

Application Layer Time Synchronization Utilizing Symbol Timing Recovery in Wireless Sensor Networks

S. M. Usman Hashmi
Electronics Engineering,
Iqra University,
Islamabad, Pakistan.

Ammar Ajmal, Arsalan Akhter,
Shehzad Khalid, Waleed Manzoor
Computer Engineering,
Bahria University,
Islamabad, Pakistan

Abstract—Information sharing, scheduling and organization in a Wireless Sensor Network (WSN) is highly dependent upon achieving the same notion of time. Hence timing synchronization becomes a vital concern in most of the distributed wireless networks. Similarly, different layers of a network also need to synchronize among themselves for efficient performance and energy optimization. Bandwidth, energy consumption and storage capacity along with the high density of nodes are some of the practices limitations present in the WSNs. Symbol timing recovery (physical layer time synchronization) is used in this work for time synchronization at the application layer to achieve higher energy conservation.

Keywords - time synchronization, wireless sensor network, clock correction, symbol timing recovery.

I. INTRODUCTION

Recent advances in the field of Micro-Electro-Mechanical systems (MEMS) have directed the active research towards designing highly distributed, low-power, low-cost, tiny unattended sensing devices capable of sensing, processing and communicating data over wireless networks [7] [8] [9] [10] [11] [4] [5]. A Wireless Sensor Network (WSN) consists of a distributed network of nodes to sense, collaborate and route data in a certain environment [12] [13]. The main component of a WSN is a base station which behaves as a gateway to collect the data from sensors, and in turn relays it to a back-end server for further processing via a wired backbone. WSNs are employed in a wide range of diverse applications like environmental and battlefield monitoring, robotics, tracking and object localization [14] [15] [6]. Like other distributed networks synchronization is a major concern in WSNs. Devices have to be perfectly time synchronized with each other through clocks which are electronic counters to count the oscillations accurately at a particular frequency [16] [17] [18]. These clocks may have a phase offset due to the difference in values and a frequency offset due to the difference in their speeds. Clock timing synchronization is of utmost importance and is required at different layers while designing a WSN, which implicates both the phase and frequency synchronization [19]. This paper deals with the clock synchronization at the physical and application layers, with the help of the hardware clock to achieve cross-layer synchronization. At the physical layer, it includes clock

signal estimation aligned both in phase and frequency with the transmission node clock [19] [20]. At the application layer, time synchronization is crucial for the network in removing timing ambiguity which is also known as clock synchronization.

II. RELATED WORK

Time Synchronization is an essential component of WSNs as it is a basic requirement for any coordination in the network. Different WSNs have different clock precision and accuracy requirements. A few important parameters which are considered for time synchronization are:

Precision: refers to the maximum error that a clock can have with respect to the master clock in a network.

Efficiency: corresponds to the time, energy and computation complexity required to perform the synchronization. Energy efficiency is a key factor in enhancing life time of a node in a WSN. So, a synchronization protocol with a very few computations results in higher energy efficiencies.

Lifetime: refers to the maximum time it requires to resynchronize the nodes present in a WSN

Cost: plays a major role since there are large number of devices in a WSN. Time Synchronization helps to reduce the overall cost of the WSN.

For time synchronization at application layer, many protocols are used for wired as well as wireless sensor networks. One of the most widely used protocol for time synchronization in wired networks is Network Time Protocol (NTP) [21]. In NTP, a hierarchical tree of time servers is created. Primary server synchronizes with the global clock. This timing information is then distributed to the secondary servers and later to the clients by the primary server. In contrast to NTP, another protocol called Precision Time Protocol (PTP) distribute the clock information among the nodes of a wired network using a master-slave architecture. These architectures, however, cannot be applied directly to wireless sensor networks due to the different requirements in WSNs such as energy efficiency, end to end latency, infrastructure and reliability [16] etc.

Using receiver-to-receiver synchronization, Reference Broadcast Synchronization (RBS) attempts to decrease non-deterministic latency among the nodes of a WSN to achieve

time synchronization [16] [22] [23]. Depending only on the packet receive time, each wireless node sends reference broadcast beacons to its neighbors using physical layer broadcasts and compute the non-determinism of packet send time, access time and propagation time. This reference broadcast packet can then be used to synchronize a set of receivers with one another [22]. All the local clock values of each node in the network are maintained in a table. RBS relates the local clocks of node with each other using this table and lets the clocks run without correction.

