CHAPTER 4

A HYBRID OPTIMIZATION TECHNIQUE FOR ROUTING IN WIRELESS SENSOR NETWORKS

4.1 INTRODUCTION

Most of the Wireless Sensor Networks (WSN), protocols have only a single research objective. The basic task of multi-objective routing is to find a route in the network which has sufficient resources to optimize some network parameters and satisfy multiple constraints. The computation of routes involves tradeoffs between energy consumption, path loss and detection accuracy. The main objective is to maximize the total detected signal energy while minimizing the energy consumption and path loss. A new hybrid multi objective decision making algorithm employing weighing function and particle swarm optimization is proposed for fuzzy based selection of wireless networks.

4.2 A MOBILE AGENT ARCHITECTURE

The architecture that appears in Figure 4.1 is comprised of both static and mobile agents; static agents provide resources and facilities to mobile agents and mobile agents move between network domains taking advantage of these resources to fulfill their goals. A domain (Min Chen, 2006) is a logical boundary used to delimit nodes, agents and resources into manageable and distinct entities. It is envisaged that mobile agents will possess the ability to migrate to the most appropriate location. The behavior segment of an agent will describe the functionality of the agent, for example a resource discovery agent or a link maintenance agent. The knowledge-base will contain the "intelligence" of the agent, that is, its experiences, its goals and its beliefs. This segment will also contain temporary information that the agent is working upon before it is returned to the user for appraisal.
4.2.1 Domain Agent

The domain agent is a static agent, that coordinates the activities that occur within a domain. The domain agent offers a number of services related to information resources, users and agents within a given domain:

1. It provides a migration service to mobile agents wishing to leave the domain. The domain agent is responsible for ensuring, that the mobile agent is transferred successfully. At this level, there is negotiation, security and fault tolerance to take into account; by contrast, the Agent Service layer is concerned with merely transporting an agent from one location to another.

2. It authenticates and performs a validation check on mobile agents wishing to enter the domain. Agents that cannot be authenticated or fail the validation check are rejected.

3. It launches received mobile agents in a suitable run-time environment, which will be related to the amount of trust given to the mobile agent. Agents from trustworthy nodes may be allowed access to more information resources than agents from unknown nodes. A suitable run-time environment will depend upon the access level granted to the mobile agent and the functions that it wishes to perform.
4. It mediates access to information resources at a domain level. When a mobile agent enters a domain, the domain agent performs a security check upon it and allocates the mobile agent a domain-wide permission. This permission can subsequently be used by resource agents to determine whether to allow the mobile agent access to a resource, and by other mobile agents to decide whether to communicate and trade information with the mobile agent.

5. It provides a central registration area within a domain where static and mobile agents can register and advertise their resources and interests. Thus, a domain acts as a meeting point which allows the agents to gather and share information to resolve their goals.

6. It is the initial point of contact within the domain. All message interactions between static and mobile agents are initially routed through the domain agent. This allows the agents to communicate in an asynchronous fashion. However, one or two agents have become aware of each other's presence (via the domain agent) and they can initiate a direct and synchronous communication between themselves, to transfer high-bandwidth data.

7. It ensures that the domain does not become overwhelmed by mobile agents. This can be achieved by limiting the amount of agents that can exist within the system at a given time or by implementing a charge for processor cycles and resource access. This way, even if a mobile agent attempts to monopolise a resource, it could only do so for as long as its credit lasts.

8. It advertises public information resources and the presence of mobile agents to enquiring agents from both inside and outside of the domain. In this way, mobile agents can interrogate the contents of a particular domain before moving to ensure that it contains
information resources that will help it to achieve its goals. It also provides a mechanism for agents to locate each other.

9. It periodically announces its presence to the larger community of domain agents. This allows each domain agent to maintain a list of other domains that is available for mobile agents to use in determining their next jump point.

The domain agent is the central manager within a domain and is ultimately responsible for ensuring that security which is enforced within the domain and that agents can communicate with information resources and with one another.

4.2.2 Resource Agent

Resource agents are static agents that exist within a domain to provide a level of abstraction between the information resource to which they provide access and the requesting mobile agents. The purpose of a resource agent is to mediate access to a particular information resource for a mobile agent; the resource agent understands how to access the resource and also understand the permission structures associated with the resource. As such, the resource agent has a number of functions:

1. It is fully conversant in the protocols of the information resource. The information resource should be completely accessible by mobile agents, so that user intervention directly on the resource is not required. This implies that the user has a set of distributed information management mobile agents that are able to access, modify, locate and manage their information resources through the individual resource agents.

