COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

COCHIN – 682022

2011

Seminar Report

On

Computational Intelligence in Wireless Sensor Networks

Submitted By

Sruthy Suresh

In partial fulfilment of the requirement for the award of

Degree of Master of Technology (M.Tech)

In

Computer and Information Science

ABSTRACT

Wireless sensor networks (WSNs) are networks of distributed autonomous devices that can sense or monitor physical or environmental conditions cooperatively. WSNs face many challenges, mainly caused by communication failures, storage and computational constraints and limited power supply. Paradigms of Computational Intelligence (CI) have been successfully used in recent years to address various challenges such as optimal deployment, data aggregation and fusion, energy aware routing, task scheduling, security, and localization.

CI provides adaptive mechanisms that exhibit intelligent behaviour in complex and dynamic environments like WSNs. CI brings about flexibility, autonomous behaviour, and robustness against topology changes, communication failures and scenario changes. However, WSN developers can make use of potential CI algorithms to overcome the challenges in Wireless Sensor Network. The seminar includes some of the WSN challenges and their solutions using CI paradigms.

Keywords- Clustering, computational intelligence, data aggregation, deployment, design, localization, routing, scheduling, security, wireless sensor networks

Contents

1. INTRODUCTION

A Wireless sensor network is a network of distributed autonomous devices that can sense or monitor physical or environmental conditions cooperatively. WSNs are used in numerous applications such as environmental monitoring, habitat monitoring, prediction and detection of natural calamities, medical monitoring and structural health monitoring. WSNs consist of a large number of small, inexpensive, disposable and autonomous sensor nodes that are generally deployed in an ad hoc manner in vast geographical areas for remote operations. Sensor nodes are severely constrained in terms of storage resources, computational capabilities, communication bandwidth and power supply.

Typically, sensor nodes are grouped in clusters, and each cluster has a node that acts as the cluster head. All nodes forward their sensor data to the cluster head, which in turn routes it to a specialized node called sink node (or base station) through a multi-hop wireless communication. However, very often the sensor network is rather small and consists of a single cluster with a single base station .Other scenarios such as multiple base stations or mobile nodes are also possible. Resource constraints and dynamic topology pose technical challenges in network discovery, network control and routing, collaborative information processing, querying, and tasking . CI combines elements of learning, adaptation, evolution and fuzzy logic to create intelligent machines. In addition to paradigms like neuro-computing, reinforcement learning, evolutionary computing and fuzzy computing, CI encompasses techniques that use swarm intelligence, artificial immune systems and hybrids of two or more of the above.

Paradigms of CI have found practical applications in areas such as product design, robotics, intelligent control, biometrics and sensor networks. Researchers have successfully used CI techniques to address many challenges in WSNs. However, various research communities are developing these applications concurrently, and a single overview thereof does not exist. Their aim is to bridge the gap between CI approaches and applications, which provide the WSN researchers with new ideas and incentives. A discussion on yet-unexplored challenges in WSNs, and a projection on potential CI applications in WSN are presented with an objective of encouraging researchers to use CI techniques in WSN applications.

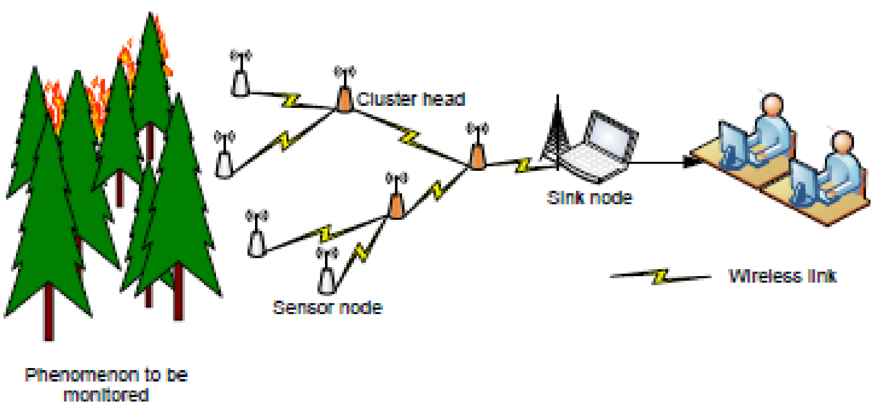

Figure 1.1 Architecture of a Typical WSN

2. CHALLENGES IN WIRELESS SENSOR NETWORKS

Real deployments of Wireless Sensor Networks (WSN) usually implement one of the three general applications:

- Periodic reporting
- Event detection
- Database-like storage.

Periodic reporting is by far the most used and the simplest application scenario, in which at regular intervals the sensors sample their environment, store the sensory data, and send it to the base station(s). Actuators such as automatic irrigation systems and alarm systems are often connected with such WSNs. This scenario is used in most monitoring applications in agriculture, microclimate and habitat surveillance, military operations , and disaster relief . The main property of periodic reporting applications is the predictability of the data traffic and volume.

In Event detection applications nodes sense the environment and evaluate the data immediately for its usefulness. If useful data (an event) is detected, the data is transmitted to the base station(s). The data traffic can hardly be predicted: events usually occur randomly and the resulting data traffic is sporadic. However, a small amount of data has to be exchanged for route management and aliveness checks even when no events are detected.

Database-like storage systems are similar to event-based systems. All sensory data (regular sampling or events) is stored locally on the nodes. Base stations search for interesting data and retrieve it from the nodes directly. The main challenge in these applications is to store the data in a smart way for fast search and retrieval.

2.1 PROPERTIES OF WSN DEPLOYMENTS

The properties of WSN can be summarized as follows:

Wireless ad hoc nature: A fixed communication infrastructure does not exist. The shared wireless medium places additional restrictions on the communication between the nodes and poses new problems like unreliable and asymmetric links. But, it provides the broadcast advantage: A packet transmitted by a node to another is received by all neighbors of the transmitting node.

