} L\chi_{elec} + L\delta_{fs}d^2, & d \leq d_0 \\ L\chi_{elec} + L\delta_{mp}d^4, & d > d_0 \end{cases} \quad (9)$$

$$h_{mi} = \theta/d_{mi} \quad (10)$$

When a collector selects a high-level cluster head to visit, it considers the following three factors: (1) the time when the data packet was generated; (2) the priority of the data packet; (3) the distance between them. If a collector g_n selects a high-level cluster head v_j , the benefit of g_n is represented by Equation (11), where $t_j, \varepsilon_j, (x_j, y_j)$ can be obtained by querying $RG_j(t_j, \varepsilon_j, (x_j, y_j))$ in request sequence set RG , d_{jn} is the distance between g_n and v_j and α, β, γ are weights of the three factors.

$$c_{jn} = \frac{\alpha}{t_j} + \frac{\beta}{d_{jn}} + \gamma\varepsilon_j \quad (11)$$

Analytic hierarchy process (AHP) is a mathematical-based technology to derive the deciding factors for complex problems [21]. According to AHP, the decision maker indicates the strength of preference by the pairwise comparison between deciding factors. By answering the two questions "Which of the two is more important?" and "By how much?" the pairwise comparison is finished [22]. In this paper, the pairwise comparison results are expressed by a matrix \mathbf{Q} in Equation (12).

$$\mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{bmatrix} = \begin{bmatrix} \alpha/\alpha & \alpha/\beta & \alpha/\gamma \\ \beta/\alpha & \beta/\beta & \beta/\gamma \\ \gamma/\alpha & \gamma/\beta & \gamma/\gamma \end{bmatrix} \quad (12)$$

The square matrix \mathbf{Q} satisfies the Equation (13), where λ is characteristic value, and the nonzero vector z is called the eigenvector of \mathbf{Q} corresponding to λ . Constructing a weight vector \mathbf{L} contains three elements of α, β, γ , as shown in Equation (14). \mathbf{L} satisfies the Equation (15), which is proved in Equation (16). Therefore, the weights of the three factors can be obtained by computing the eigenvector of \mathbf{Q} when the characteristic value λ is equal to 3. That is, to solve the homogeneous linear Equation (17), where \mathbf{E} is the unit matrix.

$$\mathbf{Q}z = \lambda z \quad (13)$$

$$(\mathbf{L})^T = [\alpha \quad \beta \quad \gamma] \quad (14)$$

$$\mathbf{QL} = 3\mathbf{L} \quad (15)$$

$$\begin{bmatrix} \alpha/\alpha & \alpha/\beta & \alpha/\gamma \\ \beta/\alpha & \beta/\beta & \beta/\gamma \\ \gamma/\alpha & \gamma/\beta & \gamma/\gamma \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} 3\alpha \\ 3\beta \\ 3\gamma \end{bmatrix} = 3 \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \quad (16)$$

$$(\mathbf{Q} - 3\mathbf{E})\mathbf{L} = 0 \quad (17)$$

2.4. The One-to-One Matching Algorithm

After the benefit matrices are built, a one-to-one matching algorithm is proposed in this section. Some definitions are given as follows.

Theorem 3. $\varphi(i)$: The identification of the high-level cluster head which pre-matches with g_i ;

Theorem 4. $\phi(j)$: The identification of the collector which pre-matches with v_j ;

Theorem 5. $D(i)$: The matching state vector of g_i ;

Theorem 6. $R(j)$: The matching state vector of v_j ;

Theorem 7. $m(i)$: A pointer to an element in $C_{1 \times M}^i$.

The steps of the one-to-one matching algorithm are as follows.

Step 1: During each collection cycle, for any collector $g_i \in G$, it computes the benefit matrix $C_{1 \times M}^i$, which is expressed as Equation (18), where c_{Mi} is defined by Equation (11). There is a special case. If a high-level cluster head has not received any data packet, the benefit for g_i selecting it is equal to 0. For any high-level cluster head $v_j \in V$, it computes the benefit matrix $H_{1 \times N}^j$, which is expressed as Equation (19), where h_{jN} is defined by Equation (10). There is also a special case. If a high-level cluster head has not received any data packet, the benefit for it selecting any collector is equal to 0.

$$C_{1 \times M}^i = [c_{1i} \quad c_{2i} \quad \cdots \quad c_{Mi}] \quad (18)$$

$$H_{1 \times N}^j = [h_{j1} \quad h_{j2} \quad \cdots \quad h_{jN}] \quad (19)$$

Step 2: For any collector $g_i \in G$, it arranges the elements in $C_{1 \times M}^i$ in descending order according to their values. Then the high-level cluster heads are sorted according to the arrangement result, which is represented by Equation (20). That is, e_1 represents the high-level cluster head which benefits g_i the most, e_2 represents the second, and so on. Initially, g_i sends invitations to the high-level cluster head represented by e_1 .

$$\mathbf{E} = [e_1 \quad e_2 \quad \cdots \quad e_M] \quad (20)$$

Step 3: Assuming the high-level cluster head which receives the invitation from g_i is v_j . The following situations are discussed.

