Techniques for Efficient Routing in Wireless Sensor Network

Keyur M Rana
Assistant Professor
Computer Engineering Department,
Sarvajanik College of Engg & Tech, Surat, Gujarat

Mukesh A. Zaveri
Professor,
Computer Engineering Department,
Sardar Vallabhbhai National Institute of Technology, Surat, Gujarat

ABSTRACT
Wireless sensor network (WSN) is a tiny sensor device about a cubic size having sensors and small battery, which enables applications that connect the physical world with pervasive networks. These sensor devices do not only have the ability to communicate information across the sensor network, but also to cooperate in performing more complex tasks, like signal processing, data aggregation and compression in the network rather than out of the network. Various routing protocols have been designed and developed for Wireless Sensor Networks because the routing in wireless sensor network is distinguished from other networks. They face various challenges. Sensor nodes are strongly energy and storage constrained and failure rate of sensor node is very high. While sending data to the sink node, some routing mechanisms which consider all these parameters, are needed to extend life of the network.

Various routing protocols have been designed and developed for Wireless Sensor Networks because the routing in wireless sensor network is distinguished from other networks. They face various challenges. Sensor nodes are strongly energy and storage constrained and failure rate of sensor node is very high. While sending data to the sink node, some routing mechanisms which consider all these parameters, are needed to extend life of the network.

Along with some energy efficient technique for routing in WSN, Genetic algorithm based approach and Ant Colony Optimization approach for energy efficient routing is discussed in this paper. Genetic algorithm is a particular class of evolutionary algorithm that uses techniques inspired by evolutionary biology such as crossover, mutation, selection, and crossover. Both techniques are biological inspired.

Keywords
Genetic Algorithm, Ant Colony Optimization, routing in wireless sensor network

1. INTRODUCTION
A wireless sensor network consists of a large number of tiny sensing devices, deployed in a region of interest. Each device has processing and wireless communication capabilities, which enable it to gather information from the environment and to generate and deliver report messages to the remote base station.

Unlike Mobile Ad Hoc networks, wireless sensor networks (WSN) are characterized by asymmetric many-to-one data flows (mainly from sensor nodes to sink node), severe energy constraints and unreliable network nodes. Therefore, most routing protocols proposed for Mobile Ad Hoc networks are not suitable for wireless sensor networks, or cannot be used in wireless sensor networks without any modification. Thus, alternative approaches need to be explored. One of the challenges of wireless sensor network routing protocols is to achieve maximal robustness against path failure with minimal energy consumption.

Many routing protocols have been designed and developed for WSNs because the routing in WSNs has many challenges which are not there in other form of networks. Few of them are: number of nodes is large, nodes are tightly constrained in terms of energy, processing, and storage capacities, node failure rate is high etc. Hence they need careful resource management.

Most of the Routing protocols for WSN support multi-hop routing. Depending on how many copies of one data packet are forwarded to the destination simultaneously, these multi-hop routing protocols can be divided into two categories: single path routing and multi-path routing. In single-path routing, for each data packet, there is only one copy traveling along one path in the network. While in multi-path routing, multiple copies of one packet are transmitted in parallel along different paths to the same destination.

2. LITERATURE SURVEY
In this paper we have discussed various methods of energy aware routing.

Energy efficient routing with guaranteed delivery in wireless sensor network, where approach of path break to repair has been discussed.

Routing protocol based on Ant colony optimization, where ability of Ants to select the shortest path among few possible paths connecting their nest to a food site is applied for routing.

In the real world, ants (initially) roam randomly, and upon finding food return to their colony while laying down pheromone trails. If other ants find such a path, they are likely not to keep traveling at random, but to instead follow the trail, returning and strengthening it if they eventually find food [15]. This concept of Ant colony is used to find the adaptive shortest path for the routing in WSN

Genetic algorithm based approach for energy efficient routing, where strongest individual survives in a generation. This is modeled after the natural process of evolution as it occurs through offspring.

2.1 Energy Efficient Routing with Guaranteed Delivery

Generally, single path routing is simple and consumes less energy than multi-path routing [15]. However, a single path failure will cause a break of transmission and hence completely ruin the delivery. Compared with other wireless networks, wireless sensor networks are subject to high node failure rate.

Sending the same data packet along two fully node-disjointed paths (if they exist) almost doubles the delivery ratio. Using k-fully node-disjointed paths (k>2) can further increase the delivery ratio but at the same time, we will have overhead to manage these redundant data. This will lead to traffic, longer delay and more collision in the network.

One intuitive approach for achieving a high delivery ratio with low energy consumption is to forward data along a single path and to repair the path whenever a break is detected. Path repair has been introduced into many wireless routing protocols. In [16], when a path break (failure) is detected, a notification is sent to the source node, which is responsible for finding an alternative path and resending the data packet. However, this kind of source-initiated path repairing approach is uneconomical, especially when a failure occurs at many hops away from the source node. In [15], a local pivot-initiated path repairing approach is proposed, whereby the node (called pivot node), which is located at the immediate upstream of a path break (from where the packet has arrived), is responsible for seeking alternative paths through a local survey. If alternative paths exist, data forwarding will proceed along the best of them without restarting from the source node.

