Volume 1, No. 4, Nov-Dec 2010
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info
ISSN No. 0976-5697

Energy Efficient Optimized Hybrid Approach for Time Synchronization in Wireless Sensor Networks

Neeta S. Bosamiya*
Department of Computer Engineering
Sarvajanik College of Engineering and Technology
Surat, India
neeta_danthara@lycos.com

Devesh C. Jinwala
Department of Computer Science and Engineering
S. V. National Institute of Technology
Surat, India
dcj@svnit.ac.in

Abstract: Hybrid Approach for synchronizing time in wireless sensor network was formulated to overcome the drawbacks of the frequently used approaches for time synchronization namely "sender to receiver" and "receiver to receiver". A Hybrid approach is based on the creation of tree structure of nodes before synchronizing nodes in the network. After surveying the traditional approaches we optimize the Hybrid approach to combine the creation of tree structure of nodes along with its synchronization. The optimized approach reduces the message communication and energy consumption of nodes. Simulations of the proposed approach under different scenarios amply demonstrate the improved performance of the same in terms of message transfer and energy consumption, as compared to the other peer approaches.

Keywords: Wireless sensor networks, Time Synchronization, sender to receiver approach, Receiver to receiver approach, Hybrid Approach

I. INTRODUCTION

In case of distributed systems, there is no global clock or common memory. Each processor has its own internal clock and its own notion of time. As a result, achieving time synchronization for those applications that depend on a synchronized notion of time is nontrivial and is already recognized as a critical problem [1].

The Wireless Sensor Networks (WSNs) comprising of geographically distributed tiny, resource constrained sensor nodes working collaboratively to achieve a specific purpose, form a special type of distributed system. Hence, time synchronization in sensor networks is also nontrivial. However, achieving sound solutions to the problem in WSNs is compounded because of their typical operation characteristics viz. *limited resources, pervasive and ubiquitous computing, data-centric multihop communication followed, on demand routing etc*. However, one of the foremost goals of any such solution has to be resource conservation due to the scarcity of the same. As we survey further in section III, one can find various attempts made in the literature to devise solutions to achieve time synchronization in case of wireless sensor networks. Generically, the same can be classified as the "sender to receiver" and "receiver to receiver" [1] and "receiver-only" [7] based approaches. However, as we elaborate further, these methodologies have some drawbacks with respect to number of message exchanged for synchronization and energy consumption. To overcome those drawbacks, a hybrid approach was introduced in [10, 11]. Hybrid approach is efficient with respect to number of message exchanged. However, as we analyze here, there is a scope for further improvement in the hybrid approach. With due justifications, we propose an optimized version of the hybrid approach here and using simulations in the TinyOS [14] environment we show that our proposed approach works better than the existing approaches. To the best of our knowledge, this is a unique attempt to demonstrate tangibly the improvement over the initial hybrid approach for time synchronization in WSNs.

The rest of the paper is organized as follows: in the next section, we discuss the related work, in section 3 we discuss the various approaches to time synchronization. In section 4 we describe the existing time synchronization protocol, Section 5 is devoted to Theoretical Analysis of Hybrid Protocol , whereas in section 6, we discuss the proposed Optimized Hybrid approach and analyze its performance, theoretically. In section 7, we describe the implementation, the experimental setup used and the methodology of evaluation of the same. We also empirically analyze its performance, whereas in section 8 we discuss its performance in terms of the energy consumption.

II. RELATED WORK

Although there are many protocols introduced for time synchronization on wireless sensor networks, most of them are not designed with focus on energy consumption. Energy consumption is very crucial parameter for power constrained sensor networks. Bharath Sundararaman et. al. in [1] and Fikret Sivrikaya in [2] survey the various approaches for time synchronization in WSNs with an emphasis on various parameters for the evaluation of the existing protocols and the common challenges for synchronization. More recently Marioti et. Al. Proposed Flooding time synchronization protocol (FTSP) [4] that synchronizes the network by successively broadcasting the synchronization messages using MAC layer time stamping and performing skew compensation based on linear regression. Time diffusion protocol (TDP) was proposed in [5]. TDP selects a set of the diffusion leaders in every level of the network considering the balance of workload and the stability of the local clocks. Noh et. Al. in [6] provides an extended version of pairwise broadcast clock synchronization (PBS) [7] for multicluster sensor networks. K.-Y. Cheng et. Al. proposed Distributed

Multihop synchronization using pairwise broadcast [8] with the goal to reduce the communication overhead by selecting the children node with maximum connectivity. The proposed algorithm outperforms the performance in comparison with extended PBS, but has a complexity of $O(n^2)$ which can be compared with many other sensor network protocols like RBS and moreover the protocol also doesn't support change in the network topology. Youngtae Jo et. Al. proposed Reference Interpolation Method of time synchronization [9] that uses the broadcast messages coming from both reference packet transmission nodes and a base station node. Each sensor node synchronizes its local time by interpolating the time difference between two packets. Robert Akl et. Al. introduce Hybrid Energy-aware synchronization algorithm in WSNs [10], providing a means of combining then existing approaches for synchronizing time in WSNs viz. the "sender to receiver" and "receiver to receiver" based approaches.

