Improving Life Time of Wireless Sensor Networks Using Neural Network Based Classification Techniques With Cooperative Routing

Sudhir G. Akojwar, Rajendra M. Patrikar

***Abstract*—** Wireless Sensor Networks are design with energy constraint. Every attempt is being made to reduce the energy consumption of the wireless sensor node. Communication amongst nodes consumes the largest part of the energy. The paper focuses on use of classification techniques using neural network to reduce the data traffic from the node and there by reduce energy consumption. The sensor data is classified using ART1 Neural Network Model. Wireless sensor network populates distributed nodes. The co-operative routing protocol is designed for communication in a distributed environment. In a distributed environment, the data routing takes place in multiple hops and all the nodes take part in communication. This protocol has been designed for wireless sensor networks. This ensures uniform dissipation of energy for all the nodes in the whole network. Directed diffusion routing protocol is implemented to carry out performance comparison. The paper discusses classification technique using ART1 neural network models. The classified sensor data is communicated over the network using two different cases of routing: cooperative routing and diffusion routing. Ptolemy-II-Visual Sense is used for modeling and simulation of the sensor network. Lifetime improvement of the WSN is compared with and without classification using cooperative routing and diffusion routing.

***Keywords*—** WSN; Neural Network; Clustering; Ptolemy-II; Visual sense, cooperative routing.

I. Introduction

Advances in sensor technology, low-power electronics, and low -power radio frequency design have enabled the development of small, relatively inexpensive and low-power sensors, called microsensors. These wireless microsensor [1]networks represents a new paradigm for extracting data from the environment and enable the reliable monitoring of a variety of environments for applications that include surveillance, machine failure diagnosis, chemical/biological detection, habitat monitoring, environmental monitoring etc. An important challenge in the design of wireless sensor networks (WSN) is that two key resources - communication bandwidth and energy - are significantly more limited than in a tethered network environment. These constraints require innovative design techniques to use the available bandwidth and energy efficiently [2]. The communication consumes the largest part of the energy budget. Hence attempt must be done to implement techniques two save energy on communications. The paper discusses real time classifier using ART1 [3] neural networks model. Real time classifier classifies the sensor readings and then only its class ID needs to be communicated further. This brings a saving of energy. The implementation of Classifier using ART1 and Fuzzy ART is discussed in detail in [4]. Ptolemy-II is used to model the sensor networks. Ptolemy-II is the software infrastructure of the Ptolemy Project. Cooperative routing and diffusion routing are implemented and simulated under Ptolemy-II environment. ART1 Classifier implemented as MATLAB code classifies this data. Ptolemy-II permits interfacing of MATLAB code within its models. The classified sensor data is then communicated further in one case with cooperative routing protocol and in other case with diffusion routing protocol. This scheme gives the wonderful advantage of improving the network bandwidth by use of classification technique and ensures uniform consumption of energy by using cooperative routing.

Manuscript received October 31, 2008: Revised version received Nov 4, 2008. (Write the dates on which you submitted your paper for review as well as the revised version).
Sudhir G. Akojwar is with the Electronics Engineering Department of Rajiv Gandhi College of Engineering Research and Technology, Chandrapur, INDIA. Phone +91 7172 256755, fax: +91 7172 225287 email: sudhirakojwar@rediffmail.com ,sgakojwar@ieee.org
Dr. Rajendra. M. Patrikar, was with Indian Institute of Technology Bombay, INDIA. He is Professor in the Department of Electronics and Computer Science, Visvesvaraya National Institute of Technology, Nagpur. Email: rajendra@computer.org .

II. WIRELESS SENSOR NETWORKS

A sensor network is composed of a large number of sensor nodes, which are densely deployed either inside the phenomenon or in its proximity. The sensor nodes may be randomly deployed in inaccessible terrains or disaster relief operations hence sensor network protocols and algorithms must possess self-organizing capabilities. One of the most constraints on sensor nodes is the low power consumption. Hence sensor network protocols focus on power conservation. Since the sensor nodes are often inaccessible, the lifetime of a sensor node must be assured. Lifetime of the sensor node depends on the lifetime of power resources. Power scarcity must be effectively managed. Fig.1. shows an architecture of a typical sensor node. It basically consist of – Sensing unit, computation or data processing unit and communication or radio unit. Sensor data is converted to digital streams using ADC. Microcomputer unit (MCU) process this data streams by executing algorithms – such as classification algorithm. Then the processed data is communicated over the network by radio unit. Power consumption can be divided into three domains: sensing, communication and data processing. Sensing power

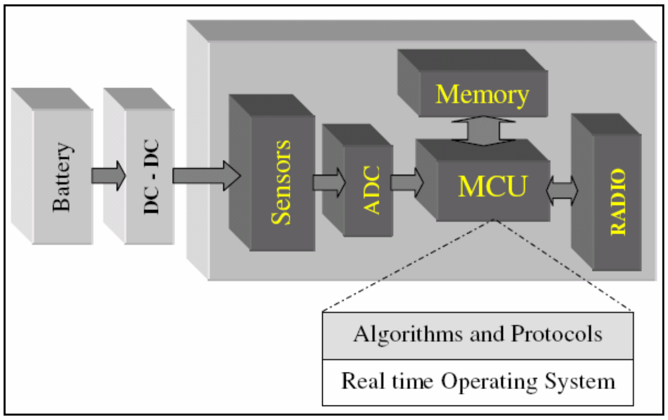


Fig.1. Architecture of a typical Sensor Node

varies with the nature of applications. Sporadic sensing might consume lesser power than constant event monitoring. Of the three domains, a sensor node expends maximum energy in data communication by radio unit This involves both data transmission and reception.

III. ARTIFICIAL NEURAL NETWORKS PARADIGM IN WSN

Wireless sensor network is highly data centric. Data communication in WSN must be efficient one and must consume minimum power. Every sensor node consists of multiple sensors embedded in the same node. Thus every sensor node is a source of data. These raw data streams cannot be straightway communicated further to the neighboring node or the base station. These sensor data streams need to be classified. A group of sensor nodes forms a cluster. Each node transfer data to a cluster head and then cluster head aggregates the data and sends to base station. Hence clustering and classification techniques are important and can give new dimension to the WSN paradigm. Hence efficient data clustering techniques must be used to reduce the data redundancy and in turn reduce overhead on communication. This can be very well accomplished by using some of the algorithms developed within the artificial neural networks paradigm, which can be easily adapted to WSN. Artificial neural network (ANN) consists of small computing units, called neurons, arranged in different layers and interconnected with each other. Simple mathematical computations are performed at each neuron. There are lots of advantages of using ANN in WSN and will meet the requirements for WSN like - simple parallel-distributed computations, distributed storage and data robustness. Thus it can reduce memory requirement to the minimum. The WSN nodes can be clustered in different groups, thereby reducing the dimension of the network. This lowers communication and memory cost.

