Optimization Framework with Reduced Complexity for Sensor Networks with In-Network Processing

Sepideh Nazemi[†], Kin K. Leung[†] and Ananthram Swami[‡]
[†]Imperial College, London, UK. Email: {s.nazemi10, kin.leung}@imperial.ac.uk
[‡]U.S. Army Research Laboratory, Adelphi, MD, USA. Email: ananthram.swami.civ@mail.mil

***Abstract*—We propose a framework with highly reduced complexity for optimizing in-network processing (INP) for quality of information (QoI) in wireless sensor networks (WSNs). INP provides a platform by processing (e.g., fusing, aggregating or compressing) the data along the transmission routes in the sensor network. This can reduce the volume of transmitted data, therefore optimizing utilization of energy and bandwidth. However, such data processing must ensure that the end result can meet given QoI requirements. We formulate the QoI-aware INP problem as a non-linear optimization problem to identify the optimal degree of data compression at each sensor node subject to satisfying a QoI requirement for the end-user. The formulation arranges all involved sensor nodes in the tree where data is transfered and processed from nodes to their parent nodes toward the root node of the tree. Under the assumption of uniform parameter setting, we show that the processing tree can be collapsed into a linear graph where the number of nodes represents the node levels of the original processing tree. Numerical example are provided to illustrate the performance of the proposed approach.**

I. Introduction

In-Network Processing (INP) is a common technique, which enables and supports possible solutions for optimising the utilization of network resources such as bandwidth and energy. Initially, INP aims to aggregate (e.g., compress, fuse and average) and route data from different sources with the objective of reducing resource consumption, particularly energy, thereby increasing network lifetime [1]. While INP can optimize network resource usage, it is very fundamental to consider how such data processing affects the Quality of Information (QoI) for the end-user. In spite of the lack of generic and comprehensive definition of QoI [2], it is agreed that QoI is reflected by a collection of attributes e.g., accuracy, completeness, reliability, timeliness, etc. [2]. In [3], QoI is defined as a multi-dimensional measure that quantitatively and qualitatively identifies the degree to which data is suited for an application or a decision-making process.

Since information is critical to many decision-making/controlling applications, information quality (e.g., accuracy, timeliness or completeness) can affect the quality of the decision or controlling process in these applications and unquestionably, processing information can decrease or increase the quality of information [4].

In recent related work, Edwards et al. [3] present the notion of QoI-aware networking. They suggest that QoI-aware networking algorithm must consider how data are transformed into information and the impact that the network may have on this transformation. [5] provides a centralized approach for network delivery and timeliness with a certain information accuracy. [6] points out that the delivered QoI is a function of attributes that sources provide as a result of observing events and channel attributes, and proposes a centralized transmission schedule to maximize the utility function of QoI. However, the existing work does not consider INP. As for distributed approaches, Eswaran et al. [7] apply the network utility maximum (NUM) framework to dynamically determine the optimal compression and fusion factors for INP and specify the nodes in a path as the optimal places for performing data processing. A common assumption among the existing NUM work is the concavity of the utility functions, which may not be valid for many communication networks and applications [8]. Applying INP, Chen et al. [9] explore joint scheduling and data aggregation by assuming that full aggregation is always possible with an additive utility. Silberstein et al. [10] present a fully distributed method for implementing many-to-many aggregation in a sensor network that minimizes the communication cost by optimally balancing a combination of multicast and in-network aggregation. However, they only consider binary aggregation options, i.e., no aggregation and full aggregation which may deteriorate the QoI at the end user. Both [9] and [10] use integer programming techniques for making data aggregation decisions. However, they do not consider any QoI metrics. Barr et al. [11] investigate the energy saving by lossless data compression. They show that the choice of how and whether to compress is not obvious and it is depends on hardware characteristics as well as software factors. [12] estimates the conditions under which aggregation is preferable to no aggregation. Sharam et al. [13] design a distributed algorithm that makes joint compression and transmission decision with the goal of minimizing energy consumption.

There are many obstacles toward designing a generic method for delivering satisfactory level of QoI to the end-users while INP applied in the network. QoI attributes can be affected indirectly/directly by other QoI attributes. i.e., there is a trade-off among different QoI metrics. For example, consider timeliness and precision as QoI attributes in a system. One way to improve timeliness is to reduce the volume of the data being transferred by reducing data resolution and as a result the data precision is decreased [3]. Furthermore, selecting appropriate attributes of QoI and range

This research was partly sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001.

of acceptance level requires consideration of the system and architectural perspective of the specific application [14].