Mobility and topology changes: WSNs may involve dynamic scenarios. New nodes may join the network, and the existing nodes may either move through the network or out of it. Nodes may cease to function, and surviving nodes may go in or out of transmission radii of other nodes. WSN applications have to be robust against node failure and dynamic topology.

Energy limitations: Nodes in most WSNs have limited energy. The basic scenario includes a topology of sensor nodes, and a limited number of more powerful base stations. Maintenance or recharging of the batteries on sensor nodes is not possible after deployment. Communication tasks consume maximum power available to sensor nodes, and in order to ensure sustained long-term sensing operation, communication tasks need to be exercised frugally.

Physical distribution: Each node in a WSN is an autonomous computational unit that communicates with its neighbours via messages. Data is distributed throughout the nodes in the network and can be gathered at a central station only with high communication costs. Consequently, algorithms that require global information from the entire network become very expensive. Thus, reticent distributed algorithms are highly desirable.

2.2 MAJOR CHALLENGES

A brief description of the major WSN challenges addressed by CI techniques is presented in the following subsections:

A. Design and Deployment

WSNs are used in vastly diversified applications ranging from monitoring a biological system through tissue implanted sensors to monitoring forest fire through air-dropped sensors. In some applications, the sensor nodes need to be placed accurately at predetermined locations, whereas in others, such positioning is unnecessary or impractical. Sensor network design aims at determining the type, amount and location of sensor nodes to be placed in an environment in order to get a complete knowledge of its functioning conditions.

B. Localization

Node localization refers to creating location awareness in all deployed sensor nodes. Location information is used to detect and record events, or to route packets using geometric-aware routing. Besides, location itself is often the data that needs to be sensed. Localization methods that use time-of-arrival of signals from multiple base stations are commonly used in WSNs.

C. Data Aggregation and Sensor Fusion

Sensor fusion is the process of combining of the data derived from multiple sources such that either the resulting information is in some sense better than would be possible with individual sources, or the communication overhead of sending individual sensor readings to the base station is reduced. Due to large-scale deployment of sensors, voluminous data is generated, efficient collection of which is a critical issue. Most widely used non-CI methods for sensor fusion include Kalman filter, Bayesian networks and Dempster-Shafer method.

D. Energy Aware Routing and Clustering

Economic usage of energy is important in WSNs because replacing or recharging the batteries on the nodes may be impractical, expensive or dangerous. In many applications, network life expectancy of a few months or years is desired. Routing refers to determining a path for a message from a source node to a destination node. In proactive routing methods, routing tables are created and stored regardless of when the routes are used. In reactive routing methods, routes are computed as necessary. In densely deployed networks, routing tables take a huge amount of memory, and therefore, hybrids of proactive and reactive methods are suitable for such networks. Another possible solution is to cluster the network into hierarchies.

E. Scheduling

In order to conserve energy, typical sensor nodes remain in sleep mode most of the time, and go into active mode periodically in order to acquire and transmit sensory data. A strict schedule needs to be followed regarding when a node should wake up, sense, transmit (or perform locomotion), ensuring maximum network lifetime. Causing the WSN nodes to take right actions at right time is the major objective of WSN scheduling.

F. Security

Wireless links in WSNs are susceptible to eavesdropping, impersonating, message distorting etc. Poorly protected nodes that move into hostile environments can be easily compromised. Administration becomes more difficult due to dynamic topology.

G. Quality of Service Management

QoS is an overused term that has various meanings and perspectives. QoS generally refers to the quality as perceived by the user/application, while in the networking community, QoS is accepted as a measure of the service quality that the network offers to the applications/users. QoS refers to an assurance by the network to provide a set of measurable service attributes to the end-to-end users/applications in terms of fairness, delay, jitter, available bandwidth, and packet loss. A network has to provide the QoS while maximizing network resource utilization. To achieve this goal, the network is required to analyze the application requirements and deploy various network QoS mechanisms.

3. PARADIGMS OF COMPUTATIONAL INTELLIGENCE

Computational Intelligence (CI) is the study of adaptive mechanisms that enable or facilitate intelligent behaviour in complex and changing environments. These mechanisms include paradigms that exhibit an ability to learn or adapt to new situations, to generalize, abstract, discover and associate. CI is also defined as the computational models and tools of Intelligence capable of inputting raw numerical sensory data directly, processing them by exploiting the representational parallelism and pipelining the problem, generating reliable and timely responses and withstanding high fault tolerance.

Paradigms of CI are designed to model the aspects of biological intelligence. CI encompasses paradigms such as neural networks, reinforcement learning, swarm intelligence, evolutionary algorithms, fuzzy logic and artificial immune systems. These paradigms are briefly introduced in the following subsections.

3.1 NEURAL NETWORKS

The human brain, which possesses an extraordinary ability to learn, memorize and generalize, is a dense network of over 10 billion neurons, each connected on average to about 10,000 other neurons. The three basic components of an artificial neuron In Figure are:

1) The links that provide weights Wji, to the n inputs of j^{th} neuron xi, i = 1, . . . , n;
2) An aggregation function that sums the weighted inputs to compute the input to the activation function uj = $\Theta_j + \sum_{i=1}^{n} x_i W_{ji},$, where Θj is the bias, which is a numerical value associated with the neuron.
3) An activation function Ψ that maps uj to vj = Ψ(uj),the output value of the neuron. Some examples of the activation functions are: step, sigmoid, tan hyperbolic and Gaussian function.