Carpenter and Grossberg developed ART; it provides a solution for the plasticity and stability dilemma [5]. ART can learn arbitrary input patterns in a stable, fast and self-organizing way, thus overcoming the effect of learning in stability that plagues many other competitive networks. ART is not, as is popularly imagined, a neural network architecture. It is a learning theory, that resonance in neural circuits can trigger fast learning. As such it subsumes a large family of current and future neural networks architectures, with many variants. ART1 is the first member, which only deals with binary input pattern

IV. CLASSIFICATION TECHNIQUES

Wireless sensor network is highly data centric. Data communication in WSN must be efficient one and must consume minimum power. Every sensor node consists of multiple sensors embedded in the same node. Thus every sensor node is a source of data. These raw data streams cannot be straight away communicated further to the neighboring node or the base station. These sensor data streams need to be classified. A group of sensor nodes forms a cluster. Each node transfer data to a cluster head and then cluster head aggregates the data and sends to base station. Hence clustering and classification techniques are important and can give new dimension to the WSN paradigm. Basically, classification system is either supervised or unsupervised, depending on whether they assign new inputs to one of a infinite number of discrete supervised classes or unsupervised categories respectively. ART1 and Fuzzy ART are unsupervised neural network models which are used for classification of sensor data. ART1 model is used for classification of Binary valued data. While Fuzzy ART model can be used for analog data, wherein the input data is fuzzy valued.

V. BASICS OF ART1 ALGORITHM

The ART1 model is described in Fig. 2[3]. It consists of three layers (basically only two layers), Layer F0, which is the input layer, which copies the inputs to the F1 layer and has N nodes (one for each binary bit of input pattern). Layer F1, which is the comparison, layer and layer F2 is the recognition or category layer. Layers F0, F1, F2 are constituted of N, N and M neurons respectively. Each node in the F2 layer represents "cluster" or "category". In this layer only one node will become active after presentation of an input pattern $I \equiv (I_1, I_2, \ldots\ldots I_N)$. The F2 layer category that will become active would be more closely represent the input pattern I. If network detects novel input for which there is no preexisting category, a new category will be formed. Each F1 node - x_i is connected to all F2 nodes - y_i through bottom up connection weights z_{ij}^{bu}, so that the input received by each F2 node y_i is given by

$$T_j = \sum_{i=1} z_{ij}^{bu} I_i \qquad (1)$$

Bottom up weights z_{ij}^{bu} take any real value in the interval [0, K], where

$K = \frac{L}{L-1+N}$, where L is a constant chosen to be greater than 1 to keep L -1 > 0. [3]

Layer F2 acts as Winner -Take-All network, i.e. a competitive layer for the outputs, so that all nodes y_i will stay inactive, except the one that receives the largest bottom up input T_j. Once an F2 winning node arises a top-down template is activated through the top-down weights z_{ji}^{td}. In the fast learning Type-3 model top-down weights z_{ji}^{td} take values '0' or '1'. Let us call this top-down template $X = (X_1, X_2, ...X_N)$. The resulting vector X is given by the equation,

$$X_i = I_i \sum_j z_{ji}^{td} y_i \quad (2)$$

Since only one y_i is active, let us call this winning F2 node y_J, so that y_J=o if j≠0 and y_J=1. In this case we can state

$$X_i = I_i z_{Ji}^{td} \quad \text{or} \quad X = I \cap z_J^{td} \quad (3)$$

where $z_J^{td} \equiv (z_{1J}, z_{2J}, ...z_{NJ})$. This top-down template will be compared against the original input pattern I according to a predetermined vigilance criteria, tuned by the vigilance parameter 0 < ρ ≤ 1, so that two alternative may occur: (1) If $\rho |I| \leq |I \cap z_J^{td}|$ the active category J is accepted and the system weights will be uploaded to incorporate this new knowledge. (2) If $\rho |I| > |I \cap z_J^{td}|$ the active category J is not valid for the actual value of the vigilance parameter ρ. In this case y_J will be deactivated (reset) making T_j = 0, so that another y_j node will become active through the Winner -Take –All action of the F2 layer. Here the notation |r| represents the cardinality of vector r, i.e. $|r| = \sum_{i=1}^{N} r_i$

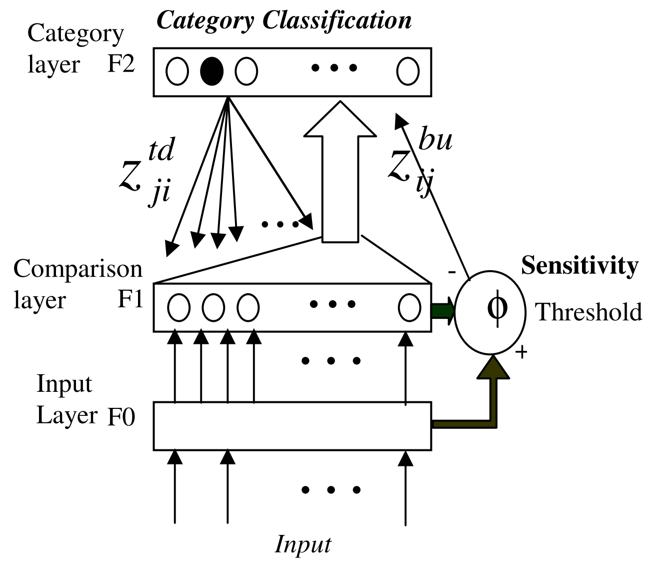
Once an active F2 node is accepted by the vigilance criterion, learning takes place. The weights will be updated according to the following algebraic equations,

$$z_J^{bu}|_{new} = \frac{L(\mathrm{I} \cap (\mathrm{z}_\mathrm{J}^{\mathrm{td}})_{old})}{L-1+|(\mathrm{I} \cap (\mathrm{z}_\mathrm{J}^{\mathrm{td}})_{old}|} \quad (4)$$

$$z_J^{td}|_{new} = \mathrm{I} \cap (\mathrm{z}_\mathrm{J}^{\mathrm{td}})_{old}$$

note that only the weights of the connections touching the F2 winning node y_J are updated.