INP and QoI have received a lot of attention in recent years. However, a gap still exist in research regarding to how dynamically control INP to satisfy QoI requirements for the end-user. In this regard, this paper concentrates on development of an optimization framework to facilitate controlling the degree of INP at intermediate nodes and guarantee that QoI required by the end-user is provided.

The rest of the paper is organized as follows. We formulate the problem in section II. The proposed solution is presented in section III and we present the numerical result and discuss the performance of the proposed approach in section IV. Conclusions and future work are presented in section V.

II. Problem formulation

Since the degree at which a node can aggregate its received data is one of the main determining factors for QoI [4], we aim to introduce an optimization framework that can determine the optimal degree of aggregation at each node, while meeting the QoI required by the end-user.

A. Assumptions

We assume that a data aggregation tree is formed among all involved nodes in the network after the user requests information from the network. The root node, *r*, of the tree is responsible for delivering the required information to the end-user. It is assumed that only leaf nodes generate data and the remaining nodes in the tree process and forward aggregated data toward the root node. Furthermore, we assume data generated at the leaf nodes has some degree of redundancy due to spatial and temporal correlations among sensors. Therefore, it is possible to aggregate data as a means to reduce the consumption of network resources.

We assume that the total energy consumption of a node consists of the energy spent in receiving, computing and transmitting its data. Although by aggregating a huge amount of data into a smaller volume, it is possible to reduce energy consumption for transmission and reception, the greater degree of data aggregation, the higher is the computational energy consumption. Therefore, an energy trade-off exists among the amount of energy that each node spends on data reception, transmission and computation. We define the ratio of the volume of the aggregated data at each node to that of all data received from its children nodes as the *data reduction rate*, which is denoted by δ lying between 0 and 1. Without considering the details of the data aggregation technique in use, the reduction rate at each node works as a regulator to justify the amount of energy that node is willing to spend on data transmission and computation subject to the end-user's QoI requirement.

B. Global Optimization

We formulate the problem of QoI-awre INP as a non-linear optimization problem. Specifically, our goal is to choose the data reduction rates at all nodes involved in the aggregation process in order to minimize the total cost for the whole aggregation tree, while ensuring that the level of QoI at the end-user exceeds a specified threshold. The global optimization problem (to be referred as the GO problem) is as follows:

$$\begin{array}{ll} \min\limits_{\boldsymbol{\delta}} & \sum\limits_{\mathrm{i=1}}^{\mathrm{N}} f_i(\delta_i, y_i) \\ \text{s.t.} & q_r(\delta_r, y_r) \geq \gamma \end{array}, \tag{1}$$

where *N* is the total number of nodes in the data aggregation tree. $f_i(\delta_i, y_i)$ is the cost function of node i, which is a function of the volume of input data y_i received from its children nodes and the data reduction rate δ_i at node i. $\boldsymbol{\delta}$ is a vector of reduction rates for all nodes. $q_r(\delta_r, y_r)$ specifies the QoI function. Since the root node, *r*, is responsible for delivering the required information to the end-user, the QoI constraint is associated only with the root node. γ indicates the QoI requirement threshold specified by the end-user. Even though the problem has only a single QoI constraint associated with the root node, the data reduction rate must be chosen optimally at every node so that the QoI constraint for the end-user can be satisfied. Since it is assumed that only leaf nodes generate data, we have

$$y_i = \sum_{\mathrm{j \in C_i}} \delta_j y_j \;\; for\ i = 1\ to\ N, \tag{2}$$

where C_i denotes the set of children nodes of node i.

Substituting (2) into the objective function and the inequality constraint in (1) reveals that the GO problem is intrinsically a nonlinear programming problem.

III. Proposed Solution

Following the assumptions made in [15], let $f_i(\delta_i, y_i)$ and $q_r(\delta_r, y_r)$ be defined as follows. Let f_i denote the total energy consumption of node i as

$$f_i = e_{iR} + e_{iC} + e_{iT}, \tag{3}$$

where e_{iR}, e_{iC} and e_{iT} denote the energy spent in receiving, aggregating (computing) and transmitting data by each node i respectively. Then, we assume

$$e_{iR} = s(y_i) = \epsilon_R y_i, \tag{4}$$

$$e_{iT} = g(y_i, \delta_i) = \epsilon_T y_i \delta_i, \tag{5}$$

$$e_{iC} = k(y_i, \delta_i) = \epsilon_C y_i l_i(\delta_i), \tag{6}$$

$$q_r = y_r \delta_r, \tag{7}$$

where ϵ_R, ϵ_C and ϵ_T are respectively the energy consumed in receiving, processing and transmitting one unit of data. $l_i(\delta_i)$ is a scaling function which is a decreasing differentiable function of the reduction rate.

