

Connectivity-Based Data Gathering with Path-Constrained Mobile Sink in Wireless Sensor Networks

Abdullah I. Alhasanat*, Khaled D. Matrouk, Haitham A. Alasha'ary, Ziad A. Al-Qadi

Department of Computer Engineering/Faculty of Engineering, Al-Hussien Bin Talal University, Ma'an, Jordan
Email: *abad@ahu.edu.jo

Received 18 April 2014; revised 18 May 2014; accepted 18 June 2014

Copyright © 2014 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

The design of an effective and robust data gathering algorithm is crucial to the overall performance of wireless sensor networks (WSN). However, using traditional routing algorithms for data gathering is energy-inefficient for sensor nodes with limited power resources and multi-hop communication protocols. Data gathering with mobile sinks provided an effective solution to this problem. The major drawback of this approach is the time and path constraints of the mobile sink, which limit the mobile sink to collect data from all sensor nodes and, then, data routing is still required for these unreachable parts by the mobile sink. This paper presents a new data gathering algorithm called Connectivity-Based Data Collection (CBDC). The CBDC algorithm utilizes the connectivity between sensor nodes so as to determine the trajectory of the mobile sink whilst satisfying its path constraint and minimizing the number of multi-hop communications. The presented results show that CBDC, in comparison with the LEACH-C algorithm, prolongs the network life time at different connectivity levels of sensor networks, varying number of sensor nodes and at different path constraints of the mobile sink.

Keywords

Wireless Sensor Network (WSN), Data Collection Algorithm, Received Signal Strength (RSS) and Mobile Sink

1. Introduction

As data aggregation is one of the primary tasks of Wireless Sensor Networks (WSN), sensor nodes with limited power resources and wireless communication range necessitate the need for energy-efficient data collection

*Corresponding author.

techniques [1] [2]. One of these techniques is to use a mobile sink which moves within the sensor field, collects the sensed data from source nodes and sends these data to application's users or network administrators. Recent research emphasized the fact that utilizing mobile sinks for data collection led to improving the performance of sensor networks in terms of network life time and the overall energy consumption [3]-[6]. However, one of the main challenges of this technique is the difficulty of the mobile sink to traverse all sensor nodes in the network, since data must be collected within a specific time deadline. This deadline is usually imposed by either the network user, the underlying application or even by the mobile sink itself, for example, due to the battery recharging cycle [7] [8].

This paper endeavors to design a data gathering algorithm (Data collection and data gathering are used interchangeably in this paper) using path constrained mobile sink referred to as connectivity-based data collection (CBDC) algorithm. With this algorithm, the major goal is to maximize the network life time whilst satisfying the time and energy constraints of the mobile sink. CBDC divides sensor nodes into clusters based on its connectivity in order to make all nodes within a single cluster fully connected. This in turn leads to increase of the number of single-hop sensor nodes¹ and therefore prolongs the network life time. Moreover, a multi-hop² communication protocol is used for those nodes not located within the communication range of the mobile sink. Hence, an energy load balancing technique is also introduced for single-hop node (also called gateway nodes) in order to prevent such nodes from quickly dying before other sensor nodes. As a result, further energy is conserved and sensor nodes of higher life times are achieved. The relation between mobile sink path constraints, network connectivity and number of sensor nodes and their impact on network life times are analyzed through extensive Monte Carlo simulation. The performance of CBDC algorithm is evaluated and compared with LEACH-C algorithm [9]. From network life time's point of view, simulation results of this paper showed that CBDC outperformed the LEACH-C for varying number of sensor nodes.

The paper is organized as follows. Related work is given in the next section. Section 3 formulates the problem statement. The CBDC algorithm is given in Section 4. Simulation results of the proposed algorithm are discussed in Section 5. Finally, the conclusion and future are drawn in Section 6.

2. Literature Review

Recent research in wireless sensor network highlighted the importance of using mobile sink techniques for data collection instead of using traditional data routing algorithms [5] [6]. Based on the type and number of mobile sinks, data gathering techniques can be applied using either random, constrained or controllable mobile sinks [3] [10] [11]. With random mobile sinks, sink nodes collect data from random locations in the sensor field. As a result of random mobility of sink node, data collection of low delivery ratio and high transfer latency is induced [12] [13]. Mounting sensor nodes on wild animals for behavior monitoring, where these animal move in an unexpected manner, are examples on such type of techniques.

More recently and with the advancement of embedded system technologies, sink nodes are supported with controlling units and, sometimes, with geographical positioning system (GPS) services. This would enable the sensor nodes to move in specific and controllable paths. In this approach, sink nodes are required to periodically move throughout in the sensor field and collect data gathered from the stationary sensor nodes. An example on such type of approaches occurs when the mobile sink placed on a shuttle for public transportation [14].

Different approaches for the mobile sink with deterministic movement pattern have been studied in literature [15] [16]. The path length and trajectory of the mobile sink are constrained by the energy resources as well as the data collection latency imposed by the underlying application or users. In literature, controlling the movements patterns of the mobile sink pave the way for more flexible design and an efficient data collection [3]. For example, traversing the whole network represent the most energy-efficient solution. However this option is restricted by the data collection latency deadlines. In this case, a fixed trajectory is determined with lower latency at the cost of higher energy consumption, since an increased number of multi-hop communication is required. The algorithm presented in this paper is actively adopting the controllable and fixed trajectory of the mobile sink.

Communication paths between source nodes and mobile sinks are either single-hop [17] [18], when these source nodes located within the communication range of the mobile sink, or multi-hop as the source node lo-

¹Single-hop nodes: those nodes which located directly in the communication range of the sink node.

