To Split or Not To Split? From the Perspective of a Delay-Aware Data Collection Network Structure

Chi-Tsun Cheng
Department of Electronic and Information Engineering,
The Hong Kong Polytechnic University,
Hunghom, Kowloon, Hong Kong
Email: chi-tsun.cheng@polyu.edu.hk

Nuwan Ganganath
Department of Computing and Information Systems,
Wayamba University of Sri Lanka,
Kuliyapitiya, Sri Lanka.
Email: manganganath@ieee.org

***Abstract*—Collecting data from massive numbers of individual nodes is always a challenging task in wireless sensor networks. The duration of a data collection process, which can greatly affect the detection capabilities of a network, should be reduced whenever possible. For scenarios where only a single cluster is allowed, the delay-aware data collection network structure can minimize the duration of a data collection process. The aim of this paper is to explore the possibilities of improving the original delay-aware network structure by splitting the single tree structure into multiple clusters. Analyses on the conditions and effects of splitting the aforementioned structure are presented. Based on the analyses, two novel network splitting algorithms using k–means clustering algorithms are proposed. Simulation results show that the proposed network splitting algorithms may further reduce the duration of a data collection process. With the help of the k–means algorithms, communication distance among sensor nodes can be further reduced especially for networks with large numbers of wireless sensor nodes.**

***Index Terms*—Wireless Sensor Networks, Delay-Aware, Data Collection Process, Resources Management**

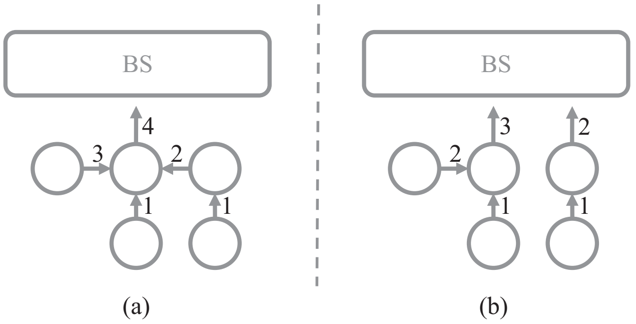
I. Introduction

In a typical wireless sensor network (WSN), a large number of wireless sensor nodes are deployed into a sensing terrain to perform close-range sensing. These nodes collect data from their surroundings and report to a base station (BS) that will further process the data. For delay-sensitive applications such as target detection, it is always desirable to shorten the duration of a data collection process (DCP).

In [1], Cheng *et al.* proposed a delay-aware data collection network structure (DADCNS), which is aimed to minimize the duration of a DCP in WSNs. The proposed structure assumes packets collected from wireless sensor nodes are highly fusible such that multiple packets can be fused into one [2], [3]. Two network formation algorithms are proposed in [1] to construct the proposed network structure in single or multiple-tree forms.

The main objective of this paper is to study the advantages, conditions, and limitations of splitting a single DADCNS into sub-clusters. Assume each data transaction will last for one time-slot and the duration for an in-network data-fusion process is negligible. Consider a network with $N = 5$ nodes (see Fig. 1(a)), by organizing the nodes into a single cluster using the DADCNS, the BS will take $T_{\text{DCP}} = \lceil 1 + \log_2 N \rceil = 4$ time-slots to complete a DCP. It is possible to further reduce T_{DCP} by splitting the network into sub-clusters. Suppose the network is divided into two as shown in Fig. 1(b), the two sub-clusters will report their data back to the BS at time-slots 2 and 3 accordingly. The overall T_{DCP} value of the network is now reduced to 3.

The rest of the paper is arranged as follow. Related work is reviewed in Section II. Analyses on splitting networks under different conditions are presented in Section III. Based on the results of the analyses, a



(a) (b)

Fig. 1. Networks with $N = 5$ nodes organized using the DADCNS (a) without splitting and (b) with splitting. Empty circles are representing wireless sensor nodes while circles with labels “BS” are representing base stations. Arrows are representing data links and the numbers next to the arrows are representing the transmission schedules.

network splitting algorithm is proposed in Section IV. In Section V, performance of the proposed algorithm is evaluated using computer simulations. The results are further studied and discussed in Section VI. Finally, conclusions are given in Section VII.