The intuition behind the definition of e_C is that (i) computation cost is proportional to input data volume y_i as entire input data needs to be scanned at least once before any computation, and (ii) typically, the greater the degree of data aggregation (i.e., the smaller amount of aggregated data produced after processing), the higher the energy consumption for computation. In addition, the influence of the reduction rate δ on e_C is highly dependent on the type of data being aggregated, the actual aggregation functions and the characteristics of computation hardware [11]. For instance, processing high-quality video usually consumes more energy than processing data such as temperature or humidity measurements. Therefore, to adequately capture these dependencies, a scaling function $l_i(\delta_i)$ is included in e_C. In this paper, we consider $l_i(\delta_i) = (\frac{1}{\delta_i} - 1)$ for $\delta_i > 0$ as the scaling function.

Clearly, it is challenging to specify a general form to model the computation energy cost. There may be a concern that the proposed assumptions are unable to adequately adjust to all the characteristics of communication and computation in the network (e.g., coding and processing functions). However, these assumptions are rational to characterize behaviour of energy spending at node *i* when INP is applied in the network. As an example, Fig.1 shows the total energy consumption of node i based on assumptions (3)-(6). As illustrated, there is a trade-off between computation and transmission costs. At $\delta_i = 1$, node i sends all received data to its parent. (i.e., $e_C = 0$). A smaller δ_i means that node *i* reduces the volume of its transmitted data. Therefore, it can reduce its total energy consumption since it needs to transmit less data. However, after a certain point, reducing more data will result in spending more energy due to the need for more computation, even though it needs to spend less energy on transmission. Note that the node parameters ϵ_C and ϵ_T can affect this trend. We will investigate other cost models in our future work.

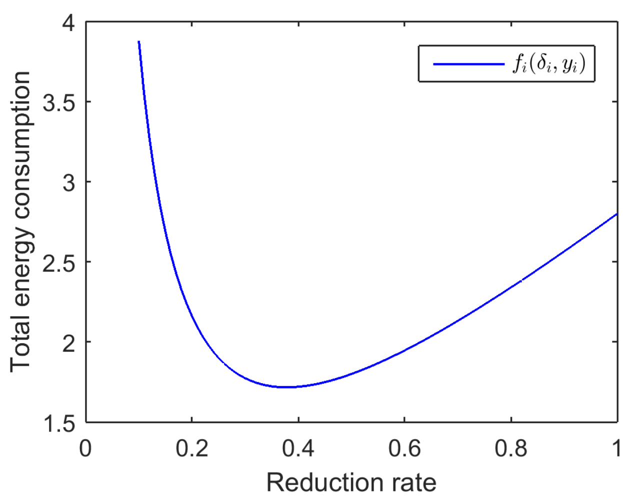

Fig. 1: Total energy consumption function associated to node i given (3)-(6), $l_i(\delta_i) = (\frac{1}{\delta_i} - 1), \epsilon_C = 0.01, \epsilon_T = 0.05, \epsilon_R = 0.02$ and $y_i = 40$.

Using (7), we restate the GO problem (1) as follows.

$$\begin{aligned} \min_{\boldsymbol{\delta}} \quad & \sum_{\mathrm{i=1}}^{\mathrm{N}} f_i(\delta_i, y_i) \\ \text{s.t.} \quad & \delta_r y_r \geq \gamma \end{aligned} . \tag{8}$$

Following the assumptions in (3) to (6), the GO (8) is a Signomial Programming (SP) optimization problem, which is an extension of geometric programming (GP) optimization problem, that in general is non-convex and NP-hard. The definition of SP is provided in Appendix C. We prove in the following that under some conditions, it is possible to significantly reduced the complexity of the GO problem (8) from the processing tree of N nodes to one with $\log_c N + 1$ nodes. Whit such complexity reduction it will not be a big burden for each node to solve a reduced version of the problem. This forms the basis of our proposed distributed approach for the optimal solution for the GO problem. Toward this end, we introduce the following Lemma and Theorem.

