Maximizing Quality of Aggregation in Delay-Constrained Wireless Sensor Networks

Bahram Alinia, Hamed Yousefi, Mohammad Sadegh Talebi, and Ahmad Khonsari

***Abstract*—In this letter, both the number of participating nodes and spatial dispersion are incorporated to establish a bi-objective optimization problem for maximizing the quality of aggregation under interference and delay constraints in tree-based wireless sensor networks (WSNs). The formulated problem is proved to be NP-hard with respect to Weighted-sum scalarization and a distributed heuristic aggregation scheduling algorithm, named SDMAX, is proposed. Simulation results show that SDMAX not only gives a close approximation of the Pareto-optimal solution, but also outperforms the best, to our knowledge, existing alternative proposed so far in the literature.**

***Index Terms*—Wireless sensor networks, quality of aggregation, combinatorial optimization, Pareto-optimality.**

I. Introduction

DATA aggregation is a key, yet time-consuming functionality introduced to conserve energy by reducing packet transmissions in WSNs [1]. Although energy efficiency is usually considered the primary concern here, recent emerging applications (e.g., habitat monitoring, forest fire monitoring, battlefield surveillance, and health monitoring) have witnessed the necessity of real-time data aggregation and allotted substantial effort to timely gathering of data. In this respect, the crucial phenomenon to tackle is communication collision as it evinces a fundamental role to introduce long latency in data aggregation through limiting simultaneous transmissions. Hence, we concentrate on the TDMA aggregation scheduling problem above the MAC layer to ensure the interference-free transmissions in a WSN rooted at the sink of aggregation.

Besides, an immediate consequence of delay-constrained data gathering is that all sensor nodes cannot participate in aggregation. Thus, there is an important trade-off between the tolerable delay and quality of aggregation, defined in the next section, obtained by the sink. In line with this objective, Hariharan and Shroff [2] have recently studied the problem of maximizing the number of nodes whose packets have been accounted for at the sink subject to deadline and interference constraints. However, they do not consider the inherent data redundancy between neighboring nodes and the way it affects the quality of aggregation. We believe that *to pick only a few sensors, those with minimum degree of spatial correlation provide more information, and so deserve higher priority to participate in aggregation.*

Taking into account the effect of data redundancy due to spatial correlation, in this letter, we consider both the number of source participating nodes and their spatial dispersion as the underlying merit factors for aggregation. This leads to the extension of the framework of [2] to a bi-objective combinatorial optimization problem. We prove that solving the formulated problem, i.e., finding a Pareto-optimal solution, is at least as hard as the Maximum Edge Weighted Clique Problem (MEWCP), which is known to be NP-hard [3]. In order to find an approximate Pareto-optimal solution, we then propose SDMAX as a heuristic distributed algorithm that (1) selects source participants and (2) assigns waiting times for a delay-constrained interference-free aggregation in tree-based WSNs. Simulation results corroborate that SDMAX appreciably approximates a Pareto-optimal solution and verify the superiority of our approach.

II. System Model and Definitions

A. WSN Model

We consider a WSN whose topology is a tree $G = (V \cup \{S\}, E)$, where S is the sink node and also the root of the tree, V is the set of sensor nodes with $|V| = N$, and E is the set of communication links each having capacity one[1]. We assume that the system is time-slotted and synchronized and that a transmission takes exactly one time slot. We consider a one-hop node-exclusive interference model [2], where simultaneous transmissions over links having a node in common cause interference. We stress that the structure as well as our assumptions is fairly typical in almost all studies on interference-free aggregation scheduling in WSNs.