(1) If v_j has not pre-matched any collector, $R(j)$ satisfies the Equation (21). The following two situations are discussed.

$$R(j) = 0 \quad (21)$$

(i) At the same time, if v_j only receives the invitation from g_i , v_j accepts this invitation. Then, $R(j)$, $D(i)$, $\phi(j)$ and $\varphi(i)$ are expressed by Equations (22)–(25), respectively.

$$R(j) = 1 \quad (22)$$

$$D(i) = 1 \quad (23)$$

$$\phi(j) = i \quad (24)$$

$$\varphi(i) = j \quad (25)$$

(ii) At the same time, if v_j receives invitations from multiple collectors, it inquires about $H_{1 \times N}^j$, and selects the collector which benefits v_j the most. If this collector is g_i , v_j accepts the invitation from g_i . Otherwise, if this collector is not g_i , assuming g_a , v_j accepts the invitation from g_a , and refuses the invitations from other collectors. After g_i is refused by v_j , it, in turn, invites the cluster heads arranged behind v_j in \mathbf{E} . Then repeat step 3 until each cluster head has pre-matched with a collector.

(2) If v_j has been pre-matched a collector, assuming g_b , $R(j)$ and $\phi(j)$ must satisfy the Equations (22) and (26). The following two situations are discussed.

$$\phi(j) = b \quad (26)$$

(i) if h_{jb} and h_{ji} satisfy Equation (27), v_j notifies g_b to cancel the original pre-match $\mu(g_b, v_j)$, and accepts the invitation from g_i . Then the pre-match $\mu(g_i, v_j)$ is established, and $D(b)$ is expressed by Equation (28).

$$h_{jb} < h_{ji} \quad (27)$$

$$D(b) = 0 \quad (28)$$

(ii) If h_{jb} and h_{ji} satisfy Equation (29), v_j refuses the invitations from g_i . Then g_i invites the cluster heads arranged behind v_j in \mathbf{E} in turn. Then repeat step 3 until each cluster head has pre-matched a collector.

$$h_{jb} > h_{ji} \quad (29)$$

THEOREM 1 According to the matching algorithm proposed in this paper, there must exist a set of stable matches for the matching problem between \mathbf{G} and \mathbf{V} , and the matching result is stable, optimal and unique.

Proof. According to the matching algorithm proposed in this paper, initially, each collector sends invitations to the high-level cluster head which benefits it the most. If a cluster head receives invitations from multiple collectors, it selects one collector which benefits it the most to accept its invitation, and rejects other collectors. Then these rejected collectors send invitations to their second preferred cluster heads. The algorithm terminates when all cluster heads are matched. Since the number of cluster heads is limited and the number of collectors is greater than or equal to the number of cluster heads, each cluster head must be able to find a collector to match it. Additionally, since the matching process is voluntary, the matching result must be consistent with the individual rationality. \square

The next step is to prove that the matching result μ obtained by the matching algorithm is stable. Suppose g_i and v_i satisfy Equations (30) and (31). g_i and v_i prefer to form a match. That is, the Equations (32) and (33) are satisfied. Thus g_i will send an invitation to v_i . The following two situations are discussed. (i) if v_i has not pre-matched any collector, v_i will accept the invitation from g_i . If it receives an invitation from $\mu(v_i)$ later, it will refuse this invitation. (ii) if v_i has had a pre-matching collector, v_i will refuse this original pre-matching object and pre-match g_i . g_i and v_i have not formed a matching pair until the matching algorithm is finished. Thus v_i must refuse the invitation from g_i , and the match (g_i, v_i) cannot block μ . Therefore, the matching result μ is stable.

$$\mu(g_i) \neq v_i \quad (30)$$

$$\mu(v_i) \neq g_i \quad (31)$$

$$g_i \succ_{v_i} \mu(v_i) \quad (32)$$

$$v_i \succ_{g_i} \mu(g_i) \quad (33)$$

Then prove that the matching result μ is optimal. Suppose there is another matching result $\tilde{\mu}$ better than μ . Then there is at least one cluster head satisfying the inequality (34), and $(\tilde{\mu}(v_i), v_i)$ must block μ . So μ is unstable. As previously proved, μ is stable. Hence this assumption does not exist, and μ is optimal.

