We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Data Reduction in Low Powered Wireless Sensor Networks

Qutub Ali Bakhtiar, Kia Makki and Niki Pissinou

Additional information is available at the end of the chapter

1. Introduction

A Wireless sensor network (WSN) [1] is a special kind of a peer to peer network where the nodes communicate with the sink wirelessly to transmit the sensed information. In contemporary world, a WSN utilizes different technological advancements in low power communications and Very Large Scale Integration (VLSI) [16] to support functionalities of sensing, processing and communications. WSN's have penetrated in all walks of life ranging from health care, environmental monitoring to defense related applications. One of the major challenges faced by WSN's is that of energy conservation. Researchers all over the world have been trying their best to make sure that WSN efficiently make use of the energy resources and increase their life span. In this Chapter one such technique for energy conservation called 'Data Reduction' is discussed.

Data reduction can be defined as the process of conversion of all the information in a finite data set into fewer subsets and later on regenerating the entire set using the reduced subset. Data reduction is often the first step to tackle a data set because it facilitates in extracting its unique features. Typically, data reduction techniques are used for data mining large data warehouses.

In WSN, data reduction forces the sensor nodes to stop transmitting the data when it is confident about regenerating the future data at the sensor sink based on the existing, past and proximity observations thereby conserving the energy resources used for transmission of data.

A common way to facilitate data reduction in WSN is by deploying adaptation and prediction mechanism at the source and the destination so as to adapt to the changing pattern of the data and to predict it. The above mechanism is efficient in conserving the communication resources involved as it requires the source to relay only a subset of the actual data [17]. Moreover, since the radio transmission at the node consumes more amount of energy than any other operation

at the node [1],[13] data reduction becomes an attractive option to conserve the limited energy resources of WSN.

Data reduction in WSN is a challenging process as data exists in the form of continuous stream (infinitely large data set) where the adaptation and prediction has to be performed online i.e. at a given instance of time not all the information is available for processing. Typically, the spatial and temporal relations among the data sources in WSN are exploited to achieve fair data reduction rates. Spatial and temporal characteristics of WSN's facilitate in adapting the environment and then predicting it thereby resulting in data reduction.

2. Data reduction systems in WSN

Data reduction systems have been deployed in many environments and scenarios to achieve conserve communication resources. Discrete fourier transforms (DFT) [21] is a classic technique used to perform data reduction for temporally related data streams. Based on DFT, researchers have more recently developed an advanced method Discrete Wavelet Transform (DWT) [5] to achieve data reduction. Also techniques such as, Singular Value Decomposition (SVD) [6] based on traditional Principal Components Analysis (PCA) [15] is an attractive data reduction technique because of its ability to provide optimal data reduction. Random projection of time series [4] is another technique which has exhibited great promise and demonstrated good results since it can provide approximate answers with guaranteed bounds of errors. Stated below are some of the prominent data reduction schemes presented in the literature for WSN.

2.1. Barbie-Q: A tiny-model query system (BBQ)

BBQ [9] is a data acquisition model for sensor networks that incorporates statistical models of real-world processes into a sensor network query processing architecture. The problem that BBQ addresses is that given a query and a model; identify a data acquisition plan for the sensor network to best refine the query answer. BBQ uses a specific model based on time-varying multivariate Gaussians in its architecture to compare it with the incoming data stream. If all of these probabilities generated meet or exceed user specified confidence threshold, then the requested readings are directly reported as the means in the probability density function. BBQ imparts confidence on the posterior density generated by time varying multivariate Gaussians and optimizes the expected benefit and cost of observing the attributes.

BBQ depends heavily on the prediction ability of the time varying multivariate Gaussians. In case of high nonlinearity among the sensed observations, the predictive ability of the multivariate Gaussians (markovian in case of dynamic environments) fails thereby responding the query incorrectly in spite of having a confidence in the model due to previous sensed observations.

On the guidelines of BBQ, the concept of multivariate Gaussians was exploited by Eiman et.al [10] in their design to propose a context (spatial and temporal) aware data cleaning mechanism. The proposed model by Eiman et. al. is more effective and efficient since it is contextually aware about the environment.

2.2. Probabilistic Adaptable Query system (PAQ)

Daniela et.al [25] proposed a general framework PAQ to efficiently answer queries at the sink based on a simple type of time-series model called an autoregressive model (AR) [21]. PAQ uses a combination of AR models to probabilistically answer queries. The model is used both globally, at the sink, to predict the readings of individual sensors, and locally, at each sensor, to detect when the sensor produces outlier readings or when the model ceases to properly fit the data. The sink maintains one AR model per geographic cluster and one cluster head in each cluster. The cluster head is responsible for communicating with the sink on behalf of its cluster. The cluster head's AR model, called the cluster model, is used to predict the values of all sensors in the cluster with an error of at most a fixed threshold over the member sensor's local models with the same confidence.

The sink maintains the coefficients associated with each of the leader's models and receives periodic readings from them. It also maintains a list of the current clusters. The cluster head's models and the cluster sets stored at the sink allow the sink to answer queries over all sensors using just the cluster models. To reduce communications, clusters are computed locally by the cluster heads and members of each cluster. Each cluster head is responsible for notifying the sink of changes in its model coefficients or in its cluster members, and for transmitting periodic readings to the sink.

Although the proposed mode achieves stable data reduction rates, one of the shortcomings of this approach is the communication of the AR parameters from the local cluster head to the sink. If the parameters do not reach the sink successfully or they get corrupted on the way then the AR models fails to synchronize at the node and the sink, ultimately causing improper prediction of the data.