²Multi-hop nodes: those nodes which are not located within the communication range of the sink node, and then require multi-hop communication protocols.

cated out of range of the mobile sink [10] [19]. In the former case, no data routing is required. However an energy efficient data routing algorithms for those nodes with multi-hop communication is needed. In addition, the issue that when and how the multi-hop nodes can communicate with the mobile sink should be also addressed [3]. To this end, we used the well-known *Stop to Collect Data* (SCD) scheme [20]. The mobile sink, in this method, stops at a pre-determined location and wait for data collection. Moreover, a proxy-based routing algorithm referred to as *Maximum Amount Shortest Path* (MASP) [10] is implemented. The single-hop nodes are selected as gateways or proxy nodes to store data from source nodes and pass them to the mobile sink at an appropriate time.

On the other hand, considering data routing from source nodes to the mobile sink, many mobile sink communication algorithms were investigated [20] [21]. These algorithms focused to prolong sensor nodes life times as well as to improve the data throughput of the sensor network. The problems of these protocols were the imbalance energy consumption of sensor nodes caused by the adopted *Shortest Path Tree* (SPT) algorithm. In [22], *MobiRoute* is suggested as a routing protocol for a path predictable mobile sink networks. With *MobiRoute*, the mobile sink holds for certain time at anchor points (*i.e.*, collection points) to collect data from sensor nodes [22].

Other solutions to the data collection problem were based on using multiple mobile sinks (or mobile elements) [8] [19] [23]. Authors in [23], for instance, suggested the *Area Splitting Algorithm* (ASA), which partitions the network into areas of a nearly equal number of sensor nodes. Then, one mobile element is assigned to each partition and entrusted with the data collection task in this partition. *Rendez-vous Design* (RD) approach is an example on using multiple mobile elements. This design was reformulated in [8] with more practical scenarios. For instance, assumptions of having several deadlines of data collection with constrained and fixed trajectories of mobile elements were considered. Although multiple mobile elements methods led to decrease the network latency, additional cost and network design complexity were produced. Hence, the current research addresses the data collection problem having single mobile sink with restricted paths.

This paper presents a clustering-based data collection algorithm, called CBDC. The proposed algorithm exploits the available information about nodes connectivity in order to maximize the number of single-hop sensor nodes at each collection point of the mobile sink, and therefore extends the life time of sensor nodes. Moreover, the proposed CBDC algorithm can be used to support multi-hop communication in sensor networks using Dijkstra algorithm with a new energy balancing scheme.

3. Problem Formulation

In this paper, data from N stationary sensor nodes need to be collected by a single node with unlimited buffer size known as mobile sink. As shown in **Figure 1(a)**, sensor nodes are randomly and uniformly distributed in the sensor field with known locations. The mobile sink must follow a limited path through a set of collection points in order to satisfy its energy and time constraints. Nodes located within the range of the mobile sink, *i.e.*, single-hop nodes, can send its data directly to the mobile sink. While other nodes, *i.e.*, multi-hop nodes, must send their data to the mobile sink by using one of the existing routing techniques, such as Dijkstra algorithm. In addition, the single-hop nodes are used to forward the data of sensor nodes to the mobile sink and hence we also refer to these nodes as gateway nodes.

Figure 1 illustrates an example on 1) Sensor nodes distribution within a sensor field including a sink node; 2) sensor nodes clustering showing single, multi-hop nodes and collection points (CP) and 3) a path of the mobile sink passing through all collection points. The main aspect that should be considered while designing data gathering algorithms is the network life time. The network life time can be defined as the time elapsed until the energy of the first node ran out [10]. In this paper, the energy model for each sensor node is given as

$$E = \frac{(P_{tx} + P_{rx})A}{R} \quad (1)$$

where P_{tx} and P_{rx} are the transmitting and receiving power, respectively. A is the data size and R is the data rate. Notice that P_{rx} is zero if the node is the source node. Obviously, E represents the energy consumed by a sensor node for one hop of data forwarding. As a total, a sensor node is consuming an energy that is given as

$$E_T(i) = E(1 + l_i) \quad (2)$$

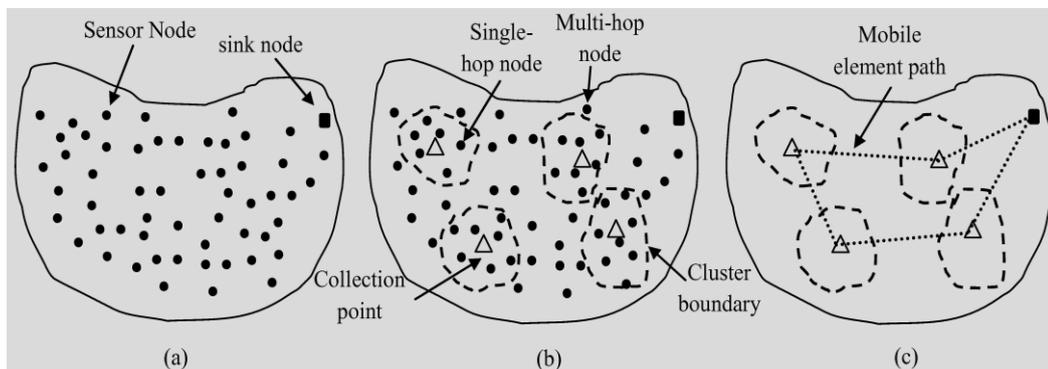


Figure 1. Examples on (a) Sensor nodes distribution within a sensor field including a sink node; (b) Sensor node clustering showing single, multi-hop nodes and collection points (CP); (c) A path of the mobile sink passing through all collection points.

where l_i denotes the number of sensor nodes which send its data through sensor i . Intuitively, increasing the ratio of number of single-hop to the number of multi-hop node certainly leads to increasing the network life time, since the number of nodes that is required to adopt a routing technique is reduced. It is clear from Equation (2), the smallest the value of l , the minimum the energy consumption. Furthermore, as shown in Figure 1, for the node that is close to the path of the mobile sink, the value of l is larger than that for distant nodes. As a consequence, in addition to using the shortest path algorithm, an adaptive energy load balancing scheme is required to reduce the data forwarding overhead of such nodes and then reduce the total energy consumption.