$$y_J = 1, \ \text{if,} \ T_j = \max_k \{T_k\} \ \text{otherwise} \ y_J = 0$$



$$I \equiv (I_1, I_2, ..., I_N)$$

Fig.2: Architecture of ART1 model

VI. VLSI FRIENDLY COMPACT ART1 ALGORITHM

The imlementation complexities can be further reduced by making some sort of normalisation of the weight templates disscussed in [6]. There are two templates of weights that have to be built. The set of bottom-up weights z_{ij}^{bu}, each of which has to store a real value belonging to the interval [0,K], and the set of top-down weights z_{ji}^{td}, each of which stores either the value '0' or the value '1'. Looking at equ. (4) it can be seen that the bottom –up set { z_{ij}^{bu} }and the top down set { z_{ji}^{td} }contain the same information: each of these sets can be fully computed by knowing the other set. It can be seen that the bottom –up set z_{ij}^{bu}, is a kind of normalised version of the top-down set z_{ij}^{bu}. This way, we can substitute the two sets { z_{ij}^{bu} } and { z_{ji}^{td} } by a single binary valued set { z_{ij} }, and modify equ.(1) to take into account the normalisation effect of the original bottom up weights,

$$T_j = \frac{LT_{Aj}}{L-1+T_{Bj}} = \frac{L\sum_{i=1}^{N} z_{ij} I_i}{L-1+\sum_{i=1}^{N} z_{ij}} \quad (5)$$

We can show use equ.(3) which can be substituted by the following equation, resulting in a system that preserve all the computational properties of the original ART1 architecture

$$T_j = L_A T_{Aj} - L_B T_{Bj} + L_M \quad (6)$$

Where $L_A > L_B$ are positive parameters that play the role of the original L and L-1 parameters. L_M >0 , is a constant parameter needed to assure that $T_j \geq 0$, for all possible values of T_{Aj} and T_{Bj}. This algorithms is imlemented in MATLAB and interfaced with sensor network model of Ptolemy- Visual sense as disscussed in section X.

VII. CLUSTERING ARCHITECTURE FOR WIRELESS SENSOR NETWORKS

The strength of the ART1 model is its unique ability to solve a *stability plasticity* dilema, in fast learning mode it take extreamly short training times, it can generate incrementally growing number of clusters based on the variations in the input data. The network runs entirely autonomously ; it does not need any outside control, it can learn and classify at the same time, provides fast access to match results, and is designed to work with infinite stream of data. All these features make it an excellent choice for applications in wireless sensor networks.

For organising the distributed data of the sensors this ART1 neural network can be used in three different clustering schemes for sensors network. *(1)One cluster head collecting all sensors data*: In this architecture the sensor nodes send the sensory reading to clusterhead (Gateway Node), where an ART1 neuron is implemented . This model, as shown in Fig. 3, brings advantages in that we need not to fix in advance the number of clusters (categories) that the network should learn to recognise. *(2) Each unit being a clusterhead clustering data with different level of details:* In this architecture each unit receives the input data from all sensor nodes in one cluster by broadcast. Then each unit classifies the sensor data with different sensitivity threshold, thus providing a general overall view on the network, Instead of having only one cluster, since the data is broadcast anyway, in this architecture all sensors node collect data from all over units and they all have Fuzzy ART implementations. So we can use different sensitivity thresholds with which we achieve different kinds of views over the same data, coarser with smaller number of categories or more detailed with bigger number of categories. *(3)Clusterhead collecting only clustering outputs from the other unit:* Each sensor node has Fuzzy ART implementations classifying only its sensor readings. One of these unit can be chosen to be a clusterhead collecting and classifying only the classifications obtained at other units. Since the clusters at each unit can be represented with binary values, the neural network implementation at the clusterhead is ART1 with binary inputs.

With this architecture[4] a great dimensionality reduction can be achieved depending on the number of sensor inputs in each unit. At the same time communication savings benefit from the fact that the cluster number is a small binary number unlike raw sensory readings which can be several bytes long real numbers converted from the analog inputs.

If the number of sensors in each unit is s , the clusterhead collects data from h units, and the number of different categories in each unit can be represented by b – byte integer, while the sensor readings are real numbers represented with p bytes, then the communication saving can be calculated as:

$$\frac{s.h.p}{h.b} = \frac{s.p}{b}.$$

Since the communication is the biggest consumer of energy in the sensor node, this leads to bigger energy savings as well.

In this model of sensor network first scheme of classificatin is implemented – One cluster head collecting all sensors data as shown in Fig.3.

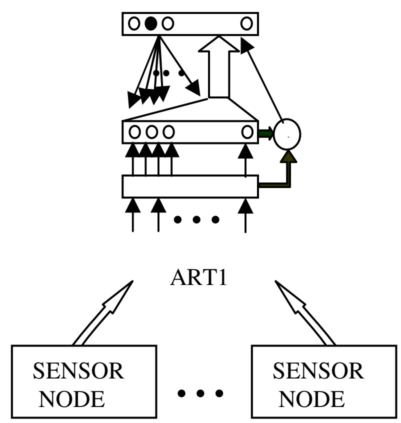


Fig.3 One cluster head collecting all Sensors data The sensor nodes send the sensory reading to one node, which is chosen to be a cluster head, where an ART1 neuron is implemented.

VIII. COOPERATIVE ROUTING

The proposed protocol aims to enhance lifetime by using sub optimal paths. While this constitutes the basis for almost all the approaches for enhancing lifetime, but the best attempt is must to ensure more equitable distribution of the energy consumption. Further, these protocols either use probabilistic method for determining a path (e.g. energy aware routing) or use the path in a round-robin fashion (e.g. directed diffusion). We introduce a deterministic method for choosing a path, with addition of updating mechanism.

In our approach to increase network lifetime, we use a completely different set of parameters to use optimal and sub optimal paths. We propose the use of local group average to make a decision regarding the rejection of an optimal path and switching over to a sub optimal path. The local average that we use is the average of the residual energies of all the directed nodes in a local group. Therefore the name of this protocol is co-operative routing protocol. The local group averages need to be updated and we propose a mechanism for these updates without spending any extra energy for communicating these updates. Thus apart from this inherent advantage of automatic update, our protocol ensures the usage of optimal path for maximum number of times without creating any hotspots.

Following definitions and terms are used for describing Co-operative routing protocol -

Network lifetime- This is the time from the setup of the network till the first node dies. This is the worst-case scenario and we assume that the network is partitioned once its first node is drained out of energy. It is assumed that the sensor nodes have non-renewable source of energy.