Lemma. *Assume a perfect c-ary aggregation tree with depth* h *where: (a) beside the leaf nodes, each node has exactly* c *of children nodes, (b) every leaf node generates the same amount of data, while all other nodes only receive data from their children nodes, process the received data and transfer the aggregated data to one's parent node, and (c) the network and energy consumption parameters in (4) to (6) are identical for all nodes. Under these assumptions, all the nodes at the same level of the tree (say level* i*) have the identical data reduction rate* δ_i *in the optimal solution for the GO problem.*

Proof. See Appendix A. ∎

The following Theorem shows that under the assumptions made in the Lemma, the optimal solution for the GO problem associated with a symmetric aggregation tree is identical to that for a linear graph of $h + 1$ nodes. For example, the optimal solution for the GO problem for the tree in Fig. 2 (a) is identical to that for the linear graph in Fig. 2 (b) which represent a very significant reduction in complexity.

Theorem. *For the perfect c-ary aggregation tree with* h *levels of nodes and the assumptions made in the Lemma, the optimal reduction rates for all nodes at the same tree level for the GO problem (8) are identical to that of the node at the corresponding level in a linear graph with* $h + 1$ *nodes. Furthermore, the optimal rates can be obtained by solving the following problem of the linear graph with* $h + 1$ *nodes:*

$$\begin{aligned} \mathop{min}_{\boldsymbol{\delta}} \quad & Y \sum_{\mathrm{j=0}}^{\mathrm{h}} \Big(\prod_{k=j+1}^{h} \delta_k \Big) \, f_j(\delta_j) \\ s.t. \quad & Y \prod_{k=0}^{h} \delta_k \geq \gamma \end{aligned} , \tag{9}$$

where $\delta_{h+1} = 1$, $Y = yc^h$ *and* $f(\delta_j) = \epsilon_R + \epsilon_T \delta_j + \epsilon_C l(\delta_j)$.

Y is the total amount of data generated at the leaf nodes. h is the number of node levels in the aggregation tree which is equal to $\log_c N$ and $\boldsymbol{\delta}$ is a vector of reduction rates associated with each level of the tree.

Proof. See Appendix B. ∎

In general, complexity of an optimization problem solver depends on number of variables of the optimization problem. Moreover, if the problem is nonconvex and solving the problem includes convexifying the problem, then the number of steps to approximate the convex form of the problem can affect the complexity of the algorithm as well. For example, to solve

SP problem, Derakhshani et al. [16] proposed an algorithm with the complexity of $O(pn^3)$ where n is the total number of variables in the problem and p is the total number of terms in all monominals and posynomials in objective and constraints functions, which is also a function of n. Accordingly, since our model reduction technique can reduce the problem of N variables to the equivalent problem of $\log_c N+1$ variables, the complexity of the optimization solver will be reduced notably.

IV. Numerical Results

In this section, we present numerical results to confirm the proposed approach. Given the assumptions (3) to (7), we consider in the first experiment 15 homogeneous nodes forming a 4-level perfect binary aggregation tree with parameters $\epsilon_C = 0.01, \epsilon_T = \epsilon_R = 0.02$ for each node. We used the Matlab optimization tools to obtain the the optimal solution for the GO problem (8). Each leaf generates $y = 15$ packets and the user's QoI requirement is considered to be $\gamma = 5$. Fig.2 (a) shows the optimal data reduction rate associated with each node for the GO problem (8). As expected, the optimal reduction rates for all nodes at the same level of the tree are identical. Total energy consumption of the network is 8.23 units and the root (node *1*) produces exactly *5* packets. Fig.2 (b) illustrates the same network after applying our model as specified in the Theorem. As shown, the optimal reduction rates are identical to the original network. Note that the result provided in Fig.2 also has been confirmed by the exhaustive search algorithm.

The significance of this proposed solution for the global-optimization problem is explained as follows. In general, the problem of in-network processing is a non-convex optimization problem, for which no exact distributed solution technique has been found. Solving the problem by a centralized method does not scale in terms of algorithm complexity and incurs significant overhead in distributed environments. Under the assumption of identical parameters for all nodes, we have proved here that the optimal data reduction rates for a symmetrical tree of N nodes can be obtained by solving the corresponding problem of a linear graph of $\log_c(N)+1$ nodes. This represents a very significant reduction in complexity such that each node can obtain its optimal reduction rate by solving the much reduced problem locally. This new approach can be viewed as a new distributed technique to the very challenging resource allocation problem.

In addition, there are many parameters that can affect the results and performance of the system. In order to get an insight into the problem, in the next experiments, we examine the behaviour of the network under different network parameter settings by utilizing our model reduction technique.