2. It provides an ontological description for each of the services offered by the resource. These services are the methods by which mobile agents interrogate, update and manage resources.

3. It mediates access to the resource at a resource level. When a mobile agent requests a service, the resource agent can either grant access
according to the mobile agent's domain-wide permission or it can impose further constraints and checks upon the mobile agent itself.

4. It advertises the presence of the information resource by registering the services available to the domain agent. In this way, mobile agents can be aware of what information resources are present within a domain before they arrive at a domain.

The resource agent, through the Agent Service layer, provides conversion between the data formats and protocols that the individual resource is using, but the Application layer is responsible for providing the ontological mapping between them.

The interaction between resource agents and mobile agents will form the crux of the distributed information management aspect of the infrastructure. The flexibility, that the mobile agents are afforded in accessing these information resources and the manner in which they interpret the results will determine their usefulness to the user.

### 4.2.3 Mobile Agent

Mobile agents, as their name suggests, are the components within the framework which can migrate between domains. They are the mechanism by which the user exercises and control over their own distributed information resources and gain access to other shared information resources.

Initially, a user launches a mobile agent within a given domain, which is called the host domain. Mobile agents are equipped with a set of user-defined goals which describe the nature and limits of their functionality; for example, a resource discovery agent is a mobile agent with a different goal set, than say, a navigation assistant agent. In addition to the limits on the functionality that users place on their mobile agents, it is probable that the mobile agents themselves will encounter limits that exist within domains. In some cases, these limits will compromise the goals that
have been given to the mobile agent. The essential functions of mobile agents will be defined by the distributed information management tasks that they are allocated. However, they have the following interactions with the framework:

1. Mobile agents determine where to migrate to, next by initially querying the local domain agent for a list of domains of which it is aware. The mobile agent can then use this information to contact each domain agent and determine which one offers a suitable set of information resources.

2. Mobile agents are authenticated by an electronic signature that they carry. This signature may be encrypted, but must certainly be verifiable with the host domain of the mobile agent or a third-party domain. Additionally, mechanisms must be employed to ensure that mobile agents are not compromised during transit or within a domain, and that the behaviour of a given mobile agent is consistent with the behaviour that it was given when it was launched. As it is impossible to guarantee these requirements, since a malicious agent may be launched from an authentic domain, mobile agents should be controlled as tightly as possible and consequently given a minimum amount of permission.

3. Mobile agents possess the characteristic of persistence; they use migration as a mechanism to achieve longevity. In this way, they are not reliant on the host domain that launched them. This is particularly useful in nomadic computing, where the user is only connected to the network for short period of time.

4. Mobile agents may communicate not only with resource agents, but also with other mobile agents to achieve their goals.

5. Mobile agents should transmit the results of their findings and actions regularly to their user on the host domain. This way, the user can ensure that the mobile agent is functioning correctly and also helps prevent the mobile agent from becoming too large to move or to process.
Mobile agents are the ultimate effectors of change within a distributed information management environment, and as such, should be treated cautiously. The possibility for mobile agents to effect changes or gain access to private information must be closely monitored by the domain agent and the individual resource agents. A comprehensive authentication, validation and access permission system is vital to ensure that malicious agents can do no harm to a system and that legitimate mobile agents can fulfill their goals.

4.2.4 User Interface Agent

The user interface agent provides a level of abstraction for the user away from the details of the mobile agent framework. It is essentially an interface agent to perform the following tasks:

1. It launches mobile agents on behalf of the user and tracks their progress and position.
2. It provides mobile agents with a communication point through which they can return the results of their tasks. User interface agents may need to possess local mobility to allow for node failures, since they need to be executing even when a user is disconnected from the system.
3. It organizes and pre-processes information returned from mobile agents into a form that is suitable for the user. This may involve filtering out replicated information, presenting urgent information to the user quickly, rendering information using the tools that the user is most familiar with, etc.
4. It provides domain agents with the authentication credentials of a user's mobile agents. It is the task of the user interface agent to ensure that the requesting entity is itself an authentic domain agent, to make certain that authentication credentials are not given to unauthorized systems.
User interface agents provide the user with a window onto their agents, their status, their results and the mobile agent framework. The creation of new distributed resource management agents could either be the task of the user to write (or reuse from existing templates with different parameters), or the task of the user interface agent to generate automatically, based upon the interpreted requirements of the user. The latter approach is most desirable, but is still a long way off giving the technology level that currently exists.

