An Adaptive Data Forwarding Scheme for Energy Efficiency in Wireless Sensor Networks

Christos Anagnostopoulos, Theodoros Anagnostopoulos, Stathes Hadjiefthymiades
Department of Informatics and Telecommunications, National and Kapodistrian University of Athens,
Communication Networks Laboratories, Pervasive Computing Research Group, Athens, Greece,
{*bleu*, *thanag*, *shadj*}*@di.uoa.gr*

***Abstract*—Applications based on Wireless Sensor Networks (WSN) are influenced by many factors such as transmission errors, network topology and power consumption. Consequently, developing such applications introduces several research challenges. In this paper, we propose an intelligent model in order to achieve energy efficient message forwarding over a WSN. Based on the proposed model, a network node evaluates a specific statistical extrapolation error through local linear regression and Lagrange polynomial methods and adaptively decides whether message retransmission is required or not. Simulation results are reported denoting the applicability of the proposed model in minimizing energy consumption on message forwarding.**

***Index Terms*—Energy efficiency, transmission control mechanism, intelligent forwarding.**

I. Introduction

Wireless Sensor Networks (WSN) are composed by a large number of nodes, which are equipped with sensing communication and minimal computation capabilities. WSNs support application areas with varying requirements and characteristics [1] (e.g., military applications, forest fire detection and traffic control). Sensor nodes measure the temporal-spatial field of a wide variety of contextual (environmental) scalar parameters, e.g., temperature, wind speed, humidity, and return their measurements to a sink (or data fusion center). This gives the maximum amount of information on the data field but at the same time requires the maximum amount of energy to collect information. Instead, it may be possible to use a reduced amount of energy to obtain sufficiently accurate approximation of the data field.

Contextual data collected by a WSN is a challenging problem. Due to very limited resources, e.g., energy, computational power, data storage and bandwidth, it is not a sound technical decision to apply strategies like forwarding any sensor data directly to a sink that does the corresponding processing. Although this problem has received considerable attention in the last years both from industry and research, it is of high importance to take into consideration the nature of the sensed data in order to avoid significant energy consumption and improve bandwidth utilization [2]. We propose, implement and evaluate an intelligent adaptive data forwarding model (ADF) that decides on forwarding measurements based on approximation techniques with reduced energy consumption. The ADF model extends the mechanism proposed in [3] to the point that it adapts its parameters to current values minimizing energy consumption while forwarding data. The adaptation rules exploit the current data accuracy, the induced approximation error and change in error, and previous behavior of the model in order to decide on forwarding a piece of data. Data is approximated using statistical extrapolation methods (Lagrange Polynomial and Local Linear Regression) based on recently received measurements. The same statistical methods are adopted for approximating data if forwarding does not take place according to a variable (variance-based) threshold. Overall, the aim of ADF is to prolong the lifetime of a WSN application. To accomplish this, unnecessary transmissions over the network are eliminated while the induced data reconstruction error (due to no-retransmission) is preserved in low levels. The rest of the paper is organized as follows: Section II discussed related work for energy efficient sensor data collection. In Section III we present the proposed model while Section IV reports a performance assessment. Conclusions and future work are summarized in Section V.

II. Related Work

The authors in [4] propose a model, which aims at avoiding unnecessary data transmissions by selecting empirically good paths, caching and data processing within the WSN. The advanced model in [6] uses occasionally sub-optimal paths to obtain energy benefits. However, the disadvantage of both models is the physical distance calculation among nodes used by the routing mechanism. The TEEN and APTEEN models discussed in [5] are the closest to our model. In both protocols the key factor is the comparison of the measured value against regularly disseminated pre-defined thresholds. Specifically, the architecture comprises cluster nodes and member nodes. A cluster node determines certain thresholds and disseminates them to the member nodes. The nodes measure all the time but forward data based on the specific threshold or when the difference between subsequent measurements is higher than a threshold. However, after a time horizon, the nodes have to change roles, re-calculate the new thresholds and disseminate them through the whole network. This inevitably imposes an additional cost in energy and bandwidth for such meta-data dissemination. In contrast, in our model the threshold is time-variant and autonomously determined locally in each node, thus, there are no further communication and energy cost. Another disadvantage of TEEN is the high dependency of the nodes on such thresholds (especially, in the case that no communication can be established). The additional feature of APTEEN [9] is the adoption of extra meta-data information such as the time schedule and the transmission frequency. This adds also more energy cost in receiving and transmitting meta-data information and necessitates strict synchronization issues among nodes. In our model, the node forwards either data or only a signal to the upstream node in order to reproduce the corresponding data without any redundant meta-data information and synchronization issues. The model in [13] exploits adaptive sensing for energy efficient data collection. However, such model assumes synchronization of nodes and high computational complexity, especially, in calculating certain coefficient measures. Such additional cost is not considered in the evaluation of the model, although it is of high importance. In addition, the model in [13] assumes sensed data with redundant information (high entropy) and the proposed optimization algorithms require additional pre-communication costs among sensor nodes and sinks. In our model all these assumptions are eliminated and, especially, the distribution of the sensed data is completely unknown.