4. The Proposed Algorithm

This paper represents a data gathering algorithm which attempts to reduce the energy consumption by maximizing the number of gateway nodes and appropriately balance the load of data routing among all sensor nodes. Our algorithm, which is referred to as connectivity-based data collection (or CBDC for short), mainly consists of three phases: clustering, path determination and data collection.

4.1. Clustering Phase

Initially, the sensor nodes are grouped into clusters based on its connectivity to each other. In this type of clustering, each cluster represents a mesh network where each node in a cluster must be located within the communication range of all other nodes in the same cluster. In other words, let G_k represent a set of nodes, such that $G_k = \{i, j | w_{i,j} = 1 \text{ for } i \neq j\}$, where $k = 1, 2, \dots, M$ and $i, j = 1, 2, \dots, N_k$. N_k denotes the number of sensor in the set G_k , M is the number of clusters in the network and W is the connectivity matrix (such that $w_{i,j} = 1$ if nodes i and j are connected; otherwise $w_{i,j} = 0$). Figure 1(a) illustrates an example on sensor nodes clustering.

It is possible that a sensor node may be appeared in more than one cluster. To avoid this scenario in CBDC algorithm, the sensor node will be removed from clusters of low N_k and preserved with the cluster of high N_k . This step is important for phase three in order to maintain clusters of unique and large numbers of sensor nodes.

4.2. Path Determination Phase

As already mentioned, the primary task of the mobile sink is to collect data from sensor nodes. Since the path length of the mobile sink is restricted due to the limited energy resources, it is difficult to visit all sensor nodes in the network. Therefore, the mobile sink must follow a specified path that satisfies the path constraint and maintains the maximum number of gateway nodes. For this reason, a set of collection points (CP) should be considered. The CP of each cluster could be any physical location within it, since each cluster represents a mesh or a fully connected network. In this algorithm, the centroid point of each cluster is computed and used as a CP for that cluster. Hence, the CP of cluster G_k is given as

$$c_k = \left[\sum_{i=1}^{N_k} \frac{x_i}{N_k}, \sum_{i=1}^{N_k} \frac{y_i}{N_k} \right]. \quad (3)$$

The advantage of selecting the centroid locations of each cluster as a CP is to maximize the contact time³ between the mobile sink and source nodes.

The Travel Sales Man (TSM) algorithm can be applied now to determine the trajectory of the mobile sink, starting by the initial location, passing through all collection points and ending up with the initial location, as shown in **Figure 1(c)**. Visiting all clusters in the network perhaps results with a tour length that exceeds the prespecified path constraints. In this case, a tour reduction mechanism should be used as will be described in the next phase of this algorithm. It is clear that reducing the tour length of the mobile sink is necessary when computed path length is longer than the prescribed path length of the mobile sink. Minimizing such tour length is achieved by removing one or more CPs from this tour. All sensor nodes in a cluster for which the collection point belongs will be considered as multi-hop nodes. Hence, the question that which CP should be removed is carefully answered realizing two main considerations. The first one is that the removed CP must lead to reduce the tour length more than other CPs. The second point focused on increasing the total number of single-hop nodes by maintaining a cluster of larger number of sensor nodes. Both considerations are integrated with the CBDC algorithm in a single benefit function which is applied to all CPs, and then the CP with the minimum benefit value is removed. This is repeated until the path constraint of the mobile sink is satisfied. The benefit function used for this purpose is given as

$$\delta(c_k) = \frac{\alpha_k}{\beta_k N_k} \quad (4)$$

where α_k denotes the difference between the tour length before and after removing the CP c_k . β_k represents the minimum value of the distance between c_k and c_{k+1} , and c_k and c_{k-1} , assuming that the optimal path is given as $c_0, c_1, c_2, \dots, c_M, c_0$, where c_0 is the location of the sink node.

Obviously, the benefit function is increased for a small value of N_k or a large value of α_k . Remember that small value of N_k means small number of cluster members. Since removing the CP of maximum benefit function δ from the optimal path would consequently reduce the total tour length of the mobile sink. This step is repeated continuously until the path constraint of the mobile sink is satisfied. The flowchart of the proposed algorithm is shown in **Figure 2**.

4.3. Data Collection Phase

As already mentioned before, each of multi-hop nodes needs to find its path to the closest CP via one of the gateway nodes. In this paper, this target is achieved using the well-known Dijkstra algorithm [24]. With this algorithm, the optimal route from a single node (*i.e.*, the source node) to a single destination (*i.e.*, the gateway node) is found. Nevertheless, some gateway nodes would be heavily incurred for packets relying more than its neighbours of gateway nodes. Taking the advantage of having within each cluster a set of gateway nodes instead of only one, such a load could be fairly distributed among these nodes when an energy balancing technique is implemented. In this regard, a threshold function is introduced with the CBDC algorithm that restricts the number of packets forwarded by one gateway node. The threshold function, say h_k , of cluster k is computed as

$$h_k = \frac{N_G}{MN_k} \quad (5)$$

where N_G represents the total number of gateway nodes, such that

$$N_G = \sum_{i=1}^M N_i \quad (6)$$

It is worth noting that Equation (5) represents an adaptive threshold value which varies from cluster to cluster based on the number of gateway nodes in the cluster itself. For example, a cluster of a large number of gateway nodes will have less h_k than a cluster of smaller number of gateway nodes. **Figure 2** shows the flowchart of the CBDC algorithm proposed in this paper.