Single hop PBS based on combination of sender to receiver and receiver only methodology is the most efficient algorithm so far with respect to the message transfer as compared to the peer approaches. But the extended version of PBS for multi cluster depends upon the creation of tree structure [12], which is very similar to the Hybrid approach. Moreover the message transfers for selecting the best children having maximum connectivity with the other children nodes in the cluster increases the complexity of the algorithm. To find the best children in a cluster, it requires all the children nodes to broad cast and acknowledges the same from its peer nodes which increases the message communication overhead. The best children node needs reconsideration in case if the tree structure is disturbed due to various attacks on sensor networks [13] or due to the problem of energy depletion in nodes, which as compared to PBS can be better handled in hybrid approach. As in hybrid approach each parent node keeps the track of number of its children nodes. In addition to it, if we compare the complexity of the extended PBS with hybrid protocol, extended PBS has a complexity of $O(n^2)$ which is the worst case complexity of hybrid protocol.

Analyzing the features and advantages of hybrid protocol over pair wise broadcast protocol we optimize the hybrid approach. Hybrid approach is efficient as compared to the peer approaches but there is an overhead in terms of message communication between the nodes for creation of tree structure and then synchronizing the nodes, which inadvertently also increases the convergence time and energy consumption. We extend the Hybrid approach with the objective to optimize the approach in terms of convergence time, reduction in energy consumption and handling energy depletion in sensor network.

Thus, our contributions in the backdrop of the existing efforts can be enlisted thus:

- The goal of optimized hybrid approach is to minimize the message communication between the nodes for synchronization in case of hybrid approach. The proposed solution to minimize the communication overhead is to combine the two phases of hybrid approach viz. "tree creation" and "synchronization", which leads to significant reduction in message communication and inadvertently also reduces the convergence time and energy consumption of nodes in WSNs.

We justify the improvement as estimated above, by implementing the Optimized Hybrid approach on TinyOS platform and tangibly show the reduction in energy consumption as obtained herein.

III. APPROACHES TO TIME SYNCHRONIZATION

There exist several approaches for synchronizing time in case of wireless sensor network. In this section we are going to discuss "sender to receiver", "receiver to receiver", "receiver only" and "Hybrid approach" that is combination of both the methodology.

A. Sender to Receiver Synchronization

In this approach sender synchronizes with a receiver by transmitting the current clock values as timestamps. This traditional approach usually happens in three steps [1].1. The sender node periodically sends a message with its local time as a timestamp to the receiver.2. The receiver then synchronizes with the sender using the timestamp it receives from the sender.3. The message delay between the sender and receiver is calculated by measuring the total round-trip time, from the time a receiver requests a timestamp until the time it actually receives a response. In this methodology due to calculation of Round trip time synchronization error gets reduced. Following are few of the limitations of this approach

- There is Variance in message delay between the sender and the receiver, the delay is due to the network delays and the load in the nodes.
- As we are considering the sender time in calculating the delay, we are not reducing the time-critical path, which is the path of a message that contributes to non deterministic errors.

B. Receiver to Receiver Synchronization

This approach exploits the property of the physical broadcast medium that if any two receivers receive the same message in single-hop transmission, they receive it at approximately the same time. So instead of exchanging message between sender and receiver, receivers exchange the time at which they received the same message and compute their offset based on the difference in reception times. In doing so it reduces the source of uncertainty at the sender end. Following figure shows the distribution of time span between sender and receiver [4].

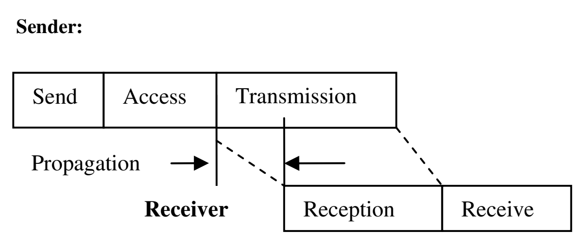

Figure 1. Decomposition of message delivery delay over a wireless link. [4]

In this methodology we are reducing the message delay variance and time critical path since we are not including sender time in the calculation. But this approach is vulnerable to the propagation delay to the various receivers and the differences in receive time.

C. Receiver only Synchronization (ROS)

A group of nodes can be simultaneously synchronized by only listening to the message exchanges of a pair of nodes. The pair of nodes can be synchronized by using sender to receiver approach. And the neighboring nodes can listen to the message transferred between the two nodes [6, 7].

D. *Hybrid Synchronization*

This approach is a combination of sender to receiver and receiver to receiver approach [10, 11]. Hybrid approach was designed basically to overcome the drawbacks of sender to receiver and receiver to receiver approach. Message transmission increases linearly in case of sender to receiver approach with increase in number of nodes in the network. So Sender to receiver approach is better approach when the numbers of nodes in the network are more as compared to receiver to receiver approach. Receiver to receiver approach is better when the numbers of nodes in the network are sparse. Hybrid approach takes the advantage of both the approach depending upon the number of nodes in the network. This methodology reduces redundant message transfer but increases the convergence time required to synchronize time in network.

