A cross-layer framework for optimal delay-margin, network lifetime and utility tradeoff in wireless visual sensor networks

M. Tahir [a,b,*], R. Farrell [c]

[a] Department of Electrical Engineering, University of Engineering and Technology, Lahore 54890, Pakistan
[b] Al-Khawarizmi Institute of Computer Science, University of Engineering and Technology, Lahore 54890, Pakistan
[c] Institute of Microelectronics and Wireless Systems, National University of Ireland Maynooth, Ireland

ARTICLE INFO

ABSTRACT

For wireless multimedia sensor networks a distributed cross-layer framework is proposed, which not only achieves an optimal tradeoff between network lifetime and its utility but also provides end-to-end delay-margin. The delay-margin, defined as the gap between maximum end-to-end delay threshold and the actual end-to-end delay incurred by the network, is exploited by the application layer to achieve any desired level of delay quality-of-service. For optimal performance tradeoff an appropriate objective function for delay-margin is required, which is obtained by employing sensitivity analysis. Sensitivity analysis is performed by incorporating delay-margin in the end-to-end delay constraints while penalizing its price in the objective function. For distributed realization of proposed cross-layer framework, the optimal tradeoff problem is decomposed into network lifetime, utility and delay-margin subproblems coupled through dual variables. The numerical results for performance evaluation show that compromising network utility does not guarantee both lifetime and delay-margin improvement, simultaneously, for the set of operating points. Performance evaluation results also reveal that the fairness among different delay-margins, corresponding to different source–destination node pairs, can be improved by relaxing the end-to-end delay threshold.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

The increasing number and decreasing prices of imaging sensors is resulting in wide spread deployment of wireless multimedia sensor networks (WMSN), which can now potentially enable a large class of applications ranging from image-based tracking and surveillance to smart environments and elderly assistance in public spaces [1]. When the nodes in these networks are battery powered then it leads to an inherent requirement for network lifetime maximization. While improving the network life-time, performance demands in WMSNs try to maximize network utilization to improve the quality-of-service (QoS). This points toward the tradeoff that exists between network utilization and its lifetime [2]. The problem of joint network utility and lifetime maximization is formulated as a cross-layer optimization problem in [3] and the distributed solution is achieved by employing dual decomposition. The interaction between energy consumption and rate allocation as a constrained optimization problem is investigated in [2], which results in an optimal lifetime maximization and network utility tradeoff. An optimal flow control based on link congestion, energy efficiency and coordination price achieving utility-lifetime tradeoff, which balances the energy consumption among sensor nodes, is proposed in [4]. In a related work [5], the flow control optimization problem for wireless sensor networks is considered, which treats network lifetime and link

* Corresponding author at: Department of Electrical Engineering, University of Engineering and Technology, Lahore 54890, Pakistan. Tel.: +92 42 99029147; fax: +92 42 99250202.

E-mail addresses: mtahir@uet.edu.pk (M. Tahir), rfarrell@eeng.nuim.ie (R. Farrell).

Nomenclature

N	set of nodes	ϵ	fixed radio wakeup cost parameter
L	set of available communication links	γ	compression offset parameter
n_k	an arbitrary node	$P_{n_k}^{(comm)}$	communication power consumption at node n_k
$I(n_k)$	set of child nodes in the subtree of node n_k	$P_{n_k}^{(comp)}$	processing power consumption at node n_k
$T(n_k)$	set of nodes along shortest path from n_k to sink	γ_{max}	maximum permissible compression ratio
R_{n_k}	uncompressed data rate at node n_k	$D_{max}(n_k)$	maximum end-to-end delay threshold
R_{max}	maximum data transmission rate capability of RF interface	α	compression algorithm specific parameter
		$E_b^{(tx)}$	per bit transmission energy cost
$r_{n_k}^{(in)}$	compressed data arrival rate at node n_k	$E_b^{(rx)}$	per bit reception energy cost
$r_{n_k}^{(out)}$	compressed data transmission rate at node n_k	$E_b^{(comp)}$	per bit processing energy cost
ρ_{n_k}	data arrival to transmission rate ratio at node n_k	β	weighting parameter for optimization objective tradeoff
ω_{n_k}	delay margin parameter at node n_k		

interference as constraints in an asynchronous setting. The utility-lifetime tradeoff discussed in [4] as well as the flow control optimization with lifetime constraint considered in [5], do not take into account the effect of the network utilization and lifetime tradeoff on the end-to-end delay incurred due to data transportation, which is of prime importance in WMSNs. As a matter of fact, one of the key application demands in WMSNs is the end-to-end delay QoS provisioning.

In addition to the above mentioned lifetime-utility tradeoff, there exists another tradeoff between network lifetime and the end-to-end delay, which is considered separately in literature. The tradeoff between network lifetime and end-to-end delay is studied in [6], by employing an objective function which minimizes energy dissipation when subjected to the delay constraints. A data aggregation mechanism based on weighted fair queuing is proposed in [7], which provides service differentiation as well as ensures delay bounds, while meeting the end-to-end delay constraints. The techniques proposed in [6,7] can achieve an optimal lifetime-delay tradeoff, but fall short in achieving an optimal utilization of the network resources. Additionally, an improvement in the delay QoS performance by reducing the end-to-end delays without a lower bound, can compromise network utilization significantly. On the other hand maximizing network utilization can result in both network lifetime as well as delay QoS performance degradation. The proposed cross-layer framework bridges the gap that exists among the existing solutions proposed in literature, by providing an optimal tradeoff among network lifetime, utilization and delay-margin.

The end-to-end delay performance demands by the application layer are matched by requiring the maximum delay thresholds to be met. But the approach based on meeting the maximum delay thresholds can lead to a solution, where some of the actual end-to-end delays may approach the maximum delay thresholds. The optimal performance achieved using this approach is at the expense of a vulnerable WMSN, which is prone to the performance degradation due to delay threshold violations. This issue is addressed in our approach by introducing *delay-margin* through parameter ω in the end-to-end delay constraints. The introduction of

delay-margin provides performance robustness which can be measured as the gap between optimal end-to-end delay and corresponding delay threshold. Our approach based on providing optimal delay-margin, in contrast to meeting the end-to-end delay thresholds, gives flexibility to the application designer to achieve any desired level of QoS provisioning for end-to-end delays. However, we mention here that the introduction of parameter ω is at the expense of slightly degraded network utilization and lifetime performance.

To realize the network lifetime, utility and delay-margin tradeoff, we have formulated a multi-objective optimization problem, which captures delay-margin in the end-to-end delay constraints through parameter ω and its price is penalized in the objective function [8]. For a given network utility objective function, we use *sensitivity analysis* to determine the optimal objective function for delay-margin. The application designer can achieve an optimal tradeoff among the contending network utility, lifetime and delay-margin objectives using the proposed cross-layer framework. It is worth mentioning that the proposed approach is more suitable for wireless sensor and actuator type of networks, where certain control actions (actuation) are required to be performed in a timely fashion. One good example is of a camera based distributed tracking network where pan-tilt-zoom of a camera as well as hand-over from one camera to another are to be performed in real-time for tracking an event of interest. The remainder of the paper is organized as follows.

The network lifetime, node power consumption and end-to-end delay models are outlined in Section 2. A cross-layer framework for optimal lifetime, network utilization and delay-margin tradeoff is constructed in Section 3. Section 3 also details problem decomposition into sub-problems and the usage of sensitivity analysis in determining the delay-margin objective function. Performance evaluation results are provided in Section 4 and we conclude our findings in Section 5.