Moreover, several models exploit pre-existing knowledge on the current WSN topology and structure in order to be energy-aware. The model in [10] proposes an energy efficient protocol based on de Bruijn and Voronoi diagrams (for routing and data aggregation). Such model requires training in order to construct the routing tables and, therefore, computational effort in order to settle such communication architecture. In addition, such computational effort is not taken into account when evaluating the proposed model in [10]. Finally, the model assumes exact knowledge on the network topology, which negatively, impacts the scalability of the model. In our model, the entire topology and the number of nodes of the WSN is completely unknown. The authors in [11] introduce an energy-efficient scheme for selective forwarding in WSN. That is, nodes transmit only the most statistically important messages and discard the least important ones. However, such model assumes extra computation cost in order to construct and infer which message is statistically important. In addition, the model does not take into consideration the complexity (energy) cost per CPU computation for evaluating the degree of importance for each received message. The authors in [12] reported that energy consumption cannot be achieved by routing through shortest path. Hence, in order to maximize network life time, instead of using the shortest path, the authors in [12] study alternative paths to avoid congestion. This implies a-priori knowledge of the exact topology and the number of nodes in WSN, which is not a requirement in our model. However, the proposed model in [12] assumes high communication cost for identifying such alternate paths since this requires the computation of all possible paths in the network. Our model considers the additional computational cost for each decision on message forwarding, takes completely online decisions for message forwarding, does not assume any knowledge in WSN topology and does not require any meta-data communication among nodes for being energy efficient.

III. ADAPTIVE ENERGY EFFICIENT MODEL

A. Model Description

We consider a WSN as a set of sensor nodes and a sink node with paths leading from the leaves nodes to the sink (root) (see Figure 1(a)). Two nodes disseminate data if they are within the communication range of each other. In our model, the nodes are not synchronized. All nodes can either sense data, or relay data, or perform both operations. A relay node i reserves memory and communication resources (receiving and forwarding) for each data flow. A sensing node reserves resources only for sensing and forwarding.

Each node adopts ADF in order to decide whether to propagate the receiving data or not. The peer node, i.e., the node found in the upstream, should be able to conceive this situation and react accordingly. Consider that a node i receives a new piece of data $p(t)$ at time t. The node i can either forward $p(t)$ to a peer node j in the upstream path or send a signal $u(t) \in \{0, 1\}$ to node j to reproduce the $p(t)$ value without, explicitly, receiving it. In the former case $u(t) = 0$ and in the latter case $u(t) = 1$. In order to achieve such decision, we can assume that the nodes have a-priori agreed on an extrapolation scheme f common throughout the WSN. This, however, is not obligatory; each node can independently perform its own extrapolation scheme. In addition, an appropriate extrapolation scheme can be chosen w.r.t. the nature of the sensed data. The node i calculates an extrapolated value $p^*(t)$ for the piece of data $p(t)$ based, also, on the previous m received measurements $p(t\text{-}m), \ldots, p(t\text{-}1)$, $m > 0$. The extrapolation scheme $f(m)$ depends on such m values. The $p^*(t)$ is compared against the actual measurement $p(t)$ and the reconstruction error $e(t) = p(t) - p^*(t)$ is obtained. The estimated error level contributes to the decision on the forwarding of $p(t)$. That is, if the data is not transmitted upstream, then the peer node j performs the same extrapolation calculation f and considers the locally estimated $p^*(t)$ as the new received measurement. This scheme is applied for all data flows handled by node i. It is worth noting that, data aggregation can be also performed when a relay node is required to propagate values from more than one data flows (possibly with aggregation).