In our delay–constrained scenario, data should be received by sink S before a pre-determined deadline of D time slots. To devise a feasible aggregation scheme, we assign a deadline of $W_i \in \{0, \ldots, D-1\}$ time slots to each node $i \in V$. Moreover, $X[i, W_i]$ denotes the maximum number of source successors[2] (including i only if i is a source) that i can account for aggregation. For each i, let $H(i) \subseteq V$ be a set consisting of node i and all its predecessors[3] (except the sink) in the aggregation tree. Moreover, let $\mathcal{K}_i$ be the set of children of node i with cardinality K_i. We use a binary decision variable T_i,

Manuscript received July 17, 2013. The associate editor coordinating the review of this letter and approving it for publication was G. Giambene.

B. Alinia is with the Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran (e-mail: b.alinia@ece.ut.ac.ir).

H. Yousefi is with the Department of Computer Engineering, Sharif University of Technology, Tehran, Iran (e-mail: hyousefi@ce.sharif.edu).

M. S. Talebi is with the School of Electrical Engineering, KTH The Royal Institute of Technology, 100–44 Stockholm, Sweden (e-mail: mstms@kth.se).

A. Khonsari is with the School of ECE, College of Engineering, The University of Tehran, and the School of Computer Science, IPM, Tehran, Iran (e-mail: ak@ipm.ir).

Digital Object Identifier 10.1109/LCOMM.2013.100413.131631

[1] This structure is mostly employed in WSNs not only because of its appropriateness for a network with one or a few sink nodes, but also because its simplicity is very attractive when network resources are limited. Moreover, it improves energy efficiency and avoids issues like double counting.

[2] Successors of each node i include all the nodes in the sub-tree with root i excluding i itself.

[3] Predecessors of each node i include all the nodes in the path from i to the sink excluding i itself.

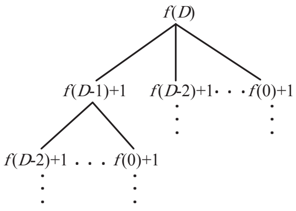

Fig. 1: $f(D)$ calculation.

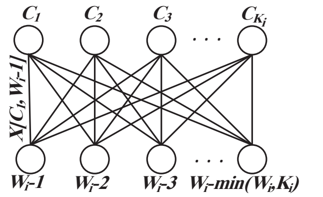

Fig. 2: MWM [2].

where $T_i = 1$ if node i is a source and $T_i = 0$ otherwise. We also introduce a binary variable n_i such that $n_i = 1$ indicates that node $i : (i, j) \in E$ is a participant (i.e., it is allowed to send data to its parent). We then define $\vec{n} = [n_i, i \in V]$. Let $V^{\text{leaf}} \subseteq V$ be the set of all leaf nodes and $V^{\text{sel-src}}(\vec{n}) \subseteq V$ be the set of source nodes *selected* for data aggregation (i.e., source participants). Indeed, $i \in V^{\text{sel-src}}(\vec{n})$ if i (with $T_i = 1$) and all of its predecessors are participating or more formally, $V^{\text{sel-src}}(\vec{n}) = \left\{ i \in V : T_i = 1 \text{ and } \prod_{j \in H(i)} n_j = 1 \right\}$.

B. Quality of Data Aggregation

We consider two metrics to quantify the quality of aggregation. The first one is the (normalized) number of source nodes that participate in data aggregation expressed as

$$J_{\text{number}}(V^{\text{sel-src}}) = \frac{|V^{\text{sel-src}}|}{N} = \frac{1}{N} \sum_{i \in V} T_i \prod_{j \in H(i)} n_j.$$

For this metric, we have the following result.

Theorem 1: For an imposed deadline D and every $V^{\text{sel-src}} \subseteq V$, we have $J_{\text{number}}(V^{\text{sel-src}}) \leq \frac{2^D - 1}{N}$.