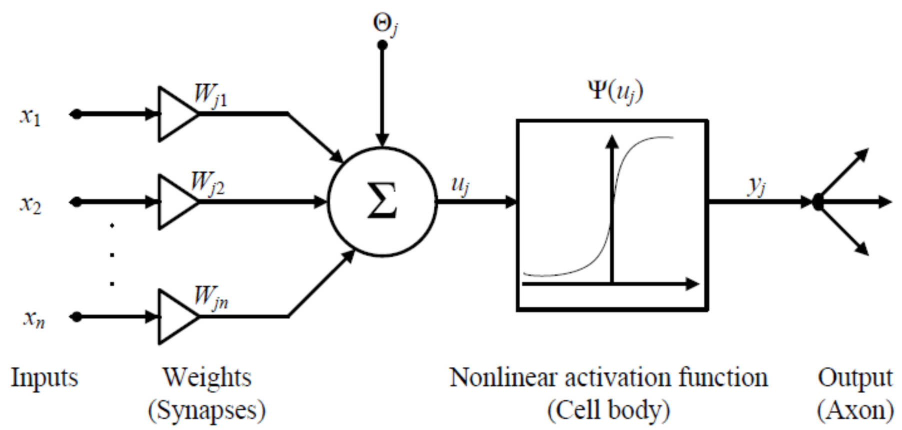

Figure 3.1 Structure of a Neuron

A NN consists of a network of neurons organized in input, hidden and output layers. In feed forward NNs, the outputs of a layer are connected as the inputs to the next layer while in recurrent networks, feedback connections are allowed.

3.2. FUZZY LOGIC

Classical set theory allows elements to be either included in a set or not. This is in contrast with human reasoning, which includes a measure of imprecision or uncertainty, which is marked by the use of linguistic variables such as most, many, frequently, seldom etc. This approximate reasoning is modelled by fuzzy logic, which is a multivalued logic that allows intermediate values to be defined between conventional threshold values. Fuzzy systems allow the use of fuzzy sets to draw conclusions and to make decisions. Fuzzy sets differ from classical sets in that they allow an object to be a partial member of a set. For example, a person may be a member of the set tall to a degree of 0.8 . In fuzzy systems, the dynamic behaviour of a system is characterized by a set of linguistic fuzzy rules based on the knowledge of a human expert.

Fuzzy rules are of the general form:

if antecedent(s)

then consequent(s), where antecedents and consequents are propositions containing linguistic variables. Antecedents of a fuzzy rule form a combination of fuzzy sets through the use of logic operations. Thus, fuzzy sets and fuzzy rules together form the knowledge base of a rule-based inference system as shown in Figure.

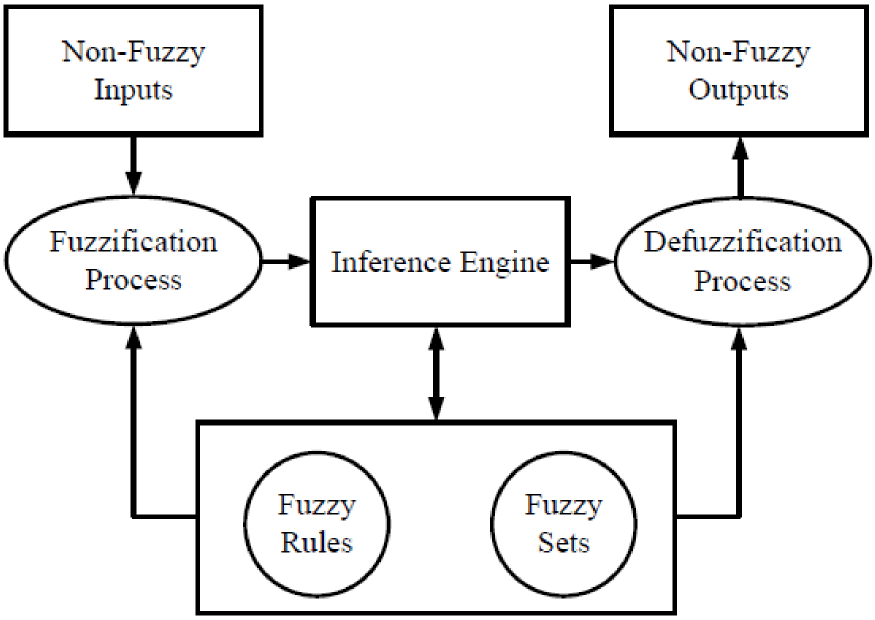

Figure 3.2 Fuzzy Inference Engine

3.3. EVOLUTIONARY ALGORITHMS

Evolutionary algorithms model the natural evolution, which is the process of adaptation with the aim of improving survival capabilities through processes such as natural selection, survival-of-the-fittest, reproduction, mutation, competition and symbiosis. EC encompasses a variety of EAs that share a common underlying idea of survival-of-the-fittest. EAs use a population of solution candidates called chromosomes. Chromosomes are composed of genes, which represent a distinct characteristic. A fitness function, which the EA seeks to maximize over the generations, quantifies the fitness of an individual chromosome. Process of reproduction is used to mix characteristics of two or more chromosomes (called parents) into the new ones (called offspring). Offspring chromosomes are mutated through small, random genetic changes in order to increase diversity. Some fittest chromosomes are selected to go into the next generation, and the rest are eliminated. The process is repeated generation after generation until either a fit-enough solution is found or a previously set computational limit is reached.

Following are the major classes of EAs.

- Genetic algorithms (GA), which model genetic evolution
- Genetic programming whose individual chromosomes are computer programs
- Evolutionary programming which model adaptive behaviour in evolution
- Evolutionary strategies which model strategy parameters that control variation in evolution
- Differential evolution which is identical to GA except for the reproduction mechanism
- Cultural evolution which models the evolution of culture of a population and culture's influence on genetic and adaptive evolution of individuals
- Co evolution in which initially "dumb" individuals evolve through cooperation or competition and become fit enough to survive

Successful applications of EA include planning, design, control, classification and clustering, time series modelling, music composing etc.