IV. TIME SYNCHRONIZATION PROTOCOLS

Application of sensor networks like military surveillance and environmental monitoring are closely tied with the real world and hence different networks will have different synchronization needs depending upon its application. There exists various time synchronization protocol designed for several need of wireless sensor networks. In this section we are going to discuss widely used synchronization protocols namely Time Synchronization Protocol for sensor Network (TSPN), Reference Broadcast protocol (RBS), Pairwise broadcast protocol and protocol namely Hybrid protocol that is combination of TSPN and RBS. TSPN is based on sender to receiver approach whereas RBS is based on receiver to receiver approach.

A. *Time Synchronization Protocol for Sensor Networks*

The objective of the protocol is to establish a unique global timescale by creating a self-configuring hierarchical structure in a wireless network. A node in this structure can simultaneously act as a synchronization server to a number of client nodes and as a synchronization client to another server node. In case of TSPN synchronization accuracy doesn't degrade as the number of nodes deployed gets increased.

TSPN protocol works in two phase level discovery phase followed by synchronization phase

- Level Discovery Phase: In this phase root node is assigned a level 0, this node initiates this phase by broadcasting a level-discovery packet that contains the identity and the level of the sender. Immediate neighbors that receive this packet assign themselves a level that is one greater than the level in the packet received. After this step these neighbors broadcast a new level discovery packet with their own level. This process is continued until each node has a level id assigned.
- Synchronization Phase: In this phase a child node sends a synchronization pulse to its parent on level one greater than itself. The parent node receives the pulse and sends back an acknowledge packet to the child node. Child node calculates the clocks offset and delay as below [1].

$$\text{Offset} = ((T2-T1) - (T4-T3))/2 \quad (1)$$

$$\text{Delay} = ((T2-T1) + (T4-T3))/2 \quad (2)$$

Where T1 is the time when the child will send the synchronization pulse to its parent, T2 is the time when the parent receives the synchronization pulse, T3 is the time when the parent will send back the acknowledgement, and T4 is the time when the child node receives the acknowledgement packet.

TSPN is scalable and synchronization accuracy does not degrade significantly as the size of the network is increased.

Following are the drawbacks of TSPN protocol

- The hierarchical structure that the protocol imposes on the network makes the protocol vulnerable to node failures.
- There is the possibility that the node at level (i-1) will not have synchronizing server at level (i). In such retransmission of level discovery should be done, where retransmission can become an over head.
- Energy conservation is not very effective because it requires a physical clock correction to be performed on local clocks of sensors.
- Protocol requires a hierarchical infrastructure that makes it unsuitable for applications with highly mobile nodes.

B. *Reference Broadcast Protocol*

The Reference Broadcast Synchronization (RBS) protocol is so named because it exploits the broadcast property of the wireless communication medium [1]. According to this property, two receivers located within listening distance of the same sender will receive the same message at approximately the same time. In other words, a message that is broadcast at the physical layer will arrive at a set of receivers with very little variability in its delay. If each receiver records the local time as soon as the message arrives, all receivers can synchronize with a high degree of precision by comparing their local clock values when the message was received. This protocol uses a sequence of synchronization messages from a given sender in order to estimate both offset and skew of the local clocks relative to each other. The protocol exploits the concept of time-critical path, that is, the path of a message that contributes to non-deterministic errors in a protocol. Nondeterministic transmission delays are detrimental to the accuracy of a synchronization protocol because they make it difficult for a receiver to estimate the time at which a message was sent and vice versa. In general, the time involved in sending a message from a sender to a receiver is the result of the following four factors, all of which can vary non deterministically.

1. Send time: The time spent by the sender for message construction and the time spent to transmit the message from the sender's host to the network interface.
2. Access Time: The time spent waiting to access the transmit channel.
3. Propagation time: The time taken for the message to reach the receiver, once it has left the sender.
4. Receive time: The time spent by the receiver to process the message.

By considering only the times at which a message reaches different receivers, the RBS protocol directly removes two of the largest sources of non-determinism involved in message transmission, namely the send time and the access time. Thus, this protocol can provide a high degree of synchronization accuracy in sensor networks. In case of

RBS largest sources of error (send time and access time) are removed from the critical path by decoupling the sender from the receivers.

Following are the drawbacks of RBS protocol

- Reference node is left unsynchronized in case of RBS protocol.
- Message transfer is of the order O (n*n), where n is the total number of nodes in the network and hence in case of large network the number of message transfers increases.
- Convergence time is more in case of large network.
- A broadcast medium is mandatory as RBS is based on the broadcast property

C. Pairwise Broadcast Synchronization (PBS)

PBS is based on the idea that while two nodes performing synchronization using two-way message exchanges, other nodes lying nearby can over hear the messages and can also synchronize themselves. PBS efficiently combines the merits of two different basic synchronization methodologies namely the sender to receiver and receiver only synchronization. The single hop based PBS assumes that every node in the network should be located within the communication of the leader nodes. The same idea is extended in [6] for multi hop communication. The extended version of PBS for multi hop scenario is based on group wise pair selection algorithm (GPA). In GPA instead of discovering the entire network connectivity, every parent node only investigates the connectivity among its children nodes. Therefore, the reference node does not need to find the pairwise synchronization sequence of the entire network, but only needs to find the pairwise synchronization sequence among its children, and other parent nodes successively perform the same connection searching procedure as the reference node.