2. System model

The network is modeled as a tree graph $G(N,L)$ with $|N| - 1$ sensing nodes and a single sink node n_{sink} (will also

be termed as root node), and L is the set of existing communication links among all node pairs. The notation $|x|$ is used to represent the number of elements in set x. Data aggregation being an essential part of wireless sensor networks in general and that of WMSNs in particular can be implemented effectively by using a tree topology for the network. It is also observed that tree based routing is efficient mechanism for data transportation from the sensing nodes to the sink node [9–11]. For instance, the authors in [10] have proposed a centralized lifetime performance improvement algorithm that employs minimum spanning tree to perform routing while achieving data aggregation. The disadvantage of centralized implementation in [10] has been addressed in [9] by constructing approximate minimum spanning tree topologies while either using the position or the distance metric for one-hop neighbors in a distributive manner. Based on this approximate spanning tree the actual routing tree is constructed for data gathering. We refer the reader to [11] for a survey of tree based data gathering schemes.

For each sensor node n_k, $n_k \in N$, the set $I(n_k)$ represents the collection of nodes in the subtree of node n_k, while $T(n_k)$ is the set of nodes along the shortest path from n_k to the n_{sink}. A node n_k is termed as a *leaf* node if $I(n_k)$ is an empty set and a *relay* node otherwise. Each sensor node n_k, has an uncompressed data arrival rate, R_{n_k}, from its on board sensors. The node compresses the data before transmission to exploit the sensor data correlation and the resulting compressed data rate is $r_{n_k}^{(in)} \in \mathbf{r}^{(in)}$. The compressed data from the local sensor as well as from the sensors in the node's subtree can be transmitted at a transmission rate of $r_{n_k}^{(out)} \in \mathbf{r}^{(out)}$. We use a variable data transmission rate, as most of the wireless transceivers today provide a range of data transmission rates. For instance, Texas Instruments CC1101 has data transmission rates from 1.2 kbps to 500 kbps. Same is true for IEEE 802.11 data transceivers.

To evaluate network lifetime we need to find maximum power consumption at a sensor node. To do that, power consumption at each sensor node, n_k, is obtained from communication power $\left(P_{n_k}^{(comm)}\right)$ and computation power $\left(P_{n_k}^{(comp)}\right)$ costs. The communication power consumption, for a tree like network topology, is given by

$$P_{n_k}^{(comm)}(\mathbf{r}^{(out)}) = E_b^{(tx)} r_{n_k}^{(out)} + E_b^{(rx)} \sum_{n_j \in I(n_k)} r_{n_j}^{(out)} + \epsilon. \quad (1)$$

In (1), $E_b^{(tx)}$ and $E_b^{(rx)}$ are the per bit data transmission and reception energy costs, respectively and ϵ is a constant representing the fixed radio wakeup cost. The per bit energy consumption costs are assumed fixed using the fact that all the nodes have uniform transmission ranges and no RF power control for data transmission is performed. In defining the computational power consumption $\left(P_{n_k}^{(comp)}\right)$ model at node n_k, we consider the raw sensor data compression cost as the main contributor to the computational power consumption. It is straight forward to realize that $P_{n_k}^{(comp)}$, will increase with an increase in the uncompressed data rate R_{n_k}. In addition, achieving a higher compression ratio, $R_{n_k}/r_{n_k}^{(in)}$, for a given distortion tolerance level, will require more processing resulting in an increase in $P_{n_k}^{(comp)}$

[12,13]. This is can be justified as each of the uncompressed raw sensor data block is at least scanned once, for the purpose of frequency transformation, by the data compression algorithm being used. Further processing of transformed data will be a function of the compression ratio to achieve the desired compression level. The collective power consumption, in the two data processing steps involving frequency transformation and achieving desired compression level is modeled as

$$P_{n_k}^{(comp)}(\mathbf{r}^{(in)}) = R_{n_k} \left[\left(\frac{R_{n_k}}{r_{n_k}^{(in)}} \right)^\alpha - 1 \right] E_b^{(comp)} + \gamma \quad \forall n_k. \quad (2)$$

In (2), $\alpha > 0$ depends on the choice of the compression algorithm as well as the selected hardware architecture, γ is the compression offset parameter and $E_b^{(comp)}$ is the per bit processing energy cost. The parameters α and γ are obtained off-line for a given compression algorithm and the hardware platform chosen.

The computational power consumption model in (2) is validated by using the practical compression results of [14,15]. In [14], variable rate wavelet transform for JPEG2000 is employed while adaptive parameter JPEG compression is proposed in [15]. The model parameters α and γ are tuned to provide best match between the computational power consumption model and the results of [14,15] (see Fig. 8 of [14] as well as Fig. 8 of [15]). The model based computational power consumption, $P_{n_k}^{(comp)}$, is compared with the empirical results of [14,15] in Fig. 1. From the results in Fig. 1, we observe that the model in (2) matches well with the results of [14,15]. Combining the computation and communication power costs, the total power consumption for node n_k, P_{n_k}, is given by [13]

$$P_{n_k}(\mathbf{r}^{(out)}, \mathbf{r}^{(in)}) = P_{n_k}^{(comm)}(\mathbf{r}^{(out)}) + P_{n_k}^{(comp)}(\mathbf{r}^{(in)}). \quad (3)$$

The average end-to-end delay experienced by each data packet is obtained by accumulating the link delays along the shortest path from source node to the sink. In our link delay model, we assume that nodes have only finite storage space available for data buffering. Depending on whether the transmitting node is a leaf node or a relay node, we have adopted two different models for the link delay as discussed below.

- *Link node transmitter being a leaf node*: At a leaf node all the data arrivals are from local sensor data compression and no data aggregation is performed. This can be modeled using $M/M/1/K$ queuing model [16], which takes into account the fact that each of the sensor nodes is equipped with a finite buffer of size K.
- *Link node transmitter being a relay node*: Relay node is not only responsible for transmitting compressed data from its local on board sensors but also from the child nodes in its subtree. Since the data arrives at a relay node from multiple sources and data aggregation is performed before its transmission, we can model it as bulk arrival process using $M^{[x]}/M/1/K$ queue. But the limitation of this model is that a closed form solution does not exist [17] and as a result we can not use it directly in our problem formulation. However, this scenario can be significantly simplified by assuming that each of the
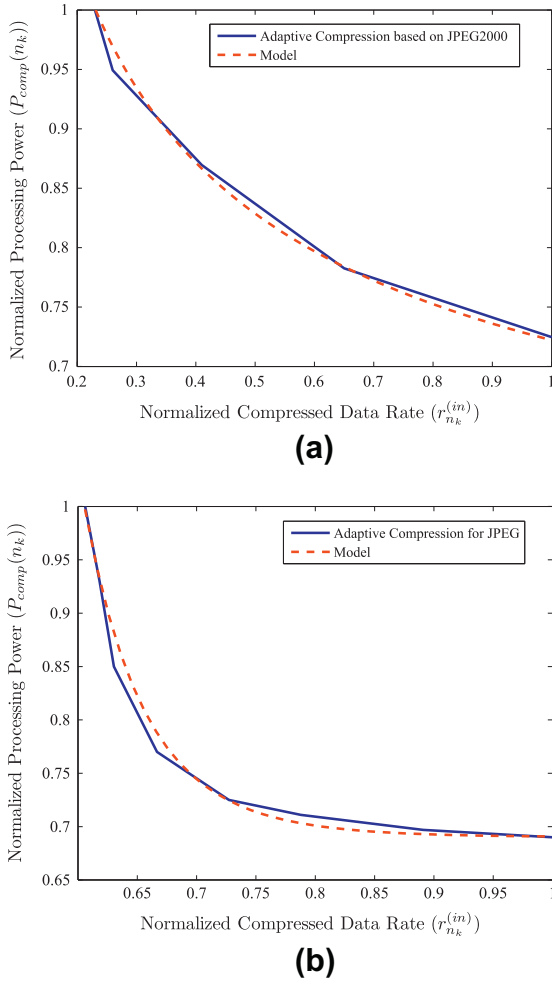
Fig. 1. Comparison of computational power consumption of the proposed node power consumption model with (a) adaptive wavelet transform based compression of [14] and (b) adaptive parameter JPEG compression proposed in [15]. The matching parameters for the model are $\gamma = 0.5$ and $\alpha = 0.51$ for [14] and $\gamma = 0.6$ and $\alpha = 12$ for [15].