Once the data collection round starts, the mobile sink begins to move from the sink node location passing through the selected collection point with speed of q m/s. Hence, l/q seconds is required to complete a single

³The contact time is defined as the amount of time which is utilized for data transfer between source nodes and the mobile sink [3].

data collection round. During each round, each sensor node should send its data to the mobile sink. Two

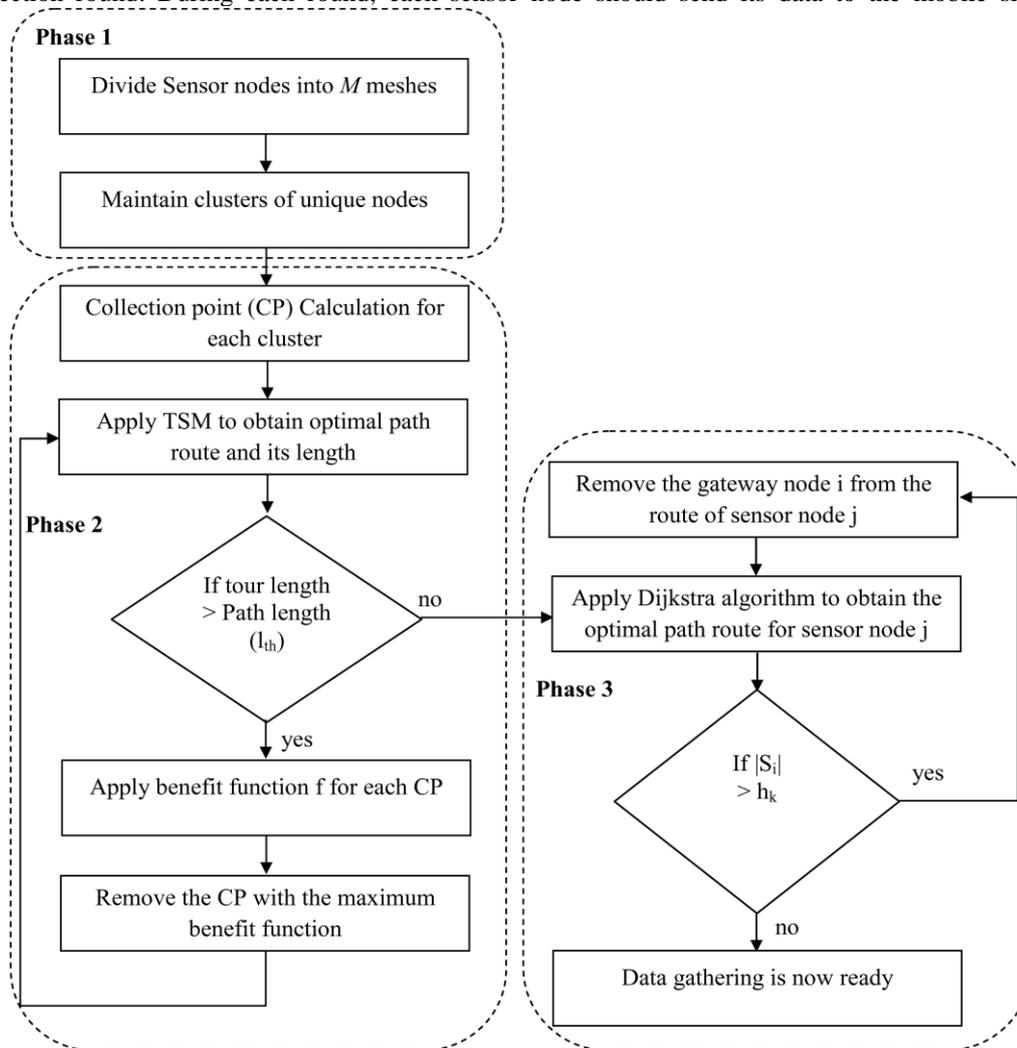


Figure 2. Flow chart of the connectivity-based algorithm discussed in this paper.

approaches are suggested to organize data collection and to prevent out-of-synchronization problem.

In the first one, sensor nodes are designed to send its data every $1/q$ seconds. The problem of out-of-synchronization still occurs when for instance the mobile sink is slowed down or even delayed somewhere in the network. In the second approach, *Stop to Collect Data* (SCD) algorithm [20] is used. Hence, once the mobile sink arrived at the CP of a cluster, it sends a data request to sensor nodes through one of the gateway nodes belonging to this cluster. This request can pass through the route designed for data collection. Then sensor nodes reply with the requested data through the same route.

5. Performance Evaluation

In this section, we study the performance of the connectivity-based data collection algorithm (CBDC). The network life time of this algorithm was evaluated for different path constraints of the mobile sink and at varying number of sensor nodes. In addition, the efficiency of our algorithm is examined for networks of different levels of connectivity.

5.1. Simulation Scenario

In this simulation, sensor nodes are randomly and uniformly deployed within 400×400 meter square area.

Transmission rate of all sensor nodes including gateway nodes is set to 200 Kbps. The mobile sink moves at speed of 1 m/s. The initial energy of each node is assumed to be 10 Joules. The transmission power and reception power are set to 21 and 15 milliwatt, respectively. In this experiment, energy consumption is considered for data transmission and reception only; while during other time periods, sensor nodes are supposed to be in the sleep mode for which no energy consumption is assumed.

In our simulation scenario, the connectivity of sensor nodes is measured as [25]

$$\text{connectivity} = \frac{\sum_{i=1}^N \sum_{j=1}^N w_{i,j}}{N^2 - N} \times 100\% \quad (7)$$

Notice that $w_{i,j}$ mostly depends on the Received Signal Strength (RSS) sensitivity threshold RSS_{th}^4 of the sensor node, or equivalently the communication range d_{th} . Since network connectivity is an RSS strictly dependant parameter, it cannot be readily controlled. For this reason, in this simulation we used a range of RSS_{th} values and records the corresponding network connectivity values. **Table 1** lists the RSS_{th} , the corresponding d_{th} and network connectivity.