To achieve clock synchronization, Romer's protocol [24] uses an innovative time transformation algorithm. Maximum time difference between two communicating nodes is estimated by the lower bound and the upper bound round trip time of a message. This time difference is then used by Romer's protocol to synchronize the nodes. Like RBS[22] Romer's protocol also allows the clocks to run independent of each other.

Mock et al. [25] introduced a protocol that spreads the correction of clock over a finite interval and attempts to achieve continuous clock synchronization. Clock rate is gradually speed up or slowed down depending on the required clock time correction [16]. This technique provides fault tolerance as well as minimal message complexity [16]. Furthermore, it avoids unpredictable instantaneous corrections of clock values [25].

Network-wide Time Synchronization, Delay Measurement Time Synchronization, Probabilistic Clock Synchronization and Time-Diffusion Protocol (TDP) are some of the other protocols discussed by [16], [22] and [26]. Network-wide Time Synchronization is capable of holding large node density. Delay Measurement Time Synchronization proves to be an energy-efficient protocol but does not provide a high level of accuracy as compared to RBS [23] protocol. Probabilistic Clock Synchronization provides probabilistic bounds on the accuracy of clock synchronization scheme used in RBS. Time-Diffusion Protocol (TDP) involves all the nodes in the synchronization process by diffusing messages throughout the WSN. All these protocols [21] [23] [24] [25] [26] [3] [2] [1] have their own merits and demerits and hence can be used for application layer clock synchronization in any wireless distributed network (e.g. Wireless Sensor Network) [16] [22] depending on the environment and requirements at hand.

Time synchronization in the physical layer is also extensively discussed in the literature. Symbol timing recovery has been discussed by Ghaffer et al.[19] [20] Timing synchronization can also be done by extracting phase and frequency of the transmitter's timing clock at the receiver end. [20] [27] [28]. Another technique is to apply feed-forward or feedback systems [27]. Feedback systems can operate at rates equal to or higher than the symbol rates by adjusting the synchronization parameters in a recursive manner using error signals [28]. Interpolation controllers, loop filters, timing er-

ror detectors and different types of interpolations are used by error feedback systems to achieve multiple synchronization schemes. Interpolants from given number of symbols can be computed using multiple techniques such as polynomial interpolation and polyphase filterbank interpolation. Linear, quadratic as well as cubic interpolations can also be computed using Farrow structures. Farrow Structures are less computationally intensive and hence are more feasible from a practical implementation point of view [20].

Different types of timing error detectors are also discussed by [20] and [27]. Slopes of the symbols alongwith sign correction is employed by Maximum likelihood timing error detectors (ML-TED) to get the error signal. The slope required by ML-TED is computed by Early-Late timing error detector (EL-TED) using the time differences between the symbols. Zero crossing timing error detector (ZC-TED) uses the eye-diagram to find the zero crossing, but it only operates at two samples per symbol [20]. Similar methods are used by Gardner timing error detector (G-TED) but G-TED uses different modulation schemes. Phase and frequency errors can be tracked using Loop Filters, which act as phase lock loops (PLLs) and their parameters can be controlled to adjust the tracking performance and acquisition times. [20] also discusses some interpolation controls that can be used with the feedback synchronization systems. In the situations where interpolants are required after every fixed number of samples, Modulo-1 counter interpolation controls can be used. Recursive interpolation control can also be employed for symbol synchronization using feedback. Harris et al [29] performed the interpolations required for symbol timing synchronization in a sampled-data receiver using polyphase filterbank.

III. SYNCHRONIZATION METHODOLOGY

An Application layer clock is usually defined by two registers which keep on counting the oscillations of crystals as well as how many oscillations of the crystal are equal to one clock tick in application layer [16]. The same quartz crystal is also used to maintain the clock at the physical layer, which then is used for symbol timing synchronization. Therefore both the clocks are derived from the oscillation of the same quartz crystal, as the devices with one hardware oscillator implement all the clocks within that device. Mathematically, the clock is defined as:

$$C(t) = k \int_{t_0}^t \omega(t)dt + C(t_0), \quad (1)$$

Where t denotes real-time, k is the constant of proportionality and $\omega(t)$ denotes angular frequency of the oscillator [24]. This fact is exploited to perform the synchronization between the physical layer and the application layer.