Residual Energy (Re) - At any given instance of time, the energy present in a sensor node battery will be called its residual energy. It is a measure of node's health in terms of available energy at that instance. The node with a higher residual energy is able to perform more operations

Cluster- It is a collection of some number of nodes which form a network among them, as shown in Fig.6, and are responsible for sensing the desired events in their environment. They are controlled by a special node called the cluster head or more commonly the Gateway node. Under normal circumstances, this node demands data from the nodes in its cluster as and when required.

Local group - A Local group of a node is defined as the collection of nodes which lie inside the transmission range of that node. Thus we can say that a node can transmit and receive data to and from the nodes in its local group only.

Received energy (Er) - It is the intensity with which a signal from gateway is received by a node in its cluster. This is the most important parameter, which resolves the directivity issue in co-operative routing protocol.

NODE ID - For identification purpose, each node is assigned a unique number, which plays the role of an identifier for the node in the local group.

Following are important assumptions for this implementation.

(1) While designing Co-operative routing protocol, we have assumed that the Gateway node has renewable energy resources thereby has the power to perform unlimited number of operations.

(2)The nodes in the network are stationary between two setup phases.

(3)The transmission range of the Gateway node is large enough to cover the whole cluster.

(4) All the nodes are having equal and fixed transmission range.

(5) The initial residual energy of all the nodes in a cluster (except the gateway node) is assumed to be equal.

(6) Energy required to transmit over a constant range is constant.

(7) Each node is having two radios, one is the normal data transmission radio which operates at a higher bit rate and its operation consumes most of the energy spent in communication and the other is MAC radio which operates at a lower bit rate and its operation consumes very less energy as compared to the normal transmission radio.

The last assumption is very significant. The MAC coordinates channel assignment such that each node gets a locally unique channel for transmission, while the channels are globally reused. There is also a global broadcast channel that is used for common control messages and for waking up nodes.

Each node has two radio receivers, one of which runs at 100% duty cycle, but is at very low bit rate and consumes very little power. The second radio runs at very low duty cycle (~1%) and is switched on only when the node needs to receive or transmit data. This is a higher rate radio (~10kbps) and consumes more power.

To send data, the MAC layer sends a wake-up signal on the broadcast channel. The ID of the node to which it needs to send data is modulated with the wake-up. Access to the broadcast channel is CSMA/CA. On receiving this message, the node to which this is addressed powers on its main radio and communication begins. Since each node has a locally unique channel, there is no problem of collisions occurring during data transmission. Thus the MAC layer enables deep sleep of the nodes, which leads to substantial power savings.

Apart from channel allotment there is one more important function of this low power operated radio. Since this radio is on all the time, it is able to receive any kind of transmission done by a node whose local group contains this node. This inherent property has been used in the update phase of our protocol

The working of protocol is described in following phases

- Setup phase
- Communication phase
- Update phase

A. *Setup Phase*

The gateway node initiates this phase by sending an initiate signal, which is received by all the nodes in the cluster. The received strength of this signal is extracted from the channel and stored by every node as Er. After this a time slot is allotted to every node during which it sends a "hello" packet to all the nodes in its transmission range. This “hello” packet contains the Er and the Node ID. All the nodes, which receive this packet thereby, come to know that this node is in their local group (assumption 4). These nodes make a table and register the Er against the Node ID of all such transmissions heard by them. Apart from Er and node ID the initial residual energy is also stored for all the nodes in the local group. Thus the purpose of setup phase is to make all the nodes aware of their local group and as a result a table is formed in every node having the Er, Node ID and the initial residual energy of all the nodes in a particular node’s local group.

B. Communication Phase

After the setup phase, gateway sends a query signal, which has to be answered by the node having its ID in this signal. By the use of MAC radio, this node is identified and it starts sending the data towards the gateway. For making a routing decision, it just needs to decide the next node to which it will transmit the data. Now the issue of directivity becomes a problem. Based on the local knowledge, a node needs to decide the correct direction in which it has to send the data towards the gateway. Taking into consideration the Er of different nodes in its table solves this problem.

Intensity of an electromagnetic signal goes on decreasing with the distance from the source; a node having a higher value of Er will be closer to gateway than a node with lower value of Er. By scanning the table, a node can identify which node in its cluster is closest to the gateway. Thus with a fixed transmission range, selecting a node with highest value of Er will ensure an optimal hop. This process is followed at every node till the data finally reaches the gateway node. Thus we say that our algorithm is a hop programmable algorithm.

After a certain number of transmissions, the nodes falling in this path tend to die out sooner thereby creating hotspots along this optimal path. To avoid the creation of hotspots, we use the residual energy column of the table to calculate the average of the residual energies of all the nodes in the local group and check whether the node selected on the basis of maximum Er is having its Re. above the average or not. If this condition is satisfied, then the node is selected as the next hop else it is rejected. If the node is rejected, this condition is checked for the node with second highest value of Er and so on.

Different local groups tend to have the same average which is the cluster average. This is ensured by the fact that every node in the cluster is a part of many local groups. This tends to normalize all the group averages and hence the cluster average, resulting in the degradation of whole cluster in a very graceful manner.

C. Update Phase

This phase is not a different phase as far as its run time is concerned. It runs simultaneously with the data communication phase and provides automatic updating of the required parameter.

For making right decisions on rejecting an optimal hop, a node needs to calculate the latest average of residual energies of nodes in its table. As an example, for deducting node A’s Re, all the other nodes only need to know that A has transmitted or received. This knowledge can be had using the MAC radio. Whenever a node makes a data transmission, it will also send its ID and receiver’s ID on the low powered MAC radio. All the other nodes in its local group will hear this transmission on the MAC radio. All these nodes will then reduce a certain amount of energy from the transmitter and receiver’s residual energy. All these nodes in calculating the averages of their group’s average and in making their own routing decisions will use this newly updated energy. In this fashion, an automatic updating mechanism can be employed without caring for the extra overhead.

IX. MODELING AND SIMULATION

Lifetime of a network is defined as the time after which certain fraction of the network runs out of battery and therefore ceases to function properly, resulting in a failure in transmission of data. Recent advances in the embedded systems have managed to accommodate sensor nodes in such remote environments where refuelling them is out of scope. Even in accessible networks, replacing the used up nodes in due course of time is a much cheaper option as compared to replenishing the batteries. The apriori knowledge of the replacement time is therefore essential. The modelling of lifetime of such WSNs has therefore attained great importance. The reason for keen interest among the research fraternity as well as the commercial groups regarding the techniques to enhance the lifetime of such inaccessible sensor networks is obviously the same.