arrivals from a leaf node is Poisson and meets the Kleinrock independence principle, which requires that data arrivals from one of the child nodes is independent of the data arrivals from other child nodes in the subtree of the relay node and heavy traffic is experienced by the relay nodes. This can be modeled by using the fact that sum of x independent Poisson processes with rates $r_{n_k}, k = \{1, 2, \ldots, x\}$ is another Poisson process with rate $\sum_{k=1}^{x} r_{n_k}$. Using this fact, we can model the data aggregation behavior at relay nodes by using $M/M/1/K$ queuing model as well, but the arrival rate is different from leaf nodes.

We can now model the mean data transmission and queuing delay [16] for finite buffer size nodes, in case of leaf as well as relay nodes, as

$$\frac{H}{r_{n_k}^{(out)}} \frac{1 + K\rho_{n_k}^{K+1} - (K+1)\rho_{n_k}^{K}}{(1 - \rho_{n_k})\left(1 - \rho_{n_k}^{K+1}\right)}. \tag{4}$$

In (4), H is packet length used for transmission and ρ_{n_k} is the ratio of compressed data arrival to data transmission rate and is defined as

$$\rho_{n_k} = \begin{cases} \dfrac{\left(r_{n_k}^{(in)} + \sum_{n_j \in I(n_k)} r_{n_j}^{(out)}\right)}{r_{n_k}^{(out)}} & \text{for } \textit{relay} \text{ node} \\[4mm] \dfrac{r_{n_k}^{(in)}}{r_{n_k}^{(out)}} & \text{for } \textit{leaf} \text{ node} \end{cases}. \tag{5}$$

From the expression in (5), it can be observed that data aggregation is only required at relay nodes (for leaf nodes the source of data arrivals is local sensor only, so we do not need any data aggregation). In (5), the term $\left(r_{n_k}^{(in)} + \sum_{n_j \in I(n_k)} r_{n_j}^{(out)}\right)$, represents the data aggregation at relay node n_k. We have employed a simple data appending mechanism (when constructing the payload of data transmission packets) for the purpose of data aggregation at relay nodes. We would like to mention here that a more sophisticated approach would try to exploit the spatial data correlation present in the sensor data of child nodes of node n_k.

The variable ρ_{n_k} is bounded as $0 < \rho_{n_k} < 1, \forall n_k$ and provides a tradeoff between packet drops and data distortion. For the case of $\rho_{n_k} \to 1$ packets are dropped due to buffer overflow. On the other hand in case of $\rho_{n_k} \to 0$ requires $r_{n_k}^{(in)}$ to be very small (as $r_{n_k}^{(out)}$ can not be too large since it is upper bounded by the maximum data transmission rate of the communication interface), which in turn requires high compression ratio leading to data distortion. The fraction $\frac{H}{r_{n_k}^{(out)}}$ in (4) is simply the packet transmission time. The resulting end-to-end delay constraint, in a multi-hop setting from source node n_i, is given by

$$\sum_{n_k \in T(n_i)} \frac{H}{r_{n_k}^{(out)}} \left(\frac{1 + K\rho_{n_k}^{K+1} - (K+1)\rho_{n_k}^{K}}{(1 - \rho_{n_k})\left(1 - \rho_{n_k}^{K+1}\right)} \right)$$
$$\leqslant D_{max}(n_i) \quad \forall n_i. \tag{6}$$

In (6), each term in the summation, along the shortest path $T(n_i)$, is link delay towards n_{sink} and $D_{max}(n_i)$ is the maximum end-to-end delay threshold.

3. Optimal lifetime utility and delay-margin tradeoff

To formulate optimal lifetime, utility and delay-margin tradeoff problem we decompose the end-to-end delay constraint in (6), using auxiliary variables $d_{n_k} \in \mathbf{d}$, into multiple link delays along the shortest path $T(n_i)$ as

$$\sum_{n_k \in T(n_i)} d_{n_k} \leqslant D_{max}(n_i), \tag{7}$$

$$\frac{H}{d_{n_k}} \leqslant r_{n_k}^{(out)} \left(\frac{(1 - \rho_{n_k})\left(1 - \rho_{n_k}^{K+1}\right)}{1 + K\rho_{n_k}^{K+1} - (K+1)\rho_{n_k}^{K}} \right). \tag{8}$$

Decomposing the end-to-end delay into multiple link delays enables us to achieve a distributed implementation of optimal lifetime, utility and delay-margin tradeoff problem. Now we introduce delay-margin parameter, $\omega_{n_i} \in \mathbf{\Omega} : \omega_{n_i} \in [0, 1) \forall n_i$, to achieve optimal tradeoff by modifying (7) as

$$\sum_{n_k \in T(n_i)} d_{n_k} \leqslant (1 - \omega_{n_i})D_{max}(n_i), \tag{9}$$

The parameter ω_{n_i} is upper bounded by 1 due to the fact that the data transmission rate $r_{n_i}^{(out)}$ can not be increased arbitrarily and the lower bound ensures that the delay threshold $D_{max}(n_i)$ is not satisfied impractically by using negative values of ω_{n_i}. Now the lifetime, utility and delay-margin tradeoff problem is formulated as

$$\textbf{maximize} \left\{ \min \frac{\beta}{P_{n_i}} + \sum_{n_i} \left(\beta\theta(\omega_{n_i}) + (1-\beta)U\left(r_{n_i}^{(in)}\right) \right) \right\} \tag{10}$$

$$\textbf{s.t.} \sum_{n_k \in T(n_i)} d_{n_k} \leqslant (1-\omega_{n_i})D_{max}(n_i) \quad \forall n_i \tag{11}$$

$$\frac{H}{d_{n_k}} \leqslant r_{n_k}^{(out)} \left(\frac{(1-\rho_{n_k})\left(1-\rho_{n_k}^{K+1}\right)}{1+K\rho_{n_k}^{K+1}-(K+1)\rho_{n_k}^K} \right) \quad \forall n_k \tag{12}$$

$$\rho_{n_k}r_{n_k}^{(out)} = \left(r_{n_k}^{(in)} + \sum_{n_j \in I(n_k)} r_{n_j}^{(out)} \right), \quad \forall n_k \tag{13}$$

$$0 < \rho_{n_i} < 1, \quad R_{n_i}/r_{n_i}^{(in)} \leqslant \gamma_{max} \ \forall n_i \tag{14}$$

$$r_{n_i}^{(out)} \leqslant R_{max}, \quad 0 \leqslant \omega_{n_i} < 1. \ \forall n_i \tag{15}$$

In (10), $\theta(\cdot)$ and $U(\cdot)$ are, respectively, delay-margin and utility objective functions, R_{max} is maximum data transmission rate capability of RF interface and β is weighting coefficient to achieve the desired level of tradeoff. We also impose an upper bound, γ_{max}, on compression ratio, $R_{n_i}/r_{n_i}^{(in)}$, to limit data distortion due to lossy compression.