II. Related Work

In [4], Strasser *et al.* proposed an alarm forwarding algorithm, which is operating across data-link, network, and transport layers. The proposed algorithm tries to find a delay-aware path, from a source node to the BS, based on the communication distance and delay associated with the path. In [5], Djukic and Valaee studied the delay due to sub-optimum transmission schedules in time division multiple access (TDMA) wireless networks. By formulating the transmission orders of the links into a cost function, the link scheduling problem is treated as a min-max optimization problem. Their proposed scheduling method is operating at the data-link layer.

For continuous monitoring applications such as tracking, the total number of samples received within a duration can be equally important as the delay of a data collection process. In [6], Cheng *et al.* studied the possibility of overlapping transmission schedules in order to reduce the time for q consecutive DCPs. For transmission schedules to be partly overlapped, some connections in the original DADCNS in [1] are regarded as conflicting and needed to be removed. In-network data fusion may not be applicable in all scenarios. In [7], Cheng *et al.* proposed another delay-aware network structure for applications with partially fusible or even non-fusible packets. Shen *et al.* tried to create a minimum spanning tree that can minimize energy consumption of DCPs while satisfy some given delay conditions [8]. Similarly, Sivaranhani *et al.* proposed an adaptive data aggregation technique that will construct paths according to the delay and energy constraints [9].

III. Analyses

In this section, conditions and effects for splitting networks with the DADCNS will be studied. Consider a network with N nodes that is managed using the DADCNS. Suppose $2^{k-1} < N \leq 2^k$, its BS will take $k+1$ time-slots to finish a DCP.

Example 1: Consider a network with $N = 8$ that is managed using the DADCNS (see Fig. 2(a)). Since $2^{3-1} < 8 \leq 2^3$, its BS will take $k+1 = 3+1 = 4$ to finish a DCP.

Theorem 1: For a network with $2^{k-1} < N \leq 2^k$ nodes (where $k > 1$) that are organized using the DADCNS, splitting the network into two sub-clusters with N_1 and N_2 nodes, where $N_1 + N_2 = N$ and $|N_1 - N_2| \leq 1$, will not increase the overall T_{DCP} value of the network.

Proof: For a network with $2^{k-1} < N \leq 2^k$ nodes (where $k > 1$) that is organized using the DADCNS, its T_{DCP} is expressed as

$$T_{\text{DCP}} = \lceil 1 + \log_2 N \rceil = k + 1. \tag{1}$$

Consider cases when N is an odd number. Splitting the network can yield two sub-clusters with N_1 and N_2 nodes. Without loss of generality, assume $N_1 = \frac{N-1}{2}$ and $N_2 = \frac{N+1}{2}$, such that $N_1 < N_2$. Suppose the sub-clusters are also arranged into the DADCNS, using (1), the corresponding $T_{\text{DCP}i}$ (where $i = 1, 2$) of the sub-clusters are expressed as

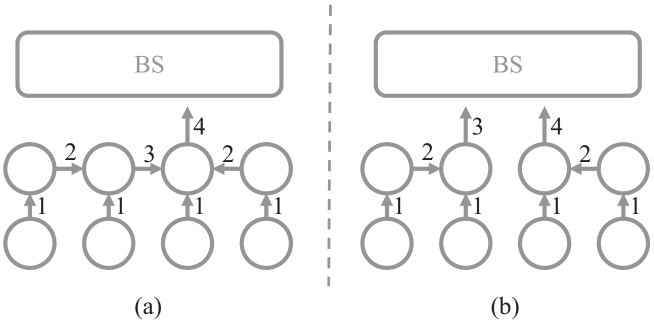
$$T_{\text{DCP1}} = \lceil 1 + \log_2 N_1 \rceil \leq k,$$

$$T_{\text{DCP2}} = \lceil 1 + \log_2 N_2 \rceil = k.$$

If $T_{\text{DCP1}} < T_{\text{DCP2}}$, the cluster heads (CHs) of the two sub-clusters can access the BS at different time-slots. The overall T_{DCP} value is equal to that of the larger sub-cluster (i.e. $T_{\text{DCP}} = T_{\text{DCP2}} = k < k+1$). However, if $T_{\text{DCP1}} = T_{\text{DCP2}} = k$, the corresponding $T_{\text{DCP}i}$ (where $i = 1, 2$) of the sub-clusters are expressed as