4.2.5 Gateway Agent

A gateway agent is a static agent that provides access to and from the domains. It allows a number of domains, related perhaps by geographic, commercial or other commonalities, to be treated as a single domain, called a gateway domain. Gateway agents offer the following services to the domains within its locale:

1. It provides firewalls for a corporation or organization where information within the gateway domain, is of a protected or secure nature. Mobile agents can be restricted both on entry to and exit from the collective domain.

2. It equips mobile agents entering a gateway domain with the suitable components to allow the mobile agent to communicate within the collective domain.

3. It provides a hierarchy of domains where queries from domains lower in the hierarchy can be dealt with and resolved by domains higher in the hierarchy. It is envisaged that the hierarchy will be similar in nature, though not restricted to an agent-like DNS, where gateway agents provide access to countries at the root level, regions at the next level and so on down to the constituent domains at the leaves.

At the Application layer, gateway agents are the ideal place at which corporations, governments, countries, etc. can ensure that local ordinances are observed within their domains. For example, the gateway domain for the Ministry
of Defence will have tighter security restrictions than, say, the gateway domain of the United Kingdom. They also provide a mechanism for ensuring that undesirable material does not enter the domain and restricted information does not leave the domain. Through the developed hierarchy of gateway domains, gateway agents add scalability to the system, since messages that are for agents within a given domain or gateway domain only travel within that domain. The burden of dealing with messages to destinations unknown can be spread across the hierarchy, thus alleviating the pressure on individual gateway and domain agents.

4.3 ROUTING IN MOBILE AGENTS

The MADSN consists of three types of components: Fusion Center (FC), sensor nodes, and communication network and it is shown in Figure 4.2. The FC and sensors are usually interconnected via a wireless communication network. A group of neighboring sensor nodes that are commanded by a single FC whenever it forms a cluster.

![Figure 4.2 Routing in MADSN](image)
4.3.1 Sensor Nodes

A sensor node is the basic functional unit for data collection in a MADSN. A sensor node may have several channels with different sensory devices connected to each sensor. Sensor nodes are geographically distributed and collect measurements of different modalities such as acoustic, seismic, and infrared from the environment. The data acquisition is controlled by a sampling subsystem, which provides the acquired data to the main system processor. The signal energy from each channel is detected individually and processed in the analog front end. A mobile agent migrates among the sensor nodes via the network, integrates local data with a desired resolution sequentially, and carries the final result to the originating FC.

The fused data is used to derive appropriate inferences about the environment based on the application. The setup time of a FC accounts for loading the mobile agent code and performs the other initialization tasks. The amount of signal energy that reaches an individual sensor is an effective indicator of how close the node is to a potential target. Once the signal is captured and processed by a sensor node, the strength level of the detected signal is broadcast to the whole cluster so that a FC in any location has knowledge of various levels of signal energy detected by all active sensor nodes within the coverage area. In target detection applications, a leaf node with higher signal energy carries more information and should have higher priority of being visited.

4.3.2 Communication Links

Wireless communication links need to be established between neighboring nodes as the mobile agent migrates along a route. The embedded RF modems on sensor nodes provide such a networking capability with low power requirement. The different clusters select different —networknumbers,— which correspond to separate hopping pseudo noise sequences to avoid interferences. The message transmission time between two sensor nodes depends not only on the physical distance between them, but also on the channel bandwidth and the data packet loss.
rate as well as the size of the message, which includes partially integrated data and mobile agent code itself. In general, the electromagnetic propagation time is almost negligible in short-range wireless communication. Hence, the physical distance is not explicitly considered but is incorporated as a part of the path loss representing the signal attenuation.

A mobile agent is dispatched from a processing element and is expected to visit a subset of sensors within the cluster to fuse data collected in the coverage area. Generally, the more sensors visited, the higher the accuracy achieved using any reasonable data fusion algorithm. But, it is important to select an appropriate route so that the required signal energy level can be achieved with a low cost in terms of total energy consumption and path loss. The routing objective is to find a path for a mobile agent that satisfies the desired detection accuracy while minimizing the energy consumption and path loss. The required path is computed based on the current state of DSN and the mobile agent traverses the nodes along the path while performing the fusion operation. The energy consumption depends on the processor operational power and computation time and the path loss is directly related to the physical length of the selected path.