Fig. 3 to Fig. 8 present variation of the reduction rates associated with the nodes located at different levels of the aggregation tree under different network parameter settings. A perfect binary aggregation tree with 1023 homogeneous nodes is considered where the root and leaf nodes are located at level *0* and *9*, respectively. Each leaf generates 15 packets. The QoI threshold is assumed to be 5 packets.

Typically, data transmission consumes more energy than data processing. Fig. 3 illustrates this condition. Since data processing is less expensive than transmission, therefore it is beneficial to reduce data at lower levels of the tree gradually as Fig. 3 presents. In Fig. 4, we consider that processing the data is more expensive than transmitting it. Similar to the previous experiment, the nodes at each level gradually reduce amount of received data. However, the amount of data reduction is not the same as in the previous experiment due to the high cost of data processing. The total energy consumption and provided QoI corresponding to each experiment can be viewed in Table I. In Fig. 5 we assumed that $\epsilon_T = \epsilon_C$. Fig.6 to Fig. 8 illustrate the extreme cases. That is when there is a big difference between the cost of processing one unit of data and transmitting it.

Figure Num.	Total Energy Consumption	Provided QoI
2	537.87	5
3	536.42	39.92
4	741.44	5
5	4547.14	5
6	3072	7680
7	155.6	5

TABLE I: Total energy consumption and provided QoI associated with data reduction rates given in Fig.3 to Fig.8

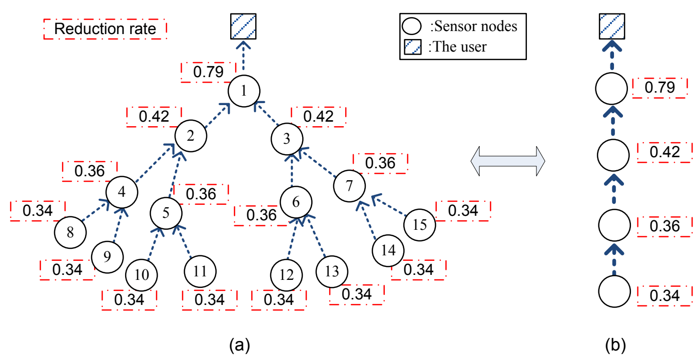

Fig. 2: (a) Reduction rates for all nodes in a symmetric aggregation tree. (b) Network topology after model reduction.

V. Conclusion

We formulate the problem of QoI-aware INP as a nonlinear optimization problem. The practical network assumptions revealed that the INP problem is a nonconvex problem which is an NP-hard. However, we prove that the optimal data reduction rates for a c-ary perfect tree of N homogeneous nodes can be obtained by solving the corresponding problem of a linear graph of $\log_c(N) + 1$ nodes. This represents a very significant reduction in complexity such that each node can obtain its optimal reduction rate by solving the much reduced problem locally. This new approach can be viewed as a new distributed technique to the very challenging resource-allocation problem. Future work includes the possible extension of the new technique to non-symmetrical settings and the continuation of our effort in pursuing the distributed solutions for the in-network processing problem.