$$T_{\text{DCP}i} = \lceil 1 + \log_2 \tfrac{N}{2} \rceil = k, \quad i = 1, 2.$$

Both CHs of the sub-clusters will try to access the BS at the same time-slot. To avoid collisions, one of these CHs will postpone its transmission by 1 time-slot. Therefore, the overall T_{DCP} value of the network will equal to $k+1$. The same situation applies to cases when N is an even number, such that $N_1 = N_2 = \frac{N}{2}$. The overall T_{DCP}



(a) (b)

Fig. 2. Networks with $N = 8$ nodes organized using the DADCNS (a) without splitting and (b) with splitting. Empty circles are representing wireless sensor nodes while circles with labels "BS" are representing base stations. Arrows are representing data links and the numbers next to the arrows are representing the transmission schedules.

value remains the same as that of the network before splitting. The theorem is proved. ■

According to *Theorem 1*, splitting a network into two sub-clusters with a size difference less than one will not impose extra delay to its DCP if the sub-clusters are organized using the DADCNS. In some cases, splitting a network may even yield a lower overall T_{DCP} value.

Example 2: Consider the network in *Example 1* (see Fig. 2(a)). Splitting such network into two sub-clusters with a size difference less than one (i.e. $N_1 = N_2 = \frac{N}{2}$) will result in the network shown in Fig. 2(b). To avoid collision, the two CHs of the sub-clusters will try to access the label BS at time-slots 3 and 4, respectively. The overall T_{DCP} value of the network remains unchanged.

Theorem 2: For a network with $N = 2^{k-1} + 1$ nodes (where $k > 1$) that are organized into the DADCNS, splitting the network into two sub-clusters with N_1 and N_2 nodes, where $N_1 = \frac{N-1}{2}$ and $N_2 = \frac{N+1}{2}$, can reduce the overall T_{DCP} value of the network.

Proof: From *Theorem 1*, it is observed that the overall T_{DCP} value of the network can be reduced if $T_{\text{DCP1}} < k$ (i.e. $T_{\text{DCP1}} = k - 1$), such that

$$\begin{array}{rcl} T_{\text{DCP1}} & = & k-1, \\ \lceil 1 + \log_2 \frac{N-1}{2} \rceil & = & k-1, \\ \log_2 \frac{N-1}{2} & \leq & k-2, \\ N-1 & \leq & 2^{k-1}, \\ N & \leq & 2^{k-1}+1. \end{array}$$

Given the condition $2^{k-1} < N \leq 2^k$ (where $k > 1$), $N = 2^{k-1} + 1$. The theorem is proved. ■

With *Theorem 2*, it is observed that splitting a network of size $N = 2^{k-1} + 1$ into two sub-clusters with a size difference less than one can reduce T_{DCP}, provided that the sub-networks are also organized using the DADCNS. Recall the earlier example given in Section I (see Fig. 1(a)), splitting a network of size $N = 2^{3-1} + 1 = 5$ into two sub-clusters with sizes $N_1 = \frac{5+1}{2} = 3$ and $N_2 = \frac{5-1}{2} = 2$ reduces the overall T_{DCP} value from 4 to 3 (see Fig. 1(b)).

Theorem 3: For a network with $N = 2^{k-1}$ nodes (where $k > 1$) that are organized into the DADCNS, splitting the network into two sub-clusters with N_1 and N_2 nodes, where $N_1 = N_2 = \frac{N}{2}$, will not reduce the overall T_{DCP} value of the network.