[20] and [28] discuss in detail time synchronization at the physical layer. Multiple symbol timing synchronization systems with different modulation schemes, loop filters,

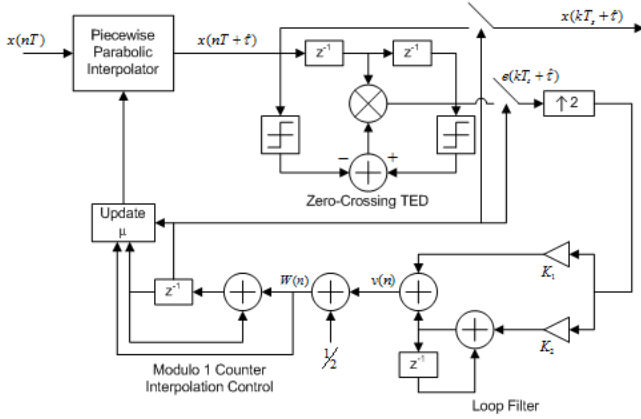


Fig. 1. Symbol timing synchronization system for physical layer

interpolators, timing error detectors, interpolation control are discussed in [20]. An example symbol timing synchronization system [20] is shown in Fig. 1

This synchronization system is based on zero-crossing timing error detector (ZC-TED) and is used for binary pulse amplitude modulation (PAM). This system uses a proportional plus integrator loop filter alongwith a piecewise parabolic interpolator and tracks and compensates for the frequency and phase error of the clock.

The piecewise parabolic interpolate receives the output of the matched filter. Fractional change and basepoint index are computed by Modulo-1 counter interpolation control, which is then fed to the interpolator. The output of the interpolator is then fed to a ZC-TED operating at 2 samples/symbol. The interpolator computes the zero crossing in the eye-diagram [20] and outputs the error to be equal to zero when perfectly aligned. The next component is the loop filter whose output is given to Modulo-1 interpolation control to compute the estimates for basepoint index and fractional change. Using the above iterative synchronization system, the variations of fractional change μ can be used to compute the actual frequency offset. Different methods for computation of frequency offset are illustrated by [20] [27] [28]. In [20] the frequency offset is computed using the slope of the fractional change and new sampling rate can be computed as given by the equation below to achieve time synchronization at the physical layer.

$$f_s = (2 + m) f_d, \quad (2)$$

Where f_s is new sampling rate, f_d is symbol rate and m is the slope of the fractional change.

Time synchronization at application layer can simply be modeled as exchange of time stamps between nodes, estimation of offsets and correction of clock phase and frequency. Least square (LS) estimate can be applied to the time stamps exchanged and used to compute the frequency offset at the application layer. In LS an equation of straight line needs to

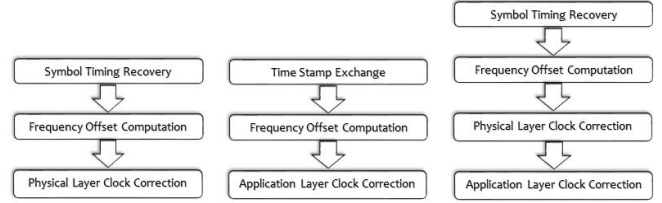


Fig. 2. (a) Physical Layer Time Synchronization, (b) Application Layer Time Synchronization, (c) Cross Layer Time Synchronization

be computed using the time stamps as data points and error is minimized between data points and straight line given by

$$e = \sum_{i=1}^N (y_i - \hat{y}_i)^2, \quad (3)$$

Where y_i are data points and \hat{y}_i are points of computed straight line given by $y = mx + c$ and to find m and c the least square estimate of a straight line is

$$\begin{bmatrix} m \\ c \end{bmatrix} = (X^T X)^{-1} X^T Y, \quad (4)$$

The cross layer approach can be illustrated as shown in Fig. 2 which clearly state that physical layer time frequency offset can be applied for the physical layer clock correction as well as application layer clock correction as it always comes out to be same, which is also shown in results section.