3.4 SWARM INTELLIGENCE

Swarm Intelligence (SI) originated from the study of collective behavior of societies of biological species such as flocks of birds, shoals of fish and colonies of ants. SI is the property of a system whereby collective behaviors of unsophisticated agents interacting locally with their environment cause coherent functional global patterns to emerge. While graceful but unpredictable bird-flock choreography inspired the development of particle swarm optimization , impressive ability of a colony of ants to find shortest path to their food inspired the development of ant colony optimization . The honey bee algorithm mimics foraging behavior of swarms of honey bees

.

Particle Swarm Optimization: The basic PSO consists of a population (or a swarm) of s particles, each of which represents a candidate solution. The particles explore an n-dimensional space in search of the global solution, where n represents the number of parameters to be optimized. Each particle i occupies position xid and moves with a velocity vid, $1 \le i \le s$ and $1 \le d \le n$. Particles are initially assigned random positions and velocities within fixed boundaries, i.e.,

$x_{min} \le x_{id} \le x_{max}$ and $v_{min} \le v_{id} \le v_{max}$ (in most cases $v_{min} = -v_{max}$).

Fitness of a particle is determined from its position. The fitness is defined in such a way that a particle closer to the solution has higher fitness value than a particle that is far away. In each iteration, velocities and positions of all particles are updated to persuade them to achieve better fitness. The process of updating is repeated iteratively either until a particle reaches the global solution within permissible tolerance limits, or until a sufficiently large number of iterations is reached. Magnitude and direction of movement of a particle is influenced by its previous velocity, its experience and the knowledge it acquires from the swarm through social interaction.

In the gbest version of PSO, each particle has a memory to store $pbest_{id}$, the position where it had the highest fitness. Besides, each particle can access the position $gbest_d$, the position of the particle having the maximum fitness. The gbest particle represents the best

solution found as yet. At each iteration k, PSO adds velocity v_{id} to each position xid and steers the particles towards its $pbest_{id}$ and $gbest_d$ using (1) and (2).

$v_{id}(k + 1) = w \cdot v_{id}(k) + c1 \cdot rand1 \cdot (pbest_{id} - x_{id}) + c2 \cdot rand2 \cdot (gbest_d - x_{id})$-------- (1)
$x_{id}(k + 1) = x_{id}(k) + v_{id}(k + 1)$ ....(2)

Here, rand1 and rand2 are random numbers having uniform distribution in the range (0, 1). The velocity update equation (1) shows that a particle is influenced by 3 components of acceleration. The first term involves the inertia coefficient w, $0.2 < w < 1.2$, and it denotes the inertia of the particle . The second term involves the cognitive acceleration constant c1. This component propels the particle towards the position where it had the highest fitness. The third term involves the social acceleration constant c2. This component steers the particle towards the particle that currently has the highest fitness. The velocity of a particle is bounded between properly chosen limits v_{max} and v_{min}. Similarly, the position of a particle is restricted between properly chosen constants x_{max} and x_{min}. Several variants of PSO have been devised and applied to optimization problems in power systems, stock markets, antenna control and WSNs.

Ant Colony Optimization: ACO was introduced as a metaheuristic for solving combinatorial optimization problems. Foraging ants initially explore surroundings of their nest in a random manner. When an ant finds a source of food, it evaluates quantity and quality of the food and carries some food to the nest. While returning to the nest, the ant deposits a trail of chemical pheromone, which guides other ants to the food source. This characteristic of ant colonies is exploited in artificial ant colonies to solve combinatorial optimization problems. Consider that two paths A and B exist between a nest and a food source, and nA(t) and nB(t) number of ants use them at time step t respectively, then the probability of ant choosing path A at the time step t + 1 is given by

$$P_A(t+1) = \frac{(c+n_A(t))^\alpha}{(c+n_A(t))^\alpha + (c+n_B(t))^\alpha} = 1 - P_B(t+1)$$

where c is the degree of attraction of an unexplored branch, and α is the bias to using pheromone deposits in the decision process.

An ant chooses between the path A or path B using the decision rule: if $U(0, 1) \leq PA(t+1)$ then choose path A otherwise choose path B. The main idea of the ACO metaheuristic is to model the problem as a search for the best path in a “construction graph" that represents the states of the problem. Artificial ants walk through this graph, looking for good paths. They communicate by laying pheromone trails on edges of the graph, and they choose their path with respect to probabilities that depend on the amount of pheromone previously left.

4. CI BASED SOLUTIONS FOR WSN CHALLENGES

In this section we can see how Computational Intelligence techniques can be used as solutions for wireless sensor network challenges. CI techniques can be very helpful in the process of designing and planning the deployments of sensor networks and there have been many efforts to apply them in this context.

4.1 EVOLUTIONARY ALGORITHM FOR NETWORK DESIGN

A decision support system based on Genetic Algorithm is proposed. The decision support system is meant for the use of a process engineer who interacts with it to determine optimal sensor network design as outlined in Figure.