Proof: To prove the theorem, we start by finding $f(D)$ such that $X[S, D] \leq f(D)$. Considering the deadline and interference model, the maximum number of source participants is attained if all nodes are sources and for each node $i \in V \backslash V^{\text{leaf}} \cup \{S\}$ with the parent P_i (if i is not sink) we have $K_i \geq W_i$, where $W_i < W_{P_i}$. This guarantees that in each time slot, we have the maximum number of simultaneous transmissions. With $K_i = W_i$, for each i, the tree has the minimum number of nodes while the upper bound of source participants is still achievable. Hence, the sink should have D source children with assigned waiting times $D - 1, \ldots, 0$, where they should support participation of $f(D-1)+1, \ldots, f(0)+1$ nodes in data aggregation, respectively. Therefore, to calculate $f(D)$, we obtain a recursion as shown in Fig. 1.

Each node $i \in V \backslash V^{\text{leaf}}$ with waiting time $W_i < D$ can choose at most W_i children whose waiting times are distinctly chosen from $\{0, 1, \ldots, W_i - 1\}$. Thus, for any $D > 0$, we can write $f(D)$ recursively as $f(D) = [f(D-1)+1] + [f(D-2)+1] + \cdots + [f(0)+1]$. By induction on D, we then show that $f(D) = 2^D - 1$. For $D = 0$, we get $f(0) = 2^0 - 1 = 0$, so the statement for $D = 0$ is true. Now, suppose that $f(D-1)$ is true for $D > 0$, i.e. $f(D-1) = [f(D-2)+1] + \cdots + [f(0)+1] = 2^{D-1} - 1$. We thus have

$$\begin{aligned} f(D) &= [f(D-1)+1] + [f(D-2)+1] + \cdots + [f(0)+1] \\ &= 2f(D-1) + 1 = 2 \times 2^{D-1} - 1 = 2^D - 1. \end{aligned}$$

Finally, division by N gives the desired result. ■

The second metric for quality of aggregation is (normalized) spatial dispersion defined as follows. A widely accepted model characterizes data correlation between nodes i and j as $\rho_{ij} = e^{-\alpha d_{ij}^{\beta}}$, where d_{ij} is the Euclidean distance between i and j, and $\alpha \in [0, 1]$ and $\beta \in [1, 2]$ are the correlation parameters [4]. Here, we define the joint participation score of two nodes i and j in aggregation as $\frac{T_i T_j}{\rho_{ij}}$.

Definition 1: For the set of selected source nodes $V^{\text{sel-src}}$, we define the *spatial dispersion* as

$$\delta(V^{\text{sel-src}}) = \sum_{i,j \in V^{\text{sel-src}}, i<j} \frac{1}{\rho_{ij}}.$$

In light of Definition 1, we consider the normalized spatial dispersion as $J_{\text{dispersion}}(V^{\text{sel-src}}) = \frac{\delta(V^{\text{sel-src}})}{\delta_{\max}(V)}$, or equivalently

$$J_{\text{dispersion}}(V^{\text{sel-src}}) = \frac{1}{\delta_{\max}(V)} \sum_{i,j \in V, i<j} \frac{T_i T_j}{\rho_{ij}} \prod_{u \in H(i)} n_u \prod_{v \in H(j)} n_v,$$

where $\delta_{\max}(V)$ is the maximum value of spatial dispersion, $\delta(.)$, over all feasible sets of source nodes.

III. Optimization Problem

Inspired by the problem formulation presented in [2], our goal is to find $V^{\text{sel-src}}(\vec{n})$ so as to simultaneously maximize the two metrics given aggregation constraints. We cast this as the following bi-objective optimization problem[4]:

$$\begin{aligned} Z: \ & \text{maximize (w.r.t. } \mathbb{R}^2_+) \quad \left(J_{\text{number}}(V^{\text{sel-src}}), J_{\text{dispersion}}(V^{\text{sel-src}}) \right) \\ \text{s.t.} \quad & \forall i \in \{S\} \cup V \backslash V^{\text{leaf}} : \forall C \subseteq \{(j,i) : (j,i) \in E\}, \\ & \sum_{j:(j,i) \in C} n_j + \min_{j:(j,i) \in C} W_j \leq W_i, \\ & W_i \in \{0, 1, \ldots, D-1\}, \quad \forall i \in V, \\ & W_S = D, \qquad \vec{n} \in \{0, 1\}^N, \end{aligned}$$