B. The Adaptive Mechanism

A node i either forwards the piece of data $p(t)$ or a data reproduction signal $u(t)$ to peer node j w.r.t. the reconstruction error $e(t)$. The extrapolated $p^*(t)$ values is generated by $f(m)$ based on the history of the last m measurements (experimentally the mean value of $m < 6$). The value of m, also called as sliding window, plays significant role in order to extrapolate the $(m+1)$th piece of data. Such sliding-window model is motivated by the assumption that recent data is more useful and pertinent for extrapolation than older data. Evidently, the value of m is not a-priori known and, additionally, there is no knowledge about the received data distribution. For that reason, we propose a controller A(m), which adjusts the history length m referring to the m previous measurements. Specifically, at time $t+1$ the value of m is based on: (i) the reconstruction error $e(t)$, (ii) the change in error $e(t)$, i.e., $\Delta e(t) = e(t) - e(t-1)$, and, (iii) the behavior of the previous decision on m, i.e., $\Delta m(t) = m(t) - m(t-1)$. We assume a unity change $a(t)$ in value of m, $a(t) \in \{-1, 0, 1\}$, which denotes increase, decrease or leave constant the value of m. The $a(t)$ value results to the value of $m(t+1)$ at the next time $t+1$. In other words we have the following adaptation rule:

$$m(t+1) = m(t) + a(t) \qquad (1)$$

The controller A(m) through a set of decision rules produces a value for $a(t)$. Hence, the decision on the next value of m, which hopefully, minimizes the reconstruction error $e(t)$ of $f(m)$, is based on the previous decision on the value of m along with the observed error induced by the extrapolation. The change in m, Δm, and the change in error, Δe, are the basic directives, which determine the direction (positive, zero or negative) of a in the decision for the next value of m. Specifically, if $\Delta e < 0$, A(m) should reward the past decision on m since there is an improvement depicted by the negative change in error. $\Delta e(t) < 0$ denotes that the induced error $e(t)$ is smaller than the previous $e(t-1)$. Hence, the value of a represents a reward on the previous decision Δm. That is, if $\Delta m(t) = 1$ then A(m) should continue to increase m ($a = 1$); otherwise A(m) should continue to decrease m ($a = -1$). If $\Delta e > 0$, A(m) should not reward the previous decision on m since there is an error increase. A(m) should apply a penalty on the past decision to reverse the current trend (i.e., $a = 1$ if $\Delta m(t) = -1$ and the inverse). If $\Delta e = 0$, A(m) deals with zero improvement or deterioration of the estimation process. This means that the previous decision for m was not successful, thus, A(m) remains idle and tries an opposite decision at the very next time instance. Especially, if $\Delta m = 0$ and $\Delta e = 0$, A(m) randomly selects a non-zero value for $a \in \{-1, 1\}$. The same holds for $\Delta e > 0$ and $\Delta m = 0$. Finally, when $\Delta e < 0$ and $\Delta m = 0$, i.e., improvement without changing behavior, A(m) leaves m unchanged ($a = 0$). The decision rules for a are depicted in tabular format in Table I. The (x, y) element in Table I is a value of a for certain Δe (x-index) and Δm (y-index) value.

The adjusted value of m is used by $f(m)$ for estimating $p^*(t)$ producing error $e(t)$. Such error is compared with a time-variant decision threshold $v(t)$. Hence, it holds that:

- If $e(t) > v(t)$ then, the node i forwards $p(t)$ to the peer node j and the A(m) adjusts accordingly (through Eq(1)) the value of m for future extrapolations (see Figure 1(b)).
- Otherwise, the node i sends a data reproduction signal $u(t) = 1$ to node j. The node j upon receiving $u(t)$ reproduces locally $p^*(t)$ with the induced error $e(t)$. In this case, the A(m) leaves the m value as is, since no error has been encountered ($e(t) \leq v(t)$). Evidently, node j behaves in a similar manner.

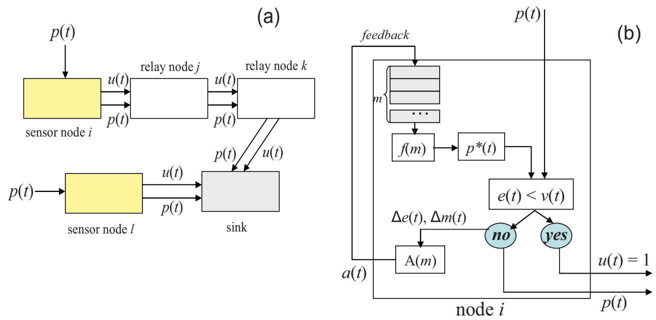

Figure 1(a): Each node sends either data *p* or data reproduction signal *u* to its peer node towards the sink node. Figure 1(b): Node *i* decides whether to forward *p(t)* or send a data reproduction signal *u(t)* to a peer node w.r.t. the induced extrapolation error. On error, the A(*m*) adjusts the length *m*.