The multi-objective function in (10) with weighting parameter β provides the tradeoff of network utility with delay-margin and network lifetime. The first constraint in (11), achieves the delay-margin for every source node n_i. The constrains in (12) and (13) are, respectively, the transmission plus queuing delay and the flow conservation constraint for each node n_i. The constraints in (14) and (15) are the block constraints for optimization variables.

The multi-objective formulation in (10) is not a convex function due to the first term, i.e. the max–min function. However, we can obtain an equivalent convex formulation by rewriting the max–min function, in the objective function, in epigraph form. This is achieved by introducing a slack variable t such that $P_{n_i} \leqslant 1/t \forall n_i$, and the resulting optimization problem in (10)–(15) is given by,

$$\textbf{maximize} \left[\beta t + \beta \sum_{n_i} \theta(\omega_{n_i}) + (1-\beta) \sum_{n_i} U(r_{n_i}^{(in)}) \right]$$

$$P_{n_i} \leqslant 1/t \forall n_i, \quad \text{constraints } (11)-(15) \tag{16}$$

The problem in (16) is decomposed for distributed realization by using dual variables $\lambda_{n_i} \in \Lambda$ and $\psi_{n_i} \in \Psi$, to obtain

$$L(t, \mathbf{r}^{(in)}, \mathbf{r}^{(out)}, \mathbf{d}, \Omega, \Lambda, \Psi) = \textbf{maximize}$$

$$\left\{ \beta t + \sum_{n_i} \left(\beta\theta(\omega_{n_i}) + (1-\beta)U\left(r_{n_i}^{(in)}\right) + \lambda_{n_i}(1/t - P_{n_i}) \right) \right.$$

$$\left. + \sum_{n_i} \psi_{n_i} \left((1-\omega_{n_i}) - \frac{\sum_{n_k \in T(n_i)} d_{n_k}}{D_{max}(n_i)} \right) \right\}$$

$$\textbf{s.t.} \quad \text{constraints } (12)-(15). \tag{17}$$

The maximization problem in (17) is decomposable into delay-margin, lifetime and joint utility and node power allocation subproblems as shown in Fig. 2. The associated dual problem is given by

$$\textbf{minimize} \quad g(\Lambda, \Psi)$$

$$\textbf{s.t} \quad \lambda_{n_i}, \psi_{n_i} \geqslant 0 \quad \forall n_i. \tag{18}$$

In (18) $g(\Lambda, \Psi) = L(t^*, \mathbf{r}^{*(in)}, \mathbf{r}^{*(out)}, \mathbf{d}^*, \Omega^*, \Lambda, \Psi)$ and $t^*, \mathbf{r}^{*(in)}, \mathbf{r}^{*(out)}, \mathbf{d}^*$ and Ω^* are optimal primal variables obtained by solving (17). Next we describe the solution approach for each of the subproblems.

3.1. Delay-margin sub-problem

The delay-margin sub-problem from (17), also depicted in Fig. 2, is given by

$$\textbf{maximize} \quad \sum_{n_i}(\beta\theta(\omega_{n_i}) + \psi_{n_i}(1 - \omega_{n_i}))$$

$$\textbf{s.t.} \quad 0 \leqslant \omega_{n_i} < 1. \tag{19}$$

For a given concave utility function $U(r_{n_i}^{(in)})$, the objective function $\theta(\omega_{n_i})$ is responsible for modulating the delay-margin tradeoff with network lifetime and utility. To achieve an optimal tradeoff, the objective function $\theta(\omega_{n_i})$ should be chosen appropriately. As a first step towards this objective, we fix $\beta = 0$ in (10) and use sensitivity analysis to study the effect of perturbing the end-to-end delay threshold $D_{max}(n_i)$ on the optimal network utility.

3.1.1. Sensitivity analysis

We perform the sensitivity analysis by perturbing the end-to-end delay constraint for node n_i by an amount of $\xi_{n_i}, \xi_{n_i} \in \Xi$. By solving a perturbed optimization problem, we observe the effect of end-to-end delay constraint variations on the optimal network utility as:

$$\eta^*(\Xi) = \textbf{maximize} \quad \sum_{n_i} U\left(r_{n_i}^{(in)}\right)$$

$$\textbf{s.t.} \frac{\sum_{n_k \in T(n_i)} d_{n_k}}{D_{max}(n_i)} \leqslant \xi_{n_i} \quad \forall n_i$$

$$\text{and constraint in } (16). \tag{20}$$

In (20), $\eta^*(\Xi)$ denotes the optimal network utility for the perturbed problem with $\xi_{n_i} = 1 \forall n_i$, being the unperturbed problem. The $0 < \xi_{n_i} \leqslant 1$ and $\xi_{n_i} > 1$, respectively, correspond to tightening and relaxing the delay constraint corresponding to n_i^{th} node. If $\psi_{n_i}^*$ corresponding to $\eta^*(\Xi = 1)$ represents the optimal value of the Lagrange multiplier associated with the unperturbed end-to-end delay constraint then fractional change in $\eta^*(\Xi = 1)$ due to n_i^{th} constraint perturbation is obtained as

$$\frac{\eta^*(\Xi) - \eta^*(\mathbf{1})}{\eta^*(\mathbf{1})} = \frac{\eta^*(\xi_{n_i}\mathbf{e}_{n_i}) - \eta^*(\mathbf{1})}{\eta^*(\mathbf{1})},$$

$$\frac{\Delta\eta^*(\Xi)}{\eta^*(\mathbf{1})} = (\xi_{n_i} - 1)\frac{\partial\eta^*(\mathbf{1})/\partial\xi_{n_i}}{\eta^*(\mathbf{1})} + o(\xi_{n_i}),$$

$$= (\xi_{n_i} - 1)(\psi_{n_i}^*/\eta^*(\mathbf{1})) + o(\xi_{n_i})$$

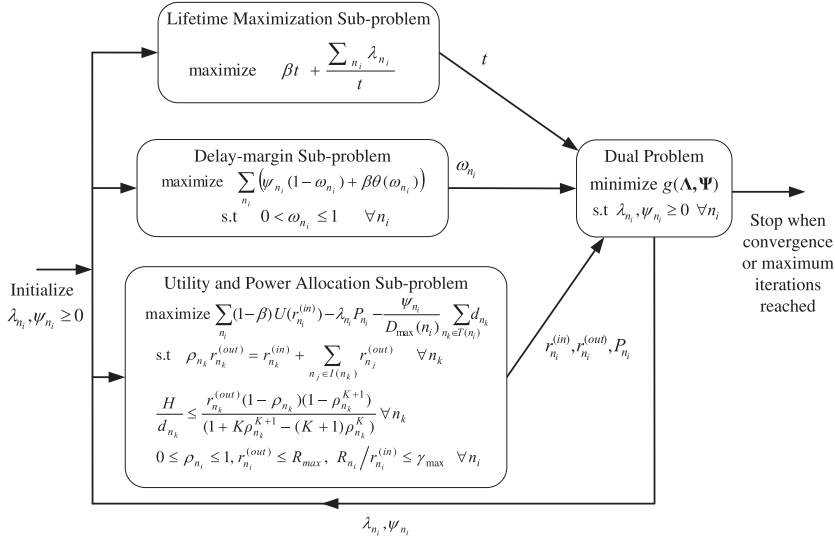$$\approx (\xi_{n_i} - 1)\frac{\psi_{n_i}^*}{\eta^*(\mathbf{1})}. \tag{21}$$

Fig. 2. Block diagram for distributed realization of cross-layer framework.