2.3. Similarity-based Adaptive Framework (SAF)

Daniela et.al modified PAQ to incorporate the spatial attributes of sensor networks by using similarities among the node to propose SAF [24]. The premise behind SAF as similar to PAQ is to build local prediction models at each node, transmit them to the root of the network and use them to answer user queries. SAF also provides a mechanism to detect data similarities between nodes and organize nodes into clusters at the sink at no additional communication/complexity expense on the nodes which is absent in PAQ. This is achieved by exploiting properties of local time series models, and by means of utilizing data similarity between nodes that is based on the prediction values.

The prediction models in SAF are developed using lightweight linear time series models built by each node from a small number of readings (enabling models to be quickly re-learned) and stored at the sink. Sensor nodes and the sink communicate occasionally to exchange models or answer queries that require more accuracy than the stored models can provide. SAF attempts to group sensor nodes into clusters under dynamic conditions, which is a challenging problem as continuous monitoring and adaptation of cluster membership is required. The clustering algorithm used in SAF has another benefit that it does not require nodes in the same cluster to be geographically co-located, and does not require nodes to communicate at all with each other, thus making the clusters highly adaptable.

SAF is based on sound mathematical principles. However, it uses prediction variables to cluster the nodes according to the similarity based on prediction. Any misconduct exhibited by the temporal behaviour (a common feature in noisy wireless channels and hardware inefficient sensing devices) of the sensed phenomena introduces a certain amount of uncertainty while identifying the commonalities among the data, causing the entire model to fail. Also the exchange of time series parameters is initiated from the nodes and then passed on to the sink i.e the complexity due to the adaptation of the pattern is forced on to the nodes and moreover there is no mechanism present in the model that ensures that the time series prediction parameters have successfully reached the sink.

2.4. Adaptive Model Selection (AMS)

Santini et. al. proposed a lightweight, online algorithm AMS [22, 23] that allows sensor nodes to autonomously determine a statistically good performing model among a set of candidate models. AMS exploits the fact that, gathered sensor data is usually accepted to lie within a known error bound. A sensor node regularly collecting local measurements can fit a prediction model to the real data and communicate it to the sink, which can then use the model to compute estimates of future sensor readings. The rationale of this approach is to use complex prediction models only if they prove to be efficient both in terms of computation and achievable communication savings, and otherwise to rely on simpler models. AMS maintains a database of complex models at the sink and a mechanism to identify appropriate models depending on the computational ability and the requirement of the application.

AMS fails to demarcate the differences between adaptation and prediction mechanisms employed. The model uses a single complex algorithm to perform both the tasks although there is a significant difference between them. A faster adaptation algorithm and a robust prediction algorithm can always be used to overcome this disadvantage. The other drawback of AMS is that no clear information about the number and kind of complex models that are supposed to be stored has been specified. In reality it is not possible to store all the models that have the ability to perfectly emulate the environment that is being sensed.

2.5. Ken

Ken [7] is robust approximate technique that uses replicated dynamic probabilistic models to minimize communication from sensor nodes to the network's PC base station. Ken focuses on to intelligently exploit spatial correlations across sensor nodes without imposing undue sensor-to-sensor communication burdens to maintain the models. The basic premise behind Ken is simple: both source and sink maintain a dynamic probabilistic model of how data evolves, and these models are always kept in sync. The sink uses the data value(s) predicted by the model as the approximation to the true data, and the source, who knows the predicted value by virtue of running a copy of the model, makes sure that the predicted data values satisfy the required bounded-loss approximation guarantees, by communicating some information to the consumer as required. Ken uses disjoint clique approach to exploit the spatial correlations that exist among the nodes to achieve data reduction.

The main drawback of Ken is that complexity analysis has been completely ignored and is there no clear methodology to adaptively generate a probabilistic model. Ken justifies the

claims made by using a Markovian approach where the information is incorporated in the model by conditioning (conditional probability). Ken achieves good results if the incoming data stream is linear in nature as it completely relies on conditional probability concept. However in reality the dynamics of the sensing information in its worst case are so strong that no fixed model is in a position to predict it. The non linearity exhibited by the sensed information causes the probability distribution to change and many times conditioning is not a reasonable approach to forecast the data successfully. In fact, incase of high nonlinearity there might be instances where conditioning further worsens the prediction ability of the model.

2.6. Prediction based Monitoring (PREMON)

PREMON [11] uses the coding concepts of MPEG that restores the quality of image using the partial image parameters. The work focuses on monitoring a sensor network in an energy efficient way by predicting all the readings which can be forecasted within a specified threshold of error. This work marked the beginning energy efficient monitoring of WSN by using prediction. The other novelty of this work is the usage of MPEG concepts used to increase the quality of ill formulated figure by utilizing efficient compression and coding techniques.

The entire process initiates by nodes sending the updates to the prediction model generator at the sink. The sink generates the model and passes the details of the model to the nodes which predict the sensed stream of data and halt the radio transmission saving the energy resources.

2.7. Dual Kalman Filter

Jain et.al proposed Dual Kalman Filter (DKF) [14] architecture as a general and adaptable solution to stream resource management problem. The architecture and the process is similar to PREMON as proposed by Goel et. al. the major difference between the usage of Kalman filter instead of MPEG concepts. The DKF model contains kalman filters at the sink synchronized with the kalman filter at each of the node. A major assumption made here is that the filter parameters are the same at the node and the sink at any given instance of time. The assumption is unreasonable for wireless and sensitive nodes as the communicating medium and radio resources are far from ideal.

Another drawback of the kalman filters is that the prior information about the input stream has to be provided as the input in order to train the filter. The size of training sequence is critical as in case of online data streaming systems the entire past data is not available at your disposal.