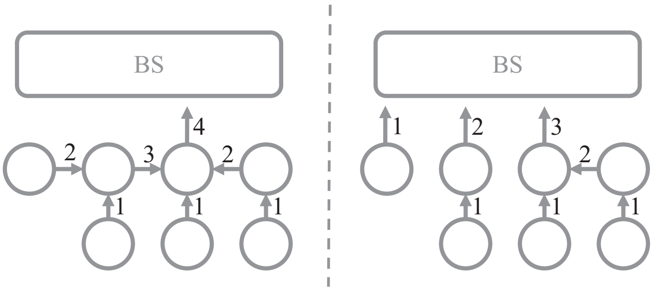
Proof: For a network with $N = 2^{k-1}$ nodes (where $k > 1$) that are organized into the DADCNS, its T_{DCP} is expressed as $T_{\text{DCP}} = \lceil 1 + \log_2 N \rceil = k$. Split the network into two sub-clusters with sizes $N_1 = N_2 = \frac{N}{2}$. The T_{DCP} values of the sub-clusters are expressed as $T_{\text{DCP1}} = T_{\text{DCP2}} = k - 1$. The CHs of the two sub-clusters have to access the BS at different time-slots and the overall T_{DCP} value of the network remains as $T_{\text{DCP}} = k$. The theorem is proved. ■

Based on *Theorem 3* and *Example 2*, splitting a network with $N = 2^{k-1}$ nodes (where $k > 1$) will not yield any improvement to the overall T_{DCP} value. Nevertheless, splitting the aforementioned network may, under some circumstances, reduce the total communication distance. The effects of splitting and not splitting a network with $N = 2^{k-1}$ nodes will be evaluated and discussed shortly.

IV. The Proposed Algorithm

From *Theorem 1*, it is observed that splitting a network into two with size difference less than one will not increase the overall T_{DCP} value of the network. Therefore, it is possible to further split the smallest sub-cluster without having any negative impact on T_{DCP} as long as the size of the smallest sub-cluster is nonzero. Furthermore, according to *Theorem 2*, by further splitting the smallest sub-cluster, it is possible to reduce the overall T_{DCP} if $N_1 = 2^{k-1} + 1$ (where $k > 1$).

Example 3: Consider a network with $N = 7$. By organizing the nodes into the DADCNS with a single tree (see Fig. 3(a)), the BS takes $T_{\text{DCP}} = \lceil 1 + \log_2 7 \rceil = 4$ to complete a DCP. By splitting the network into sub-clusters with $N_1 = 3$ and $N_2 = 4$, the CHs of the two sub-clusters will try to access the BS at the same time-slot. To avoid collision, one of them will postpone its transmission by one time-slot. Therefore, the overall T_{DCP} value of the network remain as 4. Splitting the smallest sub-cluster again (i.e. the sub-cluster with $N_1 = 3$), the network will result in three sub-clusters (see Fig. 3(b)). The sub-clusters will access the BS at time-slot 1, 2, and 3, respectively. The overall T_{DCP} value of the network is reduced to 3.



(a) (b)

Fig. 3. Networks with $N = 7$ nodes organized using the DADCNS (a) without splitting and (b) with splitting (2 times). Empty circles are representing wireless sensor nodes while circles with labels "BS" are representing base stations. Arrows are representing data links and the numbers next to the arrows are representing the transmission schedules.

Based on these observations, a network splitting algorithm is designed as follows.

Step–1 Given a set of nodes S. If $|S| = 1$, terminate the algorithm. Otherwise continue to Step–2.

Step–2 Divide S into two sets, S_1 and S_2, using a k–means clustering algorithm, such that $S_1 \cap S_2 = \emptyset$ and $S_1 \cup S_2 = S$. With loss of generality, assume $|S_1| \leq |S_2|$.

Step–3 If $|S_2| - |S_1| > 1$, move $\lfloor \frac{|S_2|-|S_1|}{2} \rfloor$ nodes from S_2, which are closest to the centroid of S_1, to S_1. Otherwise, continue.

Step–4 Organize S_2 into the DADCNS using the top-down approach proposed in [1].

Step–5 Set $S_1 \rightarrow S$ and return to Step–1.