The vector $\mathbf{e}_{n_i}$ in the first equality of (21) has its n_i^{th} entry equal to one while all other being zero. The second equality of (21) follows from the Taylor series expansion, where $o(\xi_{n_i})$ represents the second and higher order terms lumped together. The third equality of (21) is based on the fact that $\partial \eta^*(\mathbf{1})/\partial \xi_{n_i} = \psi_{s_i}^*$ (see [18], Section 5.6), which is followed by first order approximation in the last expression of (21).

3.1.2. Choosing delay-margin objective function

For optimal choice of the delay-margin objective function, we differentiate the objective in (19) with respect to ω_{n_i} and set it equal to zero to obtain

$$\left. \frac{\partial \theta(\omega_{n_i})}{\partial \omega_{n_i}} \right|_{\omega_{n_i}=\omega_{n_i}^*} = \frac{\psi_{n_i}^*}{\beta}. \tag{22}$$

From (22) we have $\psi_{n_i}^* = \beta \partial \theta(\omega_{n_i})/\partial \omega_{n_i}|_{\omega_{n_i}=\omega_{n_i}^*}$. Now in (21), substituting $\xi_{n_i} - 1$ with ω_{n_i} and using $\psi_{n_i}^* = \beta \partial \theta(\omega_{n_i})/\partial \omega_{n_i}|_{\omega_{n_i}=\omega_{n_i}^*}$, we have

$$\frac{\Delta \eta^*(\mathbf{\Xi})}{\eta^*(\mathbf{1})} \approx \omega_{n_i} \frac{\psi_{n_i}^*}{\eta^*(\mathbf{0})} = -\beta \left(\frac{\omega_{n_i}}{\eta^*(\mathbf{0})} \right) \left. \frac{\partial \theta(\omega_{n_i})}{\partial \omega_{n_i}} \right|_{\omega_{n_i}=\omega_{n_i}^*}, \tag{23}$$

where $\eta^*(\mathbf{0})$ is obtained by mapping $\eta^*(\mathbf{1})$ form ξ_{n_i} to ω_{n_i} domain. By mapping ξ_{n_i} to ω_{n_i} in the interval (0,1], we ensure that $D_{max}(n_i)$ is not violated. Let $\delta_{n_i} = \left| \frac{\Delta \eta^*(\mathbf{\Xi})}{\eta^*(\mathbf{1})} \right|$ represent the magnitude of the fraction of network utility objective, which is available for tradeoff with the end-to-end delay, then from (23) we have

$$\delta_{n_i} = \omega_{n_i} \frac{\beta}{\eta^*(\mathbf{0})} \frac{\partial \theta(\omega_{n_i})}{\partial \omega_{n_i}}, \tag{24}$$

and integrating (24) gives

$$\theta(\omega_{n_i}) = \frac{\eta^*(\mathbf{0})}{\beta} \delta_{n_i} \log(\omega_{n_i}). \tag{25}$$

Using the result in (25), delay-margin sub-problem becomes

maximize $\sum_{n_i} (\eta^*(\mathbf{0})\delta_{n_i} \log(\omega_{n_i}) + \psi_{n_i}(1 - \omega_{n_i}))$

s.t. $0 \leqslant \omega_{n_i} < 1.$ (26)

Considering the unperturbed problem in (20), the optimal objective of network utility is $\eta^*(\mathbf{1}) = \sum_{n_i} U\left(r_{n_i}^{*(in)} \right)$ and for the n_i^{th} node the optimal compressed data rate is $r_{n_i}^{*(in)}$. We denote by δ_{n_i}, the maximum fraction of network utility objective, which may be made available by the application for tradeoff with end-to-end delay, and define $\delta_{n_i} = ar_{n_i}^{*(in)}$ with parameter a bounded as $\frac{0.1R_{n_i}}{\gamma_{max}} < a < \frac{0.9R_{n_i}}{\gamma_{max}}$ is the scaling constant. This choice of parameter a allows a 10–90% of the normalized optimal rate $r_{n_i}^{*(in)}$ to be traded off with end-to-end delay for the desired level of delay-margin. This problem can be solved in ω_{n_i} using efficient algorithms available for convex optimization [18].

3.2. Joint utility and power allocation subproblem

From (17), the joint utility and node power allocation subproblem, is formed by collecting the network utility, power allocation and the the end-to-end delays in the objective function resulting in a multi-objective optimization problem. This subproblem tries to maximize the network utility while minimizing the node power consumption as well as end-to-end delays simultaneously while meeting the respective performance constraints and is formulated as

$$\textbf{maximize} \quad \sum_{n_i} \left((1-\beta)U\left(r_{n_i}^{(in)}\right) - \lambda_{n_i}P_{n_i} - \frac{\psi_{n_i}}{D_{max}(n_i)} \sum_{n_k \in T(n_i)} d_{n_k} \right)$$

$$\textbf{s.t.} \quad \frac{H}{d_{n_k}} \leqslant r_{n_k}^{(out)} \left(\frac{(1-\rho_{n_k})\left(1-\rho_{n_k}^{K+1}\right)}{1 + K\rho_{n_k}^{K+1} - (K+1)\rho_{n_k}^{K}} \right) \quad \forall n_k$$

$$\rho_{n_k} r_{n_k}^{(out)} = \left(r_{n_k}^{(in)} + \sum_{n_j \in I(n_k)} r_{n_j}^{(out)} \right), \quad \forall n_k$$

$$0 < \rho_{n_i} < 1, R_{n_i}/r_{n_i}^{(in)} \leqslant \gamma_{max} \quad \forall n_i$$

$$r_{n_i}^{(out)} \leqslant R_{max}. \quad \forall n_i \tag{27}$$

For a fixed buffer size K, $\rho_{n_i} \to 1$ will lead to higher packet drops resulting in an increased delay. On the other hand, $\rho_{n_i} \to 0$ requires a higher compression ratio, which will increase data distortion. The best compromise between higher packet drops demanding an increase in $D_{max}(n_i)$, and a higher compression ratio requiring an increase in γ_{max} will be obtained for optimal value of ρ_{n_i}. This optimal tradeoff is achieved by solving the optimization problem in (27).

The problem in (27) is a generalized geometric program due to the ratio of posynomials in the expression for end-to-end delay constraint [19]. For distributed realization, we transform the generalized geometric end-to-end delay constraint in (27) to an equivalent convex constraint. For that purpose we define $x_k = \log\left(\rho_{n_k}\right)$ and transform the end-to-end delay constraint using log transformation, which results in

$$\{\log(He^0 + HKe^{(K+1)x_k} - H(K+1)e^{Kx_k})$$
$$- \log(e^0 - e^{x_k} - e^{(K+1)x_k} + e^{(K+2)x_k})$$
$$- \log(r_{n_k}^{(out)}) - \log(d_{n_k})\} \leqslant 0. \tag{28}$$

The log-sum-exponent form in (28) makes the link delay constraints to be convex [20,19]. A similar procedure is employed for the rate conservation constraint in (13). Using the transformed constraints, the problem in (27) leads to an equivalent convex formulation and can be realized efficiently in a distributed manner.