$$\tilde{\mu}(v_i) \succ_{v_i} \mu(v_i) \quad (34)$$

Finally, prove that the matching result μ is unique. As previously proved, μ is optimal. Suppose there is another matching result $\tilde{\mu}$, and $\tilde{\mu}$ is another optimal match different from μ . Then there is at least one cluster head v_i satisfying the inequality in Equation (35). Since μ is optimal, the inequality in Equation (36) holds. Since $\tilde{\mu}$ is optimal, the inequality in Equation (34) holds. Since \succ means strict preference, the relationship between $\tilde{\mu}(v_i)$ and $\mu(v_i)$ must satisfy Equation (37). This conclusion contradicts the assumption that $\tilde{\mu}$ is different from μ . Therefore, μ is unique.

$$\tilde{\mu}(v_i) \neq \mu(v_i) \quad (35)$$

$$\mu(v_i) \succ_{v_i} \tilde{\mu}(v_i) \quad (36)$$

$$\tilde{\mu}(v_i) = \mu(v_i) \quad (37)$$

In conclusion, this matching result μ obtained by the matching algorithm proposed in this paper is stable, optimal and unique.

The pseudo code of the matching algorithm is given as follows.

Algorithm: The one-to-one matching algorithm

```

1 for  $g_i \in G$  do
2    $C_{1 \times M}^i = [ c_{1i} \ c_{2i} \ \dots \ c_{Mi} ]$ 
3    $\mathbf{E} = [ e_1 \ e_2 \ \dots \ e_M ]$ 
4    $\text{sort}(C_{1 \times M}^i, \text{'descend'}) = \mathbf{E}$ 
5    $D(i) = 0$ 
6  $m(i) = 1$ 
7 end for
8 for  $v_j \in V$  do
9    $H_{1 \times N}^j = [ h_{j1} \ h_{j2} \ \dots \ h_{jN} ]$ 
10  $R(j) = 0$ 
11 end for
12  $n = 0$ 
13 while  $n < M$  do
14   for  $g_i \in G$  do
15     if  $D(i) = 0$  then
16        $g_i$  invite the cluster head  $v_j$ 
17       if  $R(j) = 0$  then
18          $\varphi(i) = j$ 
19          $\phi(j) = i$ 
20          $D(i) = 1$ 
21          $R(j) = 1$ 
22          $n = n + 1$ 
23          $m(i) = 1$ 
24       else
25         if  $h_{jb} > h_{ji}$  then
26           if  $m(i) < M$ 
27              $D(i) = 0$ 
28              $m(i) = m(i) + 1$ 
29         else

```

```

30          $g_i$  doesn't invite any cluster head
31     else
32          $\varphi(i) = j$ 
33          $D(b) = 0$ 
34          $D(i) = 1$ 
35          $m(i) = 1$ 
36     else
37          $g_i$  doesn't invite any cluster head
38     end if
39 end if
40 end if
41 end if
42 end for
43 end while

```

3. Simulation Results

The performance of the matching algorithm proposed in this paper is evaluated by simulations in this section. In our simulations, sensor nodes are distributed uniformly in a square field without obstacles. The sink is located in the center of the area. Sensor nodes are event triggered. The parameter values used in the simulation are listed in Table 1. The simulations are performed using MATLAB R2017a. The communication range of each sensor node is set to 30 m. When a collector needs to communicate with a cluster head, if it is not within the communication range of the cluster head, it moves to the range first. Then it stops and sends an invitation message to the cluster head. If the cluster head accepts the invitation, a connection will be established between them. Then the cluster head starts sending data packets to the collector. After the collector has received the data packets, it notifies the cluster head to release the connection. Then the collector moves to another collection point. The matching game-based data collection algorithm for wireless sensor networks proposed in this paper is called MGDC. The matching game-based data collection algorithm (MGDC) is compared with the region based data collection algorithm (SOC-RDC) and the time based data collection algorithm (SOC-TDC) proposed in [20].

Table 1. Simulation parameters.

Parameters	Values
radio bandwidth of sink (Mbps)	1
sensing range of a sensor node (m)	20
transmission range of a mobile collector (m)	35
speed of a mobile collector (m/s)	5
event packet size (bytes)	100
request packet size (bytes)	25
packet head size (bytes)	25
simulation time (s)	40,000

In the simulations shown in Figures 2–6, Equation (38) is satisfied. Figure 2 shows the relationship between the average tour length of collectors and the size of the monitored area during the simulation time. In the simulation, the number of collectors is 8, and the number of sensor nodes is 100. It can be observed from the figure that, with MGDC, the average tour length of collectors is smaller than that of SOC-RDC and SOC-TDC, and with the increase of the size of the monitored area, the gap between them increases. There are two reasons for this result. One reason is that, using SOC-TDC, the data packets generated at the same time are collected by one collector. These data packets may be scattered over different regions, so the collector moves a long distance to collect them. According to the MGDC, during a collection cycle, a collector only visit a high-level cluster head, so it moves a

short distance. Another reason is, according to SOC-RDC, data packets generated at the same time are collected by several collectors, not one collector. Some collectors may select the same region. Under this circumstance, the sink randomly selects one of these collectors to collect data packets in this region. The choice may not be optimal. However, the matching result of MGDC between collectors and high level cluster heads is optimal. With the increase of the size of the monitored area, the distribution of sensor nodes is becoming more and more dispersed, so the advantages of MGDC are becoming more and more obvious.