The selected alternative path may not be optimal from the view of the source node. But the energy is conserved by using already established route by previous transmission efforts from source to the pivot node. Rather than letting source know the failure notification and source again initiate route establishment (where energy is unnecessarily wasted), the alternative path is searched locally (which may be non-optimal) by the pivot node. Such energy saving compensates the additional energy caused by using a non-optimal path.

2.1.1 Establishment of Optimal Path and Data Forwarding

Before data transmission, an optimal path from each sensor node to each sink node needs to be established. When a data packet needs to be delivered to a sink node, $ID_{downstream}$ recorded in the routing tables of the source node will route the packet to move down from the source node along its optimal path to the destination. This will further be repeated by all its successor nodes to move the packet further down towards the destination node along its optimal path.

When a data packet is delivered along the lowest cost path, any node failure or channel error along the path may cause a break in transmission. Each node, which forwarded a data packet, is responsible for confirming that its successor has successfully received the packet, to guaranty a packet delivery. This may be implemented by the transmitter monitoring the packet just sent out to the downstream node and overhearing if that node has passed it on within a predefined time period.

2.1.2 Local Path Repair

As described in the previous section, each intermediate node is responsible for confirming that its successor has successfully received the data packet it just sent out. When the transmitter detects a failed receipt, it assumes its downstream node has been found out of order (called escaped node). Then it starts finding alternative path that bypass this failure route.

Seeking an alternative path starts by the immediate upstream node of the broken link (called helped node). This helped node will broadcast a Help Request (HREQ) message.

After receiving this message, each alive neighbor node performs the following comparisons sequentially based on the information previously stored and does the corresponding process according to the comparison result [15].

(a) If the downstream node of the receiver, denoted as *j*, is the helped node or the escaped node, the message is discarded, because the purpose of alternative path seeking is to bypass the escaped node.

(b) The message is discarded if the same packet had traversed the current node *j* and tried in useless to move along further.

(c) If the comparison results are not the above two cases, the current node *j* compares its cost with that of the escaped node. If its cost is equal to or smaller than HREQ's Cost of Escaped node, the current node believes that its optimal path to the destination will not go through the escaped node. Therefore, it issues a Help Response (HREP) message and sends it back to the helped node along the reverse path traveled by the Help Request message. Otherwise, the current node j passes the Help Request message on to its downstream node and that node will do the same comparisons and process as described in (a) to (c).

There are many improvements possible, such as, loop prevention and limiting repair number. Loop prevention can be achieved by introducing TTL counter as a field. Limiting repair number can be used to improve performance, as this is some pre defined number and repair operation will be performed this many times only.

2.2 Routing Protocol based on Ant Colony Optimization

Considering relatively large number and tightly energy constrained sensor nodes, frequent change in topology due to high rate of failure and broadcasting data in wireless sensor network, we require a routing protocol which saves energy. This can be achieved by considering path delay, node energy and the frequency of a node acting as a router, in routing decisions.

The problem in existing energy aware routing protocols is that they try to discover an optimal path and then frequently use the optimal path for every communication, which leads to rapid energy depletion of the nodes on the path. To overcome this problem, Xie Hui et al. [2] proposed a routing protocol which considers not only the path delay but also the node energy and the frequency a node acting as a router to increase network lifetime as long as possible.

This helps to prolong the life of the network while routing. Considering node energy and the frequency of node being a router can balance the WSNs node power consumption and increases overall network lifetime. The main goal is to prolong the network lifetime as well as to find the best path from the source node to the sink based on Ant Colony Optimization (ACO).

Ants communicate with each other using pheromones. Like other insects, ants perceive smells with their long, thin and mobile antennae. The paired antennae provide information about the direction and intensity of smells. Since most ants live on the ground, they use the soil surface to leave pheromone trails that can be followed by other ants. In a class that hunts in groups, a hunter that finds food marks a trail on the way back to the colony; this trail is followed by other ants, these ants then reinforce the trail when they go back with food to the colony. When the food source is exhausted, no new trails are marked by returning ants and the smell slowly disappears. This behavior helps ants deal with changes in their environment. For instance, when an established path to a food source is blocked by an obstacle, the hunters leave the path to explore new routes. If an ant is successful, it leaves a new trail marking the shortest route on its return. Successful trails are followed by more ants, reinforcing better routes and gradually finding the best path [1]. Thus, when one ant finds a short path from the colony to a food source, other ants are more likely to follow that path, and positive feedback eventually leads all the ants following a single path.

At the beginning, no pheromone is laid on the branches and the ants do not have any information about the branches length. However, since one branch is shorter than the other, the shorter branch receives pheromone at a higher rate than the longer one. Ants can smell pheromone, and their probabilistic decisions are based in favor of paths marked with higher amount of pheromone. Eventually, the shorter path will be selected by almost all ants of the colony. These complex global behaviors are the result of self-organizing dynamics driven by local interactions and communications among a number of relatively simple individuals.