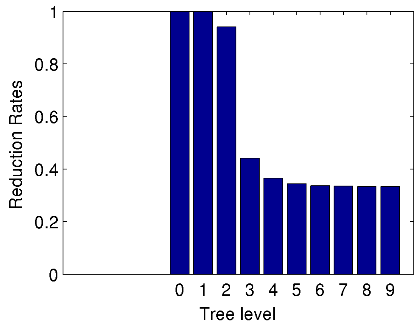

Fig. 3: $\epsilon_c = 0.01$,$\epsilon_r = \epsilon_t = 0.02$

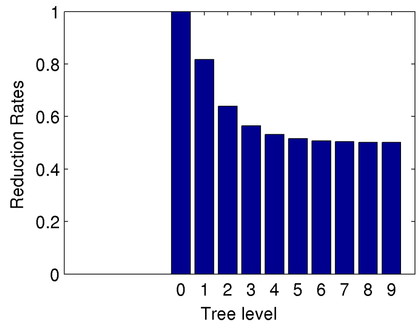

Fig. 4: $\epsilon_c = 0.02$,$\epsilon_r = \epsilon_t = 0.01$

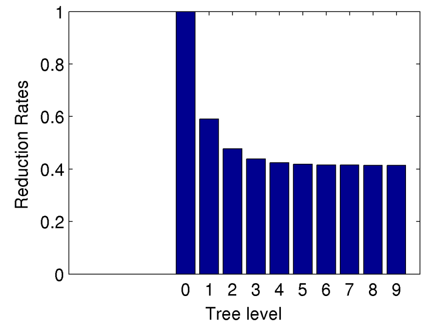

Fig. 5: $\epsilon_c = 0.02$,$\epsilon_r = \epsilon_t = 0.02$

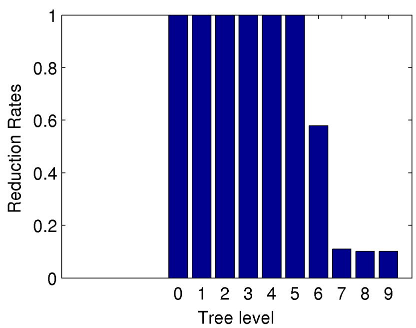

Fig. 6: $\epsilon_c = 0.01$,$\epsilon_r = \epsilon_t = 0.4$

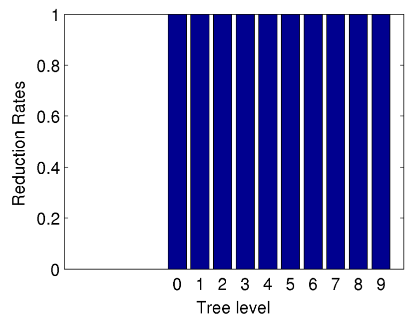

Fig. 7: $\epsilon_c = 0.4$,$\epsilon_r = \epsilon_t = 0.02$

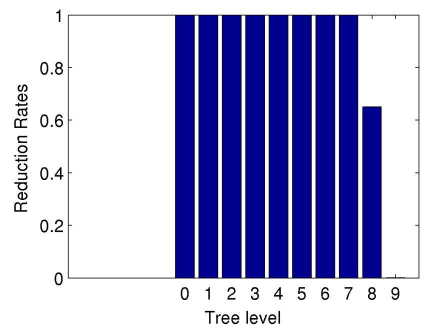

Fig. 8: $\epsilon_c = 0$,$\epsilon_r = \epsilon_t = 0.02$

APPENDIX

A. Proof of Lemma

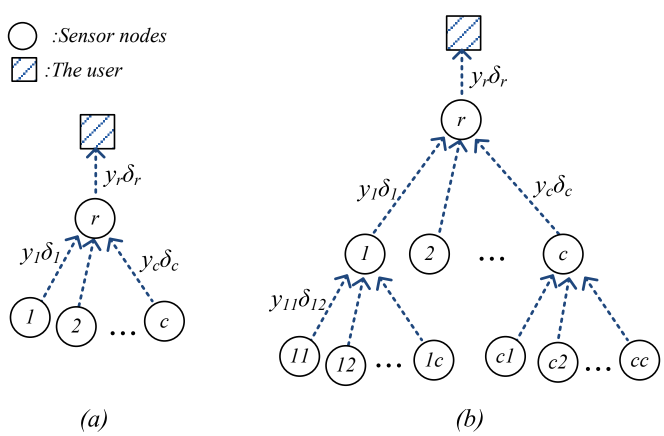

Fig. 9: (a) One level ($h = 1$) and (b) two level ($h = 2$) c-ary aggregation tree.

- Let $h = 1$ and c denote the number of children of the root node r as shown by Fig.9 (a). So that $N = c + 1$. Then the corresponding GO problem can be written as follows.

$$\min_{\boldsymbol{\delta}} \quad F(\boldsymbol{\delta}, y) = y\sum_{\text{i=1}}^{\text{c}} (\epsilon_R + \epsilon_T\delta_i + \epsilon_C l(\delta_i)) + y(\sum_{\text{i=1}}^{\text{c}}\delta_i)(\epsilon_R + \epsilon_T\delta_r + \epsilon_C l(\delta_r)),$$

s.t.

$$\delta_r(\sum_{\text{i=1}}^{\text{c}}\delta_i) \geq \gamma \quad (*)$$

At the optimum assuming that is at a point with $0 < \delta_k <= 1$),

$$\frac{\partial F(\boldsymbol{\delta}, y)}{\partial \delta_k} = y(\epsilon_T + \epsilon_C l'(\delta_k)) + y(\epsilon_R + \epsilon_T\delta_r + \epsilon_C l(\delta_r)) = 0,$$

$$for\ k = 1 \quad to \quad c$$

and (10)

$$\frac{\partial F(\boldsymbol{\delta}, y)}{\partial \delta_r} = y(\sum_{\text{i=1}}^{\text{c}}\delta_i)(\epsilon_T + \epsilon_C l'(\delta_r)) = 0\,. \qquad (11)$$

By assumption, $y \neq 0$ and $\delta_i > 0$. Considering (11) we have:

$$(\epsilon_T + \epsilon_C l'(\delta_r)) = 0\,.$$

Since we assume that $l(\delta)$ is a monotonic decreasing function, the optimal reduction rate δ_r^* must be unique. From (10) we can then conclude that δ_k^* is the same for each $k = 1 \ \ to \ \ c$, since only a unique value can satisfy (10).