2.8. Approximate replication of data

Chris Olston in his PhD dissertation [17] proposed a data reduction method that operates on approximations of the sensor data distributions. The general approach presented by him achieves a substantial energy savings for the network as the data requirement is semi accurate in nature. The approximate mode of processing is extremely useful for two reasons. Firstly, it allows answering queries fast and cheaply, since all the nodes of the network relevant to the query need not be visited in order to get the answer. Secondly, it enables the execution of

queries that would otherwise not be able to answer without consuming a lot more resources. The approximate requirement of the application also sheds off some load on the complexity and accuracy requirement of the prediction mechanism used.

Olston et. al. [18–20] proposed TRAPP (Trade off in replication precision and performance) architecture which maintains synchronous interval ranges at the sink and the node within the scope of the sensed observation. Each node is assigned a burden based on the data observation it senses. The burden represents the cost required to transmit the data to the sink. A similar approach has been used in WMA architecture for data cleaning where each node is assigned a weight based on its ability to provide a clean data.

Primarily, TRAPP deals with optimizing the precision and performance of the approximate data replication process between the node and the sink. The precision of the replicated data is inversely proportional to the interval where the value is detected to lie in. Although TRAPP has been developed for handling continuous queries its adaptive algorithm is also well equipped to handle the unprecedented one time aggregate queries that involves sum and the average functionalities. The adaptive algorithm reduces/increases the burden by increasing/decreasing the interval limit associated with each node and is said to converge in a steady state if the burden on all the nodes is uniform.

Although TRAPP provides huge energy savings for WSN, incase of one time queries that requires a response within a precision there is a high probability that it may fail to respond.

3. Data reduction in WSN using adaptive filters

Bakhtiar et. al. [2, 2] in their work demonstrated that data reduction in WSN can be achieved by deploying coordinating adaptive filters at the node and the sink to perform the operation of adaptation and prediction. The filters initially adapt to the varying nature of the sensed data until they converge. Convergence means that the error between the filter input and the filter output is within a specified threshold and the filters are ready to predict the future information.

After a successful convergence the node stops transmitting the data and the model moves to the prediction mode where the filters at both the ends begin to predict the future data. During the prediction mode the output of the filter at the sink is considered as the estimated sensed data, whereas the output of the filter at the node is compared with the sensed data to verify the prediction performance of the filter. The entire mechanism moves from the prediction phase to the adaptation phase if the error between the sensed data and the output of the filter at the node goes above a specified threshold.

In this Section, different adaptive filter algorithms presented in the literature [12] are used for Data Reduction in WSN. Primarily, the use of different classical estimation/prediction algorithms at the sink employed to reduce the set of transmitted data from the node is focussed. A special emphasis is also given on the initialization vector of the filters and a method to initialize the filters during the adaptation and the prediction phase is discussed. The presented model as stated in [2, 3] is tested using Least Mean Square LMS, Normalized Mean Square (NLMS), Recursive Least Squares (RLS) algorithms [12] at the sensor sink and the sensor node. Some modifications in the LMS algorithm employed during the prediction

mode are also discussed and justification is made why the new modifications achieve better results than the traditional algorithm using appropriate simulations.

4. Model

Assume that there is a stream of data $\mathbf{x}(k)$ that is to be transmitted from a node to the sink

$$\mathbf{x}(k) = x(k) \; x(k-1) \ldots x(k-N) \tag{1}$$

Consider a N tap filter with the weights

$$\mathbf{W}(k) = \mathbf{w}_1(k) \; \mathbf{w}_2(k) \ldots \mathbf{w}_N(k) \tag{2}$$

where $\mathbf{w}_N(k)$ is the set of window taps at the k_{th} input sample. Let the desired signal be denoted by $d(k)$ which is the output value of the filter at $(k-1)_{th}$ instance during prediction and the sensed data during adaptation. The error in prediction is given by $e(k) = d(k) - y(k)$ where $y(k) = \mathbf{w}(k)\mathbf{x}^T(k)$ is the output of the filter.

Algorithm	Tap Update Function
LMS	$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k)\mathbf{x}(k)$ $e(k) = [d(k) - y(k)]$
NLMS	$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu_n}{\|\mathbf{x}\|^2} e(k)\mathbf{x}(k)$ $e(k) = [d(k) - y(k)]$
RLS	$\mathbf{K}(k) = \frac{\rho^{-1}\mathbf{P}[k-1]\mathbf{x}(k)}{1+\rho^{-1}\mathbf{x}^H(k)\mathbf{P}[k-1]\mathbf{x}(k)}$ $\mathbf{P}[k] = \rho^{-1}\mathbf{P}[k-1] + \rho^{-1}\mathbf{K}(k)\mathbf{x}(k)$ $\mathbf{w}(k) = \mathbf{w}(k-1) + \mathbf{K}(k)e(k)$ $e(k) = [d(k) - y(k)]$

Table 1. Adaptive algorithms

The model discussed here is similar to that as presented in [3, 7, 23, 25] where coordinating filters are deployed at the sink and the node to adapt and predict the sensed information. The model is primarily divided into two modes, the adaptation mode and the prediction mode.