The ant colony optimization approach exhibits quick problem solving and high degree of self organization which is similar to the requirements of low power, self organization and quickly routing in WSN.

The ACO routing protocol is composed of Neighbor Discovery, Routing and Data Transmission and Route maintenance phases [2] as follows:

Neighbor Discovery — Neighbor Discovery is initiated by the destination node. When the destination node receives a request, the node launches a neighbor discovery mechanism. Broadcast packets are flooded through the entire network until it reaches the source node to find all the routes from destination to source. This is when routing tables are built up [2].

Routing and Data Transmission — Data is sent from source to destination, using the information from the earlier phase. This is when paths are chosen probabilistically according to the path delay, the node energy and the frequency a node acting as a router.

Route Maintenance — In Route maintenance phase, from the destination node to the source node, inquiry is flooded, to maintain the activities of all paths and update their routing tables. If the node energy below a certain threshold, it can be made on standby.

In the first phase of neighbor discovery, the destination node instigates. During this phase, broadcast packets are exchanged between nodes. The broadcast packet contains these informations: Source Id, Send Time, Receive Tim, Sequence Numbe, Required Energy Threshold, Remained Energy Level and Destination Id.

In the process of routing and data transmission, when node s sends data towards the destination node d, node i chooses the next node j to send data according to a probabilistic decision rule (equation 1), where λ_{ij} is the value of pheromone, η_{ij} is the value of heuristic related to energy and λ_{ij} is the path delay. α, β and γ are three parameters that control the relative weight of pheromone path, energy heuristic value and path delay respectively [1, 2].

$$P_{ij} = \frac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}[\frac{1}{\lambda_{ij}}]^{\gamma}}{\sum_{l \in Ji}[\tau_{il}]^{\alpha}[\eta_{il}]^{\beta}[\frac{1}{\lambda_{il}}]^{\gamma}} \qquad (1)$$

After a transfer completed, pheromone loss is caused, and then each ant (node) deposits a quantity of pheromone on each path that it has used.

Pheromone update is done as follows [1]:

$$\tau_{ij} = (1-\rho)\tau_{ij} + \Delta\tau_{ij} \qquad (2)$$

where ρ is the rate of pheromone evaporation and $\Delta\tau_{ij}$ is the amount of pheromone deposited.

$$\Delta\tau_{ij} = \begin{cases} 1/L & \textit{If } (i,j) \in T^k \\ 0 & \textit{otherwise} \end{cases} \qquad (3)$$

Where T^k is the tour done by ant k at iteration T, and L is its length [10]. Energy heuristic η_{ij} is determined from the ratio of energy level of the node *j* to the sum of energy level of all neighbor nodes of node i as in equation (4). It can also be taken as energy level of the current node. This enables a node to make a decision according to neighbor's node energy levels. It means that if a node has lower energy level, it has lower probability to be chosen.

$$\eta_{ij} = \frac{e_j}{\sum_{n \in N_i} e_n} \qquad (4)$$

After updating the pheromone and energy levels, the routing table is recalculated. The routing table of each node is described as probability. On the basis of this probability, neighbor node is selected to send data.

Due to the choice of route is derived from these parameters dynamically; the routing protocol has a very good adaptive performance.

It can automatically adapt to the energy of each node of wireless sensor networks and pheromone changes to achieve dynamic routing choice to increase the network life time to the maximum.

2.3 Genetic Algorithm based Routing Algorithm

To achieve fault tolerance and to extend network lifetime, relay nodes can be used for balanced data gathering. Relay nodes are some special nodes which can be used as cluster heads in hierarchical sensor networks. They are the nodes having higher energy as compared to the sensor nodes.

In a two-tiered network architecture, where higher powered relay nodes act as cluster heads and sensor nodes transmit their data directly to their respective cluster heads. However, the relay nodes are still battery operated and hence, power constrained. Total depletion of the power of a relay node severely impacts the functionality of the network, since all sensor nodes belonging to the cluster of the depleted relay node will be unable to send their data to the base station and the entire cluster becomes inoperative. This may also put additional load on the surviving relay nodes and will cause faster depletion of the batteries of other relay nodes. figure 2.1 shows example of two-tiered sensor network.