5.2. Scalability of Sensor Nodes

An important factor that should be considered while designing a sensor network is the scalability, which measures how efficient a WSN is for small and large numbers of sensor nodes. The scalability of CBDC algorithm is evaluated by testing the algorithm at varying number of sensor nodes (from 100 to 500 nodes). The simulation is repeated for mobile sinks of different path constraints (*i.e.*, $P_{th} = 800, 1000, 1200$ and 1400 m). The number of collection points (CPs), the percentage of gateway nodes and the network life time are shown in **Figures 3(a)-3(c)**, respectively. The network life time is computed as the time elapsed until the energy of the first node in the network ran out.

Generally speaking, when the number of sensor nodes increased larger number of clusters with fewer members (*i.e.*, gateway node), is formulated. This, in turn, leads to increase the number of CPs and decrease the percentage of gateway nodes in the network; particularly for mobile sinks of long path constraints (e.g., 1200 and 1400 m), as shown in **Figure 3(a)**, **Figure 3(b)**, respectively. As a result of reducing the percentage of gateway nodes, the percentage of multi-hop nodes is increased and therefore the network life time is shortened, as shown in **Figure 3(c)**. It is also clear from this figure that the increase of the mobile sink path constraint leads to prolonging the life time of sensor nodes. This is due to the reason that increasing the path constraint allowed the mobile sink to t area in traverse larger area of the sensor field and hence reduced the number multi-hop sensor nodes. Another trend can be seen from this figure is that for longer path constraints, the network life time is highly dependent on the number of sensor nodes compared to that for shorter path constraints. Consequently, these results emphasize the scalability of the CBDC algorithm in spite of using a mobile sink of limited path constraints.

Table 1. Typical RSS sensitivity values RSS_{th} , the corresponding communication range d_{th} and the connectivity percentage used in the simulation experiments.

$RSS_{th} d_B$	d_{th} meter	connectivity 100%
-75	33	0.01
-80	54	0.1
-85	90	0.15
-90	148	0.30
-95	245	0.60

⁴RSS sensitivity is a threshold value specified by the transceiver circuit of the sensor node. If the RSS of the received packet is less than the RSS_{th} value, the packet will be discarded. Note that the corresponding communication range d_{th} is computed as $d_{th} = d_0 10^{\frac{P_0 - RSS_{th}}{10\nu}}$ [26], where d_0 is the reference distance, P_0 is the reference power and ν is the path loss exponent (in this paper we used $d_0 = 1$, $P_0 = -40$ dB and $\nu = 2.3$ [26]).

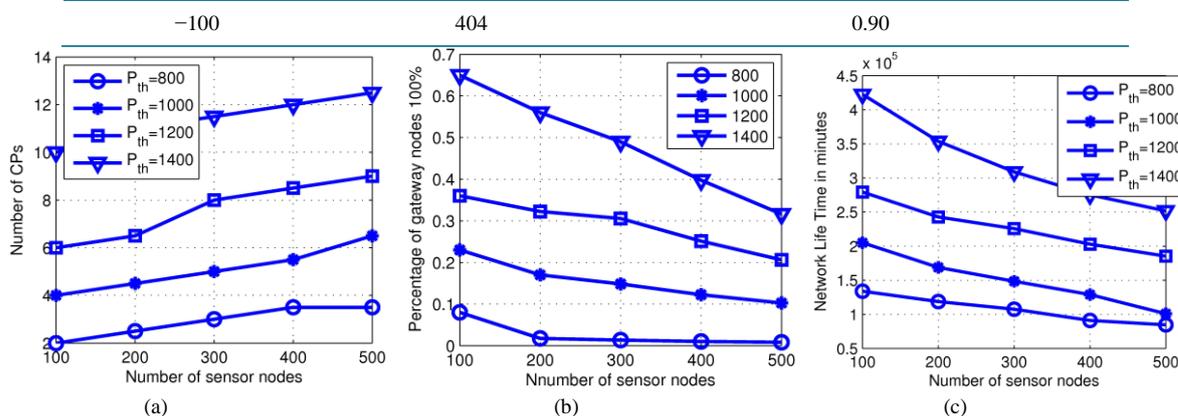


Figure 3. The impact of increasing number of sensor nodes on the performance of CBDC algorithm for different path constraints and in terms of (a) Number of collection points; (b) Percentage of gateway nodes of the total number sensor nodes; (c) Network life time in minutes.

5.3. Connectivity of Sensor Nodes

Next, we study the performance of CBDC algorithm as a function of network connectivity. It is already mentioned that the network connectivity percentage is varied based on the selected value of RSS_{th} , as given in [Table 1](#). In contrast to the previous results, it is apparent from [Figure 4\(a\)](#), that sensor networks of high connectivity include less CPs than sensor network of low connectivity. This, as an effect, increases the number of gateway nodes in these networks as shown in [Figure 4\(b\)](#). As the path constraint increased the percentage of gateway nodes (single hop sensor nodes) is increased and hence the network life time is exponentially increased, as shown in [Figure 4\(c\)](#). In the same manner, extending the network life time is actually restricted by the path constraint of the mobile sink. It can be also observed that networks of weak connectivities almost have the same life time irrespective to path constraints even though using long path constraints, e.g., 1400 m. Unsurprisingly, the effectiveness of using long path constraints is much clearer at high network connectivity, which results with networks of extremely long life times. Notice that a connectivity of 100% yields a network with only single CP located within the communication range of all sensor nodes and, hence, no multi-hop routing protocols are needed.