Once the hierarchy of the whole network is formed and the connectivity within every group of nodes is established, the children nodes in each group synchronize with the parent node using either pairwise synchronization or ROS. In order to minimize the total number of synchronization messages for the whole network, the timing message exchanges in each group should be reduced. The children node (j) selected for pair wise synchronization with the parent (i) in any group should be selected with following criteria. A node containing the maximum number of nodes in its common coverage region that is represented as below [6]

$$\hat{J}=\arg\max_{j} N^{i,j}_{ROS} \quad (3)$$

In the above formula i represents the group id and $N^{i,j}_{ROS}$ indicates the number of nodes synchronized in group i with the exchange of message between parent node and jth child node in group i. By selecting the child node with maximum connectivity in group, the maximum number of children nodes in a group i can be synchronized using ROS.

The connection discovery procedure in GPA consists of the following steps

- Select a reference node using an appropriate leader election algorithm or picks up a node having the highest priority and assign it to level zero.
- The reference node broadcasts a level discovery packet containing the identity and the level of packet
- Every node who receives a level discovery packet assigns its level in increasing order and send a new level discovery packet attaching its own level
- Once a hierarchical tree is established, every parent children group performs the following operations: every child node broadcasts a connection discovery packet to other children nodes and sends back acknowledgement packets upon receiving other connection discovery packets, connection discovery packets from any child node belonging to other groups will be discarded.

Expected source of errors in PBS algorithm can be skew in the receiver's local clock and variable delays on the receivers end.

Following are the drawbacks of PBS protocol

- The message transfers for finding the best children node to synchronize with the parent node is an overhead and increases the complexity of the algorithm, since it requires all the children nodes in the group to broadcast the message and also acknowledge the same.
- The children node selected as the best node considering its maximum connectivity amongst the other children node in the group, might need reconsideration if the tree structure of the nodes is disturbed due to various attacks on the network as described in [13] or due to energy depletion problem.

D. The Hybrid Protocol

The time synchronization protocols based on sender to receiver and receiver to receiver approaches have unique strengths when dealing with energy consumption. RBS [1] is most effective in networks where transmitting sensors have a few receivers, whereas TSPN [1] excels when transmitters have many receivers. As compared, the Hybrid algorithm which is motivated by the RBS and TSPN protocols, aims to minimize power regardless of the network's topology. It does so by choosing the best synchronization technique depending on the number of children nodes connected to the transmitter, as explained further.

Hybrid algorithm [3, 10, 11] is divided into below two phases.

- Network flooding: In this phase each sensor node floods the network by sending flood request. Root node initiates the flooding phase; all nodes that receive the flood packet will set their level one greater than the root node and rebroadcast flood request with its current level. Each node will broad cast acknowledgement packet with current node Id. Nodes receiving acknowledgement packet will count the child node based on the number of acknowledgement packet it receives.
- Time synchronization phase: In this phase based on the number of children that was calculated by the above phase synchronization protocol will be decided. If the number of child node is less than threshold value then RBS synchronization will be carried out. If the child node is greater than threshold value then TSPN synchronization will be carried out between parent node and child node.

V. THEORETICAL ANALYSIS OF HYBRID PROTOCOL

In this section we analyze the Hybrid protocol in terms of total messages exchanged. We also analyze the computation of threshold value required for number of child nodes. The threshold value is calculated by equating the number of message exchanged for TSPN and RBS. Threshold value is calculated for an ideal value of child nodes where the performance of TSPN and RBS are equal [10].

A. Calculation of threshold value for Hybrid Protocol

In case of sensor networks, sensors are dispersed at random. And hence there will be patch that will be densely populated whereas there can be patch that is lightly populated. Time synchronization protocols like TSPN and RBS perform differently for both the kind of patches. TSPN excels where the number of receivers are more whereas RBS excels where the number of receivers are less. The threshold value of receivers will decide upon TSPN and RBS time synchronization protocol. To calculate the threshold value, we need information about the energy consumption for reception, transmission and the number of transmission and reception in case of TSPN and RBS time synchronization algorithm. It is assumed that the energy required to receive the message is usually half than the energy required to transmit the message. Number of messages that are transmitted and received for TSPN time synchronization protocol are as follows where TX_{TSPN} are total number of messages transmitted and RX_{TSPN} are total number of messages received [10].

$$TX_{TSPN} = n + 1 \quad (4)$$

$$RX_{TSPN} = 2n \quad (5)$$

For RBS time synchronization the number of messages that are transmitted and received are as follows [10]

$$TX_{RBS} = n \quad (6)$$

$$RX_{RBS} = n + (n*(n-1))/2 = (n*n + n)/2 \quad (7)$$

Combining (4), (5), (6), and (7) to calculate threshold value, where α is the ratio of reception to transmission power and n is the number of child nodes [10].

$$TXRBS + \alpha * RXRBS = TXTSPN + \alpha * RXTSPN$$

$$n + ½(n*n + n)/2 = n+1 + ½(2*n)$$

$$n*n - 3*n - 4 = 0$$

$$(n-4)(n+1) = 0 \quad (8)$$

Equation (8) shows that the energies used by RBS and TSPN are equal where there are 4 receivers per transmitter, so the threshold value is set to 4. When there are fewer receivers than 4, RBS is more efficient whereas if number of receivers is more than 4 TSPN time synchronization should be selected.