$$\alpha:\beta:\gamma = 1:1:1 \quad (38)$$

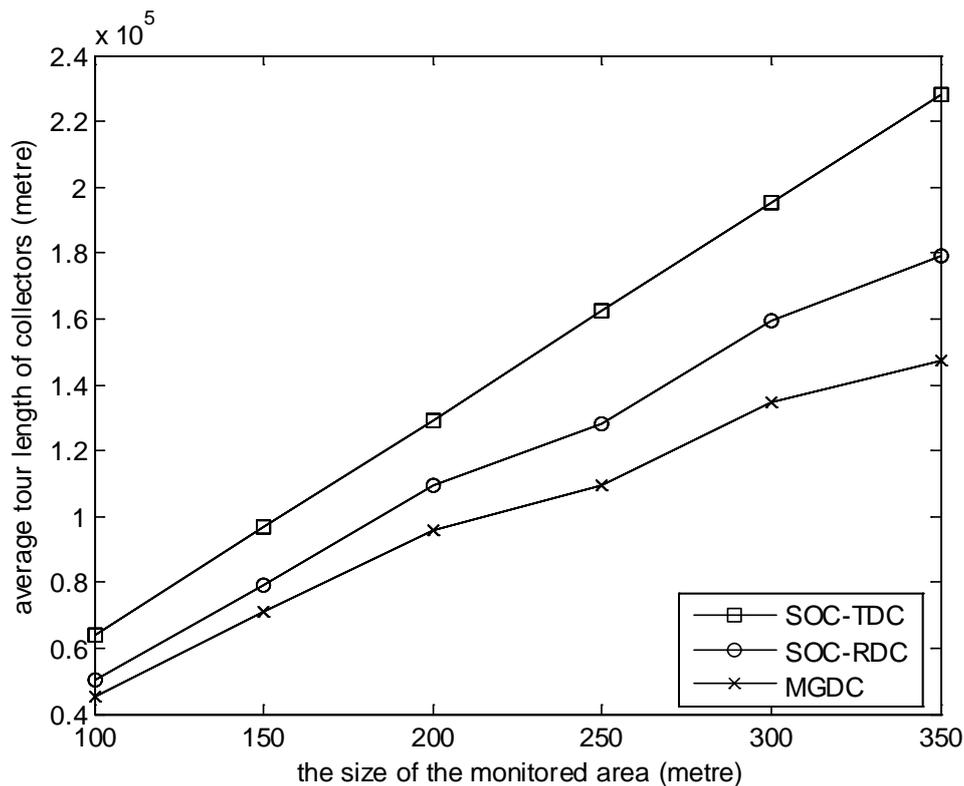


Figure 2. The size of the monitored area versus average tour length of collectors.

Figure 3 shows the relationship between the average data delay and the size of the monitored area during the simulation time. Data delay is the time difference between the data packet generated and the data packet collected by a collector. In the simulation, the number of collectors is 8, and the number of sensor nodes is 100. It can be seen from the figure that the average data delay of MGDC is smaller than that of SOC-RDC and SOC-TDC. And as the size of the monitored area increases, the gap between them increases. One reason is that, using MGDC, the average tour length of collectors is smaller than that of SOC-RDC and SOC-TDC. Another reason is that, using MGDC, the matching process between collectors and cluster heads takes into account several factors, such as time and position, rather than considering a single factor.

Figure 4 focuses on the effect of the number of collectors on the average tour length of collectors. In this experiment, the number of sensors is 100, and the size of the monitored area is 100 m. We can observe that when the number of collectors is small, the advantage of MGDC is not obvious. This is because, during the simulation time, when the number of high-level cluster heads which have received data packets is greater than the number of collectors, one collector needs to visit several cluster heads. The average tour length of collectors is long. Furthermore, when the number of collectors is small, the options of cluster heads are few, and the difference between SOC-RDC and MGDC is not obvious. However, when the number of collectors increases, the combinations between cluster

heads and collectors increase, and the advantages of MGDC become more obvious. Since the data packets generated at the same time are collected by one collector according to SOC-TDC, as the number of collectors increases, some collectors are idle. Hence, for SOC-TDC, as more collectors are added, the impact on average tour length of collectors is not significant.