where decision variables are $\vec{n}$ and $\vec{W}$. The inequality constraint is directly imposed by the node-exclusive interference model and forces a parent node to receive at most one packet from its children at each time slot. Let $\mathcal{F}_Z \in \mathbb{R}^2_+$ be the set containing all feasible objective values of Z. An optimal solution to Z is a maximum element of $\mathcal{F}_Z$. Existence of such a maximum element depends on the choice of underlying parameters of Z and is not guaranteed in general. Thus, Z may not possess an optimal solution and we resort to finding its Pareto-optimal solutions, which correspond to the maximal elements of $\mathcal{F}_Z$. Potentially, problem Z has several Pareto-optimal solutions, where in terms of the objectives involved, all of which are equally good[5]. Here, we exploit Weighted-sum solution notion to obtain a scalarized variant of Z. Hence, we concentrate on NP-hardness with respect to Weighted-sum solution notion and prove that the Weighted-sum scalarized problem of Z is NP-hard, which gives rise to the NP-hardness of Z with respect to Weighted-sum solution notion.

[4] Indeed, problem Z is a combinatorial vector optimization problem that involves the cone $\mathbb{R}^2_+$ [5].

[5] Pareto-optimal solutions of a multi-objective optimization problem can be obtained by turning it into a single objective one through scalarization approaches such as Weighted-sum and Lexicographical (see, e.g., [6]). These approaches are indeed particular *solution notions* for multi-objective optimization problems, where each solution notion induces a corresponding NP-hardness notion. We refer the reader to [6] for a detailed discussion.

IV. Approximate Solution

In this section, we analyze the complexity of problem Z and propose a distributed algorithm to find its approximate Pareto-optimal, hence Pareto-suboptimal, solution.

Theorem 2: Problem Z is NP-hard with respect to Weighted-sum solution notion.

Proof: We show that solving Z with Weighted-sum notion is generally at least as hard as solving Maximum Edge Weighted Clique Problem (MEWCP). Indeed, scalarized Z with respect to Weighted-sum includes, as special case, the problem of finding a set of participating source nodes with the maximum spatial dispersion. The latter leads to solving MEWCP, which is a well-known NP-hard problem [3]. Now, consider MEWCP in a complete graph G_c with n vertices $v_1, \dots, v_n$ and let c_{ij} be the weight of edge between nodes v_i and v_j. The goal of MEWCP is to select at most b nodes from G_c such that the sum of weights of the selected edges is maximized. Here, b is given by the problem and an edge is said to be selected if its connecting nodes are chosen. Polynomial-time reduction of MEWCP to Weighted-sum scalarization of Z can be accomplished by constructing network graph G from G_c as follows: G has a star topology of $N = n$ sensors where sensor s_i corresponds to vertex v_i in graph G_c and is directly connected to the sink. Without loss of generality, suppose that d_{ij} is equal to c_{ij}, $\forall i, j \in \{1, \dots, n\}$. All the sensors are source nodes and the sink placed in an arbitrary position imposes a deadline of D, where D is the maximum size of clique in MEWCP. Thus, using a node-exclusive interference model, at most D sensors can participate in data aggregation. Now, we need to select the sensors such that according to the spatial dispersion definition, the pairwise sum of joint participation scores is maximized. Here, finding a set of D participants with the maximum spatial dispersion is equal to finding a set of D nodes whose sum of weights of corresponding edges in G_c is maximized; this is exactly what MEWCP aims to solve. Finally, the solution to scalarized Z yields the solution to MEWCP. This completes the proof. ■

Based on the algorithm of [2], we propose SDMAX as a distributed heuristic algorithm to find an approximate Pareto-optimal solution of Z. SDMAX (shown as Algorithm 1) works in two main phases. The first phase is a bottom-up procedure that calculates a scalar $X[i, W_i]$ as well as a matrix $Y[i, W_i]$ for each node i and $0 \leq W_i \leq D-1$. It does the same for the sink for $0 \leq W_S \leq D$. Here, $Y[i, W_i]$ is a $X[i, W_i]$-by-2 matrix to store the location information of $X[i, W_i]$ source participating successors of node i in data aggregation tree.

