Understanding Linear Regression for Wireless Sensor Network Time Synchronization

Liangping Ma, Hua Zhu, Gayathri Nallamothu, Bo Ryu, Heidi R. Howard

***Abstract*— Linear regression is used by a number of time synchronization protocols to predict other nodes' clocks as the basis of long-term time synchronization for energy constrained wireless sensor networks. A common understanding is that the more frequently the time synchronization messages are exchanged, the more precise the time synchronization will be, because the samples used by the linear regression are expected to be more temporally correlated. Based on rigorous theoretical analyses and extensive simulation and experimental results, however, we show that the time synchronization frequency does not necessarily have a monotonic impact on synchronization precision. Rather, under certain conditions, the synchronization precision achieved by linear regression appears to be invariant of the synchronization frequency. We explain this in terms of the complex relationship between clock sampling intervals (inverse of the synchronization frequency) and the linear regression model. Consequently, a low synchronization frequency may achieve the same synchronization precision as a high synchronization frequency does, while being much more energy efficient.**

***Index Terms*—Time synchronization, linear regression, wireless sensor networks, synchronization frequency, synchronization precision.**

I. INTRODUCTION

TIME synchronization is critically important to wireless sensor networks. For one thing, a global time reference is necessary for a sensor network to track moving targets. For another, to conserve energy in energy-constrained sensor networks, duty-cycling (i.e., the radios are not always on, but rather being on and off in an alternating fashion) is used, and time synchronization is important for sensor nodes to coordinate wireless communications among themselves. Energy efficient and robust time synchronization can be achieved through the use of time synchronization protocols. In contrast, the two alternative approaches, namely, GPS and atomic clocks, are less desirable. The GPS signal is not always available in a deployed wireless sensor network due to shadowing and jamming. Atomic clocks, although extremely stable and accurate, consumes too much energy, and thus are not suitable for typically energy-constrained wireless sensor networks.

There are two types of time synchronization approaches: instant synchronization, and predictive synchronization. In the instant synchronization approach, a node tries to correct its clock reading by adopting that of its neighbor. If two clocks have the same oscillation frequency, a single instant synchronization point will suffice. However, clocks run at different oscillation frequencies, and the difference accumulates over time. To bound the accumulative time difference, we need frequent resynchronization, especially for time stringent applications. Instant synchronization protocols include RBS [3], TPSN [4], and time diffusion [5]. In the predictive synchronization approach, one node tries to establish a relation between its local clock and a target clock. A simple model for this relation is a linear model, which can be optimized by linear regression [1]. The predictive synchronization approach makes it possible to greatly reduce the number of resynchronization points, even for clocks exhibiting dramatic differences in oscillation frequency. Predictive synchronization protocols include FTSP [7], and RATS [8].

Linear regression [1] serves as the basis of such predictive time synchronization protocols as FTSP [7] and RATS [8], and yet its performance has not been closely examined in the literature. In this paper, we conduct a rigorous theoretical analysis of linear regression, and in particular, investigate how the synchronization precision is affected by the synchronization frequency. In addition, we carry out extensive QualNet simulations and hardware experiments, which corroborate our theoretical analysis.

The remainder of this paper is organized as follows. Section II describes the clock model and the justification of using linear regression for time synchronization.

This work is supported in part by Army Corps of Engineer's SBIR program under contract W9132T-06-C-0020. Liangping Ma, Hua Zhu, Gayathri Nallamothu, Bo Ryu are with San Diego Research Center, Inc., 6696 Mesa Ridge Rd., San Diego, CA 92121. (e-mail: lma@sdrcinc.net, hua.zhu@sdrcinc.net, gayathri.nallamothu@sdrcinc.net, bo.ryu@sdrcinc.net).

Heidi R. Howard is with ERDC-CERL, Army Corps of Engineers, 2902 Farber Drive, Champaign, IL 61822 (e-mail: h-howard@cecer.army.mil).

Section III provides the theoretical analysis of linear regression. Section IV presents simulation and experimental results, and Section V concludes this paper.

II. Clock Model

A. Clock Model for Individual Clocks

A quartz clock keeps time by counting the cycles of a sinusoidal voltage signal. The clock reading (not time deviation from "true time") of a quartz clock can be described by the following model as a function of "true time" t [3]:

$$x(t) = x_0 + (y_0 + 1)t + 1/2Dt^2 + \xi(t) \qquad (1)$$

where x_0 is the time offset, y_0 is the frequency offset, D is the frequency drift, and $\xi(t)$ is the random noise. The quadratic term *D,* while it is important to precision applications, can be neglected, since the error is dominated by the other terms [2].