In Step–2, a k–means clustering algorithm is used to divide the set S into two clusters based on the Euclidean coordinates of the nodes. Noted that although k–means clustering algorithms tend to form clusters of similar sizes, they cannot guarantee the exact sizes of the resulting clusters. To ensure $|S_1|$ and $|S_2|$ will not be deviated by more than one node, a greedy-based refinement process is introduced in Step–3. The refinement process tries to obtain the desirable cluster sizes of S_1 and S_2 by moving $\Phi = \lfloor \frac{|S_2|-|S_1|}{2} \rfloor$ nodes from S_2 to S_1. To avoid largely increase the communication distance among the nodes in S_1, these Φ nodes should have a minimum Euclidean distance to the centroid of S_1. Due to the tendency of forming clusters of similar sizes, the value of Φ is expected to be small. Therefore, results obtained from Step–3 should not deviate a lot from their optimum values. After dividing S into two clusters, the larger cluster (i.e. S_2) will be organized into a DADCNS using the top-down approach proposed in [1]. The result will be a tree structure with its CH communicating with the BS directly. The smaller cluster S_1 will becomes S and the splitting algorithm will continue. The algorithm will terminate when $|S| = 1$. The flow chart of the proposed network splitting algorithm is shown in Fig. 4.

As suggested in *Theorem 3*, the sub-splitting process will not yield any reduction in T_{DCP} if $N = 2^{k-1}$ (where $k > 1$).

Example 4: Consider a network with $N = 2^{3-1} = 4$ as shown in Fig. 5(a). With the DADCNS, the BS takes 3 time-slots to complete a DCP. Splitting the network into two sub-clusters, such that $N_1 = N_2 = 2^{2-1} = 2$. The CHs of the sub-clusters will access the BS at time-slots 2 and 3, respectively. Further splitting the smaller sub-cluster will obtain the structure shown in Fig. 5(b). The three clusters will access the BS at time-slots 1, 2, and 3, respectively. The overall T_{DCP} value remains unchanged.

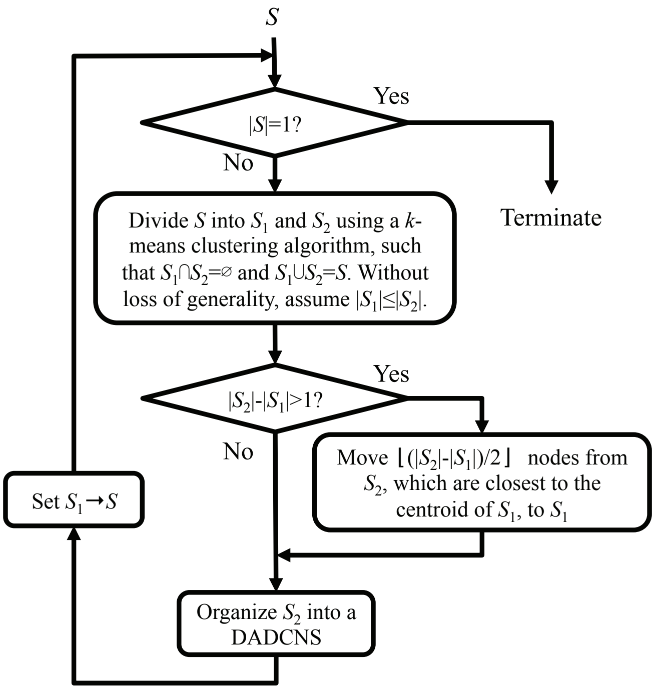

S
Yes
|S|=1?
No
Terminate
Divide S into S1 and S2 using a k-means clustering algorithm, such that S1∩S2=∅ and S1∪S2=S. Without loss of generality, assume |S1|≤|S2|.
Yes
|S2|-|S1|>1?
No
Move ⌊(|S2|-|S1|)/2⌋ nodes from S2, which are closest to the centroid of S1, to S1
Set S1→S
Organize S2 into a DADCNS

Fig. 4. The flow chart of the proposed network splitting algorithm.

Based on the above observations, the proposed network splitting algorithm can therefore be terminated earlier with the following modifications to Step–1 of the original algorithm.

Step–1' Given a set of nodes S. If $|S| = 2^{k-1}$, where $k > 1$, organize S into the DADCNS using the top-down approach proposed in [1] and terminate the algorithm. Otherwise continue to Step–2.