The overall mechanism is illustrated in Figure 1. It is worth noting that, the lower the $v(t)$ gets the more critical is for A(m) to adjust the value of m. The threshold $v(t)$ is based on the current standard deviation $s(t)$ of the received data. Specifically, a high value of $s(t)$ indicates high frequencies in the data stream. In this case, the extrapolation $f(m)$ has to be very precise to capture such high frequencies. This implies a strict threshold $v(t)$ over the induced reconstruction error $e(t)$. On the other hand, a low value of $s(t)$ denotes low variability in the data stream, thus, a relaxation of $v(t)$ can be obtained. For that reasons we define $v(t)$ as an inverse function of $s(t)$, i.e., $v(t) = b.s(t)^{-1}$, $b \in (0, 1]$. A low b value indicates that the WSN application has strict requirements for reproducing the data stream through estimations. We cannot calculate the standard deviation of the whole data stream since we do not know all the measurements a-priori. Hence, we adopt the incremental algorithm proposed in [7] for estimating the current $s(t)$ on the so far received data.

Table I. The decision rules of A(*m*) controller for the value of *a*

a		**Δ*e***		
		negative	**zero**	**positive**
Δ*m*	**-1**	-1	0	+1
	0	0	random {-1, 1}	random {-1, 1}
	1	+1	0	-1

C. The Extrapolation Mechanism

We now elaborate on the extrapolation component $f(m)$ of our model as illustrated in Figure 1. We use Local Linear Regression (LLR) and a Lagrange Polynomial (LP) for implementing $f(m)$.

<u>LLR:</u> The estimating local regression function $f(X)$ over a real domain X fits a function f separately at each target point x_0. This

is achieved by using only those observations –points- close to x_0 to fit the model in a way that the resulting estimated function f^* is *smooth* (has derivatives of all orders) in the domain X. This localization is based on a weighted function or kernel $K_\lambda(x_0, x)$. The kernel assigns weights to x_i based on its distance from x_0. The typical smoothing parameter λ indicates the width of the neighborhood. A larger λ implies lower variance. f^* is structured by a set of m observations and requires no training. The locally weighted regression solves a separate weighted least squares problem at each target point x_0 for a set of m observations (x_i, $f(x_i) = y_i$), for further reading see [14]. It should be noted that although we fit an entire linear model to the data in the region of m observations, we only use it to evaluate the fit at x_0. If we define a vector-valued function $k(x)^{\mathsf{T}} = (1, x)$, the regression matrix $\mathbf{K}$ (of dimensions $m\times 2$) with ith row $k(x_i)^{\mathsf{T}}$ and $\mathbf{W}(x_0)$ the $m\times m$ diagonal matrix with ith diagonal element $K_\lambda(x_0, x_i)$; $\mathbf{W}(x_0) = \mathrm{diag}(K_\lambda(x_0, x_i))_{m\times m}$, then (in closed form),

$$f^*(x_0) = k(x)^{\mathsf{T}}(\mathbf{K}^{\mathsf{T}}\mathbf{W}(x_0)\mathbf{K})^{-1}\mathbf{K}^{\mathsf{T}}\mathbf{W}(x_0)\mathbf{y} \qquad (2)$$

where $\mathbf{y} = (f(x_1), f(x_2), \ldots, f(x_m))^{\mathsf{T}}$. Hence, $f^*(x_0) = g_1(x_0)y_1 + \cdots + g_m(x_0)y_m$. This gives an expression for the LLR estimate and highlights the fact that the estimate is *linear* in the y_i, i.e., the $g_i(x_0)$ does not involve $\mathbf{y}$ [8]. The weights $g_i(x_0)$ combine the kernel $K_\lambda(x_0,x_i)$. Hence, the extrapolated $p^*(t)$ is generated by m previous observations $p(t\text{-}1)$, …, $p(t\text{-}m)$, that is, $p^*(t) = f^*(t) = g_1(t)p(t\text{-}1) + \cdots + g_m(t)p(t\text{-}m)$. The kernel K_λ might be, for instance, the Epanechnikov quadratic kernel [14], defined as:

$$K_\lambda(x_0, x_i) = D\left(\frac{|x - x_0|}{\lambda}\right), D(u) = \begin{cases} \frac{3}{4}\left(1-u^2\right), |u| \le 1 \\ 0, \textbf{otherwise} \end{cases}$$

According to $K_\lambda(x_0, x_i)$, as we "move" the point x_0 from left to right, neighboring points x_i enter the neighborhood initially with weight zero, and then their contribution slowly increases.