Assuming *D=0,* if two clocks have the same frequency offset, the time difference between them will remain constant except a random variable. Otherwise, the time difference will grow approximately linearly over time.

B. Relation between Different Clocks

Let the clock readings of two clocks be $X(t)$ and $Y(t)$, respectively. We assume that the relation between the two variables can be expressed by a differentiable function $Y=f(X)$. By Taylor expansion, there exists a neighborhood of X_0 such that within this neighborhood $Y \approx f(X_0) + f'(X_0)(X - X_0)$. This justifies a linear model for the relation between two different clocks:

$$Y = A + BX \qquad (2)$$

where parameters A and B may be functions of time. In general, X is a random variable. Suppose X is Gaussian distributed, then it can de decomposed as $X=X_d+X_r$, where X_d is deterministic, and X_r is zero mean Gaussian. Thus, (2) can be written as

$$Y = A + BX_r + \varepsilon \qquad (3)$$

where the Gaussian distributed ε accounts for randomness, $\varepsilon \sim N(0,\sigma^2)$. For presentation convenience, we drop the subscript r of X_r in (3), i.e.,

$$Y = A + BX + \varepsilon \qquad (4)$$

C. Linear Regression

In linear regression, one tries to obtain the maximum likelihood estimates of A and B based on a set of data points (X_i,Y_i), $i=1,2,\ldots n$. A data point is obtained by letting two sensor nodes time stamp an event common to both nodes. For example, node 1 time stamps an outgoing message, and upon receiving the message, node 2 generates another time stamp. These two time stamps are meant to the same event, the transmission of this message, after offsetting the propagation delay and transmission delay.

By linear regression theory [p.479, 1], the predicted Y, (denoted as Y^*) that corresponds to a future X^* is given by

$$Y^* = \hat{A} + \hat{B}X^* \qquad (5)$$

where

$$\hat{B} = \frac{n\sum X_iY_i - \sum X_i \sum Y_i}{n\sum X_i^2 - (\sum X_i)^2} \qquad (6)$$

$$\hat{A} = \frac{\sum Y_i}{n} - \hat{B}\frac{\sum X_i}{n} \qquad (7)$$

and the true Y (corresponding to X^*) will fall within the following interval

$$(Y^* - w, Y^* + w) \qquad (8)$$

with probability 100(1-α)%, where

$$w = t_{\alpha/2,n-2}\sqrt{\frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{n-2}\left(1 + \frac{1}{n} + \frac{(X^* - \sum X_i/n)^2}{\sum_{i=1}^{n}(X_i - \sum X_i/n)^2}\right)} \qquad (9)$$

and

$$\hat{Y}_i = \hat{A} + \hat{B}X_i\ . \qquad (10)$$

The number $t_{\alpha/2,n-2}$ defines the interval around the mean of a Student distribution (with freedom n-2) such that the area over this interval is equal to (1-α). For α=0.05, i.e., 95% confidence interval, this number is equal to 4.303.

III. Theoretical Analysis

In this section, we study how the synch frequency (at which the time stamp data pairs are generated) affects the time synchronization precision. Consider node 1 and node 2. Node 1's clock reading is $X(t)$, and node 2's is $Y(t)$. The synchronization problem is formulated as follows: node 2 predicts its own clock reading that corresponds to a future clock reading of node 1. The future clock reading of interest could be the next wake up time of node 1.

Proposition 1: Under linear clock model in (4), if A and B are both time invariant and linear regression is used, then the synchronization frequency does not affect the synchronization precision.

Proof: The confidence interval w in (9) is a measure of the synchronization error. More precisely, it measures the uncertainty of the current prediction of a future clock reading. We show that $E[w^2]$ is invariant at different synchronization frequencies.

Assuming that the values of the random term ε in (4) at different time instants are independent, we define $e_i = Y_i - \hat{Y}_i$, and evaluate $E[\sum_{i=1}^{n} e_i^2]$.