One part of the node that consumes a large share of the battery power present with the node is its transceiver. Apart from this, the data processing unit of the sensor node constitutes a big quota of consumed power. That is why the network lifetime calculations need to be based on both the routing protocol as well the data processing units. A lot of work in the UbiSens research project discussed by Saket Sakunia et al.[7] was aimed at inventing a routing protocol that would minimize the transceiver consumption. Lifetimes of WSNs have been studied earlier by Konstantinos et al.[8] and Gracanin et al.[9]. This paper focuses on modeling the network lifetime and further evaluating the efficiency of the Co-Operative Routin and diffusion routing with classification technique. We are assuming uniform distribution of the sensor nodes

A. Introduction to Ptolemy-II

Ptolemy II is the current software infrastructure of the Ptolemy Project. It is published freely in open-source form. Ptolemy II is the third generation of design software to emerge from UC Berkley. The Ptolemy-II is very helpful to study heterogeneous modeling, simulation, and design of concurrent systems as discussed by Y. Xiong et al.[10]

Most of the models of computation in Ptolemy II support actor-oriented design. This contrasts with object-oriented design by emphasizing concurrency and communication between components. Components called actors execute and communicate with other actors in a model. Like objects, actors have a well-defined component interface. This interface abstracts the internal state and behaviour of an actor, and restricts how an actor interacts with its environment. The interface includes ports that represent points of communication for an actor, and parameters that are used to configure the operation of an actor. Central to actor-oriented design are the communication channels that pass data from one port to another according to some messaging scheme. Whereas with object-oriented design, components interact primarily by transferring control through method calls, in actor-oriented design, they interact by sending messages through channels. The use of channels to mediate communication implies that actors interact only with the channels that they are connected to and not directly with other actors. The external interface consists of external ports and external parameters, which are distinct from the ports and parameters of the individual actors in the model. The external ports of a model can be connected by channels to other external ports of the model or to the ports of actors that compose the model. External parameters of a model can be used to determine the values of the parameters of actors inside the model.

B. *Visualsense*

VisualSense is a modeling and simulation framework for wireless and sensor networks that builds on and leverages Ptolemy II. Modeling of wireless networks require sophisticated modeling of communication channels, sensors, ad-hoc networking protocols, localization strategies, media access control protocols, energy consumption in sensor nodes, etc. This modeling framework as discussed by P. Baldwin et al.[11] is designed to support a component-based construction of such models. It supports actor-oriented definition of network nodes; wireless communication channels, physical media and wired subsystems. Custom nodes can be defined by sub-classing the base classes and defining the behavior in Java or by creating Composite models using any of several Ptolemy II modeling environments. Custom channels can be defined by sub-classing the Wireless Channel base class and by attaching functionality defined in Ptolemy II models.

X. IMPLEMENTATION OF COOPERATIVE ROUTING

This section is arranged as the description of the network topology, the implementation of setup phase, the communication phase and updating network.

A. *TOPOLOGY OF THE NETWORK*

Sensor Network with 50 nodes population is implemented in Ptolemy-II is shown in Fig. 4. Ptolemy-II plays an important role in the placement of the nodes. A randomizer has been used to set the locations of the nodes. This experiment is repeated for different seeds of randomization. The nodes are

thus not arranged in a manner to suite the algorithm but are

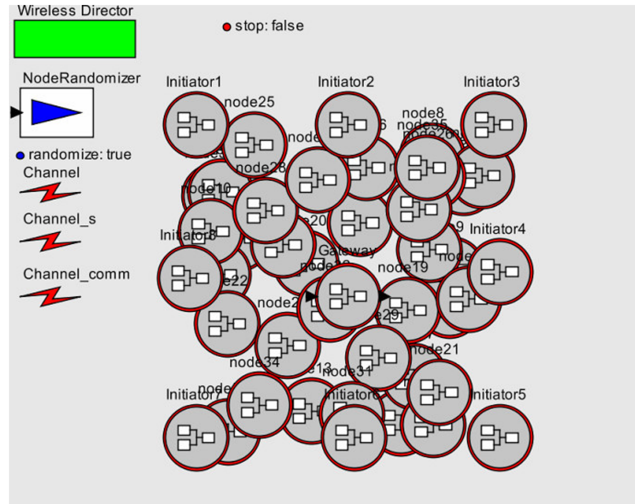


Figure 4. Pre-initialized Sensor Network

arranged illegibly. This plus point of Ptolemy-II ensures an unbiased comparison of algorithms.

The Gateway node is the central controller of the network. It is assumed as to be omni-powerful and it could transmit over its entire cluster. Again, there are no power constraints on the Gateway. It is the controller of the network in the sense that it controls the establishment of communication between the nodes. It is also the node to which the data is to be communicated. Therefore it is centrally placed to provide symmetrical access. Assuming a single sink also helps in checking the reliability of the routing algorithm. The gateway performs the important task differentiating the phases of operation of the network.

Another important feature of this implementation is that the entire process is assumed to be source initiated. This is deliberately done because the destination initiated processes are sparsely spaced in time. Again the routing mechanism is independent of the initiation of transmission, thus making the query initiated implementation trivial, though possible.

B. *IMPLEMENTING THE SETUP PHASE*

The complete internals of the sensor node implemented in Ptolemy is described in Fig. 5. It consists of different

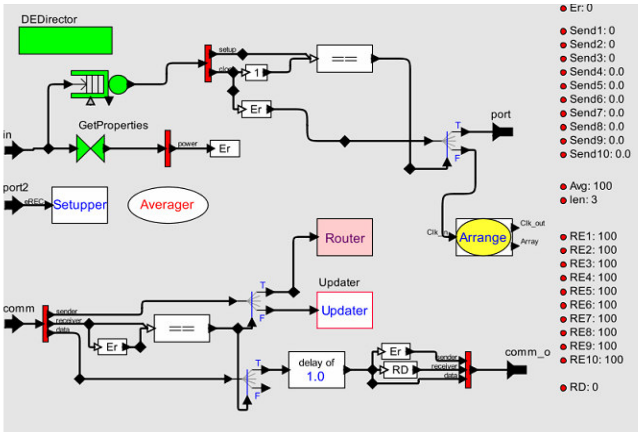


Figure 5. Internals of a Sensor Node

functional block implemented by using different Ptolemy actors - Setupper, Averager, Arranger, Router, and Updater.