B. Analysis of message transmission for Hybrid protocol

In general, the Hybrid algorithm is divided in two phase's viz. the *Network flooding phase* and the *Time synchronization phase*. The first phase aims at creation of tree structure and the goal of second phase is to synchronize the nodes in the created tree structure. In the first phase a hierarchical tree structure is created and calculation of child nodes for each node is done. In the second phase synchronization is carried out either by RBS or TSPN time synchronization protocol. With n, the number of broadcast messages by the parent nodes and m the number of unicast replies by the children nodes. The total message transmission that takes place for the first phase "network flooding" is n broadcast messages + m unicast replies. Where m is the summation of child nodes for each level in the tree structure and n is the maximum number of nodes in the tree structure.

Thus, the total message transmission for synchronizing is the total number of message for synchronization (TSPN or RBS) + n + m.

VI. Optimized Hybrid Protocol: Proposed Approach

The tradeoff between the accuracy and energy consumption is the most crucial factor in designing time synchronization protocols for WSNs. Hence it is very important to keep the balance between satisfactory accuracy level and energy consumption. In this section, we propose a new approach that improvises the existing Hybrid approach in terms of message communication and Energy consumption.

A. The Optimized Hybrid Protocol and its analysis

The basic Hybrid algorithm suffers from the overhead due to creation of the tree structure, since the second phase of synchronization is based on the calculation of child nodes computed from the first phase. However, if the first and second phases are combined, the message transmission will be reduced by the maximum nodes in the tree structure i.e. n. In addition if the children nodes are equal to or more than the threshold value, the total transmission will be further reduced by the summation of the children nodes at each level i.e. m. The worst case complexity of our algorithm is O (n^2) for sparse network, when RBS time synchronization is carried on. And otherwise O (n) when TSPN time synchronization is carried out, n denotes the number of receiver nodes. The resulting algorithm is shown below.

B. The Optimized Hybrid protocol Algorithm

```
Begin

1. Set num_receivers to 0
2. If current_node is root node
     -Broadcast flood_sync packet with level no. and time
3. Else If current_node receives flood_sync packet
     -Set parent of current_node to source of broadcast
     -Set current_node level to parent's node level + 1
     -Rebroadcast flood_sync request with current_node ID,
       timestamp and level
     -Broadcast ack_flood_sync packet with current_node ID
       and local timestamps
     -Ignore subsequent flood_sync packets
4. Else If current_node receives ack_flood_sync_packet
     -Increment num_receivers
          5. If num_receivers greater then threshold value
               -Send acknowledgement for ack_flood_sync
                 with timestamp of receiving, sending the
                 ack_flood_sync
          6. Else If
                 For each receiver
                      -Record local time of reception for
                      flood_sync packet
                      -Broadcast observation_packet
                      -Receive observation_packet from
                       other receivers
          Endif
Endif
```

As can be observed from the pseudo code, we combine the creation of hierarchical tree structure along with time synchronization in a single phase. This eventually reduces the number of messages required to create the tree structure. Thus if k is the total number of nodes in the structure, then we are reducing k broadcast messages that are required to create the tree structure. In addition, with m being the number of children nodes and is greater than the threshold value, we are also reducing m unicast replies from each of those child nodes.

When the broadcast message is transmitted from the root the message contains the level number and the time stamp. The child node that receives the message sends back the acknowledgement message back to the root node. The root node counts the number of message acknowledgements from all the children nodes and if the value exceeds the threshold value, it replies back the with current time stamp. Each child node then calculates the offset and the round trip delay to set its own clock. If the acknowledgement received is less than threshold value, it waits for a predefined time. We estimate and set the predefined time heuristically using the simulation of the algorithm in TinyOS[14]. After waiting for this predefined time if still the acknowledgement is less than threshold value it performs RBS time synchronization between the child nodes. But even in this case we are improving over the basic hybrid approach since we are reducing the message transfer required for synchronization at parent nodes.

VII. IMPLEMENTATION

The event-driven nature of sensor networks makes testing of an individual mote insufficient. Programs must be tested at scale and in complex to capture a wide range of interactions. Deploying hundreds of motes is intimidating task; simulator can deal with these difficulties, by providing controlled environment. In this section we describe the methodology followed by us for implementing the Optimized Hybrid protocol.

A. *Methodology*

The Optimized Hybrid Protocol is implemented on TinyOS 1.x platform using the TOSSIM simulator [14] and simple radio model. TinyOS is an operating system specifically designed for sensor networks. It has a component-based programming model, provided by the nesC language [15]. The testing of algorithm is carried out using various values of children nodes in the network viz. value less than threshold value and value greater than threshold to check the selection of Time Synchronization Protocol amongst TSPN and RBS. If the child node is less than the threshold value RBS synchronization is selected otherwise TSPN is selected. The simulation is carried out for single hop communication with a single parent node and three or more child nodes. The call graph for our implementation of Optimized Hybrid protocol (opthybridM) is as shown in Fig. 2. Since TOSSIM 1.x doesn't provide support for energy analysis, we use the Avrora emulator [16] for the Mica2 sensor motes for energy profiling along with monitors as energy and packets. Avrora is an open source cycle accurate simulator for embedded sensing programs. Avrora can emulate two typical platforms Mica2 and MicaZ. Avrora along with monitors as energy prints the usage of various components over the simulation period. It prints the energy consumed by each of the following components CPU, LEDS. External Flash, Radio.