IV. SIMULATION RESULTS

The simulation consists of two synchronization systems, one at the physical layer and one at the application layer. The physical layer clock simulation is done using Fig. 1. Binary PAM is used as the modulation scheme at the transmitter side with symbols +1 and -1. 3000 random symbols are generated with a symbol rate of 1000 symbols/sec. The sampling rate is set to be 8000 samples/sec. The samples per symbol are computed to be 8. The binary PAM symbols are up-sampled by 8 and pulse shaped using a square root raise cosine with 50 percent excess bandwidth and transmitted over the perfect channel. Various frequency offsets are simulated between two nodes. At the receiver side, the data after matched filter is down-sampled to 2 samples/symbol and fed to symbol timing synchronization system using ZCTED, piecewise parabolic interpolator, proportional plus integrator loop filter and modulo 1 counter interpolator control given by Fig. 1.

Simulation I: It is assumed that a -ve frequency offset of -4.9751×10^{-3} is present between the sampling clocks of transmitter and receiver. The simulation results are shown in Fig. 3 which are in term of fractional interval and timing error. The simulation result shows the plot of fractional interval μ and now the frequency offset can be computed as in [20] that is equal to the slope $-1/101 = -9.90099 \times 10^{-3}$

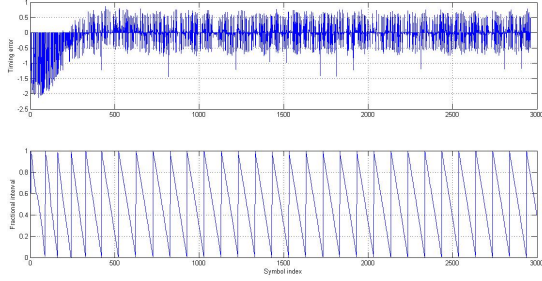


Fig. 3. Timing error and fractional change over 3000 symbols with frequency offset of -4.9751×10^{-3}

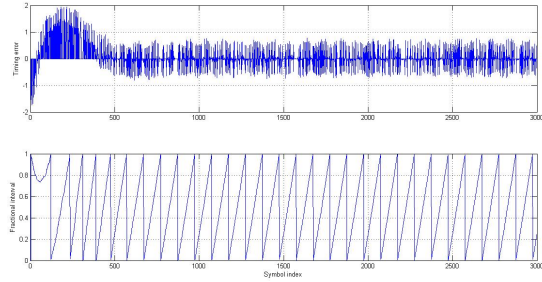


Fig. 4. Timing error and fractional change over 3000 symbols with frequency offset of 5.0000×10^{-3}

divided by the up-sampling factor which is 2 here and hence the frequency offset is computed to be -4.9505×10^{-3} .

Now same transmitter and receiver with this frequency offset are considered for the application layer clock. Application layer clock of the transmitter is assumed to be run for 500 seconds. The transmitter application layer clock starts from 1 and counts till 100000 within 500 seconds whereas the receiver application layer clock starts from 1 and counts till 100499 and they keep exchanging there clocks. Using LS estimate, slope of the estimated line is computed to be $((10000 - 1)/(100499 - 1))$ in presence of no noise and frequency offset is $1 - ((10000 - 1)/(100499 - 1)) = -4.9653 \times 10^{-3}$ which is the same as of physical layer frequency offset. Hence this simulation shows that the frequency offsets at the physical layer (-4.9505×10^{-3}) and application layer (-4.9653×10^{-3}) between clocks are related and very much closed to it as they both are based on same hardware oscillator within a node of network.

Simulation II: It is assumed that a positive frequency offset of 5.0000×10^{-3} is present between the sampling clocks of transmitter and receiver. The simulation results are shown in Fig. 4 which are in term of fractional interval and timing error. The simulation result shows that the frequency offset is computed to be 5.0000×10^{-3} . Considering the same scenario for the application layer time synchronization that is discussed in the Simulation I, the frequency offset is

$1 - ((10000 - 1)/(99502 - 1)) = 5.0050 \times 10^{-3}$ which is the same as of physical layer frequency offset. Hence this simulation shows that the frequency offsets at the physical layer (5.0000×10^{-3}) and application layer (5.0050×10^{-3}) between clocks are related and very much closed to it as they both are based on same hardware oscillator within a node of network.

The results presented in the above mentioned simulations can be used to extract the error which is computed as a difference between the offset of the physical layer and the application layer. Based on these simulation the offsets present between the nodes and the computed offsets at physical and application layers is shown in table. It must be noticed that the physical layer frequency offsets are approximately same as of application layer frequency offsets.