The setup phase of the network, as described earlier, is the stage of network establishment. In real life situations, the gateway initiates this stage on getting a trigger from some external controller. But for simplicity of simulations the gateway initiates the setup at time 0. This is indeed true for real life systems as well, since the network lifetime starts with the network being recognized and formed.

The gateway initiates this phase by setting a global variable 'Setup' 1. The status of this variable is globally transmitted and then the nodes act accordingly. The gateway is the one to stop the phase as well. This it does by simply resetting the 'Setup' flag.

The user-defined actor called 'Setupper' performs the setup function in the nodes. When the setup flag is transmitted by the gateway, the nodes' receivers find out the received energy. The 'getProperties' actor is used to find the received energy. The received energy is stored in a variable called 'Er'. The nodes check if the 'Setup' flag is unity, and if true, transmit a packet containing their ID and Received Energy. If the setup period is going on, the receiving node disassembles the packet and stores the ID in its list of neighbours. This leads to a new definition of setup phase, which states that setup is the phase of discovery of neighbours.

In this algorithm, the routing is done through forwarding tables. Therefore we are interested only in the neighbors with higher directivity. The received energy is symbolic of directivity. So, during the setup itself we reject the neighboring nodes with lesser directivity. Not storing the ID previously saved is also of prime importance because replication of IDs leads to faulty routing. Therefore the 'Setupper'has to ensure a lot of selectivity. Equally important is to avoid overwriting the already existing links. The 'Setupper' takes precautions for this as well.

Another functionality that we have embedded into the setup phase is the arrangement of the IDs of a node's neighbors. This solves a lot many problems, foremost of which is the complexity of the scan during the actual routing process. For a stationary network, the setup phase occurs only once while the routing is everlasting, at least till the network dies out. Thus reducing the complexity of the route-time scan saves a lot of computation energy over a large time slice. The arrangement that is most suitable is the one where IDs are arranged in descending order of their directivities. The actor called 'Arranger' performs this job. The arranger is deliberately detached from the 'Setupper' so that it can be used anywhere to get a descending sequence.

The added function that the arranger does is to report the number of non-zero IDs. This function is of specific use during routing. Data is not transmitted to a non-existing node, and thereby making certain that there are no loop-holes in the routes established.

C. IMPLEMENTING THE COMMUNICATION PHASE

The setup phase ends when the gateway resets the 'Setup' flag to 0. This also marks the beginning of the communication phase. The communication phase implies that all the nodes have discovered the forward links and if the data arrives at any of these nodes, it can be easily routed to the gateway. After the completion of setup phase, the Poisson clock in the nodes is triggered. This is same as data generation. This helps in verifying the correct disposal packets.

Another actor called the 'Router', shown in Fig.5, performs the job of routing the packets. The router is the most important part of the design as it is the one taking decisions based on the proposed algorithm. As defined in the algorithm, the router has to find out the most cost effective link. Such link is the one, which is most directed as well as farthest from dying out. Computation of cost of the links is based on the Er of the node and the difference between the residual energy of the node and the average energy of all the forward nodes. Therefore the router simply has to select the node which has maximum directivity among the nodes with residual energy greater than the average energy. Now the significance of arranging the nodes in descending order of directivity becomes obvious. The router simply scans through the list or the forwarding table and selects the first node that is found to be above the average.

Another important block of the design is the 'Averager'. It finds the average of the residual energies of all the nodes in the forwarding table. The nodes with zero ID are not to be counted for calculating the average. The 'Averager' output is fed to the router enabling it to take a correct decision.

D. UPDATING THE NETWORKS

The controlling parameter in all the routing decisions that are being made through the proposed algorithm is the average energy of the forward paths in a particular local group. This energy has to be calculated often. For this purpose, the residual energies of the neighbours need to be known. This is done in the 'Update Phase'.

In most of the existing algorithms that are somehow interested in the residual energies of the updating of the remaining power is done through a specialized update cycle. In the update cycle, each node communicates to each of its neighbours, and informs them of its own residual energy. For a densely populated network, the number of such communications is very large. With such large number of communications just for the sake of updating, the energy lost in this mechanism is tremendously high. This simply increases the burden of the routing protocol. Again the delays involved in such an update are much higher for the proposed algorithm.

While developing the Co-Operative Routing protocol, we focused on improving the methodology of updating as well. An important arrangement in improving the lifetime of the network was obtained by making use of the low-power MAC radio for updating the energy of node. As described earlier, we make use of two radios in the sensor node. The communication radio is the one consuming major share of power and therefore is maintained at a very low duty cycle. The other radio is the MAC radio, which is ultra-Low, powered as compared to the previous one. This radio is kept continuously ON to take Medium Access Control decisions. To take the MAC decisions polling for the channel occurs. So every node in the vicinity knows the ID of the node, which presently holds the channel. Thus it becomes clear that the MAC is indicative of the nodes involved in transmission. The energy of a node can be updated only if it is transmitting. So the function of detecting whether a node is transmitting or not is handed to the MAC radio, and with this knowledge, the residual energy of that node, which is assumed at a certain reasonable value during the initialization, is decremented by a certain amount to account for the energy usage during transmission. In most cases, the above-mentioned approximation simply proves to be an over-precautious way of updating energies and therefore is rather helpful in improving the real lifetime of the network. This functionality is implemented in the above model using an actor called ‘Updater’ as shown in Fig. 5.

XI. DIRECTED DIFFUSION ROUTING

The routing is the most important operations in a Wireless Sensor Network. The energy for communication exceeds the energy for computation. Typically the power required[11] for single instruction is about 10-20 pWatt while the energy for transmission is about 10-20 nJ/bit. A good routing protocol ensures lesser number of transmissions and enhanced lifetime of the network.

Data generated by sensor nodes is named by attribute-value pairs. A node requests data by sending interests for named data. Data matching the interest is then “drawn” down toward that node as depicted in Figure 6. (Intermediate nodes can cache, or transform data and may direct interests based on previously cached data. The human operator’s query would be transformed into an interest that is diffused (e.g., broadcasted, geographically routed) toward nodes in region When a node in that region receives an interest, it activates its sensors which begin collecting information about events. When the sensors report the occurrence of events, this information returns along the reverse path of interest propagation. Intermediate nodes might aggregate the data, e.g., more accurately pinpoint the exact event by combining reports from several sensors. An important feature of directed diffusion is that interest and data propagation and aggregation are determined by localized interactions (message exchanges between neighbours or nodes within some vicinity).