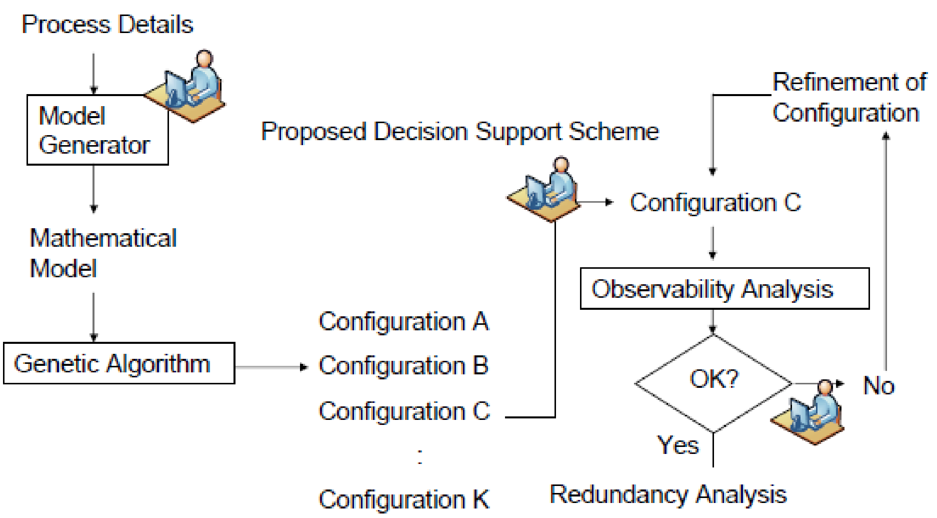

Figure 4.1 Role of GA in Decision Support System

The process engineer introduces information about the process under stationary operating conditions through a rigorous module called model generator, which involves linear functionalities and non-linear equations. Then the engineer chooses the initial sensor network from the candidates. The chosen configuration undergoes observability analysis which determines which of the unmeasured variables are observable. Based on the results, the engineer decides whether the information achieved from the configuration is satisfactory or not. If not, the sensor network needs to be improved by adding more instruments before the analysis is repeated. Traditionally, the analysis begins from an initial sensor network with a few instruments chosen by the process engineer based on his/her skills and experience. Then, an iterative process takes place; the process engineer runs the tools in the decision support system and refines the configuration until satisfactory instrumentation is achieved. In the case of complex industrial processes, a complete iteration involves a great deal of expert examination and several executions of the analysis software tools. Therefore, a good initial configuration assures lesser number of iterations before the final configuration is reached.

Each chromosome used in GA is a sequence of length l which represents number of variables in the mathematical model of the network, like in 10001100. Here, a 1 represents the presence of a sensor to measure the variable at that position in the mathematical model, and l is the number of variables in the mathematical model. GA uses binary tournament to select individuals to go to next generation maintaining the best up-to-the-moment individual with the elitist approach.

The fitness function, which GA seeks to maximize, is defined as

$$f(i) = N_R(i) + N_{obs}(i) + 1 - N_C(i)$$

where $N_R(i)$, $N_{obs}(i)$ and $N_C(i)$ are the normalized values corresponding to the reliability, observability and cost terms, respectively.

The proposed scheme is tested on a ammonia synthesis plant, where all the compositions and critical variables were set as measured. The performance of an initial sensor configuration suggested by the GA reported to be more cost effective and more reliable than the initial configuration determined by the process engineer. The initial configuration suggested by GA is reported to have achieved a cost saving of 60%, and a smaller amount of time to complete the whole observability analysis.

4.2 FUZZY LOGIC FOR DEPLOYMENT

This technique assumes that the area to be monitored by a sensor network is divided into a square grid of subareas, each having its own terrain profile and a level of required surveillance (therefore, its own path loss model and required path loss threshold). The proposed technique uses fuzzy logic to determine the number of sensors n(i) necessary to be scattered in a subarea i.

For a sub-area i, path loss PL (i) and threshold path loss PL_{TH} are normalized on a scale 0 to 10, then divided into overlapping membership functions low, medium and high. This requires $3^2 = 9$ rules. The system computes an output weight (i), from which the number of sensors is determined as

$$n(i) = \frac{weight(i)}{\sum_j weight(j)}$$

Membership Functions are defined to form three Fuzzy sets Low, Medium and High depending on the values of PL and PL_{TH}

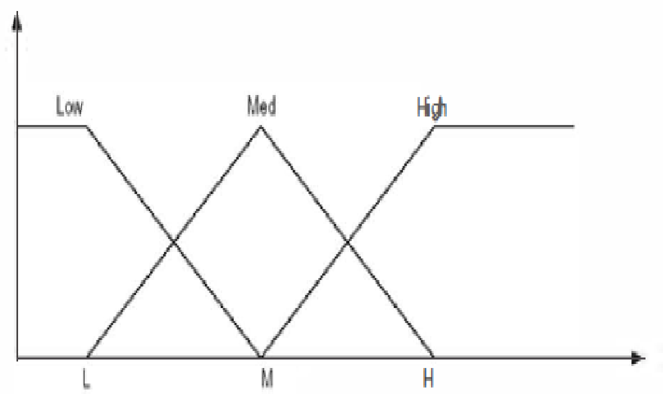

Figure 4.2 Membership Functions for PL and PL_{TH}

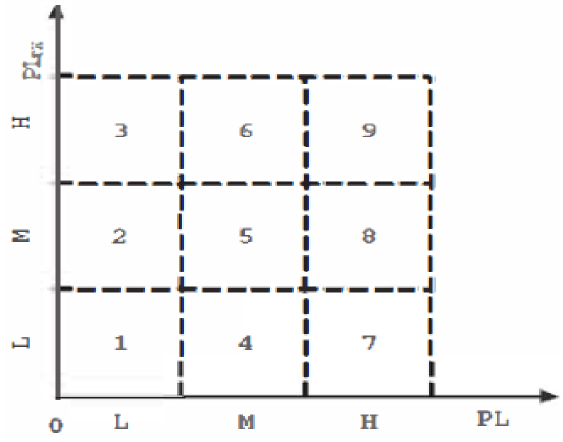

Figure 4.3 Rule base for Deployment

Membership functions provides $3^2 = 9$ rules. The rules are designed such as:

eg.R9: IF PL(i) is high and PL_{TH}(i) is high,THEN weight(i) is 9

Based on the simulations, we can conclude that using fuzzy logic to find the optimal number of nodes in each sub area is very beneficial because the path loss observed at worst points are reduced.