The problem of finding $X[i, W_i]$ is a Maximum Weighted Matching (MWM) problem in a bipartite graph with sets of vertices U and R (see Fig. 2 in the previous page) [2]. Here, $U = \mathcal{K}_i = \{C_1, \dots, C_{K_i}\}$ is the set of the children of node i and $R = \{W_i - 1, W_i - 2, \dots, W_i - \min(W_i, K_i)\}$ includes the possible waiting times. The edge connecting node $u \in U$ to waiting time $r \in R$ has weight $X[u, r]$. For waiting time assignment, the MWM problem is formalized as follows:

$$\mathsf{MWM}(U, R, X): \quad \max_{\text{matching } \mathcal{M} \subset U \times R} \sum_{(u,r) \in \mathcal{M}} X[u, r]$$

In the second phase of SDMAX, the waiting times are assigned using the top-down procedure. Here, each parent node i assigns an initial schedule of waiting times to its children such that the maximum number of source participants in its sub-tree is taken (MWM problem). The rationale is that based on our simulations in Section V, in 83% of scenarios, the maximum value of spatial dispersion is achieved by one of the feasible sets of source participants with maximum size. However, there may be more than one feasible schedule resulted in this maximization, each of which with different set of nodes and different spatial dispersion. The assignments can be accomplished in $\binom{K_i}{W_i} \times W_i!$ and $K_i!$ different ways for $K_i > W_i$ and $K_i \leq W_i$, respectively. In order to make different assignments comparable, we first need to quantify the objective of problem Z. Hence, given a parameter $\lambda \in [0, 1]$, we define QoA as the scalarized objective of Z as follows:

$$QoA(V^{\text{sel-src}}, \lambda) = \lambda J_{\text{number}}(V^{\text{sel-src}}) + (1 - \lambda) J_{\text{dispersion}}(V^{\text{sel-src}}).$$

Having obtained an initial assignment meanwhile in top-down procedure, we proceed to find a better assignment in terms of QoA via local search. We restrict the local search to 1-*diff alternative* assignments defined next. Given a waiting time assignment A_0, we let a 1-diff alternative to A_0 be a feasible assignment in which the waiting time of each child differs at most by one time slot from A_0.

Algorithm 1: SDMAX

```
Input: Tree $G$, deadline $D$, weight $\lambda$
Output: $\vec{n}$, $\vec{W}$
//Initialization
for all $l \in V^{\text{leaf}}$ and $W_l = 0, \dots, D-1$ do
    $X[l, W_l] \leftarrow T_l$
    if $T_l == 1$ then
        $Y[l, W_l] \leftarrow$ position of node $l$

//Bottom-Up Procedure
for all $i \in V \backslash V^{\text{leaf}}$ do
    for $W_i = 0, \dots, D-1$ do
        Find $X[i, W_i]$ by solving $\mathsf{MWM}(\mathcal{K}_i, R, X)$, where
        $R = \{W_i - 1, W_i - 2, \dots, W_i - \min(W_i, K_i)\}$.
        Merge all $Y[j, W_j]$s, where $j$ is a selected child of $i$ to
        produce $Y[i, W_i]$.

Calculate $X[S, r]$ and $Y[S, r]$, $\forall r \in \{0, 1, \dots, D\}$.
$D \leftarrow D - \max\{0 \leq r \leq D : X[S, D] = X[S, D - r]\}$
$W_S \leftarrow D$
//Top-Down Procedure
for all $i \in \{S\} \cup V \backslash V^{\text{leaf}}$ do
    Use the solution of
    $\mathsf{MWM}(\mathcal{K}_i, \{W_i - 1, W_i - 2, \dots, W_i - \min(W_i, K_i)\}, X)$ to
    find initial values of $\{W_u : u \in \mathcal{K}_i\}$.
    Form $A_0 = [C_1, \dots, C_{\min(W_i, K_i)}]$ of initially selected children
    such that $\forall l, C_l \in \mathcal{K}_i$ and $W_{C_l} > W_{C_{l+1}}$.
    for $j = 1, \dots, \min(W_i, K_i)^2$ do
        Generate $A_j$ as a 1-diff alternative to $A_0$.
    Assign waiting times of children by finding $A_l^\star$, where
    $A_l^\star = \arg\max_{l=1,\dots,\min(W_i,K_i)^2} QoA(A_l, \lambda)$.
```