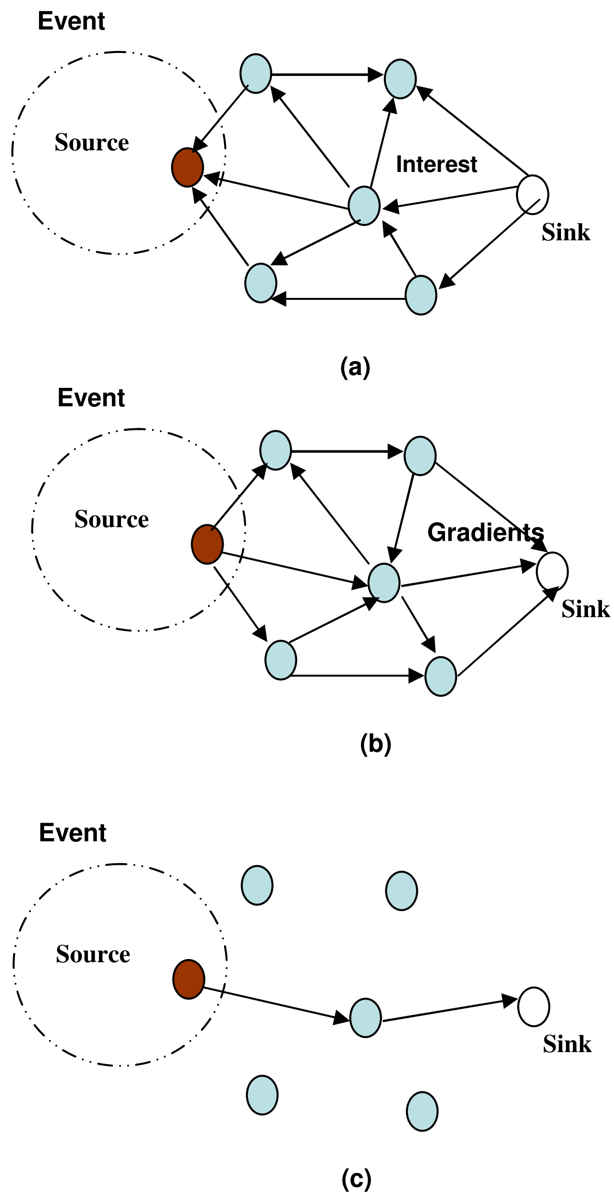


Figure 6: Simplified schematic for Directed Diffusion. (a) Interest Propagation. (b) Initial Gradients Setup. (c) Data Delivery along Reinforced Path.

Directed diffusion is discussed by Chalermek Intanagonwiwat et. al,[12] is significantly different from IP-style communication where nodes are identified by their end-points and inter-node communication is layered on an end-to-end delivery service provided within the network by using directed diffusion, one can realize robust multipath delivery, empirically adapt to a small subset of network paths, and achieve significant energy savings when intermediate nodes aggregate responses to queries.

Directed diffusion consists of several elements: interests, data messages, gradients, and reinforcements. An interest message is a query or an interrogation that specifies what a user wants. Each interest contains a description of a sensing task that is supported by a sensor network for acquiring data. Typically, data in sensor networks is the collected or processed information of a physical phenomenon. Such data can be an event, which is a short description of the sensed phenomenon. In directed diffusion, data is named using attribute-value pairs. A sensing task (or a subtask thereof) is disseminated throughout the sensor network as an interest for named data. This dissemination sets up gradients within the network designed to "draw" events (i.e., data matching the interest). Specifically, a gradient is direction state created in each node that receives an interest. The gradient direction is set toward the neighbouring node from which the interest is received. Events start flowing toward the originators of interests along multiple gradient paths. The sensor network reinforces one or a small number of these paths

XII. RESULT AND ANALYSIS

The comparison Table-1 presents the network lifetime of the proposed cooperative routing algorithm in comparison with the commonly used Directed-Diffusion algorithm. The comparisons have been made for same number of nodes. The difference in all the sets of simulation is the seed used for randomization. The seeds for randomization determine the network topology. The centralized clock in the Gateway node that measures the time right from setup phase till the first node dies, that calculates the lifetime. Classification technique with ART1 neural net work model is implemented with both the routing algorithms and life time is counted by the central clock in the gateway node. Table-2 & 3 shows the comparison of life time with and without classification techniques.

It is clearly evident from Table-1 that there is an appreciable improvement in the network lifetime. The improvement varies according to the network topology. It is clearly visible that the improvement in lifetime is consistently around 31%. The maximum network lifetime improvement is found to be 37%

Table 2 describes the Lifetime of Network for Diffusion Routing With and Without Classification. Here with classification the network life time is improved by around 41%.

Table 3: describes the Lifetime of Network for Co-operative Routing With and Without Classification. Here with classification the network life is improved by around 45%. When the network lifetime for cooperative routing with classification is compared with diffusion routing without classification, the life time count is significantly higher at about 97%.

Table 1: Lifetime of Sensor Network with two different routing algorithms

Number of Nodes	Life time with Co-Operative Routing	Life time with Diffusion Routing	Performance Improvement (%)
50	172.99	132.83	30.23
50	165.40	120.63	37.11
50	185.40	137.04	33.34
50	188.62	126.00	27.44
50	170.51	128.12	33.08
50	166.25	129.34	28.53

Table 2: Lifetime of Network for Diffusion Routing With and Without Classification

Number of Nodes	Life time with Diffusion Routing		Performance Improvement (%)
	Without Classification	With Classification	
50	132.83	188.92	42.23%
50	120.63	170.36	41.23%
50	137.04	196.28	43.23%
50	126.03	177.94	41.23%
50	128.12	181.44	41.62%
50	129.34	186.75	44.39%

Table 3: Lifetime of Network for Co-operative Routing With and Without Classification

Number of Nodes	Life time with Co-Operative Routing		Perform-ance Improv-ement (%)	Perform-ance Improv-ement (%)With Ref. to Diffusion Routing
	Without Classifi - cation	With Classifi-cation		
50	172.99	251.38	45.32%	89.26%
50	165.4	240.35	45.32%	99.25%
50	185.4	272.96	47.23%	99.19%
50	188.62	270.51	43.42%	114.70%
50	170.51	246.59	44.62%	92.47%
50	166.25	245.75	47.82%	90.00%

XIII. CONCLUSION

The sensor network is populated with 50 nodes. Communication over the network is carried out by cooperative routing in one case and with diffusion routing in other case. Table.1. shows the count of lifetime of the network. The comparisons have been made for same number of nodes. The difference in all the sets of simulation is the seed used for randomization. The seeds for randomization determine the network topology. The central clock in the Gateway node that measures the time right from setup phase till the first node dies, that calculates the lifetime. The time is shown as unit scale of clock.