$$
\begin{aligned}
&E[\sum e_i^2] \\
&= E[\sum (Y_i - \hat{Y}_i)^2] \\
&= \sum E[((A + BX_i + \varepsilon_i) - (\hat{A} + \hat{B}X_i))^2] \\
&= \sum E[((A - \hat{A}) + (B - \hat{B})X_i + \varepsilon_i)^2] \\
&= \sum [\mathrm{var}(\hat{A}) + \mathrm{var}(\hat{B})X_i^2 + \mathrm{var}(\varepsilon_i)] \\
&= n\,\mathrm{var}(\hat{A}) + \mathrm{var}(\hat{B})\sum X_i^2 + n\,\mathrm{var}(\varepsilon_i) \\
&= n\frac{\sigma^2 \sum X_i^2}{n\sum_{i=1}^{n}(X_i - \sum_{j=1}^{n} X_j / n)^2} + \frac{\sigma^2}{\sum_{i=1}^{n}(X_i - \sum_{j=1}^{n} X_j / n)^2}\sum X_i^2 + n\sigma^2 \\
&= \frac{2\sigma^2 \sum X_i^2}{\sum_{i=1}^{n}(X_i - \sum_{j=1}^{n} X_j / n)^2} + n\sigma^2
\end{aligned}
$$

which is invariant when we change $(X_i - X_{i-1})$, i=2,…,n, from d to γd. Note that for a fixed synch frequency, $(X^* - \sum X_i / n)^2 / \sum_{i=1}^{n}(X_i - \sum X_i / n)^2$ depends only on the window size n. Therefore, by (9), $E[w^2]$ is invariant when $(X_i - X_{i-1})$ is changed from d to γd.

Note that the net effect of applying different synch frequencies is that the difference between adjacent time stamps, X_i - X_{i-1}, is different. Thus, $E[w^2]$ is invariant of synchronization frequencies. **Q.E.D.**

Note: (1) Although we assume that the noise sequence ε_i are independent, the actual input sequence to the linear predictor X_i are still highly correlated due to non-stationarity, and so are Y_i. **(2)** If parameters A and B in the clock model (4) change quickly over time, then higher synch frequency may improve the performance of linear regression, since the data in the time stamp window is fresher and more relevant.

We define the *prediction interval* as the time interval between the time when the prediction is made and the time when the predicted event occurs. That is, the longer the prediction interval, the further in the future the linear regression predictor predicts. The following proposition quantifies the impact of the prediction interval on the synchronization precision.

Proposition 2: Under the linear clock model in (4), the time synchronization precision w depends on the ratio of the prediction interval p and the time synchronization interval d. If p>>d, the synch error increases almost linearly with p.

Proof: The X^* in (9) can be written as

$$X^* = X_n + p \tag{11}$$

Since $X_i = X_1 + (i-1)d$ (12)

we have

$$\sum_{i=1}^{n} \frac{X_i}{n} = X_1 + \frac{(n-1)}{2}d \tag{13}$$

Therefore

$$\frac{(X^* - \sum X_i / n)^2}{\sum_{i=1}^{n}(X_i - \sum X_i / n)^2} = \frac{3(n - 1 + 2p/d)^2}{(n-1)n(n+1)} \tag{14}$$

which depends only on p/d, and thus w in (9) depends only on p/d. It is clear that if p>>d, p becomes the only dominant term and thus from (9), w increases almost linearly with p. **Q.E.D.**

Note: The above proposition clearly states that the synchronization precision is directly affected by how far the linear regression predictor predicts.

IV. Simulation and Experimental Results

We implemented linear regression for time synchronization in a Mica2 microsensor network, where the Mica2 [9] nodes run Sensor Operating System (SOS) [10]. The Mica2 nodes use CC1000 low power 916MHz radios, and use the 8-bit, 7.37-MHz Atmega 128L microprocessor as the CPU, has 128KB program memory, and a maximum data rate of 38.4kbps.

Due to the lack of support for double float computation on the Atmega processor, we resort to the QualNet network simulator for more testing of linear regression.

A. Experimental Results

The experiment description:

- The experimental set up is shown in Figure 1. There is a laptop, a 4-port USB serial adaptor, and two Mica2 sensor nodes. The 4-port USB Serial adapter acts as a wired communication hub for the laptop (using a USB port) and the 2 Mica2 sensor nodes (each using a serial port).
- The two Mica2 node exchange time synch messages.
- Each node can print information in a Cygwin window on the laptop. The information includes: neighbor list, values for the parameters in the linear regression process, its own time, and the prediction of its neighbors' time.
- A time synch message is sent every 30 seconds.

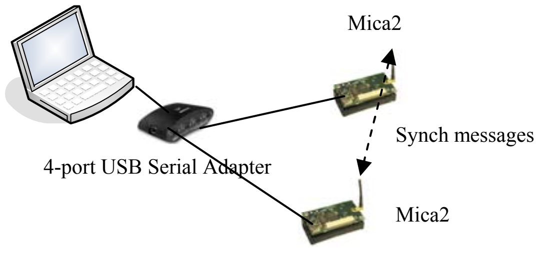

Figure 1 Experiment set up

Figure 2 shows the experimental results. The blue line represents the actual prediction error, and the red line represents the measure of synchronization precision, i.e., the 95% confidence interval w in (9).