5.4. Energy Balancing Technique

The importance of using the energy balancing technique, considered in this paper, for the CBDC algorithm is assessed through Monte Carlo simulation. The network life time as a function of increasing number of sensor nodes with and without the adopted energy balancing technique is shown in [Figure 4\(d\)](#). Obviously, the performance of CBDC with the proposed energy balancing scheme is substantially improved at different numbers of sensor nodes. This is due to the reason that with this scheme the energy consumption is appropriately distributed among the gateway nodes, rather than using Dijkstra algorithm solely for multi-hop data routing. In this section, the performance of CBDC algorithm is compared with the LEACH-C algorithm [9]. In this experiment, each sensor node is required to send as a total of one MByte of data to the sink node. The maximum message size is set to 100 Bytes that should be transmitted every 6 seconds. The communication energy parameters, as given in [9], are: $E_{elec} = 50$ nJ/bit, $\epsilon_{fs} = 10$ pJ/bit/m², $\epsilon_{mp} = 0.0013$ pJ/bit/m² and the energy for data aggregation is set as $E_{DA} = 5$ nJ/bit/signal. For consistent comparison, the simulation parameters listed above were used for the two algorithms alike. Thereafter, 200 independent simulation runs were executed for both algorithms. At each simulation run, the network density is kept constant at 0.25 node/m² and the sensor nodes were randomly and uniformly distributed in $\left(\frac{N}{0.25}\right)$ square area. The network life time, energy consumption and the number of dead nodes were recorded at the end of the experiment. The results for the two algorithms are shown in [Figure 5\(a\)-5\(c\)](#), respectively.

It can be observed from [Figure 5](#), CBDC algorithm performs better than LEACH-C in particular for small

number of sensor nodes. However, as the number of sensor nodes increases the network life time is reduced and the network life time difference between the CBDC and LEACH-C is almost diminished. This, in consequence,

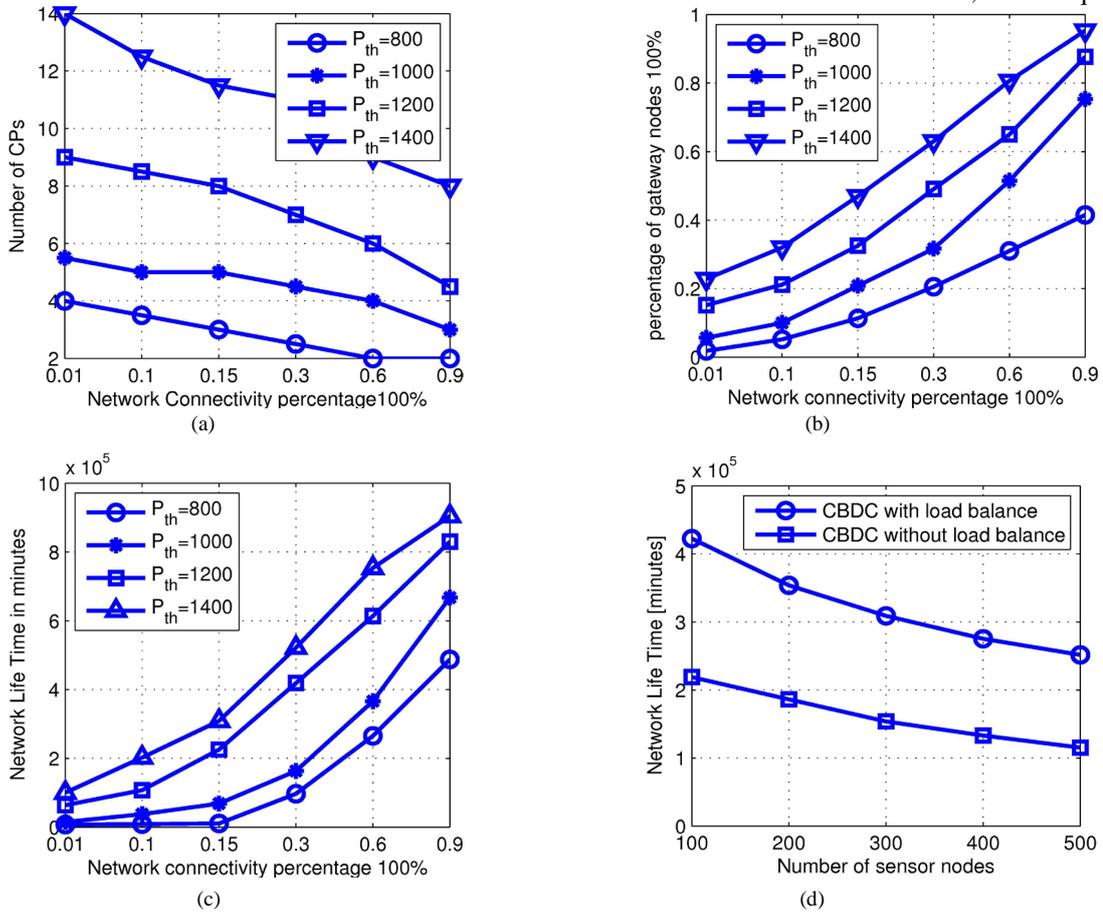


Figure 4. The performance of CBDC algorithm as a function of network connectivity and for different path constrains and in terms of (a) Number of collection points; (b) Percentage of gateway nodes; (c) Network life time in minutes; (d) The network Life Time of CBDC algorithm as a function of increasing number of sensor nodes with and without using the load balancing scheme proposed in this paper.