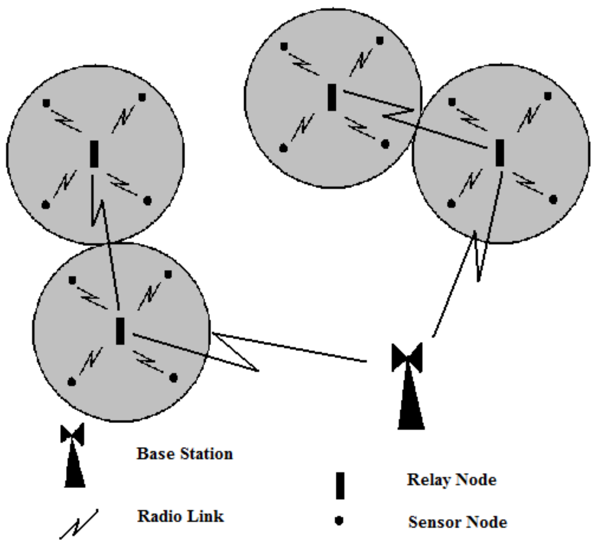


Figure 2.1 A two-tiered Network

In this model, the lifetime of a network is determined mainly by the lifetimes of these relay nodes. An energy-aware communication approach can greatly extend the lifetime of such networks. However, integer linear program (ILP) formulations for optimal, energy-aware routing quickly become computationally intractable and are not suitable for practical networks. In [3], Ataul Bari et al., have proposed a solution, based on a genetic algorithm (GA), for scheduling the data gathering of relay nodes. This efficient solution can significantly extend the lifetime of a relay node network. For smaller networks, where the global optimum can be determined, the GA based approach is always able to find the optimal solution. Furthermore, this algorithm can easily handle large networks, where it leads to significant improvements compared to traditional routing schemes.

2.3.1 Genetic Algorithm Overview

Genetic algorithm (GA) is modeled after the natural process of evolution as it occurs through offspring. For most genetic algorithms, the main concept is that the strongest individuals survive a generation (survival of the fittest) and reproduce with other survivors, producing (hopefully) an even stronger child [4]. The Genetic Algorithm is a technique for randomized search and optimization and has been applied in a wide range of studies in solving optimization problems, especially problems that are not well structured and interact with large numbers of possible solutions. In GA the search space of a problem is represented as a collection of individuals.

The individuals are represented by character strings, which are often referred to as chromosomes. The GA starts with the set of the randomly generated possible solutions. Each solution is a chromosome. The length of the each chromosome in the population should be the same. The purpose of the use of a GA is to find the individual from the search space with the best "genetic material." The quality of an individual is measured with an objective function called as fitness function. A fitness function is provided to assign the fitness value for each individual. This function is based on how close an individual is to the optimal solution – the higher the fitness value, the closer is the solution to the optimal solution.

The individuals are represented by character strings, which are often referred to as chromosomes. The GA starts with the set of the randomly generated possible solutions. Each solution is a chromosome. The length of the each chromosome in the population should be the same. The purpose of the use of a GA is to find the individual from the search space with the best "genetic material." The quality of an individual is measured with an objective function called as fitness function. A fitness function is provided to assign the fitness value for each individual. This function is based on how close an individual is to the optimal solution – the higher the fitness value, the closer is the solution to the optimal solution.

Two randomly selected chromosomes, known as parents, can exchange genetic information in a process called recombination or crossover, to produce two new chromosomes known as child or offspring. If both the parents share a particular pattern in their chromosome, then the same pattern will be carried over to the off springs. To obtain a good solution, mutation is often applied on randomly chosen chromosomes, after the process of crossover. Mutation helps to restore any lost genetic values when the population converges too fast. Once the processes of crossover and mutation have occurred in a population, the chromosomes for the next generation are selected. To ensure that the new generation is at least as fit as the previous generation, some of the poorest performing individuals of the current generation can be replaced by the same number of the best performing individuals from the previous generation. For example, say, best 10% individuals are copied as it is in the next generation, which will replace the equal amount of the poorest individuals. This process is called elitism. This entire cycle is repeated until the stopping criterion of the algorithm is met. The steps of a standard GA [5] are outlined in Algorithm 1.

Algorithm 1: Genetic algorithm
Begin
    Generate an initial population
    Compute the fitness of each individual
  While (not stopping criterion) do
    Choose parents from population.
    Perform crossover to produce offsprings.

Perform mutations.
Compute fitness of each individual.
Replace the parents by the corresponding offsprings in new generation.
End
End

3. WIRELESS SENSOR NETWORK MODEL AND GA BASED SOLUTION

For this model, a two-tiered wireless sensor network has been considered, with n relay nodes (acting as cluster heads). They are labeled as node numbers $1, 2, 3, \ldots, n$ and one base station, labeled as node number $n+1$. This labeling is done for representing the network in form of chromosome in genetic algorithm. Let D be the set of all sensor nodes, and D_i, $1 \le i \le n$, be the set of sensor nodes belonging to the i^{th} cluster, which has relay node i as its cluster head. We assume that each sensor node belongs to exactly one cluster i.e., $D = D_1 \cup D_2 \cup \ldots \cup D_n$ and $D_i \cap D_j = \phi$ for $i \neq j$.