During the adaptation mode the filter uses the sensed information to adapt the sensing environment. It is during this mode that the sensed information is relayed to the sink and the filter at the node is idle. Once the filters are converged they begin to predict and if the difference between the predicted output and the sensed information remains below the threshold for N samples the model moves to the prediction mode. During the prediction mode the sensing nodes does not relay the sensed information to the sink, rather the estimated output at the sink is treated as the sensed data.
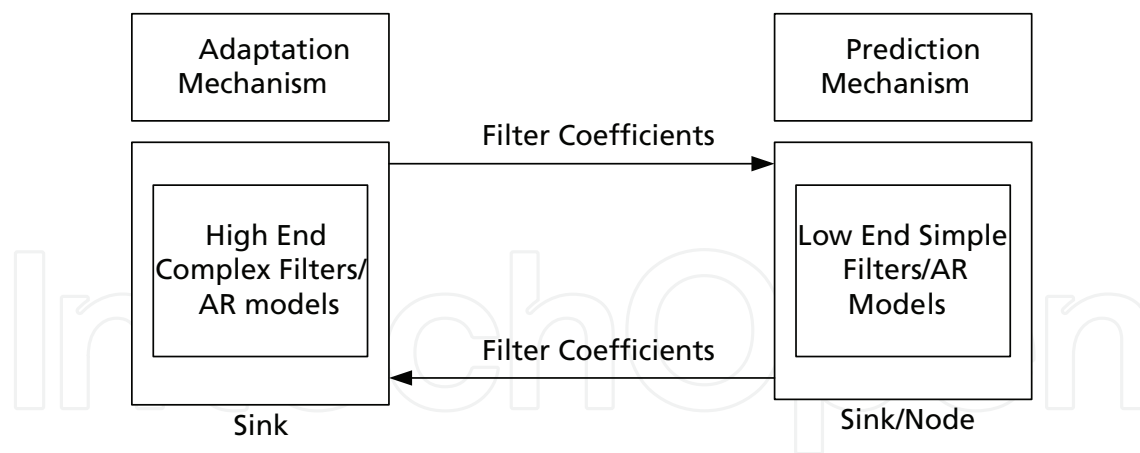
Figure 1. Flow diagram of the proposed model

In the methodology of data reduction presented in [3], the functionality of adaptation is restricted to sink only and the filter at the node remains idle until it receives the converged parameters of the filter from the sink. The authors then extend their previous work by incorporating high end complex filters at the sink during the adaptation mode [2]. The premise behind this approach is the ability of the sink to handle complex algorithms that offers fast and good convergence with an ease. Also, a simple prediction algorithm at the node to make sure that the energy resources at the node are utilized efficiently is also used in the above mentioned work.

The other major change of this work is the usage of filter coefficients during initialization of the mode. As shown in Figure 1, whenever there is a change in the mode of the model, the filter coefficients are exchanged. The filter parameters at the end of any mode are passed on to the filter of the other mode. These parameters serve as a initialization vector of the filter and assist in faster convergence.

As shown in Figure 2 the radio transmission of the node can be assumed to be controlled by the output of the prediction filter at the node. Each sensed sample is compared with the estimated output during the prediction mode and if the difference between the sensed data and the estimated output is below a specified threshold the switch is turned *off* i.e. the the node does not relay the sensed data to the sink. During this process the prediction filter at the sink estimates the sensed data and the approximate predicted output is used by the end application as the sensed information.

Incase, the difference between the predicted output and the sensed data goes above the threshold level, the node stops the prediction process and relays the sensed information to the sink. At this instance the switch is turned *on*. The sink uses the current filter parameters of the prediction filter and the sensed information to initialize the adaptation process and thus the entire model moves from the prediction mode to adaptation mode. The entire process repeats itself and the radio transmission at the node is accordingly controlled. The model is stable enough as any incapability of the prediction filter will result in the transmission of data from the node.

Another major change observed in this work is the usage of the prediction algorithm. During the prediction phase LMS algorithm is used only as it is the simplest and robust among
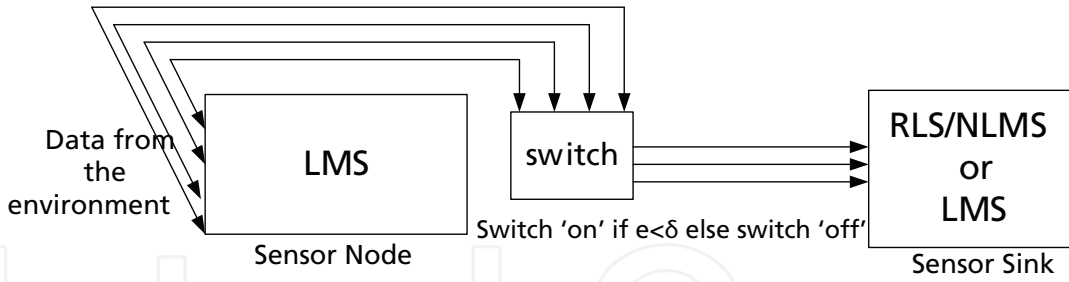
Figure 2. Transmission Control mechanism

the other liner prediction techniques. A more faster convergent algorithm can be used for prediction but faster convergent algorithms have more likelihood to diverge faster especially while performing prediction using the feedback mechanism. However, linear prediction techniques ideally forecast the nth value provided the filter has the prior information of $(n-1)$th values and if the prediction is extended to forecast the $(n+k)$th value during approximate replication using the feedback mechanism no significant change is observed in the output. The other problem that faced by LMS is that the step size parameter should have a common value during each prediction phase at the sink and the node.

In order to overcome the above mentioned drawbacks the authors modify the window tap update equation by removing the step size parameter from it. Intuitively, it can be said that the window tap update is representation of the input trend of data to the filter. It depends on the error between the output of the filter and the input at a particular instant which incase of data reduction is less than δ. The basic LMS equation is given by.