3.3. Lifetime maximization subproblem

The network lifetime maximization subproblem from the Lagrangian function in (17) is given by

$$\textbf{maximize} \quad \beta t + \frac{\sum_{n_i} \lambda_{n_i}}{t}.$$
$$\textbf{s.t} \quad t > 0. \tag{29}$$

Differentiating the objective in (29) with respect to t and setting it to zero gives

$$t^* = \sqrt{\frac{\sum_{n_i} \lambda_{n_i}}{\beta}}. \tag{30}$$

The condition $\lambda_{n_i} \geqslant 0$ in (18) can lead to small t^*, which in turn can result in arbitrary large P_{n_i} due to $P_{n_i} \leqslant 1/t$. But from a practical viewpoint, P_{n_i} can not be increased arbitrarily high and an upper bound, $P_{n_i}^{(ub)}$, is imposed by the

processor and the wireless transceiver used. This is ensured by introducing the constraint $P_{n_i}^{(ub)} \leqslant 1/t^*$, which along with the result in (30) provides the solution for t^* given by

$$t^* = \min\left\{ \frac{1}{P_{n_i}^{(ub)}}, \sqrt{\frac{\sum_{n_i}\lambda_{n_i}}{\beta}} \right\}. \tag{31}$$

The lifetime maximization subproblem effectively tries to minimize the power consumption of a highly loaded node by reducing its communication load at the expanse of an increase in computational load of some lightly loaded nodes. Our proposed solution for network lifetime maximization is valid for homogeneous case, where all the nodes have equal amount of energy available at the beginning. But introducing a weighting coefficient in $P_{n_i} < 1/t$ constraints, we can extend this model for heterogeneous case.

3.4. Dual problem

The dual problem in (18) is solved by using the following projected sub-gradient updates:

$$\lambda_{n_i}(k+1) = [\lambda_{n_i}(k) + \pi(k)(1/t - P_{n_i})]^+ \quad \forall n_i, \tag{32}$$

$$\psi_{n_i}(k+1) = \left[\psi_{n_i}(k) + \pi(k)\left((1-\omega_{n_i}) - \frac{\sum_{n_k \in T(n_i)} d_{n_k}}{D_{max}(n_i)} \right) \right]^+ \quad \forall n_i. \tag{33}$$

In (32) and (33), $[x]^+$ is defined as $max\{0, x\}$ and $(t - 1/P_{n_i})$ and $\left((1-\omega_{n_i}) - D_{max}^{-1}(n_i)\sum_{n_k \in T(n_i)} d_{n_k} \right)$ are the sub-gradients. We use variable step-size rule $\pi(k) = 1/\sqrt{k}$, to update the dual variables [21]. Choosing variable step-size rule can provide fast convergence at the start of the algorithm but results in slower convergence near optimality when compared to the convergence performance for constant step-size rule. In case of constant step-size rule, we can observe frequent sign reversals, near optimal point, for the slope of dual variable updates. From the above discussion, the final choice is based on the tradeoff that exists between faster convergence near optimality and the magnitude of the tolerance band around the optimal solution point. The lifetime, utility and delay-margin subproblems, coupled through dual variables, can be assigned to different nodes for distributed realization as discussed in the following subsection.

3.5. Distributed realization

For distributed realization, the delay-margin subproblem (to determine ω_{n_k}), the network utility and power allocation subproblem (to determine $r_{n_k}^{*(in)}, r_{n_k}^{*(out)}$ and P_{n_k}) and the dual problem (to update $\lambda_{n_k}, \psi_{n_k}$) are assigned to the respective sensor nodes. It is worth mentioning that a fully distributed realization is not possible because of the information exchange requirement among different nodes. For instance, in order to update dual variable λ_{n_k}, we require t^* and $P_{n_k}^*$ to be communicated to node n_k. Similarly the objective function for node utility and power allocation

subproblem, require optimal link delays $d^*_{n_i} : n_i \in T(n_k)$ to be available at node n_k. However, the communication overhead is reduced at a relay sensor node as it does not require optimal rates $r^{*(out)}_{n_k}$ from the nodes in its subtree, which are needed for the flow conservation constraint of utility and power allocation subproblem. This is because, the relay sensor node extracts $r^{*(out)}_{n_k}$ from the data it relays corresponding to every sensor node in its subtree. The maximal power minimization subproblem, to determine t^*, requires $\lambda_{n_k} \forall n_k$ and the most suitable candidate for this subproblem assignment is n_{sink}.

4. Results

The example network shown in Fig. 3 is chosen for studying the optimal tradeoff among network utilization its lifetime and delay-margin. To quantify network utilization, a concave function, $U\left(r^{(in)}_{n_i}\right) = \log\left(r^{(in)}_{n_i}\right)$ is chosen, which provides proportional throughput fairness among the network nodes [3]. To avoid large number of possible combinations, which may arise when different values of parameters R_{n_i}, δ_{n_i} and $D_{max}(n_i)$ are selected for each sensor node, we use $R_{n_i} = R, \delta_{n_i} = \delta$ and $D_{max}(n_i) = D_{max}, \forall n_i$. This leads to homogeneous network from the view point of these parameters. The other parametric values used in the performance evaluation of the proposed cross-layer framework are tabulated in Table 1.

To study the optimal tradeoff between network lifetime and delay-margin, the end-to-end delay threshold D_{max} is varied to change the delay-margin and is shown in Fig. 4. It is observed from the performance result in Fig. 4 that network lifetime increases monotonically with an increase in D_{max}. This is due to the fact that larger values of D_{max} allow a decrease in $r^{(out)}_{n_i}$ or an increase in $r^{(in)}_{n_i}$ results in a decrease in $P^{(comm)}$ (due to corresponding reduction in $r^{(out)}_{n_i}$) or a decrease in $P^{(comp)}$ (due to an increase in $r^{(in)}_{n_i}$), which leads to network lifetime improvement. The variation of parameter β has an interesting affect on the network lifetime. One would expect increasing β will lead to lifetime improvement, as it will assign more weight to the network lifetime (see (10)). This is true for $D_{max} > 55$ ms as observed from the result in Fig. 4. But for smaller values of D_{max} increasing β is counter productive resulting in degradation in the network lifetime. This is because increasing β also tries to improve the delay-margin by increasing the
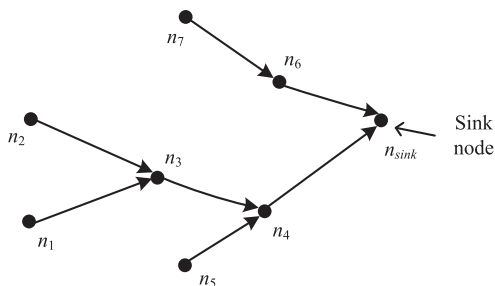
Fig. 3. An example WMSN consisting of seven sensor and single sink node.

Table 1
Network parameters used in the performance evaluation.

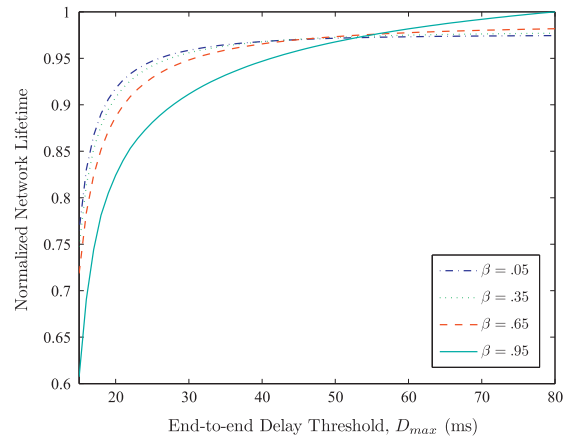| Parameters	Values
Transmission cost/bit $\left(E^{(tx)}_b\right)$	.209 μJ/bit
Reception cost/bit $\left(E^{(rx)}_b\right)$	.252 μJ/bit
Maximum transmission rate (R_{max})	500 kbps
Maximum compression ratio (γ_{max})	50
Packet length (H)	104 bytes
Computational cost/bit $\left(E^{(comp)}_b\right)$	.0023 μJ/bit
Uncompressed data rate (R)	500 kbps
Transmit power lower bound $\left(P^{(ub)}_{n_i}\right)$	10 mW
Compression parameter (α)	0.6
Compression offset parameter (ϵ)	0.5

Fig. 4. Network lifetime, $1/t$, as a function of end-to-end delay threshold, D_{max}. Normalization is done with the maximum value of $1/t$ over its entire domain. Parameter $\delta = 0.1$ is used.

delay margin. For $D_{max} < 55$ ms, the delay-margin improvement, due to an increase in β, is achieved at the expanse of both network lifetime as well as network utility. On the other hand, for a $D_{max} > 55$ ms, both delay-margin and lifetime improvements are achieved at the expanse of network utility.