A number of different metrics have been used in the literature to measure the lifetime of a sensor network. In [6], the lifetime of a sensor network has been defined as the minimum of (i) the time when the percentage of nodes that are alive (i.e., nodes whose batteries are not depleted) drops below a specified threshold, (ii) the time when the size of the largest connected component of the network drops below a specified threshold, and (iii) the time when the volume covered drops below a specified threshold. The work of [7] has defined the lifetime of the network as the lifetime of the sensor node that dies first. In [8], a number of metrics are used to define the network lifetime, e.g., N-of-N lifetime (i.e., time till any relay/gateway node dies), K-of-N lifetime (i.e., time till, a minimum of K relay/gateway nodes are alive) and m-in-K-of-N lifetime (i.e., time till, all m supporting nodes and overall a minimum of K relay/gateway nodes are alive). In this approach of routing using GA, N-of-N lifetime has been used by Ataul Bari et al. [3]. However, it can be used with other metrics by simply modifying the way the fitness of a chromosome is calculated.

3.1 Genetic Algorithm based Routing

Given a collection of n relay nodes, numbered from 1 to n, and a base station, numbered as n+1, along with their locations, the objective of the GA is to find a schedule for data gathering in a sensor network, such that the lifetime of the network is maximized. Each period of data gathering is referred to as a round [9], and the lifetime is measured by the number of rounds until the first relay node runs out of power. In other words, the N-of-N metric is used to measure the network lifetime. It is also assumed that the initial energy provisioned in each relay node is equal.

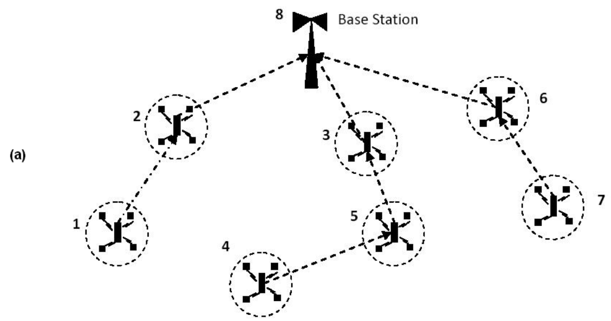
The Non-flow-splitting routing model satisfies the following characteristics, which can be used to design initial population of genetic algorithm [3].

Characteristic (i) : Each relay node receives data from the sensor nodes belonging to its own cluster, and can also receive data from any number of other relay nodes.

Characteristic (ii) : Each relay node i transmits data to one other node j (either another relay node or the base station), such that node j is within the transmission range of node i and j is either the base station, or j is a relay node that is closer to the base station than node i.

Characteristic (iii) : The base station only receives data, there is no transmission from base station to any relay node.

Now, the chromosome representation is defined, specify the initial population, describe the fitness function, and the strategies for crossover and mutation. The chromosome is represented as a string of node numbers. The length of each chromosome is always equal to the number of relay nodes. A routing scheme for a network with 6 relay nodes, and one base station, is shown in figure 3.1(a) and the corresponding chromosome is shown in figure 3.1(b). In this example, the value of the gene in position 1 is 2, indicating that node 1 transmits to node 2. Similarly, the value in position 3 is 8, indicating that node 3 transmits to node 8 (base station).



(b)	1	2	3	4	5	6	7	From
	2	8	8	5	3	8	6	To

Figure 3.1 : Representation of network graph as chromosome.

3.1.1 *Initial Population and Fitness Function*

Each chromosome in the initial population corresponds to a valid routing scheme. In our approach, the construction of this initial routing is based on the positions of the relay nodes, which is used by the base station to first create a list, Ni, $1 \le i \le n$, that contains all the one-hop neighbors j, of i, such that the link $i \rightarrow j$, $\forall j \in N_i$ can be used to route data from i towards the destination (base station) through j. For example, node *1*, from figure 3.1, will have N_i (where $i=1$) $=\{2, 4\}$ which are one hop neighbor of node 1. These are the nodes (referred as j) who leads i towards the destination. Here node *1* can reach to node 8 (destination) through node *2* or *4* only. Based on this information, possible routing schemes i.e. chromosomes, for the initial population are generated using a greedy approach, by randomly picking up the next hop node $j \in N_i$ for each source node i.

The search space for this problem is enormous. On an average, if each node has d valid one-hop neighbors, then the number of feasible routings for a network with n nodes is $O(d^n)$. In order to select a ‘‘good” energy efficient routing scheme, from such a large number of possible solutions, within a reasonable amount of time, a heuristic search technique, such as GA, is needed.

After generating each new individual, it is needed to evaluate its fitness value. Fitness value can be computed as the lifetime of the network, represented by the total number of rounds, until the first relay node runs out of battery power. As in [9], the value of the fitness function for an individual is computed as

$$L_{net} = \frac{E_{initial}}{E_{\max}} \quad (5)$$

where L_{net} is the network lifetime in terms of rounds and $E_{initial}$ is the initial energy of a relay node. We assume that the value of $E_{initial}$ is known beforehand and is the same for all relay nodes. E_{max} is the maximum energy dissipated by any relay node in the individual in one round of data collection.

It is assumed that $E_{initial}$ is known initially and will remain same. But E_{max} is the maximum power dissipated by any node of the chromosome. E.g. in figure 3.1(b), maximum energy dissipated from node 1 to 2, form node 2 to 8, from node 3 to 8, from node 4 to 5, from node 5 to 3, from node 6 to 8 and from node 7 to 6 is considered as E_{max}.

i.e. Emax = Max Energy Dissipated from node (1→2, 2→8, 3→8, 4→5, 5→3, 6→8, 7→6)

Such fitness calculation is done for each individual of the population.