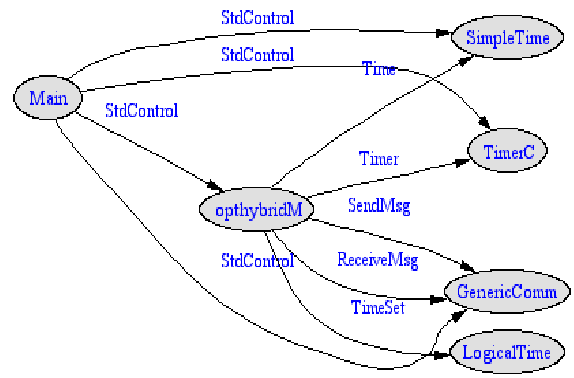

Figure 2. Call graph for Simulation of Optimized Hybrid Protocol

The above call graph provides us with the graphical representation of the component relationships within an application. StdControl is a common interface used to initialize and start TinyOS components. In our application we are wiring stdControl interface in main to the stdControl interface in opthybridM. We are also wiring RecieveMsg and SendMsg event of GenericComm with opthybridM Receive and send msg event. We are wiring Timer event from TimerC component, TimeSet and Time from LogicalTime and SimpleTime respectively.

B. *Performance Results*

In Table I below, we analyze the number of messages that are transferred between the parent node and child node, in the proposed approach and in the basic Hybrid approach. The value of child node for first table is equal to the threshold value calculated above and for the table II it is less than the threshold value. In Table I the comparison between the proposed approach and hybrid approach proves that the total number of messages communicated gets reduced by fifty percent for children nodes when TSPN protocol is followed and in table II the comparison depicts the reduction in message communication at parent nodes.

Table I Comparison of message transmitted for Hybrid and Optimized Hybrid Protocol for four child nodes and TSPN protocol

Type of node	**Protocol**	**Hybrid**		**Optimized Hybrid Protocol**	
		Broadcast	*Unicast*	*Broadcast*	*Unicast*
parent Node	TSPN	2	4	1	4
Four Child nodes	TSPN	-	8	-	4

Table II Comparison of message transmitted for Hybrid and Optimized Hybrid Protocol for Three child nodes and RBS protocol

Type of node	**Protocol**	**Hybrid**		**Optimized Hybrid Protocol**	
		Broadcast	*Unicast*	*Broadcast*	*Unicast*
parent Node	RBS	2	-	1	-
Three Child nodes	RBS	3	3	3	3

VIII. ENERGY REQUIREMENTS

As is evident, the sensor nodes operational paradigms are aimed to reduce the overall energy overhead to the extent possible. Thus, in our evaluation we use energy consumption also as one of the important metric to be evaluated. We use Avrora [16] as the Mica2 emulator on the TinyOS platform. In the tables III and IV we show the energy consumption of Optimized Hybrid Protocol as well as Hybrid protocol for children nodes less than and greater than threshold value, the energy consumption at each node is divided into various components like, Radio, and CPU.

A. Energy consumption for child nodes less than the threshold value

We selected three Children nodes for a parent node and the communication and energy consumption were traced. Table three indicates the performance of Hybrid protocol in terms of energy consumption in joules. The transmission mode zero is for initializing the nodes in the network and transmission mode fifteen indicates the actual transmission of messages for synchronizing time. Table four indicates the energy consumption for Optimized Hybrid Protocol for all the nodes in the network. In table three the energy consumption at each node including the parent node is same since each of them transmits two message including broadcast and unicast message. Parent node transmits two broadcasts and child node transmits one unicast and one broadcast message. In table four there is a reduction in energy consumption at parent node since we are reducing the transmission of one broad cast message. Hence the energy consumption at parent node reduces by half which is depicted in table five. Fig. three and four denotes the graphical presentation of reduction in energy at parent node.

Table III Energy Consumption for Hybrid Protocol three child nodes and RBS Protocol

Optimized Hybrid Protocol	CPU		Radio(joules)		
	Active	*Idle*	*Rx*	*Tx(0)*	*Tx(15)*
Node 0	0.02	0.15	0.43	0.00053	0.00177
Parent Node	0.02	0.14	0.43	0.00053	0.00088
Node 2	0.02	0.14	0.43	0.00053	0.00177
Node 3	0.02	0.14	0.43	0.00053	0.00177

Table IV Energy Consumption for Optimized Hybrid Protocol three child nodes and RBS Protocol

Hybrid Protocol	*CPU*		*Radio(joules)*		
	Active	*Idle*	*Rx*	*Tx(0)*	*Tx(15)*
Node 0	0.04	0.3	0.8	0.00107	0.00177
Parent Node	0.04	0.2	0.8	0.00108	0.00177
Node 2	0.04	0.2	0.8	0.00107	0.00177
Node 3	0.04	0.2	0.8	0.00107	0.00177