TABLE I
COMPARISON OF FREQUENCY OFFSETS

Simulation	1 (-ve offset)	2 (+ve offset)
Offset Between Nodes	-4.9751×10^{-3}	5.0000×10^{-3}
Computed Phy Layer Offset	-4.9505×10^{-3}	5.0000×10^{-3}
Computed App Layer Offset	-4.9653×10^{-3}	5.0050×10^{-3}

V. CONCLUSIONS

The simulation demonstrates that the heart of both the physical layer and application layer clock is hardware oscillator and that is the reason that frequency offset figured at the physical layer between two nodes is same as the frequency offset at the application layer and the application layer clock can be corrected straightforwardly utilizing physical layer clock frequency offset and henceforth there will be no compelling reason to utilize any message trade algorithm on the application layer to set the application layer clock frequency offsets and this will spare energy and valuable for energy constrained networks.

REFERENCES

- [1] Yildirim, K. Sinan, A. Kantarci, "Time Synchronization Based on Slow-Flooding in Wireless Sensor Networks," *Parallel and Distributed Systems, IEEE Transactions*, 2014, pp. 244-253.
- [2] Ardakani, S. Pourroostaei, J. Padget, M. D. Vos, "HRTS: A Hierarchical Reactive Time Synchronization Protocol for Wireless Sensor Networks," *Ad Hoc Networks. Springer International Publishing*, 2014, pp. 47-62.
- [3] Yildirim, K. Sinan, A. Kantarci, "External Gradient Time Synchronization in Wireless Sensor Networks," *Parallel and Distributed Systems, IEEE Transactions*, 2014, pp. 633-641.
- [4] H. Sundani, et al, "Wireless sensor network simulators a survey and comparisons," *International Journal of Computer Networks*, 2011, pp. 249-265.
- [5] K. Maraiya, K. Kamal, N. Gupta, "Wireless sensor network: a review on data aggregation," *International Journal of Scientific and Engineering Research*, 2011, pp. 1-6.
- [6] X. Hu, B. Wang, H. Ji, "A Wireless Sensor NetworkBased Structural Health Monitoring System for Highway Bridges," *Computer Aided Civil and Infrastructure Engineering*, 2013, pp. 193-209.
- [7] Shuang-Hua Yang, "Wireless Sensor Networks Principles, Design and Applications" ISBN: 978-1-4471-5504-1 (Print) 978-1-4471-5505-8 (Online) 2014.