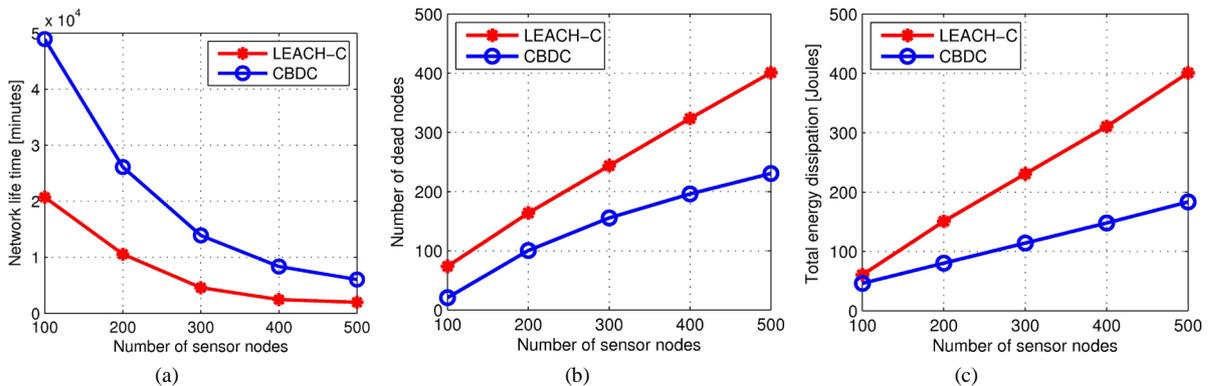


Figure 5. Performance comparison between LEACH-C and CBDC algorithms as a function of increasing number of sensor nodes at constant density of 0.25 node/m. The performance is evaluated in terms of (a) Network life time; (b) Number of dead sensor nodes; and (c) Total energy dissipation in the system.

emphasizes the efficiency of CBDC algorithm at small number of sensor nodes; while at high number of sensor nodes the two algorithms obtained almost the same performance. On the other hand, the total number of dead

nodes after 10^4 seconds of simulation time is elapsed is shown in **Figure 5(b)**. Each sensor node in this experiment is required to send one mega bytes of data to the sink node. It is apparent from this figure that larger numbers of sensor node died for LEACH-C algorithm in order to send this size of data. This number of dead nodes is approximately linear with respect to the number of nodes. In contrast, CBDC completed its task with lower number of dead nodes. Furthermore, the number of dead nodes is slightly increased with the increase in the number of sensor nodes.

Next, we studied the performance of both algorithms in term of the total energy dissipation in the system and showed the result in **Figure 5(c)**. Obviously, a significant saving of the total energy is achieved in the case of CBDC algorithm in comparison with the LEACH-C algorithm. Moreover, with the CBDC algorithm, networks of larger number of sensor nodes resulted with a further saving of total system energy. For example, 50% of the total energy is consumed at 100 sensor nodes, while 35% is only consumed for 500 sensor nodes. On the other hand, LEACH-C consumed about 80% and 70% of its total energy for 100 and 500 sensor nodes; respectively. The high energy consumption of LEACH-C algorithm is due to the dynamic transmission scheme used with this algorithm, since the transmission power model is proportional to the transmitter-receiver separation distance [9]. With the CBDC algorithm, however, a constant transmission power is used regardless of the transmitter-receiver separation distance. If a receiver node is not located in the direct communication range of the transmitter node, then a multi-hop communication is used with the energy balanced scheme discussed in this paper.

6. Conclusions and Future Work

In this paper, an energy efficient data gathering technique, referred to as connectivity-based data collection (CBDC) algorithm, is presented. The CBDC algorithm focused on reducing the number of multi-hop communication without violating the path constraints of mobile sinks. In addition, an energy balancing scheme is suggested for gateway nodes in order to distribute the energy consumption over these nodes. The simulation results presented in this paper showed that the proposed algorithm provided a desirable data gathering solution for networks of varying number of sensor nodes. Moreover, the CBDC performed well for mobile sinks of limited path constraints and energy resources. In comparison with the LEACH-C algorithm, CBDC conserved about 35% of the total system energy and hence extended the network life time substantially.

In future work, it would be worthwhile to use multiple mobile elements rather than using only a single mobile sink, in order to address the problems of buffer overflow, data latency and improve the data delivery ratio. Meanwhile, it is also necessary in this case to explore whether the potential benefit obtained would justify its additional requirements.

References

- [1] Akyildiz, I., Su, W., Sankarasubramaniam, Y. and Cayirci, E. (2002) A Survey on Sensor Networks. *IEEE Communication Magazine*, **40**, 102-114.
- [2] Li, D., Wong, K.D., Hu, Y.H. and Sayeed, A.M. (2002) Detection, Classification and Tracking of Targets. *IEEE Signal Processing Magazine*, **19**, 17-29. <http://dx.doi.org/10.1109/79.985674>
- [3] Di Francesco, M. and Das, S.K. (2011) Data Collection in Wireless Sensor Networks with Mobile Elements: A Survey. *ACM Transactions on Sensor Networks (TOSN)*, **8**, Article No. 7.
- [4] Shangguan, L., Mai, L., Du, J., He, W. and Liu, H. (2011) Energy-Efficient Heterogeneous Data Collection in Mobile Wireless Sensor Networks. *Proceedings of 5th PMECT International Workshop Co-Located with the 20th International Conference on Computer Communication Networks (ICCCN'2011)*, Hawaii.
- [5] Tang, B., Wang, J., Geng, X., Zheng, Y. and Kim, J. (2012) A Novel Reliable and Efficient Data Harvesting Mechanism in Wireless Sensor Networks with Path-Constrained Mobile Sink. *Proceedings of SPDA/NT/ICE 2012 Conference*, Hangzhou, 158-163.
- [6] Xing, G., Li, M., Wang, T., Jia, W. and Huang, J. (2012) Efficient Rendezvous Algorithms for Mobility-Enabled Wireless Sensor Networks. *IEEE Transactions on Mobile Computing*, **11**, 47-60.
- [7] Somasundara, A., Ramamoorthy, A. and Srivastava, B. (2004) Mobile Element Scheduling for Efficient Data Collection in Wireless Sensor Networks with Dynamic Deadlines. *Proceedings of the 25th IEEE International Real-Time Systems Symposium (RTSS)*, Lisbon, 05-08 December 2014, 296305. <http://dx.doi.org/10.1109/REAL.2004.31>
- [8] Xing, G., Wang, T., Jia, W. and Li, M. (2008) Rendezvous Design Algorithms for Wireless Sensor Networks with a Mobile Base Station. *Proceedings of the 9th ACM International Symposium on Mobile ad Hoc Networking and Com-*