To verify the above mentioned tradeoff, we define network throughput as $\sum_{n_i} r^{(in)}_{n_i}$, and observe the network throughput performance as a function of D_{max} and β simultaneously and is shown in Fig. 5. The performance tradeoff result in Fig. 5 shows that increasing β leads to throughput reduction, but a decrease of 0.45 Mbps at $D_{max} = 15$ ms only partially compensates for delay-margin resulting in lifetime degradation. In contrast, a decrease of 0.786 Mbps in the throughput at $D_{max} = 80$ ms not only compensates for delay-margin but also allows lifetime improvement. The above argument is further validated by observing the tradeoff between network lifetime and throughput $\left(\sum_{n_i} r^{(in)}_{n_i}\right)$, which is shown in Fig. 6. An increase in network throughput in Fig. 6 is obtained by reducing parameter β from 0.95 to 0.05. From the result in Fig. 6, we observe that for smaller values of D_{max} an increase in parameter β results in both lifetime as well as network throughput
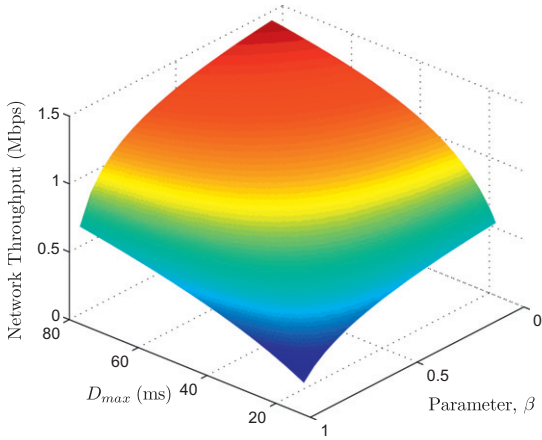
Fig. 5. Network throughput $\left(\sum_{n_i} r_{n_i}^{(in)}\right)$ performance as a function of D_{max} and weighting coefficient β. Parameter $\delta = 0.1$ is used for this result.
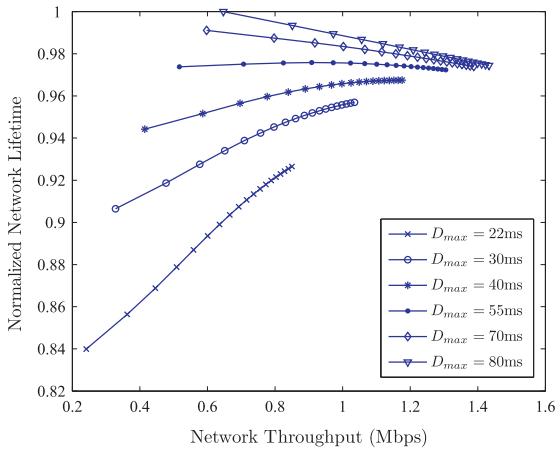

Fig. 6. Network lifetime performance as a function of throughput for different values of D_{max}. An increase in network throughput along horizontal axis is obtained by reducing parameter β from 0.95 to 0.05. Parameter $\delta = 0.1$ is used for this result.

degradation. On the other hand for larger values of D_{max} (i.e. when $D_{max} > 55$ ms approximately as observed from Fig. 4), we observe an improvement in network lifetime with an increase in parameter β. This is based on the fact that for larger values of D_{max} higher compression ratio is achievable at the sensor nodes resulting in a decrease in $P^{(comm)}$ at the expense of an increase in $P^{(comp)}$. But a dominant decrease in $P^{(comm)}$ compared to a relatively small increase in $P^{(comp)}$ reduces node power P_{n_i}, leading to an improvement in network lifetime and shows the criticality of parameter D_{max} while choosing the operating point for β.

Next we study the delay-margin performance tradeoff (corresponding to node n_4) as a function of parameter δ for different values of D_{max} and is shown in Fig. 7. From the result in Fig. 7 we observe that a significant improvement in delay-margin can be achieved by relaxing D_{max} for relatively smaller values of D_{max}, compared to small delay-margin improvement for larger values of D_{max}. Despite
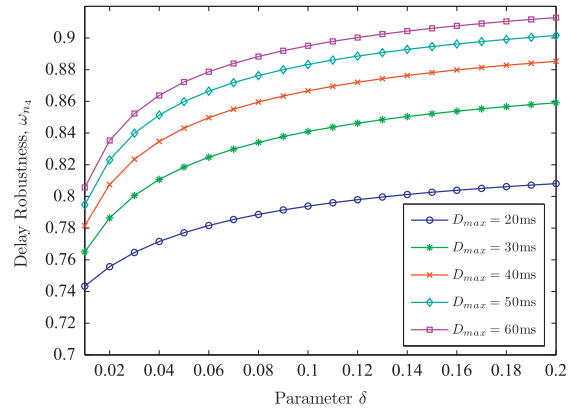

Fig. 7. Delay-margin performance for node n_4 as a function of δ for different values of D_{max}. Parameter δ chooses maximum utility fraction that we tradeoff with delay-margin. We use $\beta = 0.5$ for this result.

the fact that we have assigned same values to the parameters R_{n_i}, δ_{n_i} and $D_{max}(n_i), \forall n_i$, the optimal value of the delay-margin achieved at each sensor node is different. This is due to the fact that the available fraction of the throughput, which can be compromised for delay-margin is different for different nodes in the network because each node has different number of child nodes. To quantify this difference between the optimal values of delay-margin achieved at different nodes, which is measure of delay-margin fairness, we define normalized delay-margin-gap ($DMG_{normalized}$) as

$$DMG_{normalized} = 1 - \frac{\min\{\omega_{n_i} | n_i \in N \setminus n_{sink}\}}{\max\{\omega_{n_i} | n_i \in N \setminus n_{sink}\}} \times 100 \qquad (34)$$

The percentage variation of $DMG_{normalized}$ as a function of parameter D_{max} is shown in Fig. 8. For relatively smaller values of D_{max} the delay-margin gap is large because of the fact that highly loaded nodes have a small fraction of the throughput available that can be compromised to achieve delay-margin. On the other hand for larger values of D_{max} the parameter $DMG_{normalized}$ reduces mainly because the highly loaded nodes can compromise larger
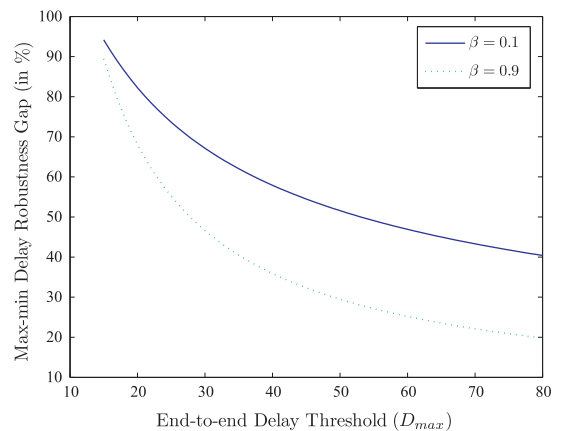

Fig. 8. Normalized delay-margin gap as a function of delay threshold (D_{max}). Parameter $\delta = 0.1$ is used for this result.

fraction of the network throughput (i.e. network utility). Ideally the delay-margin gap should be zero leading to a scenario, where all the nodes have same level of delay-margin. From the $DMG_{normalized}$ result in Fig. 8 we observe that for $D_{max} = 80$ ms increasing the parameter β from 0.1 to 0.9 reduces the normalized delay-margin gap by almost 100% from 0.4 to 0.2 (i.e. from 40% to 20%).