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k)\mathbf{x}(k) \tag{3}$$

where μ is the step size parameter which can have values ranging from [12]

$$0 < \mu < \frac{2}{NS_{max}} \tag{4}$$

where S_{max} is the maximum value of the power spectral density of the tap inputs $\mathbf{x}(k)$ and N is the filter length. Assume that the value of the step size parameter be given by

$$\mu = \frac{1}{N(\mathbf{x}(k)*\mathbf{x}^T(k))} \tag{5}$$

During the prediction phase of the data reduction the value of $\mathbf{x}(k)$ changes minimally and the difference among the elements of the array $\mathbf{x}(k)$ is less than δ. Using the above facts and substituting the value of μ in Equation 3 the following tap update equation is obtained.

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{e(k)}{N\sum \mathbf{x}(k)} \tag{6}$$

The main objective of Equation 6 is to sub optimally predict the sensed information from the environment for a longer duration of time. The stability of the modified algorithm remains the same as that of LMS as the value of the step size parameter lies within the bounds of stability and moreover is use to educate the algorithm about the conditions of approximate replication.

5. Results

In order to prove the claims made, two simulation scenarios involving real time data sensed by two nodes from Intel lab [8] are performed on Matlab. Out of 54 nodes deployed in the Intel Lab, Node-4 and Node-11 is selectd for simulation. A total number of 43790 temperature samples are used in case of node 4 and 41833 in the case of Node-11. The threshold level is set to +/-0.5. The initial tap weights are assigned to be zeros. The length of the filter is chosen to be 3 (i.e. N=3). The value of μ for LMS is selected as 7.10^{-5} [23]. Let A_n be the number of times the model moves from the prediction mode to the adaptation mode. The model is tested by using the algorithms mentioned in Table 1.

Node 4 , 43790 Samples				Node 11 , 41833 Samples			
	LMS	NLMS	RLS		LMS	NLMS	RLS
Predicted Data	42228	42534	42904	Predicted Data	40495	41136	41178
Transmitted Data	1562	1256	856	Transmitted Data	1338	697	655
A_n	866	728	653	A_n	806	615	566

Table 2. Comparison of convergence rates for Node 4 and Node 11 using different algorithms when traditional LMS is used in the prediction phase

Initialization Method	LMS	NLMS	RLS
$W_{adapt}(n) = 0$	39179	40745	40942
$W_{adapt}(n) = W_{pred}(n)$	42228	42534	42904

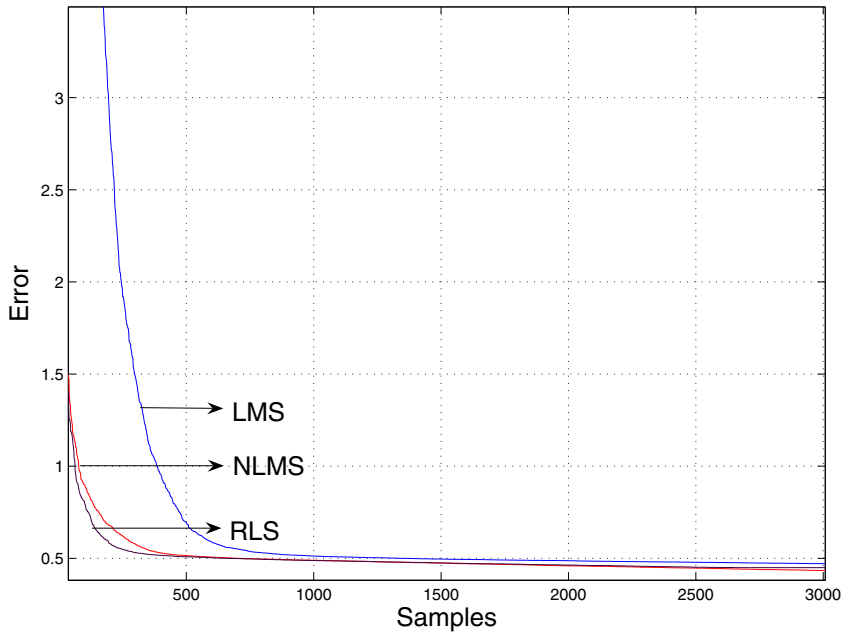**Table 3.** Comparison of different initialization methods on Node 4

Figure 3. Convergence rates of the filters during adaptation mode at the sink using different algorithms for sensor node11
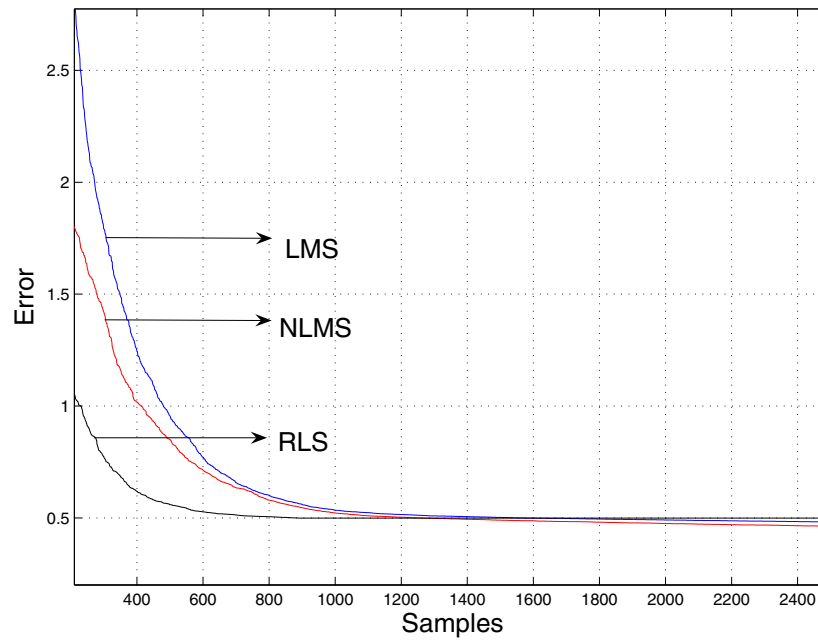
Figure 4. Convergence rates of the filters during adaptation mode at the sink using different algorithms for sensor node4