3.1.2 Selection and Crossover

Selection of individuals is carried out using the Roulette-Wheel selection method [4]. As per this method, those individuals having higher fitness value will have higher probability to get selected.

Selection strategy should be such that, strong individual should be given more chance to recombine and produce new offspring. But at the same time, it should also be noted that weak individuals are also given chance to recombine. The probability to give chance to such weak individuals should be lesser than the probability of giving chance to the strong candidates. This probability should be in proportionate to the fitness value of individuals.

Sometimes, weak individuals contain some good genetic material. If we simply discard those weak candidates and not allow them to participate in producing new offspring, then we may lose this good genetic material. By allowing them to participate in producing new offspring, we are able to retain this good genetic material in the next generation. We have used Roulette-Wheel method for such selection.

To produce new offspring from the selected parents, the uniform crossover or k-point crossover (k = 1, 2, or 3, selected randomly) can be used for each crossover operation.

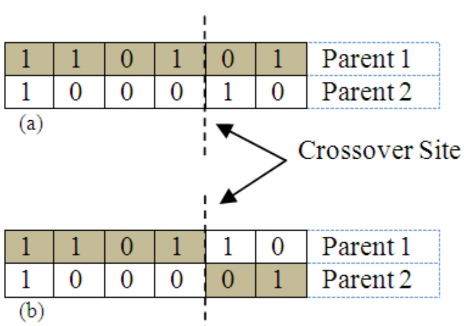


Figure 3.2 : Example of Crossover (a) before crossover (b) after crossover

As shown in figure 3.2, crossover site is chosen randomly. Figure 3.2 (a) shows two parents which are selected for crossover operation. Crossover site is shown by dotted vertical line. Figure 3.2 (b) shows chromosomes after crossover is performed. We can see that, part of string of parent 1 goes to parent 2 and vice versa.

3.1.3 Mutation

Selection and crossover alone can obviously generate an amazing amount of differing strings. However, depending on the initial population chosen, there may not be enough variety of strings to ensure the genetic algorithm sees the entire problem space. Or the GA may find itself converging on strings that are not quite close to the optimum it seeks due to a bad initial population.

1	1	0	1	0	1	Parent 1

(a)

1	1	0	0	0	1	Parent 1

(b)

Figure 3.3 :Example of Mutation (a) before mutation (b) after mutation

Some of these problems are overcome by introducing a mutation operator into the GA. For each string element in each string in the mating pool, the GA checks to see if it should perform a mutation. If it should, it randomly changes the element value to a new one. In the binary strings, 1s are changed to 0s and 0s to 1s. If it is not binary encoding of chromosome, then it should change as per the encoding is validated.

Figure 3.3 shows mutation operation. In figure 3.3 (a), it is a chromosome without mutation and in figure 3.3 (b) chromosome after mutation operation is shown. The shaded bit is altered in mutation operation.

The mutation probability should be kept very low as a high mutation rate will destroy fit strings and deteriorate the GA algorithm into a random walk, with all the associated problems. But mutation will help prevent the population from having same value and become unused. Remember that much of the power of a genetic algorithm comes from the fact that it contains a rich set of strings of great diversity. Mutation helps to maintain that diversity throughout the genetic algorithm's iterations.

3.2 Data gathering without aggregation

Previous approach of Genetic Algorithm is applicable for data gathering with aggregation. All nodes aggregates data and then send them forward.

In an application where aggregation is not possible, or, sensor node cannot wait for next packet to come, for aggregation and instead, it is to be transmitted immediately, this approach will not work.

In the figure 3.4, part of the sensor network is shown where dark circles are relay nodes and arrow shows data flow. Here node 4 takes and sends all data from node 1, 2 and 3 to node 5 and in turn, node 5 sends all these data towards destination.

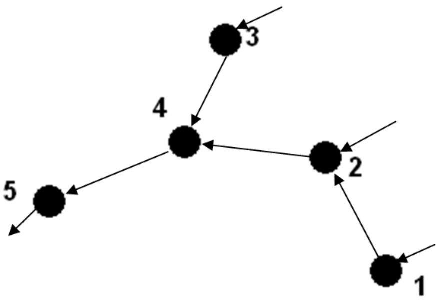


Figure 3.4 : Part of the Sensor network, showing one of the schedule for data transfer

If we consider data aggregation application, GA approach of [3] works fine but if we consider other application where aggregation is not required and instantaneously data to be sent, GA approach will not give desire result. It considers only one time data transfer from node 5 to towards destination and only one time data transfer from node 4 to 5. Hence calculation of energy dissipated will be counted only one time from node 5 to towards destination and node 4 to 5.