LP: The estimating $f^*(t)$ function in the Lagrange polynomial method is polynomial of degree less than or equal to m-1 with form: $f^*(x) = l_1(x)y_1 + \cdots + l_m(x)y_m$, with $l_j(x) = \Pi_{i=1\ldots m}((x - x_i)(x_j - x_i)^{-1})$, $i \neq j$. The extrapolated $p^*(t)$ is generated by m previous observations, i.e., $p^*(t) = f^*(t) = l_1(t)p(t\text{-}1) + \cdots + l_m(t)p(t\text{-}m)$.

It is worth noting that both versions of f (LLR and LP) induce additional computational cost in the WSN nodes, which, obviously, has to be taken into account in evaluating the efficiency of the proposed model.

IV. Performance Assessment

In this section we examine whether ADF achieves efficient data forwarding at the expense of data accuracy. ADF is compared against the Simplest Data Forwarding (SDF) model, which simply forwards all received data. We use real data streams for the contextual parameters temperature and wind speed, which are sampled at 1Hz; there are 990 measurements of each parameter. It is worth noting that the wind-speed readings demonstrate very high variability compared with the variability of the temperature readings (see Figure 2(a)). As it will be shown, the proposed model is evenly energy efficient for both types of readings. We adopted the Mica2 energy consumption model [15]. Mica2 operates with a pair of AA batteries that approximately supply 2200 mAh with effective average voltage 3V. It consumes 20mA if running a sensing application continuously which leads to a lifetime of 100 hours. The energy costs for single CPU instructions (energy per second) and transmitting/receiving either data $p(t)$ or signal $u(t)$ (energy per bit) are summarized in Table II. In addition, the packet header is 7 bytes (MAC header+ CRC) and the preamble overhead is 20 bytes. The temperature and wind speed piece of data payload is 4 bytes (float) and the signal u is only one bit.

Table II: Energy Costs

Node Operation Mode	Energy Cost
Instruction Execution	4 nJ/instruction
Idle – Stand by	9.6 mJ/s - 0.33 mJ/s
Transmitting - Receiving	720 nJ/bit - 110 nJ/bit

We also incorporate the energy cost for implementing LLR, LP and A(m). Hence, the total cost $c(t)$ in Joule at time t for a node i is accumulated as:

$$c(t) = c(t\text{ -}1) + c_{\mathrm{R}}(t) + c_{\mathrm{T}}(t) + c_{\mathrm{I}}(t) + c_0(t) \qquad (3)$$

where $c_{\mathrm{R}}(t)$, $c_{\mathrm{T}}(t)$ are receive (rx) and transmit (tx) costs either for data $p(t)$ or for signal $u(t)$, respectively, and $c_{\mathrm{I}}(t)$ is the energy cost for the CPU instructions of $f(m)$ mechanism. $c_0(t)$ is the state transition cost for node i. The concept is that we conserve energy once the $c_{\mathrm{R}}(t) + c_{\mathrm{T}}(t)$ cost refers more to signal transmissions rather than to data transmissions at the expense of additional $c_{\mathrm{I}}(t)$ and data accuracy. The data accuracy is evaluated through the reconstruction error $e(t)$. We define percentage cost gain $h(t) \in [0, 1]$ when applying ADF w.r.t. SDF, the value:

$$h(t) = \frac{c_{\mathbf{SDF}}(t) - c_{\mathbf{ADF}}(t)}{c_{\mathbf{SDF}}(t)} \qquad (4)$$

We require that $h(t)$ assumes high value. Figure 2(b) depicts the cumulative $c_{\mathrm{I}}(t)$ energy cost for the CPU instructions of LLR and LP mechanism. Obviously, both extrapolation mechanisms incur computational cost for predicting $p^*(t)$ with adjustable length m. The absolute mean reconstruction error for LLR is 33% lower than that of LP for wind-speed, and 27.68% higher than that of LP for temperature data. In addition the mean value of m for LLR is 6.12 (3.37) and for LP is 3.15 (5.28) for wind-speed (temperature) data. Figure 3 depicts the overall cost $c(t)$ of ADF(LLR) (ADF with f = LLR) and ADF(LP) (ADF with f = LP) compared with that of SDF, in Joule, for the temperature measurements. The history length m ranges in {2, …, 15} and the b threshold parameter is b = 0.3 (indicating high data accuracy requirement). The percentage of the transmitted signals ($u(t)$) is 66,8% and 49,1% for ADF(LP) and ADF(LLR), respectively, out of the total transmitted messages. This results to 44.8% and 35.3% increase in the life time of the network, respectively for ADF(LP) and ADF(LLR). Figure 3 depicts also the percentage gain $h(t)$ for ADF(LLR) and ADF(LP) w.r.t. SDF. As illustrated in Figure 3(c), the controller A(m) adjusts m resulting to minimum errors, thus, optimizing the transmissions of data values. The gain stabilizes to a certain value denoting that we obtain 48% and 31.6% more energy savings with ADF(LP) and ADF(LLR), respectively, w.r.t. SDF. Such gains demonstrate the suitability of the proposed model for the problem of energy efficient data forwarding.