From Figure 3 and Figure 4 it is evident that the filter error (i.e the difference between the sensed data and filter output) of the sink converges faster when complex algorithms like NLMS, RLS are deployed. From Table 5 is is also observed that the value of A_n (the number of times the model shifts from the prediction mode to adaptation) is small when a more complex algorithm is used for adaptation. It is due to the fact that algorithms such has NLMS,RLS have better adaptation capabilities than LMS [12]. From Table 5 it can also be inferred that the number of samples required to approximately predict the remaining samples reduces when a better adaptation mechanism is deployed at the sink. As opposed to the previous model where only a simple adaptation and mechanism using LMS was used [23] due to limited node capabilities, a marked improvement in the energy savings is observed. This is due to the deployment of NLMS and RLS that achieves a better data reduction of 306 and 706 for node 4 and 641 and 683 samples respectively.

It can also be inferred from Table 5 that the initialization vector of the algorithm during adaptation has an impact on the rate of convergence. Whenever the process moves from the prediction mode to the adaptation mode, filter parameters of the prediction mechanism at the last instance to initialize the filter during the adaptation process are used. An improvement of 3049,1789,2412 while convergence is observed for LMS, NLMS and RLS algorithms when the sensed samples at node 4 were considered.

Figures 5 & 6 depict the window tap behaviour when traditional and modified LMS is used for prediction. It is observed that when traditional LMS is used a significant variation of window tap is achieved than when compared to the modified one. The traditional LMS does not exploit the conditions of data reduction and always tries to achieve an optimal solution.
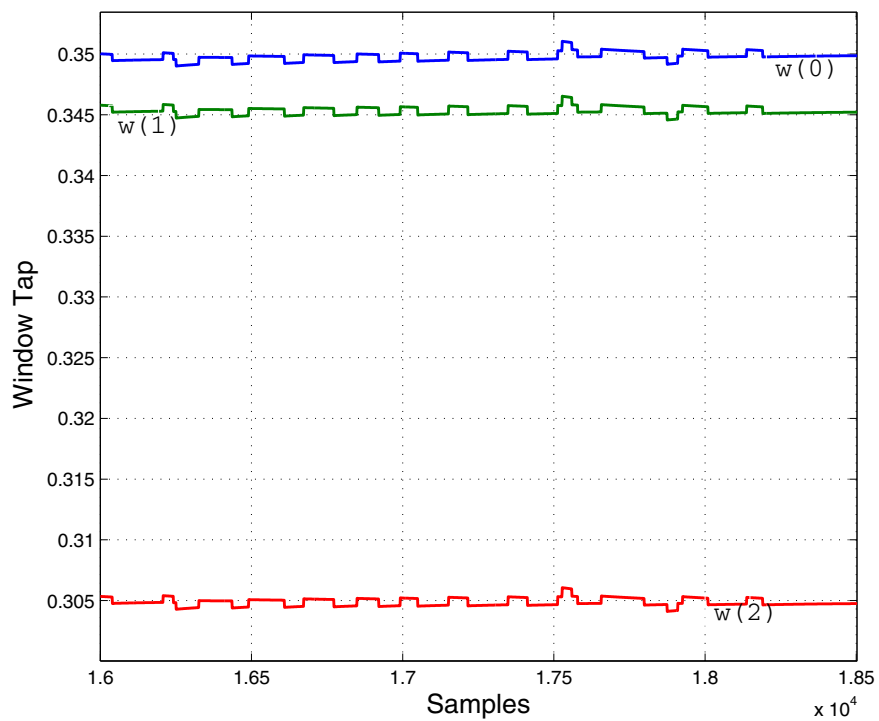
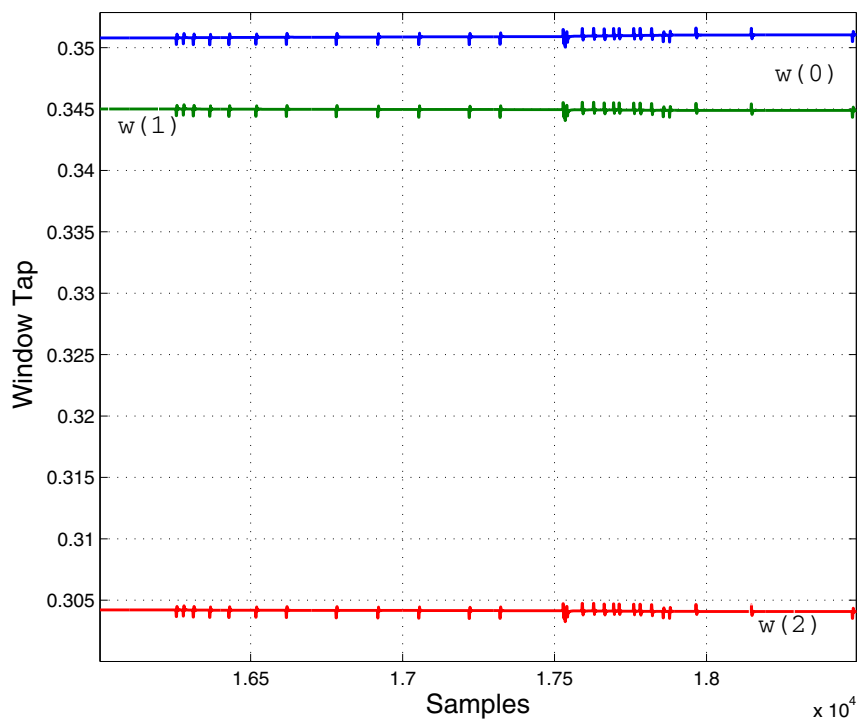**Figure 5.** Amplitudes of the window taps when LMS is used for prediction