Lemma 1: For a parent node p with waiting time W_p and K_p children, the cost of finding $O(\min(W_p, K_p)^3)$ 1-diff alternatives to an initial assignment A_0 is $O(\min(W_p, K_p)^3)$.

The proof of the above lemma can be found in the technical supplement [7]. Now, to make SDMAX more efficient, a parent only checks $\min(W_i, K_i)^2$ 1-diff alternatives produced by exchanging the initial waiting times and finally selects the one that maximizes $QoA(V^{\text{sel-src}}, \lambda)$. Thus, it is likely that

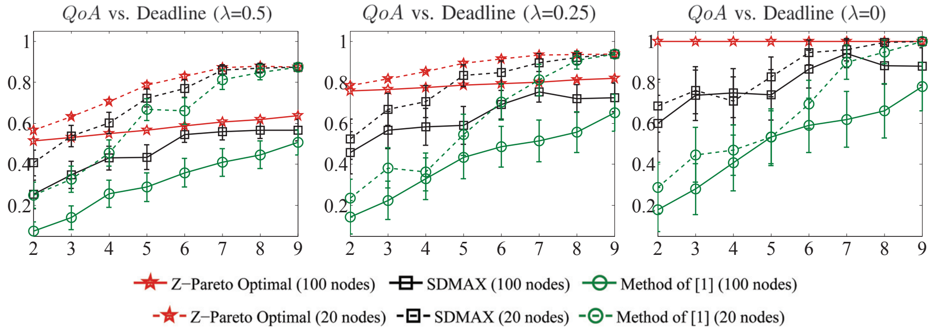

Fig. 3: QoA vs. Deadline.

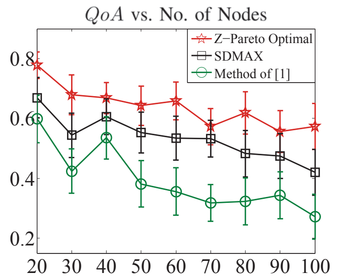

Fig. 4: QoA vs. Network Size (Number of Nodes)

such a minor modification of the initial waiting times yields another schedule that also maximizes J_{number}.

Theorem 3: Given an imposed deadline D, the time complexity of SDMAX algorithm in a graph of height h and maximum degree v is $O(hND + hv^2(D+v)\log v)$.

Proof: In the bottom-up procedure of SDMAX, calculation of $X[i, W_i]$ in each node i for $0 \leq W_i \leq D-1$ costs $O(v^2(D+v)\log v)$ [2]. Moreover, to produce $Y[i, W_i]$, we need to merge all $Y[j, W_j]$s, where j is a selected child of i. Considering that the maximum size of $Y[.,.]$ is N-by-2, $Y[i, W_i]$ is calculated at a cost of $O(ND)$ for $0 \leq W_i \leq D-1$. Since nodes at the same level of the tree can work in parallel, the total cost of bottom-up procedure will be $O(hND + hv^2(D+v)\log v)$. In the top-down procedure, for each node i, we assign initial waiting times of children by solving an MWM problem, which is already done in the bottom-up procedure. Then, by modifying the initial assignment, we produce and check $\min(W_i, K_i)^2$ possible 1-diff alternatives in $O(\min(W_i, K_i)^3)$. Thus, we pay a top-down cost of $O(h\min(D, v)^3)$ which is dominated by the bottom-up cost. Finally, SDMAX has a time complexity of $O(hND + hv^2(D+v)\log v)$. ■