4.3.3 Objective Function

In MADSNs, the sensors that are close to the target with high detected signal energies are first identified before the fusion step. The identification of such sensor nodes reduces the complexity and amount of the data for fusion. Hence, the objective of MADSNs is to compute a route for the mobile agent through such sensor nodes by using the detected signal energies of different sensors. The mobile agent visits the sensors in the route and performs data fusion at these nodes. Each sensor node in the MADSN is associated with two parameters: detected signal energy and energy consumption. These quantities determine the energy consumption and detection accuracy of the path. The path loss is a function of the distance between the sensors. Hence, the order of sensors on the path traversed by the mobile agent determines the
path loss. These quantities have a significant influence on the performance of the MADSN. The more the number of sensors visited, the higher is the data accuracy of the route. However, it is important to compute an appropriate route such that, the detection accuracy is maximized with a low cost in terms of total energy consumption and path loss. The objective function for the mobile agent routing problem is based on three aspects of a routing path: energy consumption, path loss, and detected signal energy:

4.3.3.1 Energy Consumption

The energy consumption at a sensor node is determined by the processing speed and the computation time. For wireless message transmissions, the energy consumption depends on the sensor’s transmission power and message transmission time. The energy consumption of a path $R$ is the sum of the energy expended at each sensor node along the path. If $(n_0, n_1, n_2, \ldots, n_m)$ denotes the sequence of nodes along a path $R$, then the total energy consumption $E(R)$ (R. Rajagopalan, 2005) is given by:

$$E(R) = \sum_{k=0}^{m} \left[ \left( (t_{ak} + t_{pk}) \times H_k \right)^2 + (P_{ik} \times t_m) \right] \tag{4.1}$$

where $t_{ak}$ and $t_{pk}$ indicate the data acquisition time and data processing time for node $k$, $t_m$ indicates the message transmission time, $H_k$ denotes the operational power level, and $P_{ik}$ denotes the transmission power of node $k$.

4.3.3.2 Path Loss

The path loss represents (A. Konstantinidis, 2007) the signal attenuation due to free space propagation, and should be minimized to guarantee reliable communication. The well-known Friss free space propagation model expresses the relation between the power $P_{ij}$ received by sensor $j$ and the power $P_{ii}$ transmitted by sensor $i$ as:

$$P_{ij} = \frac{P_i \times G_i \times G_{ij} \times \lambda^2}{4\pi^2 \times d_{ij}^2 \times \beta} \tag{4.2}$$
where $G_i$ is the gain of transmitting sensor $i$, $G_j$ is the gain of the receiving sensor $j$, $\lambda$ is the wavelength, $d_{ij}$ is the Euclidean distance between the co-ordinates of sensors $i$ and $j$, and $\beta$ is the system loss factor. The path loss associated with the corresponding wireless link is (in dB):

$$PL_{i,j} = 10 \times \log \left( \frac{p_i}{p_{ij}} \right)$$  \hspace{1cm} (4.3)

The total path loss along a path is the sum of the path losses associated with each link along the path. The total path loss for a path $R$ is calculated as

$$PL(R) = \sum_{i=0}^{l-1} PL_{i,j} = \sum_{i=0}^{l-1} PL_{n_i \rightarrow n_{i+1}}$$  \hspace{1cm} (4.4)

where $l$ is the total number of nodes along the path.

### 4.3.3.3 Detection Accuracy

A sensor node detects a certain amount of energy emitted by the potential target, which may be used by a mobile agent for data integration. A mobile agent always tries to accumulate as much signal energy as possible for accurate decision in target detection.

High detection accuracy is also an important goal for accurate inference about the target. Each sensor detects a certain amount of energy $e_d(u)$, emitted by a target. If $K_v$ is the energy emitted by a target at location $u=(x_t, y_t)$, the signal energy $e_i(u)$ measured by a sensor $i$ is

$$e_i(u) = \frac{K_v}{1 + \alpha d_i^p}$$  \hspace{1cm} (4.5)

where $d_i$ is the Euclidean distance between the target location and sensor location, $p$ is the signal decay exponent that takes values between 2 and 3, and $\alpha$ is an adjustable constant. The sum of the detected signal energy along a path $R$ (Q. Wu, 2004) is defined as

$$DE(R) = \sum_{j=1}^{l} e_j(u)$$  \hspace{1cm} (4.6)
4.4 HYBRID OPTIMIZATION TECHNIQUES FOR ROUTING

The main objective of this algorithm is to perform the best selection of access network and to maximize the percentage of the satisfied users. It is important to design a suitable decision making algorithm for the selection of best nodes. The proposed Optimization algorithm is as follows:

**Step1:** As discussed in section 3.5.1 the good, bad and the normal nodes are selected by the fuzzy logic controller with the inputs as the node degree, link quality, residual energy and traffic rate.