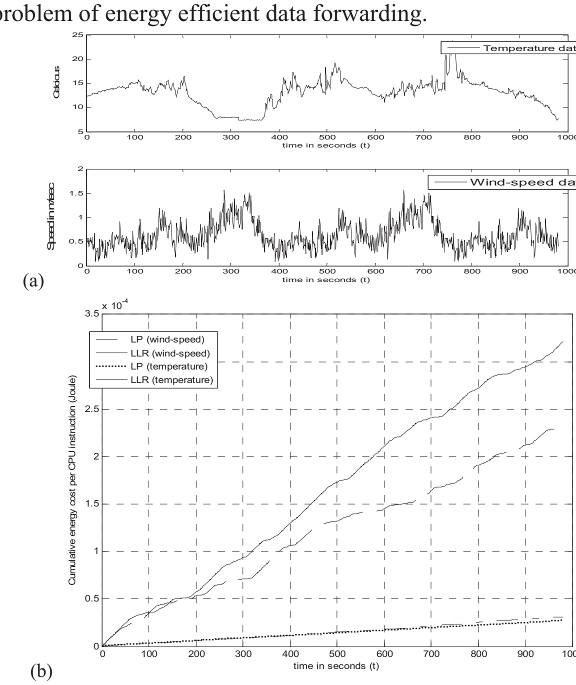

Figure 2. (a) temperature and wind-speed data, (b) the energy cost for the CPU instructions for LLR and LP.

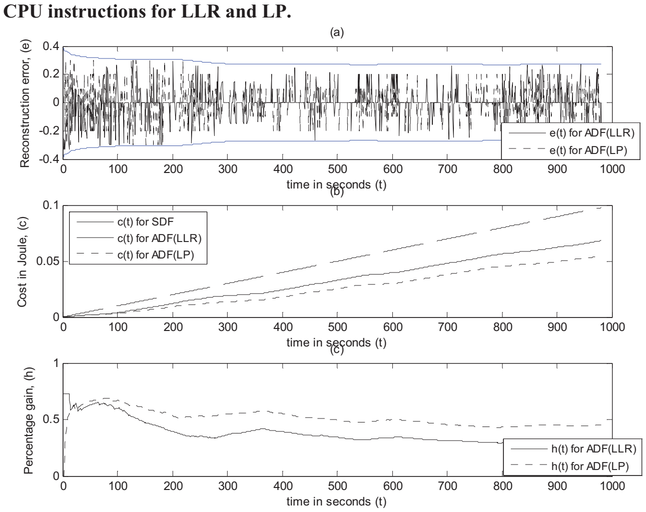

Figure 3. Temperature data: (a) Reconstruction error $e(t)$ and the threshold $v(t)$, (b) total cost $c(t)$ for ADF and SDF, (c) percentage $h(t)$ for ADF.

In addition, in Figure 3 one can observe the absolute reconstruction error and the corresponding dynamic threshold for ADF(LLR) and ADF(LP). We obtain similar benefit for wind-speed measurements having $b = 0.3$. Specifically, Figure 4 depicts the costs and gains for ADF(LLR) and ADF(LP) w.r.t. SDF. We obtain 48.6% and 57.1% increase in the life time of the network, respectively for ADF(LP) and ADF(LLR). That is ADF(LLR) (ADF(LP)) replaces 81% (72%) of the transmitted messages with data reconstruction signals ($u(t)$) between nodes. This is achieved because the A(m) is capable of dynamically adjusting the value of m resulting to more accurate extrapolations. In addition, the gain indicates 49% and 56.7% more energy savings for ADF(LP) and ADF(LLR), respectively.