4.3 SWARM INTELLIGENCE FOR LOCALIZATION

Most WSN localization algorithms share a common feature that they estimate the location of nodes using the a priori knowledge of the positions of special nodes called beacon nodes or anchor nodes. The most intuitive solution to the localization problem is to load each node with a global positioning system. But it is not an attractive solution because of cost, size and power constraints. Typically, the nodes estimate their locations using signal propagation time or received signal strength. Signal propagation time is estimated through measurement of time of arrival, round trip time of flight or time difference of arrival of two signals. The localization is formulated as a multidimensional optimization problem, and tackled with population based CI methods such as GA and PSO.

WSN localization is a two phase process which includes Ranging phase and Position estimation phase. Whatever be the ranging method, there will be measurement errors in practical localization systems that result in noisy range estimations.Here proposed an optimization approach that minimizes the error in locating the coordinates of the target Nodes based on Swarm Intelligence called Particle Swarm optimization (PSO).

<u>PSO based Localization scheme</u>

- The Position estimation of the coordinates of the target nodes can be formulated as an optimization problem, involving the minimization of an objective function representing the error in locating the target nodes.
- Distance between target node and anchor node is calculated by

$$d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$$

- Initial value of position is estimated ,And then base station runs PSO to minimize localization error with the objective function,

$$f(x,y) = \frac{1}{M}\sum_{i=1}^{M}\left(\sqrt{(x-x_i)^2+(y-y_i)^2} - \hat{d}_i\right)^2$$

where (x, y) is the node location that needs to be determined,
$M \geq 3$ is the number of anchors
(x_i, y_i) is the location of anchor i.
Here $\hat{d}_i$ is the measured value of distance di between a node and the anchor i, obtained under noise.

MATLAB simulation results shows that localization error in PSO is more than halved in all experimental setups.

4.4 NEURAL NETWORKS FOR SECURITY

Many types of DoS attacks on WSNs have been devised. In collision attacks, attackers transmit packets regardless of status of the medium. These packets collide with data or control packets from the legitimate sensors. In unfairness attacks, adversaries transmit as any packets as possible after sensing that the medium is free. This prevents the legitimate sensors from transmitting their own packets. In exhaustion attacks, adversaries transmit abnormally large number of ready-to-send packets to normal sensor nodes, thereby exhausting their energy quickly.

Multilayered perceptron and generalized neuron-based (GN) distributed secure MAC protocols are proposed in which NNs onboard WSN nodes monitor traffic variations, and compute a suspicion that an attack is underway. When the suspicion grows beyond a preset threshold, the WSN node is switched off temporarily. A node detects an attack by monitoring abnormally large variations in sensitive parameters: collision rate Rc (number of collisions observed by a node per second), average waiting time Tw (waiting time of a packet in MAC buffer before transmission), and arrival rate (R_{RTS}), rate of RTS packets received by a node successfully per second. The strategy used in both the articles is the same, but the nonlinear mapping between traffic pattern variation and suspicion factor is implemented by different types of NNs in these articles. MLP and generalized neuron are trained using PSO. These distributed methods ensure that only the part of the WSN under attack is shut down, which is an advantage. Besides, these methods have been reported to be quite effective against DoS attacks, but the effectiveness depends on the value of the threshold suspicion.

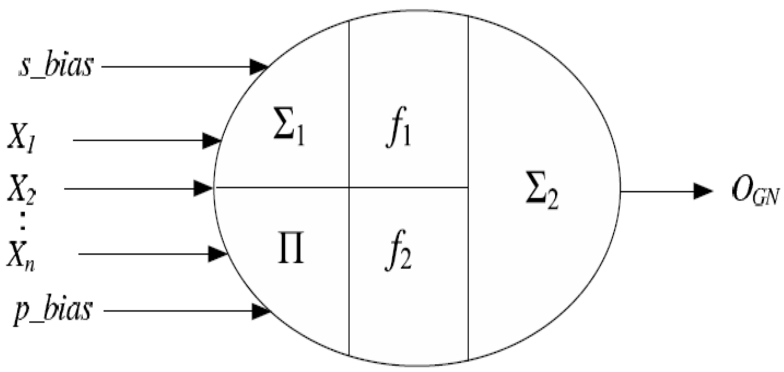

Figure 4.4 Structure of a Generalized Neuron

- GN is a neural network model that is more compact and flexible.
- A GN uses both Σ (sum) and Π (product) aggregation functions

WSN nodes monitor traffic variations, and compute a suspicion that an attack is underway. When the suspicion grows beyond a preset threshold, the WSN node is switched off temporarily.

A node detects an attack by monitoring abnormally large variations in sensitive parameters:

- Collision rate (Rc)- number of collisions observed by a node per second.
- Average waiting time (Tw)-waiting time of a packet in MAC buffer before transmission.
- Arrival rate (Rr)- rate of RTS packets received by a node successfully per second

<u>GN Based Secure MAC Protocol</u>

The GN senses the rise in critical parameters Rc, Rr and Tw and produces the output, ie used as a measure of suspicion of an attack that shuts down the node. When the suspicion of an attack exceeds a predefined threshold suspicion level ,the MAC and physical layer of the node is switched off. This results in saving of power that would have been wasted in retransmission of collided packets.

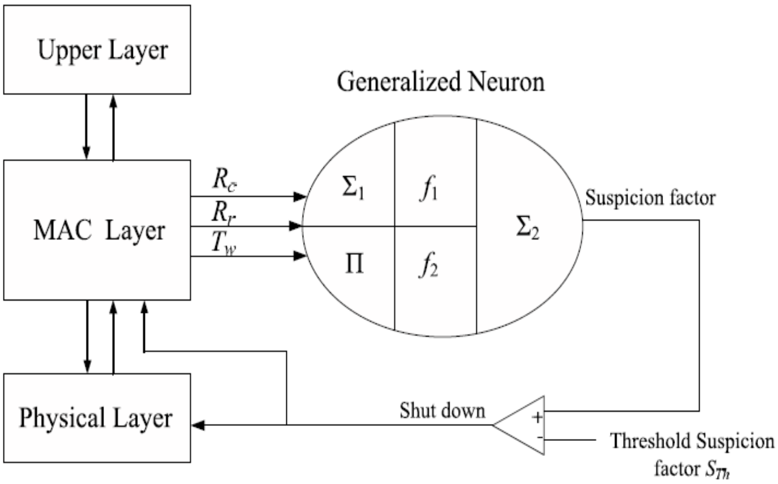

Figure 4.5 Generalized Neuron based MAC Protocol

Simulation results show that the power saving due to shutting down the attacked nodes results in reduction in power wastage, which in turn extends the network life