- puting (*MobiHoc*), Hong Kong, 26-30 May 2008, 231-240.
- [9] Heinzelman, W., Chandrakasan, A. and Balakrishnan, H. (2002) An Application-Specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Transactions on Wireless Communications*, **1**, 660-670.
- [10] Gao, S., Zhang, H. and Das, S.K. (2011) Efficient Data Collection in Wireless Sensor Networks with Path-Constrained Mobile Sinks. *IEEE Transactions on Mobile Computing*, **10**, 592-608. <http://dx.doi.org/10.1109/TMC.2010.193>
- [11] Gao, S., Zhang, H., Song, T. and Wang, Y. (2010) Network Lifetime and Throughput Maximization in Wireless Sensor Networks With a Path-Constrained Mobile Sink. *Proceeding of International Conference on Communications and Mobile Computing*, Shenzhen, 12-14 April 2010, 298-302.
- [12] Shah, R., Roy, S., Jain, S. and Brunette, W. (2003) Data MULEs: Modeling a Three-Tier Architecture for Sparse Sensor Networks. *Proceedings of the 1st IEEE International Workshop Sensor Network Protocols and Applications*, Anchorage, 11 May 2003, 30-41.
- [13] Jain, S., Shah, R.C., Brunette, W., Borriello, G. and Roy, S. (2006) Exploiting Mobility for Energy Efficient Data Collection in Sensor Networks. *Mobile Networks and Applications*, **11**, 327-339. <http://dx.doi.org/10.1007/s11036-006-5186-9>
- [14] Chakrabarti, A., Sabharwal, A. and Azhang, B. (2003) Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks. *Proceedings of the 2nd International Workshop on Information Processing in Sensor Networks (IPSN 2003)*, Palo Alto, 26-27 April 2004, 129-145.
- [15] Wang, G., Cao, G. and La Porta, T. (2006) Movement-Assisted Sensor Deployment. *IEEE Transactions on Mobile Computing*, **5**, 640-652. <http://dx.doi.org/10.1109/TMC.2006.80>
- [16] Song, L. and Hatzinakos, D. (2007) Architecture of Wireless Sensor Networks with Mobile Sinks: Sparsely Deployed Sensors. *IEEE Transactions on Vehicular Technology*, **56**, 1826-1836.
- [17] Xing, G.L., Wang, T., Xie, Z.H. and Jia, W.J. (2008) Rendezvous Planning in Wireless Sensor Networks with Mobile Elements. *IEEE Transactions on Mobile Computing*, **7**, 1430-1443.
- [18] Wang, G., Cao, G. and La Porta, T. (2006) Movement-Assisted Sensor Deployment. *IEEE Transactions on Mobile Computing*, **5**, 640-652.
- [19] Rao, J., Wu, T. and Biswas, S. (2008) Network Assisted Sink Navigation Protocols for Data Harvesting in Sensor Networks. *Proceeding of the 2008 Conference in Wireless Communication and Networking (WCNC 2008)*, Las Vegas, 31 March-3 April 2008, 2887-2892.
- [20] Kansal, A., Somasundara, A., Jea, D., Srivastava, M. and Estrin, D. (2004) Intelligent Fluid Infrastructure for Embedded Networks. *Proceedings of the 2nd International Conference on Mobile Systems, Applications and Services 2004*, Boston, 6-9 June 2004, 111-124.
- [21] Somasundara, A., Kansal, A., Jea, D., Estrin, D. and Srivastava, M. (2006) Controllably Mobile Infrastructure for Low Energy Embedded Networks. *IEEE Transactions on Mobile Computing*, **5**, 958-973. <http://dx.doi.org/10.1109/TMC.2006.109>
- [22] Luo, J., Panchar, J., Piorkowski, M., Grossglauser, M. and Hubaux, J.P. (2006) MobiRoute: Routing towards a Mobile Sink for Improving Lifetime in Sensor Networks. *Proceedings of 2nd IEEE/ACM International Conference Distributed Computing in Sensor Systems (DCOSS)*, San Francisco, 18-20 June 2006, 480-497. http://dx.doi.org/10.1007/11776178_29
- [23] He, L., Xu, J.D. and Yu, Y.T. (2009) Optimize Multiple Mobile Elements Touring in Wireless Sensor Networks. *IEEE International Symposium on Parallel and Distributed Processing with Applications*, Chengdu, 10-12 August 2009, 317-323.
- [24] Goldsmith, A. (2005) *Wireless Communications*. Cambridge University Press, Cambridge. <http://dx.doi.org/10.1017/CBO9780511841224>
- [25] Siek, J.G., Lee, L.Q. and Lumsdaine, A. (2002) *The Boost Graph Library: User Guide and Reference Manual*. Pearson Education, Upper Saddle River.
- [26] Li, X.R. (2007) Collaborative Localization with Received-Signal Strength in Wireless Sensor Networks. *IEEE Transactions on Vehicular Technology*, **56**, 3807-3817. <http://dx.doi.org/10.1109/TVT.2007.904535>

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or [Online Submission Portal](#).