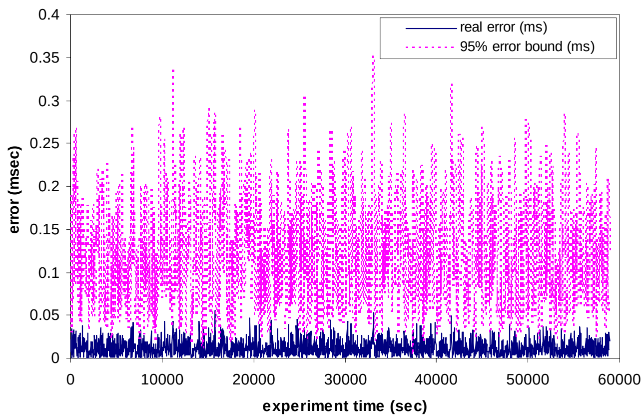

Figure 2 Mica2 clock synchronization errors

B. Simulation Results

We developed an innovative approach to enabling clock modeling in any discrete-event simulator, including QualNet. Among many options, the clock model defined in (1) has been used in the following simulation. An appropriate set of parameters that matches the clock characterization of Mica2 nodes has been identified.

In the QualNet simulation, we test four different synchronization rates, i.e. one time synch message per 120, 240, 480, 960 seconds, respectively. The actual prediction error and the 95% confidence interval for each case are shown in Figure 3 to 6.