But in reality node 4 will send data to node 5 for more than 3 times (one time each for node 1, 2 and 3) and similarly node 5 will send data towards destination for more than 4 times (one time each for node 1, 2, 3 and 4). Here, Energy consumption from node 4 to 5 will be at least 3 times more than the approach of without data aggregation. So, Energy consumption will be more in such case and routing algorithm should consider this scenario.

3.2.1 Fitness Functions for data gathering without aggregation

For genetic algorithm, encoding scheme, cross over, mutation can be derived from the technique discussed in previous chapter. Parent selection method can be selected as roulette wheel, rank based selection or tournament selection method. Termination criteria for genetic algorithm used is N-of-N metric [11]. Fitness function used in [8] is for data gathering with aggregation and what we are proposing is for data gathering without aggregation. So, Fitness function will certainly be changed and is discussed next.

Ataul Bari et. al.[3] have suggested Fitness function as discussed earlier in equation 5. This was used for application of data gathering with aggregation. This cannot be used for application of data gathering without aggregation. We propose total energy consumed by the chromosome (an individual solution in GA) as a fitness function. This total energy can be calculated iteratively. For example, in figure 3.2 (a), base station is node 7 and each node will send data to node 7. This communication can be direct or via some intermediate nodes. Using this new approach, for the solution given in chromosome format, figure 3.2(b), energy consumed can be calculated as summation of the following:

i. Energy dissipated from node 1 to 3 and node 3 to 7
ii. Energy dissipated from node 2 to 4 and node 4 to 7
iii. Energy dissipated from node 3 to 7
iv. Energy dissipated from node 4 to 7
v. Energy dissipated from node 5 to 3 and node 3 to 7
vi. Energy dissipated from node 6 to 7

4. EXPERIMENTAL RESULTS

Genetic algorithm approach was tested for different size of network. N-of-N metric was used to measure lifetime of network.

We have simulated different networks (with different numbers of relay nodes) for Warshall's shortest path first algorithm and then simulated same network with the approach of data gathering without aggregation. We have also simulated these networks using genetic approach proposed in [3].

Warshall's algorithm [12, 13] (sometimes known as the Roy–Floyd algorithm) is a graph analysis algorithm for finding shortest paths in a weighted graph. A single execution of the algorithm will find the lengths (summed weights) of the shortest paths between all pairs of vertices.

For our experiments, we have used following first order radio model for communication energy dissipation [14].

$$E_{T_i}(b_{t_i}, d_{i,j}) = \alpha_2 b_{t_i} + \beta b_{t_i} d_{i,j}^m$$

Where di,j is the Euclidian distance between node i and j, α2 is the transmit energy coefficient, β is the amplifier coefficient, bti is amount of data to transmit from node i to another node and m is the path loss exponent, 2≤ m ≤4. ETi is total transmit energy dissipated.

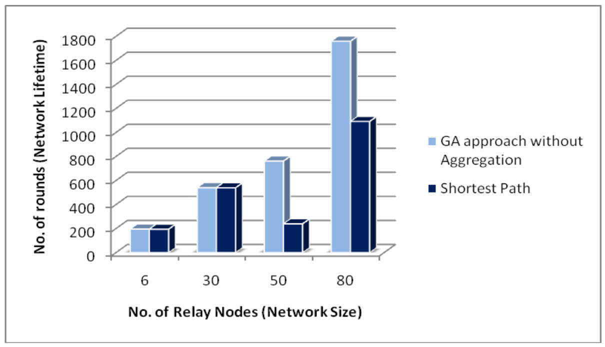


Figure 4.1 : Comparison of Warshall's shortest path and GA approach for data gathering without aggregation.

Similarly, the receive energy, ERi is calculated as follows:

$$E_{R_i}(b_{r_i}) = \alpha_1 b_{r_i}$$

Where b_{ri} is the number of bits received by relay node *i* and *α1* is the receive energy coefficient.

Hence total energy dissipated by a node *i* for data to receive and then to transmit it further is *Ei*.

$E_i = E_{Ti} + E_{Ri}$

We consider both type of energy in computation of energy consumption. For simulation, the values for the constants are taken same as in [14] as (i) α1= α2=50 nJ/bit, (ii) β=100 pJ/bit/m2 and (iii) the path loss exponent, m=4.

The initial energy of each node, E_{init}=5J.

Figure 4.1 shows comparison between GA approach with Warshall's shortest path algorithm. It can be seen that as the number of nodes in network increases, the performance gets better for GA based approach. Warshall's algorithm does not perform well when, number of nodes are more in the network. These experiments are done with a consideration of data gathering without aggregation.

4.1 Evaluation of various techniques

We have discussed various techniques for routing in wireless sensor network. These techniques mainly emphasis on energy conservation so that overall life of sensor network is extended.