Table V Percentage Reduction in Energy Consumption for Optimized Hybrid Protocol for three child nodes and RBS Protocol

Nodes	Optimized Hybrid Protocol Tx(15)
Parent Node	50%

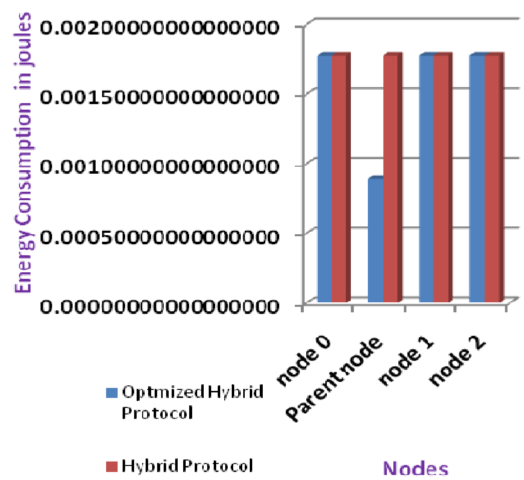

Figure 3. Energy consumption for Optimized Hybrid protocol and Hybrid Protocol with three child nodes and RBS protocol

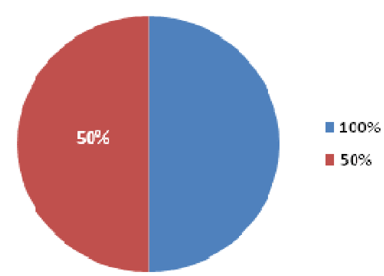

Figure 4. Fifty percent energy reduction at parent node using Optimized hybrid protocol with three child nodes and RBS protocol

B. Energy consumption for child nodes greater than or equal to threshold value

Four children nodes were selected for a parent and the communication and energy consumption were traced. Table six indicates the performance of Hybrid protocol in terms of energy consumption in joules. Table Seven indicates the energy consumption for Optimized Hybrid Protocol for all the nodes in the network. In table six the energy consumption at each child node is same since each of them transmits two unicast messages. Parent node transmits two broadcasts and one unicast message for each child. In table seven there is a reduction in energy consumption at parent node and child nodes as well since we are reducing the transmission of one broad cast message at parent level and unicast messages at each child. Hence the energy consumption at child node reduces by almost half which is depicted in table eight. Fig. five, six and seven denotes the graphical presentation of reduction in energy at child and parent node.

Table VI Energy Consumption for Hybrid Protocol four child nodes and TSPN Protocol

Hybrid Protocol	CPU		Radio (joules)		
	Active	Idle	Rx	Tx(0)	Tx(15)
Node 0	0.08	0.5	1.7	0.00107	0.00179
Parent Node	0.34	0.4	1.7	0.00110	0.00540
Node 2	0.08	0.5	1.7	0.00107	0.00179
Node 3	0.08	0.5	1.7	0.00107	0.00179
Node 4	0.08	0.5	1.7	0.00107	0.00179

Table VII Energy Consumption for Optimized Hybrid Protocol four child nodes and TSPN Protocol

Optimized Hybrid Protocol	CPU		Radio(Joules)		
	Active	Idle	Rx	Tx (0)	Tx (15)
Node 0	0.137	0.94	2.87	0.00053	0.000888
Parent Node	0.137	0.94	2.87	0.00055	0.004444
Node 2	0.137	0.94	2.87	0.00053	0.000888
Node 3	0.137	0.94	2.87	0.00053	0.000888
Node 4	0.137	0.94	2.87	0.00053	0.000888

Table VIII Percentage Reduction in Energy Consumption for Optimized Hybrid Protocol for four child nodes and TSPN Protocol

Nodes	Optimized Hybrid Protocol Tx (15)
Parent Node	17.76%
Four Child Nodes	49.60%

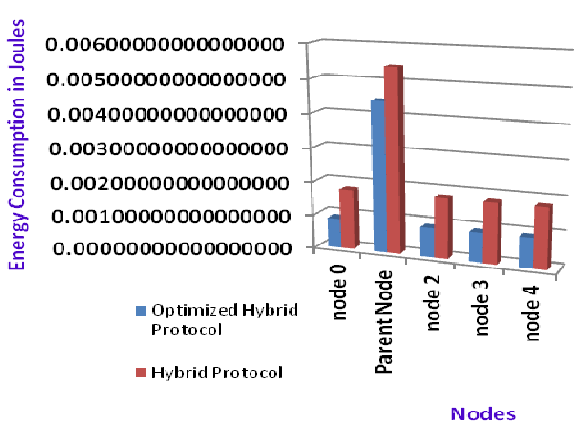

Figure 5. Energy consumption for Optimized Hybrid protocol and Hybrid Protocol with four child nodes and TSPN protocol

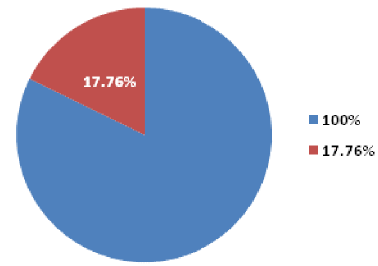

Figure 6. Energy Reduction at parent node using Optimized Hybrid protocol and TSPN protocol