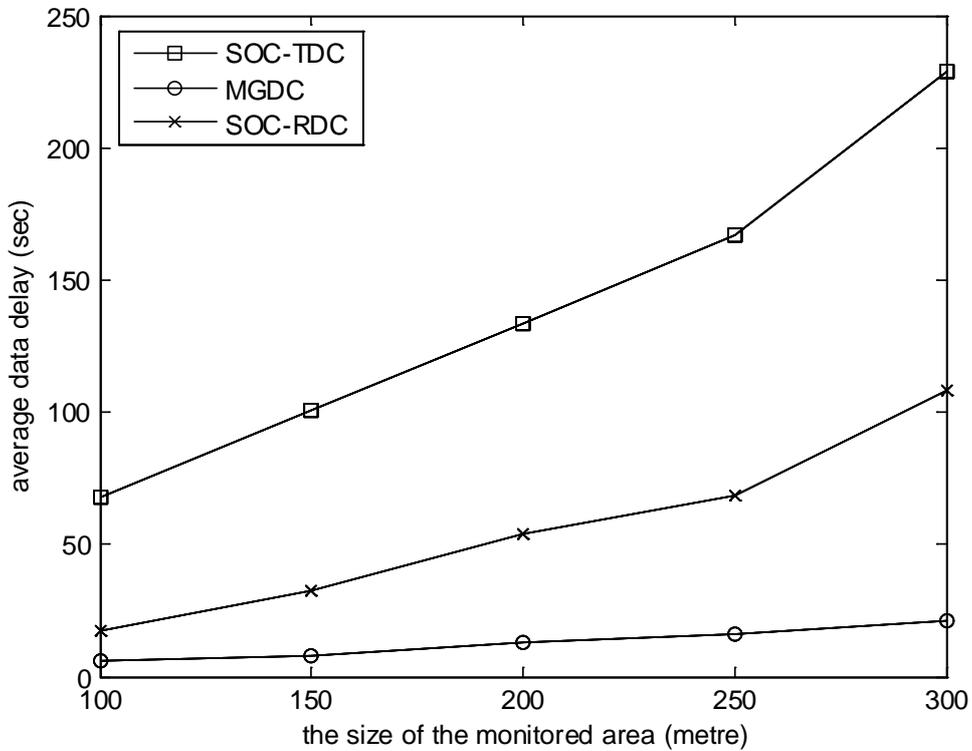


Figure 3. The size of the monitored area versus average data delay.

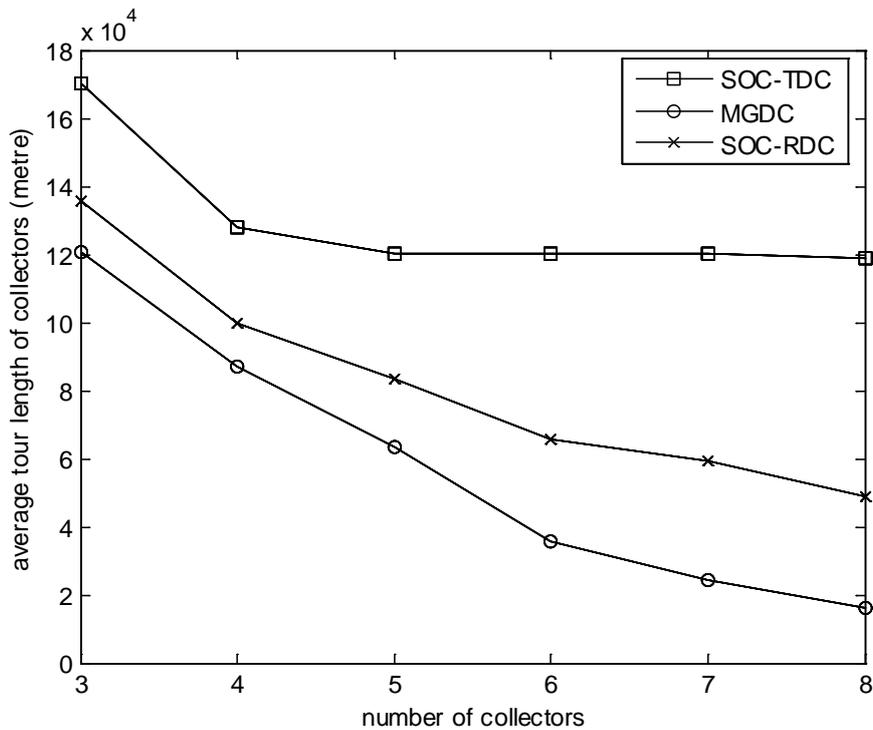


Figure 4. Number of collectors versus average tour length of collectors.