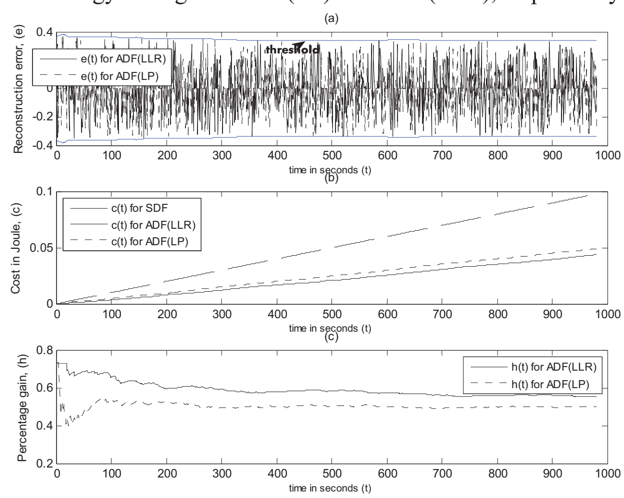

Figure 4. Wind-speed data: (a) Reconstruction error $e(t)$ and the threshold $v(t)$, (b) total cost $c(t)$ for ADF and SDF, (c) percentage $h(t)$ for ADF.

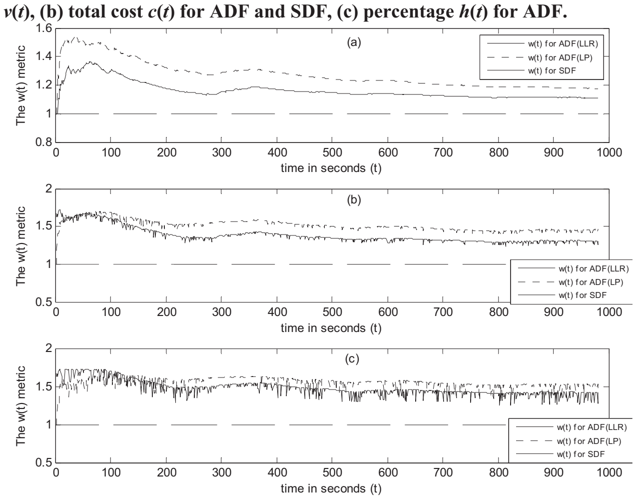

Figure 5. Temperature data: the $w(t)$ metric (a) for $b = 0.1$, (b) for $b = 0.3$, and (c) for $b = 0.5$.

We have to assess the benefit of ADF by taking into account not only the energy savings but also the induced reconstruction error. The controller A(m) tries to adjust the value of m in order for the extrapolator to predict an estimate $p^*(t)$ as close as possible to $p(t)$. For that reason, we define the metric $w(t) \in [0,$

2] which jointly indicates the percentage of gain $h(t)$ and the relative reconstruction error $e^*(t) = |e(t)|/p(t)$, that is,

$$w(t) = h(t) + \left| \frac{p(t)}{p(t)+e(t)} \right| = h(t) + \frac{1}{1+e^*(t)} \quad (5)$$

In ADF, $w(t)$ should get values close to 2, i.e., promote energy efficient ($h(t) \to 1$) with minimum error ($e^*(t) \to 0$). That is the w metric assesses the energy efficiency of the proposed model along with the corresponding reconstruction error. Evidently, for the SDF forwarding scheme we obtain $w(t) = 1$. Figure 5 depicts the w metric of ADF for values of $b = (0.1, 0.3, 0.5)$ w.r.t SDF for temperature data. We can observe that the proposed model provides energy efficiency in data dissemination with low reproduction error. Specifically, the ADF model always assumes higher values in the w metric ($w = 1.5$ for $b = 0.3$ and $b = 0.5$) even for very strict data thresholds ($w = 1.2$ for $b = 0.1$) w.r.t SDF model ($w = 1$). Obviously, the higher the b is the higher the w gets. We obtain similar results for the w metric when dealing with wind-speed data, as shown in Figure 6. Specifically, the ADF model assumes $w = 1.7$ (1.58) for $b = 0.5$ ($b = 0.3$). Especially, for very strict data thresholds, we obtain $w = 1.25$ for $b = 0.1$ w.r.t SDF model ($w = 1$).

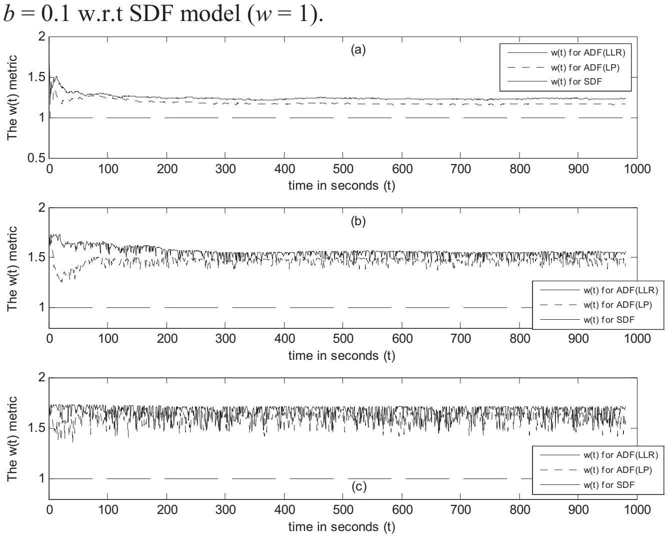

Figure 6. Wind-speed data: the *w(t)* metric (a) for *b* = 0.1, (b) for *b* = 0.3, and (c) for *b* = 0.5.