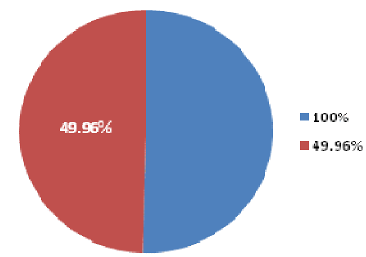

Figure 7. Energy Reduction at child nodes using Optimized Hybrid protocol and TSPN protocol

IX. CONCLUSION

Being motivated with the severe resource constraints in the sensor nodes and the limitation of the basic Hybrid protocol for time synchronization in WSNs, we propose here an Optimized Hybrid Protocol. The proposed protocol aims to reduce converge time, the number of message exchanges and the energy overhead. Using the TOSSIM as the simulator, we are aptly able to justify that there is substantial reduction in the message transfers required for time-synchronizing the sensor nodes. In addition, using the detailed energy analysis of nodes with the help of Avrora emulator in the Mica2 mode, we tangibly show that there is significant reduction in the energy consumption of nodes as well.

X. REFERENCES

[1] Bharath Sundararaman, Ugo Buy, and Ajay D. Kshemkalyani, "Clock Synchronization for Wireless Sensor Networks: A Survey", Elsevier Ad Hoc Network Journal, vol. 3, No. 3, 2005, pp. 281-323.

[2] Fikret Sivrikaya and Bulent Yener, "Time Synchronization in Sensor Networks: A Survey". IEEE Network Journal special issue on 'Ad Hoc Networking: Data Communications & Topology Control', vol. 18, No. 4, 2004, pp. 45-50.

[3] Khurram Shahzad, Arshad Ali , N.D. Gohar , "ETSP: An Energy-Efficient Time Synchronization Protocol for Wireless Sensor Networks", Proc. 22nd International Conference on Advanced Information Networking and Applications - Workshops (aina workshops 2008), 2008, pp. 971-976.

[4] Miklós Maróti, Branislav Kusy, Gyula Simon, Ákos Lédeczi, "The Flooding Time Synchronization Protocol", Proc. 2nd International Conference on Embedded Networked Sensor Systems, 2004, pp.39 - 49.

[5] W. Su and I. F. Akyildiz, "Time-diffusion synchronization protocol for wireless sensor

networks", IEEE/ACMTransactions on Networking, vol. 13, No. 2, 2005, pp. 384–397.

[6] Kyoung-Lae Noh, Yik-Chung Wu, Khalid Qaraqe, Bruce W. Suter , "Extension of pairwise broadcast clock synchronization for multicluster sensor networks", EURASIP Journal on Advances in Signal Processing, No. 71, 2008.

[7] K.-L. Noh, E. Serpedin, and K. A. Qaraqe,"A new approach for time synchronization in wireless sensor networks: pairwise broadcast synchronization", IEEE Trans. Wireless Communications, vol. 7, No. 9, 2008, pp. 2218-3322.

[8] K.-Y. Cheng, K.-S. Lui, Y.-C. Wu and V. Tam,"A Distributed multihop time synchronization protocol for wireless sensor Networks using pairwise broadcast synchronization", IEEE Trans. on Wireless Communications, vol. 8, No. 4, 2009, pp. 1764-1772.

[9] Youngtae Jo, Chongmyung Park, Joahyoung Lee, Inbum Jung, "Energy effective time synchronization in wireless sensor network", Proc. IEEE International Conference on Computational Science and its Applications (ICCSA 2007), 2007, pp. 26-29.

[10] Robert Akl, Yanos Saravanos, "Hybrid energy-aware synchronization algorithm in wireless sensor networks**",** Proc. 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07), 2007**.**

[11] Yanjing Sun, Jiansheng Qian, Jinlei Wu, "Hybrid energy-aware time synchronization protocol for WSNs in Coal Mine", Proc. *IEEE* Information Acquisition, ICIA '07, 2007, pp. 436-441.

[12] Li-Ming He, "Time Synchronization based on spanning Tree for wireless sensor networks", Proc. IEEE Wireless 4th International Conference on Communications, Networking and Mobile Computing, *WiCOM '08*, 2008, pp.12-14.

[13] Michael Manzo, Tanya Roosta, Shankar Sastry, "Time synchronization attacks in Sensor Networks", Proc. 3rd ACM workshop on Security of ad hoc and sensor networks, 2005, pp. 107 – 116.

[14] Philip Levis and Nelson Lee, TOSSIM: A Simulator for TinyOS Networks, University of California, Berkeley, Tech. Rep., 2003.

[15] Philip Levis, Nelson Lee, Matt Welsh, David Culler, "TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications", Proc. First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003), 2003, pp. 126-137.

[16] Ben L. Titzer, Daniel Lee, and Jens Palsberg, "Avrora: Scalable Sensor Network Simulation with Precise Timing", Proc. Fourth International Conference on Information Processing in Sensor Networks (IPSN '05), 2005, pp. 477-482.