- Let $h=2$ as illustrated by Fig.9 (b). Then we have:

$$\begin{aligned}\min_{\boldsymbol{\delta}} F(\boldsymbol{\delta},y) =& y\sum_{\mathrm{j=1}}^{\mathrm{c}}\sum_{\mathrm{i=1}}^{\mathrm{c}} (\epsilon_R+\epsilon_T\delta_{ji}+\epsilon_C l(\delta_{ji}))+\\ & y\sum_{\mathrm{j=1}}^{\mathrm{c}}(\sum_{\mathrm{i=1}}^{\mathrm{c}}\delta_{ji})[\epsilon_R+\epsilon_T\delta_j+\epsilon_C l(\delta_j)]+\\ & y(\sum_{\mathrm{j=1}}^{\mathrm{c}}\delta_j(\sum_{\mathrm{i=1}}^{\mathrm{c}}\delta_{ji}))(\epsilon_R+\epsilon_T\delta_r+\epsilon_C l(\delta_r))\\ \text{s.t.} & \\ & \delta_r(\sum_{\mathrm{j=1}}^{\mathrm{c}}\delta_j)(\sum_{\mathrm{i=1}}^{\mathrm{c}}\delta_{ji}) \geq \gamma\end{aligned} \quad (12)$$

At the optimum point we have:

$$\frac{\partial F(\boldsymbol{\delta},y)}{\partial \delta_r} = y(\sum_{\mathrm{j=1}}^{\mathrm{c}}\delta_j(\sum_{\mathrm{i=1}}^{\mathrm{c}}\delta_{ji}))(\epsilon_T+\epsilon_C l'(\delta_r)) = 0\ . \quad (13)$$

With the same argument as the case $h=1$, δ_r^* must be unique.

$$\frac{\partial F(\boldsymbol{\delta},y)}{\partial \delta_k} = y(\sum_{\mathrm{i=1}}^{\mathrm{c}}\delta_{ki})(\epsilon_T+\epsilon_C l'(\delta_k)) = 0\ . \quad (14)$$

By assumption $y(\sum_{\mathrm{i=1}}^{\mathrm{c}}\delta_{ki}) > 0$ and since $l(\delta_k)$ is a monotonic decreasing function, δ_k is the same for all $k=1 \quad for \quad c$.

$$\begin{aligned}\frac{\partial F(\boldsymbol{\delta},y)}{\partial \delta_{mn}} &= y(\epsilon_T+\epsilon_C l'(\delta_{mn}))\\ &+ y(\epsilon_R+\epsilon_T\delta_m+\epsilon_C l(\delta_m))\\ &+ y\delta_m(\epsilon_R+\epsilon_T\delta_r+\epsilon_C l(\delta_r)) = 0\end{aligned}\ , \quad (15)$$

Since δ_r is unique, δ_m are identical for all $m=1 \quad to \quad c$ and $l(\delta_{mn})$ is a monotonic decreasing function, the same δ_{mn} for all *m* and *n* will satisfy (15).

- **Remark:** The same argument can be applied for $h \geq 3$. Therefore, we shall omit them.

B. Proof of Theorem

At level *i*, there is c^i nodes. Each node receives a volume of data equal to $yc^{h-1}\prod_{k=i+1}^{h+1}\delta_k$ for $i=1 \ to \ h$, where $\delta_{h+1}=1$.

We assumed that energy cost per unit of data at node *j* is equal to $\epsilon_R+\epsilon_T\delta_j+\epsilon_C l(\delta_j)$. Therefore, the total cost at level *i* is:

$$\begin{aligned}\phi_i(\delta_i,y) &= c^i[yc^{h-i}\prod_{i+1}^{h+1}\delta_k](\epsilon_R+\epsilon_T\delta_i+\epsilon_C l(\delta_i))\\ &= yc^h(\prod_{i+1}^{h+1}\delta_k)f(\delta_i)\end{aligned}\ .$$

So, the total cost is

$$F(\boldsymbol{\delta_i},y) = yc^h\sum_{\mathrm{i=0}}^{\mathrm{h}}(\prod_{i+1}^{h+1}\delta_k)f(\delta_i) \quad .$$