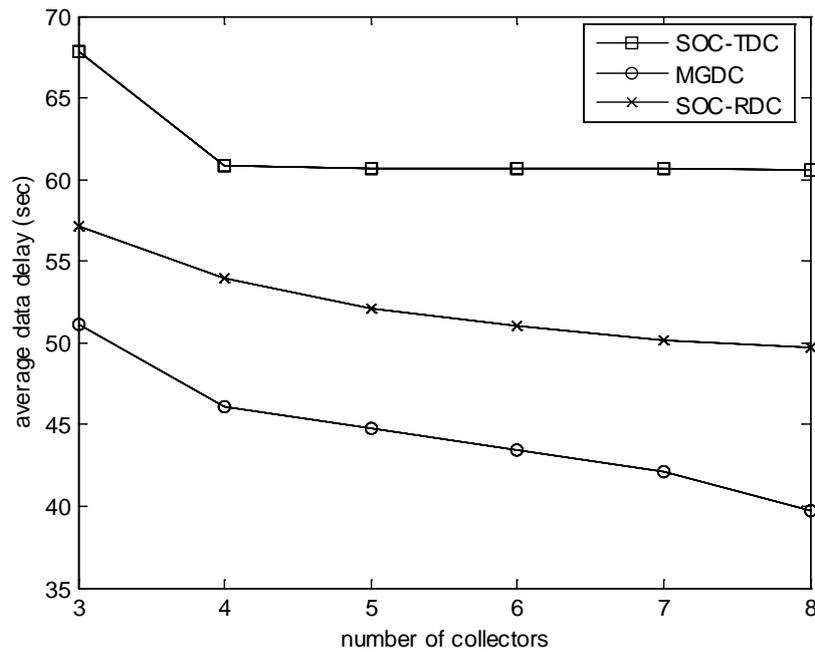


Figure 5. Number of collectors versus average data delay.

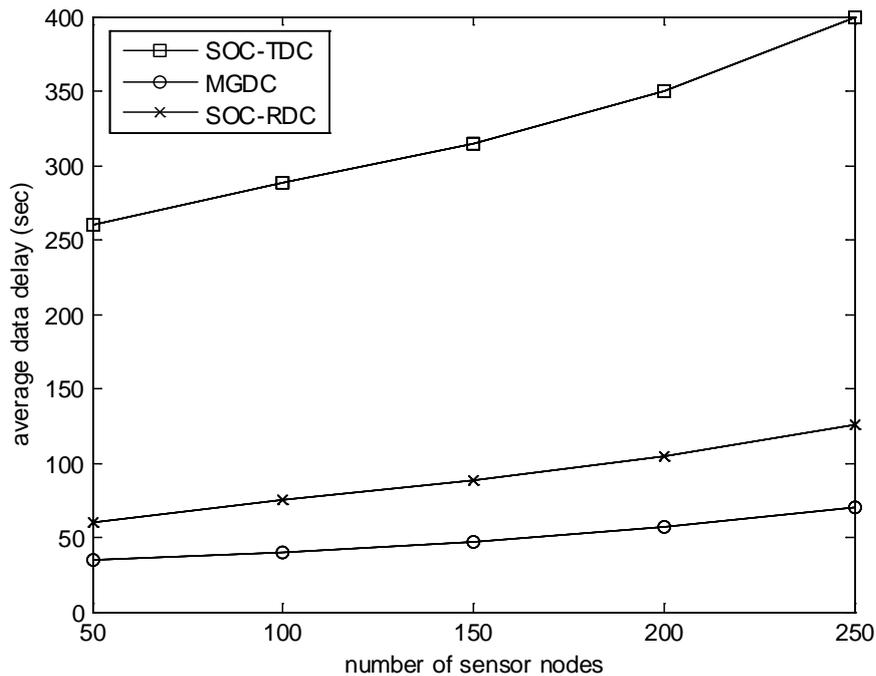


Figure 6. Number of sensor nodes versus average data delay.

Figure 5 highlights the effect of the number of collectors on the average data delay. In this experiment, the number of sensors is 50, and the size of the monitored area is 100 m. We can find that MGDC performs better than SOC-RDC and SOC-TDC. The more the number of collectors, the more obvious the advantage of MGDC.

Figure 6 evaluates the effect of the number of sensor nodes on the average data delay. In this experiment, the number of collectors is 6, and the size of the monitored area is 400 m. As illustrated in Figure 6, when the number of sensor nodes increases, the average data delay increases. The reason is that, when the number of nodes is small, the number of high-level cluster heads is small, and there are few optional objects for collectors. The gap between MGDC and SOC-RDC is not obvious. With ixithe

increase of the number of sensor nodes, the number of clusters and nodes which have monitored events increases. There are more optional objects for collectors, and the gap between MGDC and SOC-RDC becomes more and more obvious.

Figures 7 and 8 investigate the effects of the weights of three factors of MGDC. In the simulations, the number of collectors is 8. It is observed from the graph in Figure 7 that the higher the proportion of β , the shorter the average moving distance. This is because, when the proportion of β is high, according to the benefit of the collector obtained from Equation (11), collectors will give priority to the near cluster head. As illustrated in Figure 8, the higher the proportion of α , the smaller the data delay. The reason is that, when the proportion of α is high, according to the benefit of the collector obtained from Equation (11), the cluster head which has collected the data packets with the earlier generation time will be the first choice of collectors.