Other steps of the original algorithm are remaining unchanged. The flow chart of the proposed network split-

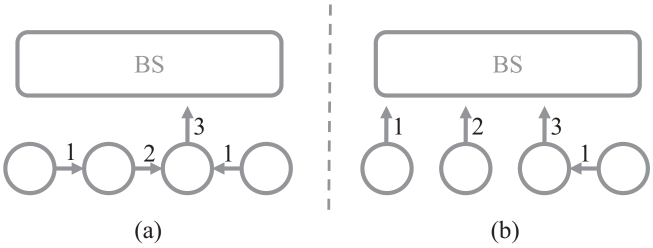


Fig. 5. Networks with $N = 4$ nodes organized using the DADCNS (a) without splitting and (b) with splitting (2 times). Empty circles are representing wireless sensor nodes while circles with labels "BS" are representing base stations. Arrows are representing data links and the numbers next to the arrows are representing the transmission schedules.

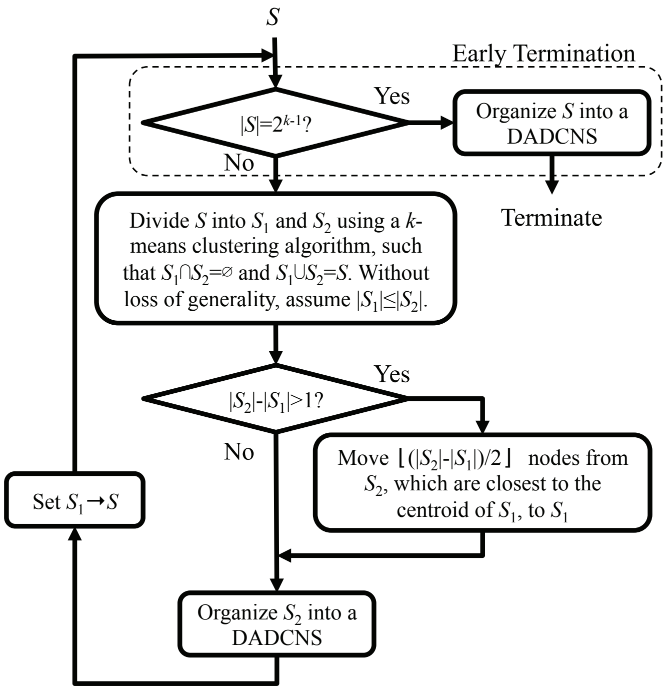

S
Early Termination
Yes
$|S|=2^{k-1}$?
Organize S into a DADCNS
No
Terminate
Divide S into S_1 and S_2 using a k-means clustering algorithm, such that $S_1 \cap S_2 = \varnothing$ and $S_1 \cup S_2 = S$. Without loss of generality, assume $|S_1| \leq |S_2|$.
Yes
$|S_2|-|S_1|>1$?
No
Move $\lfloor(|S_2|-|S_1|)/2\rfloor$ nodes from S_2, which are closest to the centroid of S_1, to S_1
Set $S_1 \rightarrow S$
Organize S_2 into a DADCNS

Fig. 6. The flow chart of the proposed network splitting algorithm with early termination.

ting algorithm with early termination is shown in Fig. 6. By not splitting networks with $N = 2^{k-1}$ (where $k > 1$), fewer clusters will be generated. As a result, fewer nodes will be involved in long distance communications (i.e. CH→BS) and thus can reduce energy consumption.

V. Simulations

The performance of the proposed network splitting algorithms is evaluated using computer simulations. In the simulations, the duration of a DCP (T_{DCP}) and the total squared communication distance (Ψ) are used as performance indicators. The duration of a DCP is expressed as the total number of time-slots required by the BS to collect data from all the nodes in the network.

The total squared Euclidean distance is expressed as

$$\Psi = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} c_{ij} d_{ij}^2 + \sum_{k=1}^{N} {c'}_k {d'}_k^2. \tag{2}$$

Here, c_{ij} is an indicator showing the existence of a connection between the i^{th} and the j^{th} nodes. If a connection exists, $c_{ij} = 1$, else $c_{ij} = 0$. Variable d_{ij} is representing the Euclidean distance between the i^{th} and the j^{th} nodes. Similarly, ${c'}_k$ indicates the existence of a connection between the BS and the k^{th} node, while ${d'}_k$ represents the Euclidean distance between the BS and the k^{th} node. The total squared Euclidean distance is a good estimation for the total energy consumption of a WSN [1].