V. Conclusions

In this paper we propose an adaptive data forwarding model (ADF) in WSN. The ADF mechanism decides on forwarding messages between network nodes based on approximation techniques over the received data. Its main objective is to achieve a balance between energy consumption and error propagation. Energy consumption is reduced by avoiding unnecessary message transmissions if the reconstruction error is acceptable. In order to retain the corresponding error in acceptable levels we propose a controller that dynamically adjusts the size of the history buffer (last m measurements) that is needed by the approximation techniques. We assess the performance of ADF w.r.t. SDF with real data streams (temperature and wind-speed contextual data) using the LLR and LP extrapolation methods. Other extrapolation methods e.g., splines, are also applicable. We show that ADF fulfils its design objective and prolongs the network life time while keeping reconstruction error at very low levels.

Our plans for future work include intelligent data aggregation schemes. These schemes may significantly reduce the upstream communication requirements by merging diverse data flows at a certain level. Such schemes, along with the transmission suppression scheme discussed in the present paper, are expected to significantly improve the operational characteristics of WSN.

References

[1] Y. Li, My T. Thai, W. Wu (Eds.), “Wireless Sensor Networks and Applications”, Springer Series on Signals and Communication Technology, Springer, e-ISBN 978-0-387-49592-7, 2008.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, “Wireless sensor networks: a survey”, Computer Networks, Elsevier, 38(4), 393-422, 2002.

[3] M.Athanassoulis, I.Alagiannis, S.Hadjiefthymiades, “Energy Efficiency in Wireless Sensor Networks: A Utility-Based Architecture”, in Proc. of the 13th European Wireless Conference, April 2007, idn: 1569014800.

[4] C. Intanagonwiwat, R. Govindan, D. Estrin, “Directed diffusion: a scalable and robust communication paradigm for sensor networks”, in Proc. of the ACM MobiCom '00, 2000.

[5] A. Manjeshwar, D. Agrawal, "TEEN: A Protocol for Enhanced Efficiency in Wireless Sensor Networks," in Proc. of the 1st Intl. Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, April 2001.

[6] R. Shah, J. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks", in Proc. of the IEEE Wireless Communications and Networking Conference, March 2002.

[7] D. Knuth (1998). *The Art of Computer Programming*, volume 2: *Seminumerical Algorithms*, 3rd edn., p. 232. Boston: Addison-Wesley.

[8] Simonoff, Jeffrey S. (1996). *Smoothing Methods in Statistics*. Springer Series in Statistics, ISBN 0-387-94716-7.

[9] A. Manjeshwar, D. Agrawal, “APTEEN: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks”, in Proc. of the Parallel and Distributed Processing Symposium, IPDPS 2002, Page(s):195 – 202

[10] H. Lee, A. Klappenecker, K. Lee, L. Lin. "Energy Efficient Data Management for Wireless Sensor Networks with Data Sink Failure", in Proc. of the IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, November 2005, 7 pp. - 210.

[11] R. Arroyo-Valles, A. Marques, J. Cid-Sueiro, *Optimal Selective Transmission under Energy Constraints in Sensor Networks*, IEEE Transactions on Mobile Computing, 8(11), pp: 1524-1538, 2009.

[12] S. Rout, A. K. Turuk, B. Sahoo. "Energy Efficiency in Wireless Network: Through Alternate Path Routing", in Proc. of the 4th Innovative Conference on Embedded Systems, Mobile Communication and Computing, 2009, pp. 3-8.

[13] C. Tung Chou, R. Rana, W Hu. “Energy efficient information collection in wireless sensor networks using adaptive compressive sensing”, in Proc. of IEEE Conference on Local Computer Networks, pp. 443-450, 2009.

[14] T. Hastie, R. Tibshirani, J. Friedman, *The elements of statistical learning: data mining, inference, and prediction,* 2nd Edition, Springer Series in Statistics, 2009.

[15] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, B. Krogh, “Energy-efficient surveillance system using wireless sensor networks”, in Proc. of 2nd Intl. Conf. on Mobile Systems, Applications, and Services, (ACM MobiSys '04), pp.270-283, June 2004.