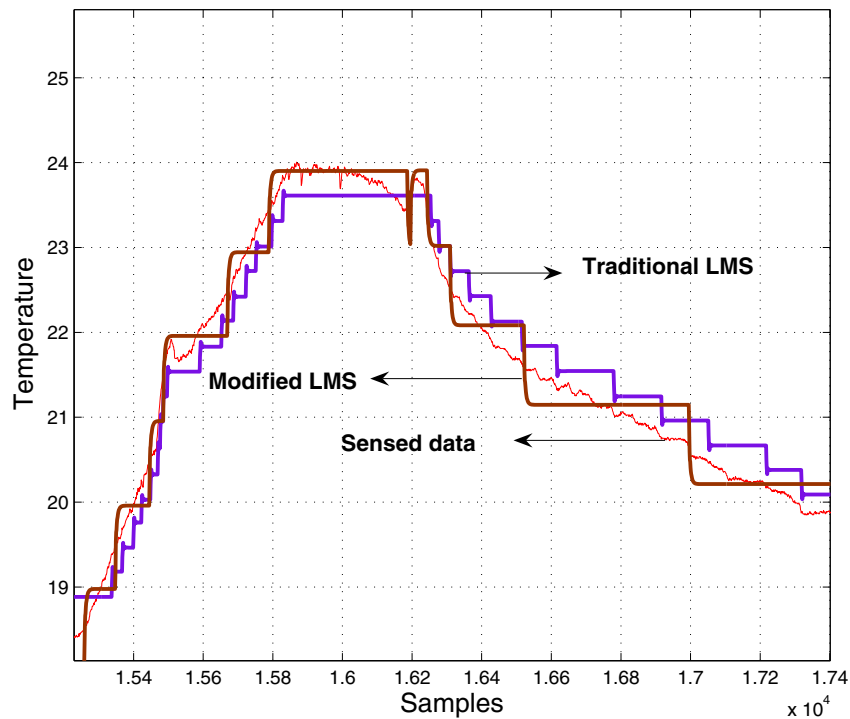**Figure 6.** Amplitudes of the window taps when modified LMS is used for prediction

Figure 7. Comparison of modified LMS and traditional LMS when NLMS is used for adaptation at the sink

Node 4 , 43790 Samples				Node 11 , 41833 Samples			
	LMS	NLMS	RLS		LMS	NLMS	RLS
Predicted Data	42270	42998	43068	Predicted Data	40498	41270	41280
Transmitted Data	1520	792	722	Transmitted Data	1335	563	553
A_n	880	617	579	A_n	803	498	494

Table 4. Comparison of convergence rates for Node 4 and Node 11 using different algorithms with modified LMS in the prediction phase

Table 5 provides the predicted and the transmitted samples when modified LMS is used during the prediction phase. On comparison of Table 5 with Table 5 it can be concluded that whenever NLMS and RLS are used for adaptation, modified LMS should be used for prediction on the other hand when LMS is used for adaptation, traditional LMS should be used for prediction. The window taps exhibit a better shift from adaptation phase to prediction phase when LMS is used in both the phases.

6. Conclusion

This Chapter provides an overview of the data reduction systems that have been proposed in the last decade for Wireless Sensor Networks. In order to further elaborate the process of data reduction for WSN this Chapter delves into the dual filter model proposed by Bakhtiar et.al. [2, 2] that emphasizes on initialization of the filters and exchange of parameters between the filters. The model is mathematically elucidated by means of equations and tables. Also

the environment is simulated and tested with real time data from Intel Labs [8]. The results obtained reaffirm the fact that data reduction obtains high communication resource savings and has a great impact in the extending the life span of the WSN. Although a significant amount of work is ongoing [26],[27],[28] on topics related to data reduction in WSN, there are open issues that can be addressed in the future, specifically when data reduction is achieved by means of adaptive filters. The issues are stated as follows

- **Synchronization** In data reduction models involving dual filters it is very important to maintain the synchronization of the filters at node and sink. By synchronization it is meant that at a given instance both the filters have to be in the same mode. In real time scenario synchronization adds an overhead to the communication process and any error or mistiming leads to the failure of the model.

- **Modeling the sensed data**: Prediction is usually carried out after modeling the data to some fixed form. There are many methods described in the literature that model the incoming data to some mathematical form. At the sink end this mathematical form can be used to predict the data.

- **Algorithms**: The present day algorithms do not fully exploit the threshold value restriction that is imposed at the sensing node. A new process which effectively exploits this condition can always be developed and deployed in the data reduction mechanism.

- **Spatial Considerations**: A filter which does the dual function of co-relation and fusion can be employed at the sensor sink by exploiting the spatial characteristics of the WSN. The correlation attribute will further enhance the data quality of the data output from the data reduction process.

Author details

Qutub Ali Bakhtiar and Kia Makki
Network Lab, Technological University of America, Coconut Creek, FL USA

Niki Pissinou
Information Technology and Telecommunications Institute, Florida International University, Miami, FL USA

Appendix: Derivation

The tradition LMS algorithm [12] can be stated as follows.

$$w(k+1) = w(k) + \mu e(k)x(k) \tag{A.1}$$

where μ is the step size parameter which can have values ranging from [12]

$$0 < \mu < \frac{2}{NS_{max}} \tag{A.2}$$

where S_{max} is the maximum value of the power spectral density of the tap inputs $x(k)$ and N is the filter length. Assume that the value of the step size parameter be given by

$$\mu = \frac{1}{N(\mathbf{x}(k) * \mathbf{x}^T(k))} \tag{A.3}$$

Let

$$\mathbf{x}(k) = \gamma \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \tag{A.4}$$

The assumption of $\mathbf{x}(k)$ is valid in the case of prediction as data changes minimally. Substituting the value of and solving the equation

$$\mathbf{w}(k+1) = \mathbf{w}(k)\frac{e(k)}{N \sum \mathbf{x}(k)} \tag{A.5}$$

The above equation is devoid of the step size parameter μ which controls the speed and optimality of convergence of the algorithm.