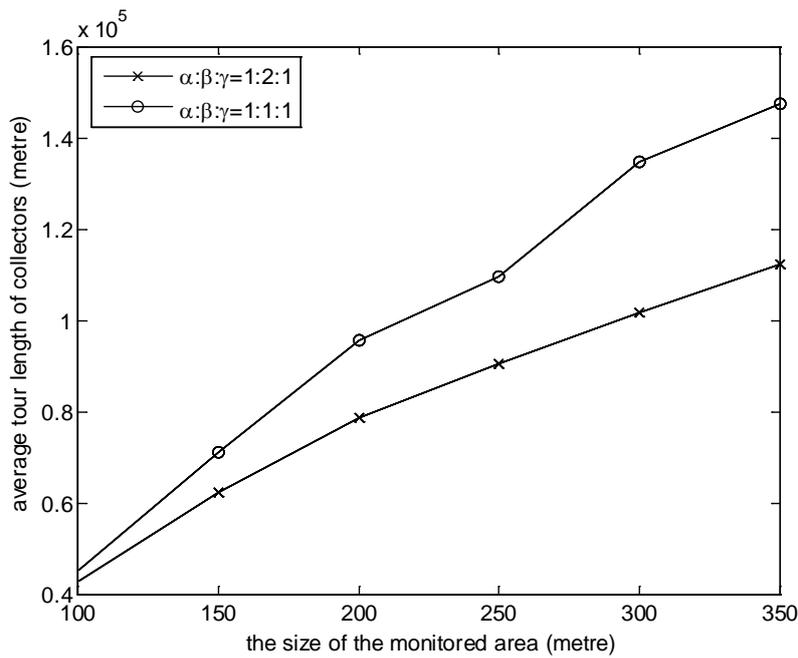


Figure 7. The effect of the weights on average tour length of collectors.

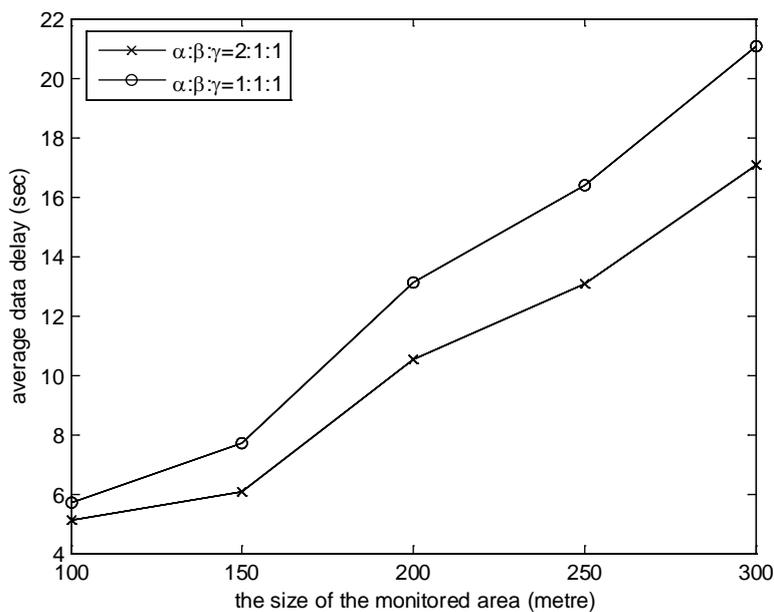


Figure 8. The effect of the weights on average data delay.

4. Conclusions

Sensor nodes consume most of energy for data transmission and the battery of sensor nodes cannot be replaced in many complex environments, therefore it is necessary to introduce mobile collectors to collect data. Since the speed of collectors is limited, it will cause data delay. Therefore, the problem of this paper is how to make multiple collectors cooperate with each other to reduce data delay. Based on the matching game theory, the problem is modeled as a one-to-one matching problem between collectors and high-level cluster heads. Then, a one-to-one matching algorithm is proposed. Simulation results show that as the size of the monitored area, the number of collectors and the number of sensors increase, the advantage of this algorithm in reducing data delay becomes more and more obvious.