Routing with guaranteed delivery		
Metric for routing	**Methodology**	**Remarks**
Hop number	Single path routing	At failure, energy is saved by taking local decision, rather than sending packet to the originator.
	Repair function	
	Local Pivot initiated approach	

Routing protocol based on ACO		
Metric for routing	**Methodology**	**Remarks**
Path delay, node energy and frequency of node acting as a router	Single path routing	Biologically inspired
	Works the way Ant finds food on a shortest path	Energy is saved by considering how many times a node acting as a router and level of node energy
	Dynamic and adaptive	

GA based approach for energy efficient routing		
Metric for routing	**Methodology**	**Remarks**
N-of-N lifetime	For hierarchical or two tiered network architecture	Biologically inspired
	Network is divided in clusters	Survival of fittest
		Fitness function and Mutation operations are designed such that energy is not wasted

5. CONCLUSION

This report focuses on routing techniques in wireless sensor network. Routing with guaranteed delivery uses local pivot initiated approach and repairs the failure at the failure point only. This saves energy to start the transmission all the way from source node. Ant colony optimization and Genetic algorithm based approach, are inspired from swarm intelligence and evolution theorem respectively. Both approaches are dynamic and adaptive. They take in to consideration, the current level of energy and overall energy usage to decide the optimal route.

These techniques are already proposed by researchers. In this paper, we have changed fitness function of Genetic Algorithm based approach for routing in WSN with data gathering without aggregation. We have compared experimental results for different size of networks using GA based approach and Warshall's algorithm. GA based solution with proposed fitness function gives extension to the sensor network life time as compared to other mentioned techniques.

6. REFERENCES

[1] Marco Dorigo, Vittorio Maniezzo, and Alberto Colorni, "Ant System: Optimization by a Colony of Cooperating Agents", IEEE Transactions on Systems, MAN, and Cybernetics-Part B Cybernetics, vol 26, no 1, February 1996

[2] Xie Hui, Zhang Zhigang, Zhou Xueguang, "A Novel Routing Protocol in Wireless Sensor Networks Based on Ant Colony Optimization", Environmental Science and Information Application Technology, 2009. ESIAT 2009. International Conference on Volume 2, 4-5 July 2009 pp. 646 – 649

[3] Ataul Bari, Shamsul Wazed, Arunita Jaekel, Subir Bandyopadhyay "A genetic algorithm based approach for energy efficient routing in two-tiered sensor networks", Ad Hoc Networks ,Volume 7 , Issue 4, Pages 665-676, ISSN:1570-8705, June 2009

[4] D.E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison Wesley, Reading, MA, 1989.

[5] Ming-Tsung Chen, Shian-Shyong Tseng, "A genetic algorithm for multicast routing under delay constraint in WDM network with different light splitting", Journal of Information Science and Engineering 21 (2005) pp. 85–108.

[6] D.M. Blough, P. Santi, "Investigating upper bounds on network lifetime extension for cell-based energy conservation techniques in stationary ad hoc networks", in: Proceedings of the 8th ACM International Conference on Mobile Computing and Networking (ACM MobiCom 2002), September 2002, pp. 183–192.

[7] R. Madan, S. Cui, S. Lall, A. Goldsmith, "Cross-layer design for lifetime maximization in interference-limited wireless sensor networks", in: Proceedings of 24th IEEE Conference on Computer Communications (IEEE INFOCOM 2005), vol. 3, 2005, pages. 1964–1975.

[8] J. Pan, Y.T. Hou, L. Cai, Y. Shi, S.X. Shen, "Topology control for wireless sensor networks", in: Proceedings of the Ninth Annual International Conference on Mobile Computing and Networking, 2003, pp. 286– 299.

[9] A. Bari, A. Jaekel, S. Bandyopadhyay, "Maximizing the lifetime of two tiered sensor networks", in: the Proceeding of IEEE International Electro/Information Technology Conference (EIT 2006), MI, May 2006, pp. 222–226.

[10] R. C. Shah and J. Rabaey, "Energy Aware Routing for Low Energy Ad-Hoc Sensor Networks", in Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), vol. 1, Orlando, FL, Mar. 2002, pp. 350.355.

[11] J. Pan, Y.T. Hou, L. Cai, Y. Shi, S.X. Shen, "Topology control for wireless sensor networks", in: Proceedings of the Ninth Annual International Conference on Mobile Computing and Networking, 2003, pp. 286– 299.

[12] "A 'layman's' explanation on Floyd–Warshall algorithm", [online]. Available : http://en.wikipedia.org/wiki/Floyd%E2%80%93Warshall_algorithm [Accessed : April 24, 2010].

[13] "An explanation on Network flows and Graphs", [online]. Available : http://www.ieor.berkeley.edu/~ieor266/Lecture12.pdf [Accessed : April 24, 2010].

[14] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, "Energy efficient communication protocol for wireless micro-sensor networks", In: Proceedings of the 33rd HICSS, Maui, Hawaii, 2000, pp. 3005–3014.

[15] Di Tian, Nicolas D. Georganas. "Energy Efficient Routing with Guaranteed Delivery in Wireless Sensor Networks" Proc. IEEE Wireless Communications and Networking Conference (WCNC2003), New Orleans, LA, USA, March 2003

[16] Robert D. Poor. "Gradient Routing in Ad Hoc Networks." http://www.media.mit.edu/pia/Research/ESP/texts/poorieeepaper.pdf