7. References

[1] Akyildiz, I. F., Weilian, S., Sankarasubramaniam, Y. & Cayirci, E. [2002]. A survey on sensor networks, *IEEE Communication Magazine* 12: 102–114.

[2] Bakhtiar, Q. A., Pissinou, N. & Makki, K. [2008a]. Approximate replication of data using adaptive filters in wireless sensor networks, *International Symposium on Wireless Pervasive Computing*, Santorini, Greece.

[3] Bakhtiar, Q. A., Pissinou, N. & Makki, K. [2008b]. Estimated replication of data in wireless sensor networks, *International Conference on Communication Systems and Middleware (Comsware 08), Bangalore, India* .

[4] Bingham, E. & Mannila, H. [2001]. Random projection in dimensionality reduction: applications to image and text data, *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, New York, NY, USA, pp. 245–250.

[5] Burrus, C. S., Gopinath, R. A. & Guo, H. [1998]. *Introduction to Wavelets and Wavelet Transforms: A Primer*, Prentice hall.

[6] Businger, P. A. & Golub, G. H. [1969]. Algorithm 358: singular value decomposition of a complex matrix [f1, 4, 5], *Commun. ACM* 12(10): 564–565.

[7] Chu, D., Deshpande, A., Hellerstein, J. M. & Hong, W. [2006]. Approximate data collection in sensor networks using probabilistic models, p. 48.

[8] Data, I. L. [n.d.]. *http://db.lcs.mit.edu/labdata/labdata.html*.

[9] Deshpande, A., Guestrin, J. H. C., Madden, S. & Hong, W. [2004]. Model-driven data acquisition in sensor networks., *30th VLDBŠ04, Toronto, Canada* .

[10] Elnahrawy, E. & Nath, B. [2003]. Cleaning and querying noisy sensors, *ACM International Workshop in Wireless Sensor Networks WSNAŠ03* .

[11] Goel, S. & Imielinski., T. [1994.]. Prediction based monitoring in sensor networks taking lessons from mpeg, *ACM Computer Communication Review* pp. 510–514,.

[12] Haykin, S. [2002]. *Adaptive Filter Theory*, 4th edn, Prentice Hall, Englewood Cliffs, NJ.

[13] Iyengar, S. & Brooks, R. (eds) [2005]. *Distributed Sensor Networks*, Chapman and Hall.

[14] Jain, A., Chang, E. & Wang, Y. [2004]. Adaptive stream resource management using kalman filters., *ACM SIGMOD/PODS Conference (SIGMOD Š04), Paris, France* .

[15] Jolliffe, I. [2002]. *Principal Component analysis*, Springer Berlin Heidelberg.

[16] Min, R., Bhardwaj, M., hwan Cho, S. & Sinha, E. S. A. [2001]. Low-power wireless sensor networks, *In VLSI Design*, pp. 205–210.

[17] Olston, C. [2003]. *Approximate Replication*, PhD thesis, Stanford University.

[18] Olston, C., Jiang, J. & Widom, J. [2003]. Adaptive filters for continuous queries over distributed data streams, *ACM SIGMOD, Intl Conference on Management of data (SIGMOD 03))* pp. 563–574.

[19] Olston, C., Loo, B. T. & Widom, J. [2001]. Adaptive precision setting for cached approximate values, *Proceedings of SIGMOD Conference* .

[20] Olston, C. & Widom, J. [2005]. Efficient monitoring and querying of distributed, dynamic data via approximate replication, *IEEE Data Engineering Bulletin* .

[21] Proakis, J. [2006]. *Digital Signal Processing*, Prentice Hall.

[22] Santini, S. [2006]. Poster abstract: Towards adaptive wireless sensor networks, *3rd European Workshop on Wireless Sensor Networks (EWSN 2006)* pp. 13–15.

[23] Santini, S. & Romer, K. [2006]. An adaptive strategy for quality-based data reduction in wireless sensor networks, *Proceedings of 3rd International Conference on Networked Sensing Systems(INSS 2006),Chicago* .

[24] Tulone, D. & Madden, S. [2006a]. An energy-efficient querying framework in sensor networks for detecting node similarities, *MSWiM '06: Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, ACM, New York, NY, USA, pp. 191–300.

[25] Tulone, D. & Madden, S. [2006b]. Paq: Time series forecasting for approximate query answering in sensor networks, *European Workshop on Wireless Sensor Networks EWSN* pp. 21–37.

[26] G. Patterson & M. Mehmet-Ali [2011] "Modeling of data reduction in wireless sensor networks," *Wireless Sensor Network*, vol. 3, no. 8, pp. 283–294, 2011.

[27] Rooshenas,A. , Rabiee,H.R. , Movaghar,A., & M.Y. Naderi,[2010] "Reducing the data transmission in wireless sensor networks using the principal component analysis," *In Proceeding of the Sixth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP'10)*, (Brisbane, Australia), IEEExplore, 2010.

[28] E.Msechu & G.Giannakis, [2012 "Sensor-centric data reduction for estimation with wsns via censoring and quantization," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 400–414, 2012.