The sensor network involves redundancy in data generation and communication, which can be efficiently deled by classification technique. Sensor data is classified at each node and then the classified data is communicated further. This effectively improves the bandwidth of the communication channel and there by reduces the energy consumption.

Two cases of routing are implemented one with cooperative routing and other with diffusion routing. The proposed scheme is modelled and simulated using Ptolemy-II : Visual sense. Ptolemy permits the interfacing of MATLAB within its actors. Visual sense is the framework of Ptolemy-II for wireless sensor network modelling. Classification techniques are implemented using ART1 model in MATLAB. The routing takes place in multiple hops and all the nodes takes part in communication. It achieves uniform dissipation of energy for all the nodes. The concept of use of two Radio – MAC radio and separate radio for data routing has also contributed to the energy saving. The use of two separate radios may seem to add to the cost of the node, but recent development in VLSI implementation can facilitate the implementation at low cost. The life time of the network is improved by 45% with cooperative routing and about 31% with diffusion routing, because of classification of sensor data when tested with 50 nodes sensor network.

REFERENCES:

[1] A Wendi B. Heinzelman, Member IEEE, Ananth P. Chandrakasan, Senior Member, IEEE, and Hari Balakrishnan, Member, IEEE, An Application –Specific protocol Architecture for Wireless Microsensor Networks, IEEE Transaction on Wireless Communications, Vol.1 No. 4 October 2002

[2] Wendi Rabiner Heinzelman, Ananth Chandrakas and Hari Balakrishnan, "Energy-efficient communication protocols for wireless microsensor networks,"in Proceedings of the Hawaii International Conference on Systems Sciences, Jan 2000.

[3] Neural networks Algorithms, applications, and programming techniques ,J.A.Freeman, David M. Skapura, Pearson Education Asia, 2001.

[4] Sudhir G. Akojwar, R.M.Patrikar ,Real Time Classifier For Industrial Wireless Sensor Network Using Neural Networks with Wavelet Preprocessors, IEEE International Conference on Industrial Technology (ICIT 2006), pp 512-517, DEC 15-17 2006, Mumbai.

[5] Teresa Serrano-Gotarrendona and Bernabe Linares-barranco, An ART1 Microchip and its use in Multi-ART1 systems, IEEE Transaction on Neural Networks, VOL 8, No. 5, Sep 1997.

[6] Gail A. Carpenter and Stephen Grossberg, A Self-Organising Neural Network For Supervised Learning, Recognition, and Prediction, IEEE Communications Magazine, Sep 1992

[7] Saket Sakunia, Shantanu Bhalerao, Abhishek Chaudhary, Mukund Jyotishi, Mandar Dixit, Raghav Jumade, Rajendra Patrikar, "UbiSens: Achieving Low Power Wireless Sensor Nodes," IEEE International Conference on Wireless and Optical Communications Networks, (WOCN 2007)

[8] Konstantinos kalpakis, Koustuv Dasgupta and Parag Namjoshi, ,"Efficient algorithm for maximum lifetime data gathering and aggregation in wireless sensor networks,"Computer Networks Volume 42, Issue 6, 21 August 2003, pp.697-716.

[9] Gracanin, D. Eltoweissy, M. Olariu, S. Wadaa, "On modelling wireless sensor networks," A. Parallel and Distributed Processing Symposium, 2004.

[10] Y. Xiong, "An Extensible Type System for Component-Based Design," Technical memorandum UCB/ERL Mo2/13, University of California, Berkely,CA94720,May1,2002.

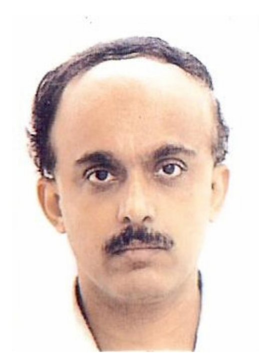
[11] P. Baldwin, S. Kohli, E. A. Lee, X. Liu, and Y. Zhao, "Modelling of Sensor Nets in Ptolemy-II," In Proc. of Information Processing in Sensor Networks, (IPSN), April 26-27, 2004

[12] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, Fabio Silva,"Directed Diffusion for Wireless Sensor Networking" IEEE/ACM transaction on networking, Vol.-11, No.-1, February 2003

Sudhir G. Akojwar. Born on 27.10.1968 in Chandrapur, India. He did Polytechnique in electrical engineering in 1987, then Bachelor degree in Engineering in 1990 and M.Tech in 1994 from Regional Engineering College (now NIT) Warangal, India.

In 1990 he joined as lecturer in the department of electronics at Rajiv Gandhi College of Engineering Research and Technology, Chandrapur, India. Currently he is Assistant Professor in the same college.

He is Ph.D. candidate at VNIT, Nagpur, India. He published about 10 papers in national and international conferences. His research interest areas are wireless sensor networks, embedded systems, neural networks and microcontroller based system design. He is life member of ISTE (India), IETE(India), IE(India). He is member of IEEE and IEEE computer society since 2000.



Rajendra M. Patrikar. Born on 14.09.1960, in Amravati, India. After completing his M. Sc. In Physics from Nagpur Univrsity he did M.Tech in Electrical Engineering from IIT Bombay in 1985 and Ph.D. from IIT Bombay in 1992.

Currently he is Professor in Electronics and Computer Science Department VNIT, Nagpur,India. He has been working in the area of VLSI Design, nanotechnology and electronics systems. He Joined as Research engineer in the Microelectronics project at IIT Bombay. He joined as faculty at IIT Bombay after working for a year at Computervision R&D Pune. Later he moved to Singapore to work in TECH Semiconductor in Advance Device Technology Department. After working there for three years he moved to Institute of High Performance Computing Singapore where he carried research work in the area of CAD for VLSI and nanoelectronics. He has also worked as Team Lead for Nanotechnology group at Computational Research laboratory, Pune.

Prof. Patrikar has published about 50 papers in International Journals, international conferences and also filed four patents. He is a fellow of IETE and Member of IEEE since 1996 and Senior Member of IEEE since 2004.