Since we assume that the root *r* is located at level 0, the amount of data received by the root is equal to $Y_r = yc^h\prod_{k=1}^{h}\delta_k$. Therefore, the constrain in problem (8) can be written as follows

$$yc^h(\prod_{k=1}^{h}\delta_k)\delta_0 = yc^h(\prod_{k=0}^{h}\delta_k) \geq \gamma\ . \quad (16)$$

C. Signomial Programming

SP is a problem of minimizing a signomial ($s_0(\boldsymbol{x})$) subject to upper bound inequality constraints on signomials ($s_k(\boldsymbol{x})$, $k=1 \ to \ K$). As follows:

$$\begin{aligned}&\min_{\boldsymbol{x}} \ s_0(\boldsymbol{x})\\ &\quad \text{s.t.}\\ &\quad\quad s_k(\boldsymbol{x}) \geq \gamma \quad k=1 \ to \ K\end{aligned}\ ,$$

where a signomial is a sum of monomials,

$$s_k(\boldsymbol{x}) = \sum_{\mathrm{i=1}}^{\mathrm{N}} c_i\prod_{j}^{n} {x_j}^{a_i}\ ,$$

where $c \in R^N, a_i \in R, \forall i,j \quad x \in R_{++}$ [17].

References

[1] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, “In-network aggregation techniques for wireless sensor networks: a survey,” *Wireless Communications, IEEE*, vol. 14, no. 2, pp. 70–87, 2007.

[2] C. Bisdikian, L. M. Kaplan, and M. B. Srivastava, “On the quality and value of information in sensor networks,” *ACM Transactions on Sensor Networks (TOSN)*, 2013.

[3] J. Edwards, A. Bahjat, Y. Jiang, T. Cook, and T. F. La Porta, “Quality of information-aware mobile applications,” *Pervasive and Mobile Computing*, 2014.

[4] S. A. Ehikioya, “A characterization of information quality using fuzzy logic,” in *NAFIPS*. IEEE, 1999.

[5] E. Ciftcioglu, A. Yener, R. Govindan, and K. Psounis, “Operational information content sum capacity: Formulation and examples,” in *FUSION*. IEEE, 2011.

[6] E. N. Ciftcioglu and A. Yener, “Quality-of-information aware transmission policies with time-varying links,” in *MILCOM*. IEEE, 2011.

[7] S. Eswaran, J. Edwards, A. Misra, and T. F. L. Porta, “Adaptive in-network processing for bandwidth and energy constrained mission-oriented multihop wireless networks,” *Mobile Computing, IEEE Transactions on*, vol. 11, no. 9, pp. 1484–1498, 2012.

[8] G. Tychogiorgos, “Non-convex resource allocation in communication networks,” Ph.D. dissertation, Imperial College London, 2012.

[9] F. Chen, M. P. Johnson, A. Bar-Noy, and T. F. L. Porta, “Convergecast with aggregatable data classes,” in *SECON*. IEEE, 2012.

[10] A. Silberstein and J. Yang, “Many-to-many aggregation for sensor networks,” in *ICDE 2007*. IEEE, 2007.

[11] K. C. Barr and K. Asanović, “Energy-aware lossless data compression,” *ACM Transactions on Computer Systems (TOCS)*, 2006.

[12] L. Galluccio, S. Palazzo, and A. T. Campbell, “Efficient data aggregation in wireless sensor networks: an entropy-driven analysis,” in *PIMRC*. IEEE, 2008.

[13] A. B. Sharma, L. Golubchik, R. Govindan, and M. J. Neely, “Dynamic data compression in multi-hop wireless networks,” in *ACM SIGMETRICS Performance Evaluation Review*. ACM, 2009.

[14] M. A. Hossain, P. K. Atrey, and A. Saddik, “Context-aware qoi computation in multi-sensor systems,” in *MASS 2008*. IEEE, 2008.

[15] S. Nazemi, K. K. Leung, and A. Swami, “A distributed, energy-efficient and qoi-aware framework for in-network processing,” in *PIMRC*. IEEE, 2014.

[16] M. Derakhshani, X. Wang, T. Le-Ngoc, and A. Leon-Garcia, “Virtualization of multi-cell 802.11 networks: Association and airtime control,” *arXiv preprint arXiv:1508.03554*, 2015.

[17] M. Chiang, *Geometric programming for communication systems*. Now Publishers Inc, 2005.