4.5 ENERGY AWARE ROUTING AND CLUSTERING

The energy consumption can be reduced by allowing only some nodes to communicate with the base station. These nodes called cluster-heads collect the data sent by each node in that cluster compressing it and then transmitting the aggregated data to the base station. Appropriate cluster-head selection can significantly reduce energy consumption and enhance the lifetime of the WSN. A fuzzy logic approach to cluster-head election is proposed here.

The base station has the global knowledge about the network. Moreover, base stations are many times more powerful than the sensor nodes, having sufficient memory, power and storage.

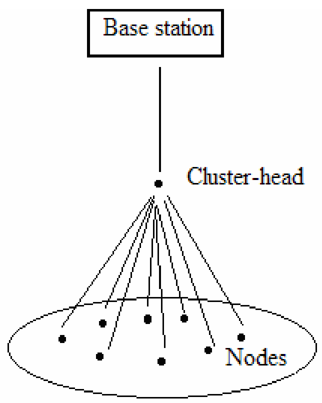

Figure 4.6 Architecture of Cluster

<u>Fuzzy Cluster-head Election Scheme</u>

Judicious cluster head election can reduce the energy consumption and extend the lifetime of the network. The operation of the scheme is divided into two rounds each consisting of a setup and steady state phase.

Setup phase- cluster-heads are determined by using fuzzy knowledge processing and then the cluster is organized.

Steady Phase- cluster-heads collect the aggregated data and is then sent to the base station.

A fuzzy logic approach based on energy, concentration and centrality is proposed for cluster head election.

- **Node Energy** - energy level available in each node, designated by the fuzzy variable 'energy'.
- **Node Concentration** - number of nodes present in the vicinity, designated by the fuzzy variable 'concentration'.
- **Node Centrality** - a value which classifies the nodes based on how central the node is to the cluster, designated by the fuzzy variable 'centrality'.

The study uses a network model in which all sensor nodes transmit the information about their location and available energy to the base station. The base station takes into account the energy each node has, the number of nodes in the vicinity and a node's distance from other nodes into account and determines which nodes should work as the cluster heads.

The base station fuzzifies the variables node energy and node concentration into three levels: low, medium and high, and the variable node centrality into close, adequate and far. Therefore, 3^3 (=27) rules are used for the fuzzy rule base. The fuzzy outcome that represents the probability of a node being chosen as a cluster head, is divided into seven levels: **very small, small, rather small, medium, rather large, large, and very large**. Triangular membership functions are used to represent the fuzzy sets medium and adequate and trapezoidal membership functions to represent low, high, close and far fuzzy sets. All the nodes are compared on the basis of chances and the node with the maximum chance is then elected as the cluster-head.

	energy	concentration	centrality	chance
1	low	low	close	small
2	low	low	adeq	small
3	low	low	far	vsmall
4	low	med	close	small
5	low	med	adeq	small
6	low	med	far	small
7	low	high	close	rsmall
8	low	high	adeq	small
9	low	high	far	vsmall
10	med	low	close	rlarge
11	med	low	adeq	med
12	med	low	far	small
13	med	med	close	large
14	med	med	adeq	med
15	med	med	far	rsmall
16	med	high	close	large
17	med	high	adeq	rlarge
18	med	high	far	rsmall
19	high	low	close	rlarge
20	high	low	adeq	med
21	high	low	far	rsmall
22	high	med	close	large
23	high	med	adeq	rlarge
24	high	med	far	med
25	high	high	close	vlarge
26	high	high	adeq	rlarge
27	high	high	far	med

Figure 4.7 Fuzzy Rule Base

The article observes a substantial increase in the network life in comparison to the network that uses the low energy adaptive clustering hierarchy approach. For a 20-node scenario in a 100m × 100m field, the number of data rounds before first-node-death in case of the proposed method is on average about 1.8 times greater than in low energy aware clustering hierarchy. Once again, the approach involves the overhead of collecting necessary information at a base station before determining cluster heads.

5. GUIDE TO CI METHODS FOR WSNs

Many CI methods have outperformed or complimented conventional methods under uncertain environments and severe limitations in power supply, communication bandwidth, and computational capabilities. However, all works presented here are not the best possible solutions and many have not been compared to traditional or to other CI approaches. Additionally, only a few researchers have evaluated their algorithms under real WSN environments like test-bed or deployments.

Findings have been summarized in Figure. The columns of the table represent the application areas in WSNs considered in this survey, while the rows represent the main CI techniques. The size of the black circles represents the number of articles surveyed in this paper for the particular combination of WSN problem and CI approach. In contrast, the shadowing of the box represents an evaluation of the applicability and suitability of the CI method for the particular problem. Of course, this evaluation is not always true: It depends highly on the exact CI algorithm, its parameters and the exact formulation of the problem. However, this overview gives a good insight about which CI method to explore first, when trying to solve a WSN problem.