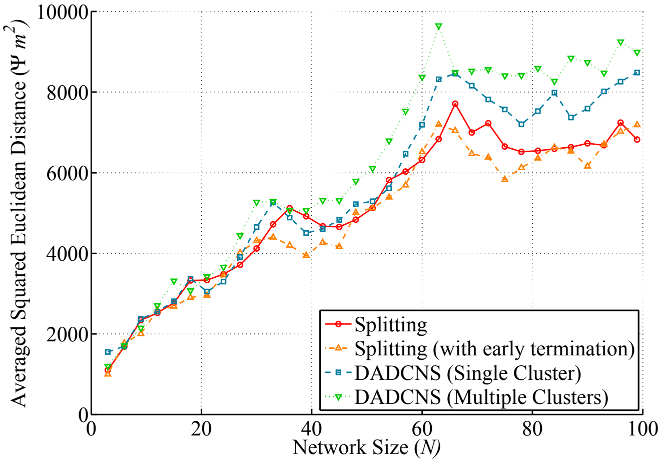


Fig. 7. The averaged squared Euclidean distance of networks with the DADCNS formed by different algorithms.

A. Simulation Settings

Simulations were conducted in Matlab. In each simulation, a network with N wireless sensor nodes are distributed randomly on a square sensing terrain with 50×50 m^2, which has its center and one of its corners located at (25, 25) m and (0, 0) m, respectively. The BS is located at the center of the terrain, which tries to collect data from all the nodes in the networks. In the simulations, performance of the original DADCNS will be used as references. The DADCNS will be constructed as a single cluster and multiple clusters using the top-down network formation approaches proposed in [1] and [7], respectively. In order to evaluate the effect from network size (N) to the performance of networks with different network structures, N is varying from 3 to 99 with a step-size of 3. In the simulations, all the network formation algorithms are implemented in a centralized manner. Results presented in this paper are the averaged values of 50 simulations.

B. Simulation Results

Simulation results are shown in Fig. 7 and Fig. 8. As expected, the Ψ values of networks with different network formation algorithms increase with N. Algorithms based on k–means clustering algorithms can, in general, obtain lower values of Ψ than their counterparts. The margin increases further as N increases. The Ψ values of networks with the two algorithms based on k–means

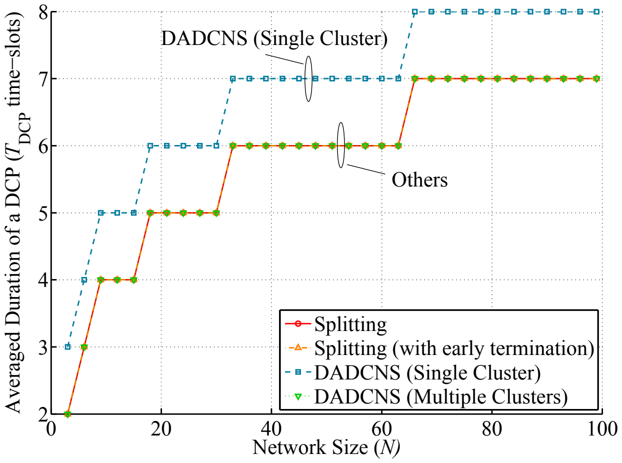


Fig. 8. The averaged duration of a data collection process in networks with the DADCNS formed by different algorithms. Note that except networks with the DADCNS (Single Cluster), results of networks with other structures are overlapping (the lower curve).

clustering algorithms are close to each other. Comparatively, networks with the proposed splitting algorithm with early termination perform slightly better. Similarly, the T_{DCP} values of all networks under test increase with N. Simulation results on the T_{DCP} values concur with the analyses in Section III. Networks with multiple clusters can achieve lower values of T_{DCP} than those with single cluster.

VI. Discussions

The averaged T_{DCP} values of networks with the proposed splitting algorithms and the DADCNS (Multiple Clusters) are, most of the time, lower than those of networks with the DADCNS (Single Cluster). Note that the T_{DCP} values of all the algorithms under test are equal when $N = 2^{k-1}$ ($k > 1$), which are not shown in Fig. 8 due to the step-size applied on N. By organizing wireless sensor nodes into multiple clusters with different sizes, their CHs can finish collecting data from their cluster members and return fused data to the BS at an interleaved-manner. From the perspective of minimizing T_{DCP}, it is desirable to form multiple clusters with different sizes whenever possible. As proved in Section III, dividing networks with $N = 2^{k-1}$, where $k > 1$, will not yield any further reduction in T_{DCP}. Therefore, the performance of the proposed splitting algorithms, with or without early termination, on reducing T_{DCP} are the same.