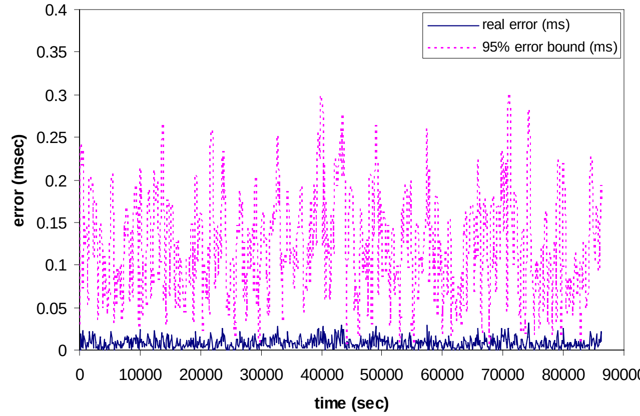

Figure 3 Sync message interval = 120 sec

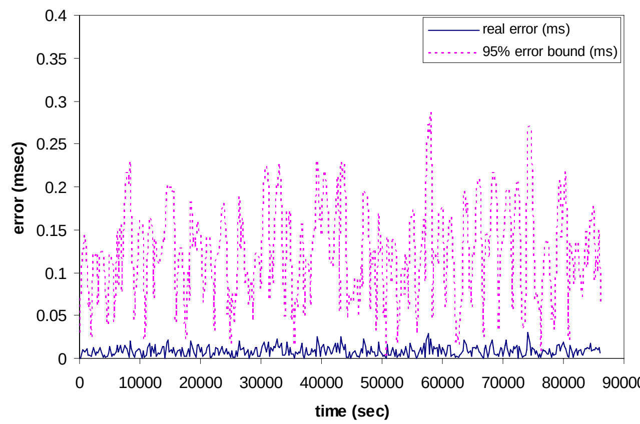

Figure 4 Sync message interval = 240 sec

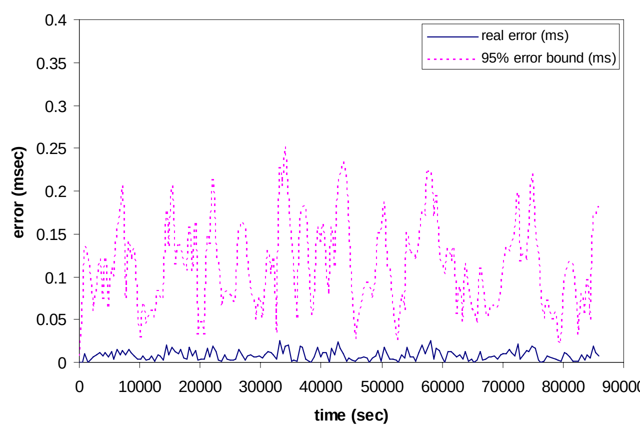

Figure 5 Sync message interval = 480 sec

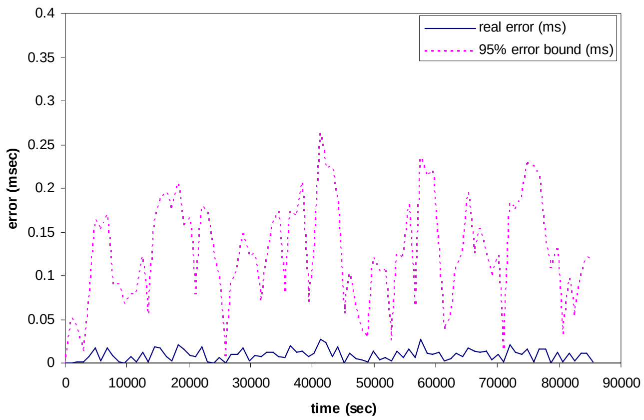

Figure 6 Sync message interval = 960 sec

The time-averages of all four cases are shown in Figure 7. Slightly counter-intuitive, the synchronization error does not monotonically increase along with the synchronization message interval. Instead, Figure 7 clear shows that the performance of clock synchronization essentially does not vary much within such range of the time synchronization rate. This is because that the relationship between two clock sources are close to linear in short to mid term. Such time-invariant clock synchronization performance has also been observed in Mica2 experiment. We plan to collect more data for extremely slow synchronization rate (i.e., > 960 seconds per message). It is possible that the relationship between two clock sources in long term may be significantly different from a time invariant linear approximation.

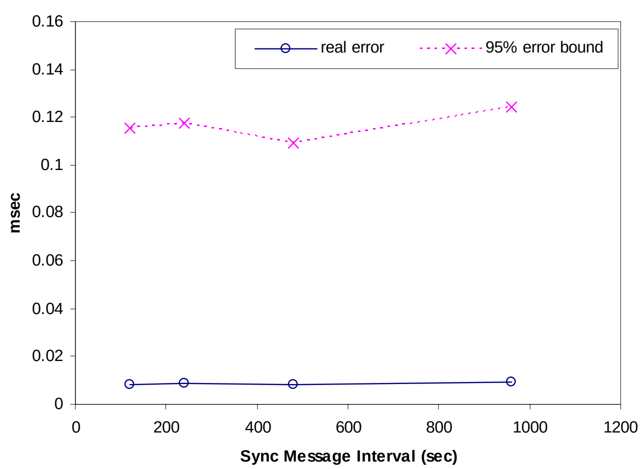

Figure 7 Sync performance v.s. sync message interval

V. Conclusion

In this paper, we thoroughly investigate the linear regression time synchronization technique for sensor networks. Based on a rigorous theoretical analysis and extensive simulation and experimental results, we show that how often the time synchronization messages are exchanged does not necessarily have a monotonic impact on the time synchronization precision.

Acknowledgment

We would like to thank Professor Mani Srivastava of UCLA and his research group, including Ilias Tsigkogiannis, David Jea, and Saurabh Ganeriwal, for their invaluable support.

References

[1] R. Larsen, M. Marx, *An introduction to Mathematical Statistics and Its Applications*, 2nd edition, Prentice Hall, 1986.
[2] D. Mills, Executive Summary: Computer Network Time Synchronization, Available at http://www.eecis.udel.edu/~mills/exec.html
[3] J. Elson, L. Girod, D. Estrin, "Fine-grained network time synchronization using reference broadcasts", Proceedings of the 5th symposium on Operating systems Design and Implementation (OSDI), 2002.
[4] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync Protocol for Sensor Networks", SenSys'03, November 5-7, 2003, Los Angeles, California, USA.
[5] Q. Li and D. Rus, "Global Clock Synchronization in Sensor Networks", IEEE INFOCOM 2004.
[6] D.W. Allan, "Time and Frequency (Time-Domain) Characterization, Estimation, and Prediction of Precision Clocks and Oscillators", IEEE Trans. Ultrasonics, Ferroelectrics, and Frequency Control, pp. 647-654, Nov. 1987.
[7] M. Maróti, B. Kusy, G. Simon and Á. Lédeczi, "The Flooding Time Synchronization Protocol", ACM SenSys, 2004.
[8] S. Ganeriwal, D. Ganesan, H. Sim, V. Tsiatsis, M. B. Srivastava, "Estimating Clock Uncertainty for Efficient Duty-Cycling in Sensor Networks", SenSys'05, November 2-4, 2005, San Diego, California, USA.
[9] Product Features: MICA2 433, 868, 916 MHz, Crossbow, available at http://www.xbow.com/Products/productdetails.aspx?sid=174
[10] Sensor Operating System, UCLA, available at http://nesl.ee.ucla.edu/projects/sos/
[11] QualNet, available at http://www.scalable-networks.com/