- [8] J. Hill, M. Horton, R. Kling, L. Krishnamurthy, "The Platforms Enabling Wireless Sensor Networks" Commun. ACM 2004, p.6, pp.41-46.
- [9] J. Kahn, R. Katz, and K. Pister, "Next Century Challenges: Mobile Networking for Smart Dust." In *Proceedings of the fifth annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp.271278, 1999.
- [10] Ill-Keun Rhee, Jaehan Lee, Jangsub Kim, Erchin Serpedin, and Yik-Chung Wu "Clock Synchronization in Wireless Sensor Networks: An Overview" *Sensors 2009* 9, pp.56-85; doi:10.3390/s90100056 sensors ISSN 1424-8220
- [11] U. Buy, A. D. Kshemkalyani, B. Sundararaman, "Clock Synchronization for Wireless Sensor Networks: A Survey," *Ad Hoc Networks*, Vol. 3, No. 3, pp. 281-323, May 2005.
- [12] J. Elson and K. Rmer: "Wireless Sensor Networks: A New Regime for Time Synchronization", *Proceedings of the First Workshop on Hot Topics In Networks (HotNets-I)*, Princeton, New Jersey. October 28-29 2002.
- [13] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, K. Pister: "System Architecture Directions for Networked Sensors". *Proceedings of Ninth International Conference on Architectural Support for programming Languages and Operating Systems*, November 2000.
- [14] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless Sensor Networks for Habitat Monitoring", *ACM International Workshop on Wireless Sensor Networks and Applications* September 28, 2002, Atlanta, GA.
- [15] A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, and J. Zhao: "Habitat Monitoring: Application Driver for Wireless Communications Technology", *Proceedings of the First ACM SIGCOMM Workshop on Data Communications in Latin America*, 2001.
- [16] U. Buy, A. D. Kshemkalyani, B. Sundararaman, "Clock Synchronization for Wireless Sensor Networks: A Survey," *Ad Hoc Networks*, Vol. 3, No. 3, pp. 281-323, May 2005.
- [17] P. Eskelinen, S. Ono, M. Kihara, *Digital Clocks for Synchronization and Communications*, Artech House, 2003.
- [18] C. Veerarittiphan, M. L. Sichitiu, "Simple, accurate time synchronization for wireless sensor networks," In *Wireless Communications and Networking*, pp. 1266-1273, New Orleans, LA, USA, 2003.
- [19] H. S. Abdel-Ghaffar, "Analysis of synchronization algorithms with time-out control over networks with exponentially symmetric delays," *IEEE Transactions on Communications*, Vol. 50, No. 10, pp. 1652-1661, Oct 2002.
- [20] M. Rice, *Digital communication: A discreet time approach*, Prentice Hall, 2008.
- [21] I. Chakraborty, N. Lynch, R. Fan, "Clock Synchronization for Wireless Networks," *Principles of Distributed Systems*, Vol. 3544/2005, No. 900, 2005.
- [22] D. Rus, Q. Li, "Global Clock Synchronization in Sensor Networks," In *INFOCOM, Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, 2004.
- [23] L. Girod, D. Estrin, J. Elson, "Fine-grained network time synchronization using reference broadcasts," In *ACM SIGOPS Operating Systems Review - OSDI 02: Proceedings of the 5th symposium on Operating systems design and implementation*, New York, USA, 2002.
- [24] K. Romer, "Time synchronization in ad hoc networks," In *MobiHoc 01 Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking and computing*, New York, USA, 2001.
- [25] R. Frings, E. Nett, S. Trikalotis, M. Mock, "Continuous clock synchronization in wireless real-time applications," In *Reliable Distributed Systems, SRDS-2000, Proceedings on The 19th IEEE Symposium*, Nurnberg, pp. 125-132, Germany, 2000.
- [26] K. Romer, J. Elson, "Wireless sensor networks: a new regime for time synchronization," *ACM SIGCOMM Computer Communication Review*, Vol. 33, No. 1, Jan 2003.
- [27] A. N. D'Andrea, U. Mengali, *Synchronization Techniques for Digital Receivers (Applications of Communications Theory)*, 1st ed. Pisa, Italy: Springer, 1997.
- [28] M. Moeneclaey, S. A. Fechtel, H. Meyr, *Digital Communication Receivers: Synchronization, Channel Estimation and Signal Processing*, 2nd ed. USA: Wiley-Interscience, 1998.
- [29] M. Rice, F. J. Harris, "Multirate digital filters for symbol timing synchronization in software defined radios," *Selected Areas in Communications*, Vol. 19, No. 12, pp. 2346-2357, Dec 2001.
- [30] V. K. Bhargava, K. B. Letaief, H. Minn, "A robust timing and frequency synchronization for OFDM systems," *Wireless Communications, IEEE Transactions*, Vol. 2, No. 4, pp. 822-839, July 2003.
- [31] K. Liu, "Architectures for symbol timing synchronization," Brigham Young University, Thesis 2004.
- [32] Q. M. Chaudhari, E. Serpedin, B. W. Suter, N. Kyoung-Lae, "Novel Clock Phase Offset and Skew Estimation Using Two-Way Timing Message Exchanges for Wireless Sensor Networks," *Communications, IEEE Transactions*, Vol. 55, No. 4, pp. 766-777, Apr 2007.
- [33] J. Yang, K. Cheun, "Improved symbol timing synchronization for IEEE 802.11a/g wireless LAN systems in multipath channels," In *Consumer Electronics, ICCE 06, 2006 Digest of Technical Papers, International Conference*, South Korea, pp. 291-292, 2006
- [34] B. Abolhassani, R. Shahbazian, "A New symbol Timing Synchronization Scheme for Multi Band OFDM Ultra Wideband (MB-OFDM UWB) systems," In *International Conference on Computer Communication and Management*, Singapore, 2011.
- [35] S. Chakrabarti, R.V. R. Kumar, D. Sen, "An Energy Based Symbol Timing Synchronization Scheme for MB-OFDM Based Ultra-Wideband Communication," 2008.