In general, the Ψ values of networks with the proposed splitting algorithms are lower than those of networks with the original DADCNS. The top-down network formation approaches used in [1] and [7] try to avoid long connections by means of finding the heaviest k–subgraph in the network. These approaches perform well for small networks. However, they tend to be trapped in local optimums as N increases. For the proposed network splitting algorithms, the k–means algorithms will first try to organize nodes located closely into clusters. Within each cluster, the nodes will then be organized into a single DADCNS using the approach in [1]. The k–means algorithms adopted in the proposed splitting algorithms can break down a large network into sub-clusters and avoid having long inter-connections within each of them. The Ψ values of networks managed by the proposed splitting algorithm with early termination are, on average, lower than those managed by the proposed splitting algorithm without early termination. With early termination, fewer sub-clusters will be formed. As a result, fewer long communication links will be formed between CHs and the BS.

VII. Conclusions

It has been shown that for networks with the delay-aware data collection network structure, splitting the networks into multiple sub-clusters may shorten the duration of its data collection process and its intra-communication distance. Conditions and limitations for splitting networks with such network structure are analyzed and presented. Two novel network splitting algorithms based on k–means algorithms are proposed. The performances of the proposed algorithms are evaluated using computer simulations. Simulation results show that the proposed network splitting algorithms can effectively reduce the duration of a data collection process and can significantly reduce the intra-communication distance of a network.

References

[1] C.-T. Cheng, C. K. Tse, and F. C. Lau, “A delay-aware data collection network structure for wireless sensor networks,” *Sensors Journal, IEEE*, vol. 11, no. 3, pp. 699–710, March 2011.

[2] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Trans. Wireless Communications*, vol. 1, no. 4, pp. 660–670, October 2002.

[3] S. Lindsey and C. S. Raghavendra, “PEGASIS: Power-efficient gathering in sensor information systems,” in *Proc. IEEE Conf. Aerospace*, vol. 3, Big Sky, Montana, USA, March 2002, pp. 1125–1130.

[4] M. Strasser, A. Meier, K. Langendoen, and P. Blum, “Dwarf: Delay-aware robust forwarding for energy-constrained wireless sensor networks,” in *Distributed Computing in Sensor Systems*, ser. Lecture Notes in Computer Science, J. Aspnes, C. Scheideler, A. Arora, and S. Madden, Eds. Springer Berlin Heidelberg, 2007, vol. 4549, pp. 64–81.

[5] P. Djukic and S. Valaee, “Delay aware link scheduling for multi-hop tdma wireless networks,” *IEEE/ACM Trans. Netw.*, vol. 17, no. 3, pp. 870–883, Jun. 2009.

[6] C.-T. Cheng and C. K. Tse, “An analysis on the delay-aware data collection network structure using pareto optimality,” in *2012 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC 2012)*, Sanya, Hainan, China, October 2012, pp. 348–352.

[7] C.-T. Cheng, H. Leung, and P. Maupin, “A delay-aware network structure for wireless sensor networks with in-network data fusion,” *Sensors Journal, IEEE*, vol. 13, no. 5, pp. 1622–1631, May 2013.

[8] Y. Shen, Y. Li, and Y. hua Zhu, “Constructing data gathering tree to maximize the lifetime of unreliable wireless sensor network under delay constraint,” in *2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC 2012)*, Limassol, Cyprus, August 2012, pp. 100–105.

[9] S. Sivaranjani, S. Radhakrishnan, and C. Thangaraj, “Adaptive delay and energy aware data aggregation technique in wireless sensor networks,” in *Mobile Communication and Power Engineering*, ser. Communications in Computer and Information Science, V. Das and Y. Chaba, Eds. Springer Berlin Heidelberg, 2013, vol. 296, pp. 41–49.