V. Simulation Results

In this section, the performance of SDMAX algorithm is evaluated via simulation. We consider two different scenarios to mainly show how SDMAX outperforms the algorithm of [2] in terms of quality of data aggregation by considering spatial dispersion. At the first scenario, we consider two random tree-based WSNs (with the same network density) comprising 20 and 100 nodes, uniformly at random scattered in square fields of $100 \times 100\ m^2$ and $500 \times 500\ m^2$ and sink locations of $(50, 100)$ and $(250, 500)$, respectively. We designated 80% of nodes as sources uniformly at random and let $\alpha = 0.05$ and $\beta = 1.5$. Fig. 3 portrays QoA against D for different values of λ along with the 95% confidence interval, where each point is obtained by averaging the results of 50 experiments. Evidently, QoA decreases as λ increases from 0 to 0.5. The main reason is that term $\lambda J_{\text{number}}$ in QoA becomes dominant when λ increases, but 20% of nodes (i.e., non-source nodes) yields a decrease. Furthermore, QoA improves as D increases as a higher deadline allows participation of more nodes. According to the results of Fig. 3, SDMAX achieves 0.78-optimal and 0.8-optimal solutions to Z in terms of QoA, and thereby outperforms the algorithm of [2] on average by 26% and 37% in cases of 20 nodes and 100 nodes, respectively.

At the second scenario, we consider different node densities in a square field of $250 \times 250\ m^2$, where sink is located at $(125, 250)$. Fig. 4 depicts QoA against the number of nodes (increasing from 20 to 100 with step 10) along with the 95% confidence interval. Each data point in this figure is the average of 50 experiments, where for each one, we have selected D and λ uniformly from intervals $[2, 9]$ and $[0, 1]$, respectively. According to this figure, in all cases, SDMAX strictly acts better than the algorithm of [2] on average by 27%.

VI. Conclusion

In this letter, we framed a bi-objective optimization problem to maximize the quality of data aggregation in delay-constrained WSNs. we proposed SDMAX as a distributed approximation algorithm. Simulation results show that our proposed approach obtains an approximate Pareto-optimal solution where, in terms of QoA, it is 0.81-optimal on average. Future work will look at extending the present study to account for reliability in noisy sensor networks.

References

[1] X. Xu, X. Li, X. Mao, S. Tang, and S. Wang, “A delay-efficient algorithm for data aggregation in multihop wireless sensor networks,” *IEEE Trans. Parallel Dist. Syst.*, vol. 22, no. 1, pp. 163–175, 2011.

[2] S. Hariharan and N. B. Shroff, “Maximizing aggregated information in sensor networks under deadline constraints,” *IEEE Trans. Automat. Contr.*, vol. 56, no. 10, pp. 2369–2380, 2011.

[3] B. Alidaee, F. Glover, G. Kochenberger, and H. Wang, “Solving the maximum edge weight clique problem via unconstrained quadratic programming,” *Eur. J. Oper. Res.*, vol. 181, no. 2, pp. 592–597, 2007.

[4] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, “On network correlated data gathering,” in *Proc. 2004 IEEE INFOCOM*, pp. 2571–2582.

[5] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[6] K. Fleszar, C. Glaßer, F. Lipp, C. Reitwießner, and M. Witek, “Approximability and hardness in multi-objective optimization,” *CiE*, pp. 180–189, 2010.

[7] “Technical Supplement 1.” Available: http://people.kth.se/∼mstms/TS1.pdf