Author Contributions: C.Z. designed the algorithm, analyzed the data, and wrote the paper; S.F. revised the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (No. 61903198).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, J.; Lin, Z.W.; Tsai, P.W.; Xu, L. Entropy-driven data aggregation method for energy-efficient wireless sensor networks. *Inf. Fusion*. **2020**, *56*, 103–113. [[CrossRef](#)]
2. Tao, L.; Zhang, X.M.; Liang, W.F. Efficient Algorithms for Mobile Sink Aided Data Collection From Dedicated and Virtual Aggregation Nodes in Energy Harvesting Wireless Sensor Networks. *IEEE Trans. Green Commun. Netw.* **2019**, *3*, 1058–1071. [[CrossRef](#)]
3. Yang, S.S.; Adeel, U.; Tahir, Y.; McCann, J.A. Practical Opportunistic Data Collection in Wireless Sensor Networks with Mobile Sinks. *IEEE Trans. Mob. Comput.* **2017**, *16*, 1420–1433. [[CrossRef](#)]
4. Heinzelman, W.R.; Chandrakasan, A.P.; Balakrishnan, H. An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans. Wireless Commun.* **2002**, *1*, 660–670. [[CrossRef](#)]
5. Younis, O.; Fahmy, S. HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Trans. Mob. Comput.* **2004**, *3*, 366–379. [[CrossRef](#)]
6. Smaragdakis, G.; Matta, I.; Bestavros, A. SEP: A stable election protocol for clustered heterogeneous wireless sensor networks. In Proceedings of the Second International Workshop on Sensor and Actor Network Protocols and Applications, Boston, MA, USA, 22 August 2004; pp. 1–11.
7. Lung, C.H.; Zhou, C.J. Using hierarchical agglomerative clustering in wireless sensor networks: An energy-efficient and flexible approach. *Ad Hoc Netw.* **2010**, *8*, 328–344. [[CrossRef](#)]
8. Oh, S.; Lee, E.; Park, S.; Jung, J.; Kim, S.H. Communication scheme to support sink mobility in multi-hop clustered wireless sensor networks. In Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, Australia, 20–23 April 2010; pp. 866–872. [[CrossRef](#)]
9. Nitesh, K.; Azharuddin, M.; Jana, P. Minimum spanning tree-based delay-aware mobile sink traversal in wireless sensor networks. *Int. J. Commun. Syst.* **2017**, *30*, 1–11. [[CrossRef](#)]
10. Jain, S.; Shah, R.C.; Brunette, W.; Borriello, G.; Roy, S. Exploiting mobility for energy efficient data collection in wireless sensor networks. *Mob. Netw. Appl.* **2006**, *11*, 327–339. [[CrossRef](#)]
11. Tashtarian, F.; Sohraby, K.; Varasteh, A. Multihop data gathering in wireless sensor networks with a mobile sink. *Int. J. Commun. Syst.* **2017**, *30*, 1–14. [[CrossRef](#)]
12. Wu, S.Y.; Liu, J.S. Evolutionary path planning of a data mule in wireless sensor network by using shortcuts. In Proceedings of the 2014 IEEE Congress on Evolutionary Computation, Beijing, China, 6–11 July 2014; pp. 2708–2715. [[CrossRef](#)]
13. Kim, D.; Uma, R.N.; Abay, B.H.; Wu, W.L.; Wang, W.; Tokuta, A.O. Minimum latency multiple data mule trajectory planning in wireless sensor networks. *IEEE Trans. Mob. Comput.* **2014**, *13*, 838–851. [[CrossRef](#)]
14. Sugihara, R.; Gupta, R.K. Optimal speed control of mobile node for data collection in sensor networks. *IEEE Trans. Mob. Comput.* **2010**, *9*, 127–139. [[CrossRef](#)]
15. Zhao, M.; Yang, Y.Y. Bounded relay hop mobile data gathering in wireless sensor networks. *IEEE Trans. Comput.* **2012**, *61*, 265–277. [[CrossRef](#)]

16. Dasgupta, R.; Yoon, S. Energy-efficient deadline-aware data-gathering scheme using multiple mobile data collectors. *Sensors* **2017**, *17*, 742. [[CrossRef](#)] [[PubMed](#)]
17. Gong, J.; Chang, T.H.; Shen, C.; Chen, X. Flight time minimization of UAV for data collection over wireless sensor networks. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 1942–1954. [[CrossRef](#)]
18. El Fissaoui, M.; Beni-Hssane, A.; Saadi, M. Energy efficient and fault tolerant distributed algorithm for data aggregation in wireless sensor networks. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 569–578. [[CrossRef](#)]
19. Sohrabi, K.; Gao, J.; Ailawadhi, V.; Pottie, G.J. Protocols for self-organization of a wireless sensor network. *IEEE Pers. Commun.* **2000**, *7*, 16–27. [[CrossRef](#)]
20. Zhang, C.; Fei, S.M.; Zhou, X.P. Energy efficient data collection in hierarchical wireless sensor networks. *China Commun.* **2012**, *9*, 79–88. [[CrossRef](#)]
21. Saaty, T.L. Fundamentals of the analytic network process-dependence and feedback in decision-making with a single network. *J. Syst. Sci. Syst. Eng.* **2004**, *2*, 129–157. [[CrossRef](#)]
22. Song, Q.Y.; Jamalipour, A. A network selection mechanism for next generation networks. In Proceedings of the IEEE International Conference on Communications, Seoul, Korea, 16–20 May 2005; pp. 1418–1422.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).