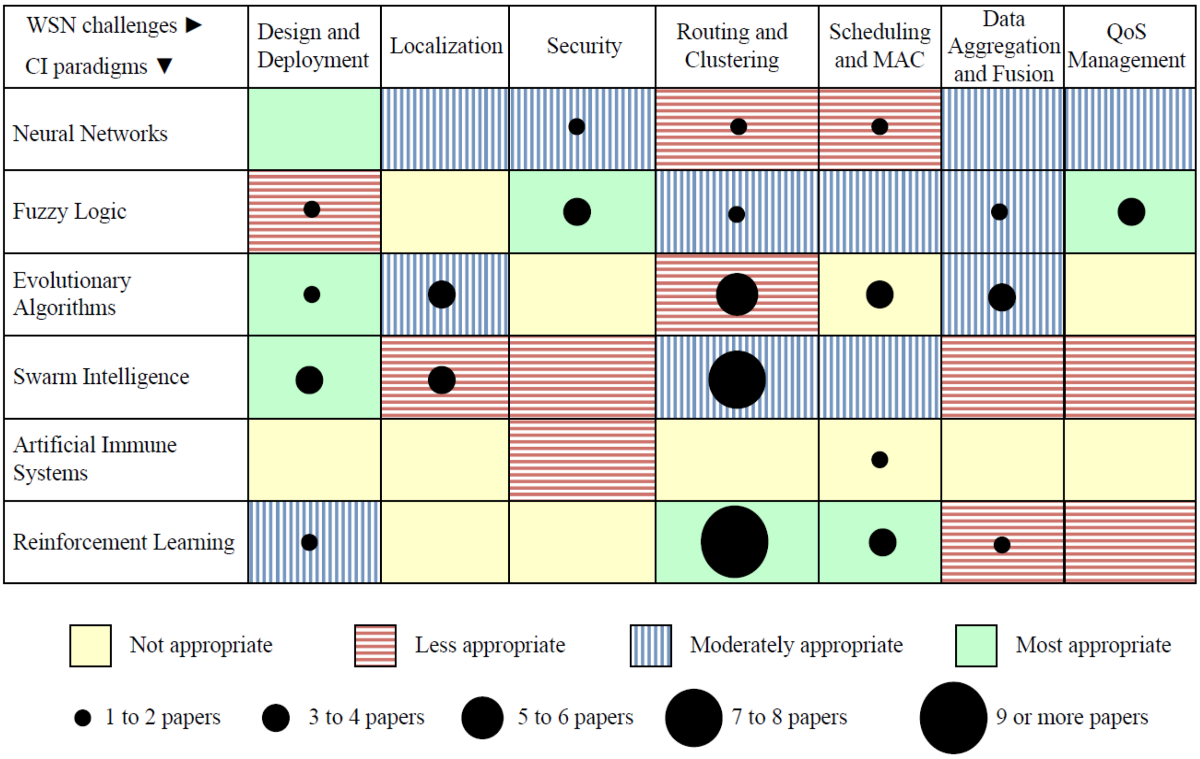

Figure 5.1 Overview of WSN Challenges and CI Paradigms to address them

6. CONCLUSION

We have discussed the innovative use of CI techniques to address WSN issues such as design and deployment, localization, security, optimal routing and clustering etc. Recent implementations of CI methods in various dynamical and heterogeneous networks are presented here. CI paradigms and numerous challenges in sensor networks are briefly introduced, and the CI approaches used by researchers to address the challenges are briefly explained. In addition, a general evaluation of CI algorithms is presented, which will serve as a guide for using CI algorithms for WSNs.

Future research is likely to focus on developing a well founded analytical approach to distributed multi-sensor estimation problem where there are time varying communication bandwidth constraints. Cross-layer design and parameter learning is envisioned to be an interesting new research area for CI in WSNs. Right now, a majority of the solutions presented here apply CI to one limited problem in areas like multicast routing, link quality, optimal clustering or placement. However, most issues arise from cross-layer incompatibility and the high human intervention needed for parameter setting and adjustment. Learning platforms and paradigms are needed rather than specialized solutions.

In spite of a multitude of successful CI applications in WSNs, the main concern is that the most of these algorithms or protocols are still in development stage, and they may remain forever in non-finalized state. Very few protocols have grown out of the simulation environment. Most of them do not even consider unreliable or asymmetric links, node failure and mobility. Besides, a common problem is the lack of comparison to conventional state-of-the-art protocols to clearly identify the advantages of introducing CI. Thus, the goal of CI research community for the future of CI in WSNs is to improve already existing solutions, refine them and define well-performing real-world protocols

7. REFERENCES

1. [1]. Raghavendra V. Kulkarni, Anna Förster and Ganesh Kumar Venayagamoorthy, “Computational Intelligence in Wireless Sensor Networks:A Survey”, *IEEE* Communications Surveys & Tutorials, VOL. 13, NO. 1,pp 63-96, First Quarter 2011.

2. [2]. N. B. B. Jessica A. Carballido, Ignacio Ponzoni, “CGD-GA: A graph-based genetic algorithm for sensor network design.” Inf. Sci., vol. 177, no. 22, pp. 5091–5102, 2007.

3. [3]. L. Zhao and Q. Liang, “Fuzzy deployment for wireless sensor networks,” in Proc. IEEE Int. Conf. Computational Intelligence Homeland Security Personal Safety CIHSPS, Q. Liang, Ed., 2005, pp.79–83.

4. [4]. A. Gopakumar and L. Jacob, “Localization in wireless sensor networks using particle swarm optimization,” in Proc. IET Int. Conf. Wireless, Mobile Multimedia Netw., 2008, pp. 227–230.

5. [5]. R. V. Kulkarni, G. K. Venayagamoorthy, A. V. Thakur, and S. K. Madria, “Generalized neuron based secure media access control protocol for wireless sensor networks,” in Proc. IEEE Symp.Computational Intelligence Multi-Criteria Decision-Making MCDM, Mar. 30-Apr. 2 2009, pp. 16–22.

6. [6].I. Gupta, D. Riordan, and S. Sampalli, “Cluster-head election using fuzzy logic for wireless sensor networks,” in Proc. 3rd Annual Commun. Netw. Services Research Conf., D. Riordan, Ed., 2005, pp.255–260.