5. Conclusions

A cross-layer framework is proposed for wireless multi-media sensor networks, which achieves an optimal trade-off among network lifetime, its utilization and delay-margin. For optimal performance tradeoff we require an appropriate objective function for delay-margin, which provides any desired level of delay QoS provisioning to the application layer. This is achieved using sensitivity analysis, which incorporates delay-margin in the end-to-end delay constraints and penalizes its price in the objective function. The multi-objective cross-layer optimization problem is decomposed into lifetime, utility and delay-margin subproblems allowing distributed realization. Performance results provide an insight into the complex coupling among the contradicting objectives and show that how the network utility can be compromised to improve the lifetime and delay-margin performance. The performance evaluation results show that compromising network utility does not ensure both lifetime and delay-margin improvement, at all operating points. The proposed framework can also be used for delay critical applications in sensor networks.

Acknowledgments

References

[1] I. Akyildiz, T. Melodia, K. Chowdhury, A survey on wireless multimedia sensor networks, Computer Networks 51 (4) (2007) 921–960.
[2] J. Zhu, S. Chen, B. Bensaou, K. Hung, Tradeoff between lifetime and rate allocation in wireless sensor networks: a cross layer approach, in: IEEE INFOCOM, 2007, pp. 267–275.
[3] H. Nama, M. Chiang, N. Mandayam, Utility-lifetime trade-off in self-regulating wireless sensor networks: a cross-layer design approach, in: IEEE International Conference on Communications, 2006, pp. 3511–3516.
[4] J. Chen, S. He, Y. Sun, P. Thulasiraman, X. Shen, Optimal flow control for utility-lifetime tradeoff in wireless sensor networks, Computer Networks 53 (18) (2009) 3031–3041.
[5] J. Chen, W. Xu, S. He, Y. Sun, P. Thulasiraman, X. Shen, Utility-based asynchronous flow control algorithm for wireless sensor networks, IEEE Journal on Selected Areas in Communications 28 (7) (2010) 1116–1126.
[6] Y. Yu, B. Krishnamachari, V. Prasanna, Energy-latency tradeoffs for data gathering in wireless sensor networks, in: IEEE INFOCOM, 2004, pp. 244–255.
[7] K. Akkaya, M. Younis, M. Youssef, Efficient aggregation of delay-constrained data in wireless sensor networks, in: ACS/IEEE International Conference on Computer Systems and Applications, 2005, pp. 904–909.
[8] M. Tahir, R. Farrell, Optimal utility lifetime and delay-robustness tradeoff in wireless multimedia sensor networks, in: IEEE International Conference on Communications (ICC), 2010, pp. 1–5.
[9] H. Tan, I. Korpeoglu, I. Stojmenovi, Computing localized power efficient data aggregation trees for sensor networks, IEEE Transactions on Parallel and Distributed Systems 22 (3) (2011) 489–500.
[10] H. Tan, I. Korpeoglu, Power efficient data gathering and aggregation in wireless sensor networks, ACM SIGMOD Record 32 (4) (2003) 66–71.
[11] R. Rajagopalan, P. Varshney, Data aggregation techniques in sensor networks: a survey, IEEE Communications Surveys & Tutorials 8 (4) (2006) 48–63.
[12] Y. Yu, B. Krishnamachari, V. Prasanna, Data gathering with tunable compression in sensor networks, IEEE Transaction on Parallel and Distributed Systems 19 (2) (2008) 276–287.
[13] M. Tahir, R. Farrell, Optimal communication–computation tradeoff for wireless multimedia sensor network lifetime maximization, in: Proc. IEEE Wireless Communications and Networking Conference, 2009, pp. 1–6.
[14] D. Lee, S. Dey, Adaptive and energy efficient wavelet image compression for mobile multimedia data services, in: Proc. IEEE International Conference on Communications, 2002, pp. 2484–2490.
[15] C. Taylor, S. Dey, Adaptive image compression for wireless multimedia communication, in: IEEE International Conference on Communications, 2001, pp. 1925–1929.
[16] S. Ross, Introduction to Probability Models, Academic Press, San Diego, 2000.
[17] D.R. Manfield, P. Tran-Gia, Analysis of a finite storage system with batch input arising out of message packetization, IEEE Transactions on Communications COM30 (3) (1982) 456–463.
[18] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.
[19] M. Chiang, Geometric Programming for Communication Systems, Now Publishers Inc., 2005.
[20] S. Boyd, S. Kim, L. Vandenberghe, A. Hassibi, A tutorial on geometric programming, Optimization and Engineering 8 (1) (2007) 67–127.
[21] N. Shor, K. Kiwiel, A. Ruszcaynski, Minimization Methods for Non-differentiable Functions, Springer-Verlag, New York, 1985.

Muhammad Tahir is an Assistant Professor at the Department of Electrical Engineering of the University of Engineering and Technology Lahore, Pakistan. Prior to that, he was a postdoctoral research fellow at the Institute of Microelectronics and Wireless Systems, National University of Ireland, Maynooth. He received his Ph.D. in Electrical and Computer Engineering from the University of Illinois at Chicago (2004–2008). Mr. Tahir is the co-recipient of the Outstanding Student Paper Award in the 21st IEEE International Conference on Advanced Information Networking and Applications (2007). Before joining University of Illinois at Chicago, he was a Lecturer in the Department of Electrical Engineering, University of Engineering and Technology, Lahore Pakistan (2000–2003). Currently his research focuses in the area of sensor integration and wireless multimedia sensor network. His research interests include delay constrained wireless networks, distributed communication network resource optimization and realtime wireless multimedia networks. He is a reviewer of numerous IEEE journals and conferences.

Farrell is from Clondalkin, Dublin, Ireland. He graduated from University College Dublin in 1993 with a B.E. and proceeded to work with ICI/Zeneca Chemicals for the next 2 years in Louisiana, USA, and on various sites in Yorkshire, England. In 1995 he returned to University College Dublin to start on a Masters on Sigma-Delta Modulators, sponsored by Analog Devices. This developed into a Ph.D. which he received in 1998.

After receiving Ph.D. Mr. Farrell joined Silicon Systems Limited (SSL), which was to become Parthus Technologies after going public. He worked within the mixed signal group designing custom silicon chips, ranging from low frequency, low power, applications to 2.4 GHz phaselock loops. His areas of expertise are in system modeling and development, digital and analog design at low and medium frequencies, and in particular data converters utilizing oversampling techniques. He left Parthus Technologies in 2001 and took an academic position with the National University of Ireland, Maynooth. Currently He is a lecturer in the Department of Electronic Engineering at NUI, Maynooth. His courses currently include Electronic Engineering Fundamentals (EE101) and ASIC Design (EE408).