**Step2:** Call Particle Swarm Optimization algorithm MOPSO (J. Kennedy, 1995) with input arguments energy consumption, path loss and detection accuracy.

**Multi-objective PSO**

1. Initialize the positions for all the particles.
2. **DO WHILE:**
   (a) Add random numbers in the interval [0 1] to position the particle.
   (b) Constrain the position so that it does not exceed the grid size.
3. Initialize the velocity of each particle.
4. Evaluate the objectives.
5. Store the position
6. Initialize the memory of each particle.
7. **WHILE:** max no of iterations has not yet reached

For each particle **DO:**

(a) Compute the velocity
\[
V_{id} = W V_{id} + C_1 \text{rand.}(P_{id} - X_{id}) + C_2 \text{rand.}(P_{gd} - X_{id})
\]  
(4.7)

(b) Compute the new position by adding the velocity to the previous position
\[
X_{id} = X_{id} + V_{id}
\]  
(4.8)
(c) Constrain the position so that it does not exceed the grid size
(d) WHILE: the particle has not formed a network
   (i) Add random numbers in the interval [0 1] to position the particle.
   (ii) Constrain the position so that it does not exceed the grid size.
(e) Evaluate the objective values of the current position.
(f) Compare the new position with members in the memory
(g) Update the memory by inserting all the current non-dominated positions and eliminate any dominated locations.
(h) When the new position dominates the local best, replace the local best
(i) Locate the member that dominates the fewest particles in this iteration as the global best.
(j) Increase the loop counter
8. Return the memory as the non-dominated solution set.

4.5 SIMULATIONS AND DISCUSSIONS

The proposed approach is developed using Matlab R2009 and executed in a PC with Pentium IV processor with 2.40 GHz speed and 256 MB of RAM. Here 60 sensors are placed within each cluster of the network. The targets are placed at random locations in the sensor field. The mobile agent parameters (R. Rajagopalan, 2005) are listed in Table 4.1. The data processing time and power level of each sensor is chosen randomly from the given range.
Table 4.1 Mobile agent parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile agent size</td>
<td>400 bytes</td>
</tr>
<tr>
<td>Average data size</td>
<td>100 bytes</td>
</tr>
<tr>
<td>Transmitter gain</td>
<td>2</td>
</tr>
<tr>
<td>Receiver gain</td>
<td>2</td>
</tr>
<tr>
<td>Channel operation frequency</td>
<td>20KHz</td>
</tr>
<tr>
<td>Operational power range</td>
<td>100-500 mw</td>
</tr>
<tr>
<td>Carrier frequency</td>
<td>2.4GHz</td>
</tr>
<tr>
<td>Transmitter power range</td>
<td>200-1000 mw</td>
</tr>
<tr>
<td>Channel width</td>
<td>16 bits</td>
</tr>
<tr>
<td>Data acquisition and Processing time</td>
<td>50-100 ms</td>
</tr>
</tbody>
</table>

Table 4.2 Genetic parameters and PSO Parameters

<table>
<thead>
<tr>
<th>Genetic Parameters</th>
<th>PSO Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Population Size: 200</td>
<td>Max. Iteration: 100</td>
</tr>
<tr>
<td>No of generations: 600</td>
<td>Swarm Size: 50</td>
</tr>
<tr>
<td>Crossover fraction: 0.9</td>
<td>Inertia constant: 1.0</td>
</tr>
<tr>
<td>Crossover fraction: 0.9</td>
<td>$c_1$ and $c_2$: 2</td>
</tr>
</tbody>
</table>

The parameters listed in table 4.2 were used to examine the output of genetic algorithm and the proposed method. Figure 4.3 shows the route obtained when running genetic algorithm. Figure 4.4 shows the route obtained by hybrid optimization technique.
Figure 4.3 Route obtained by genetic algorithm

Figure 4.4 Route obtained by hybrid optimization technique

The proposed technique makes the mobile agent capable of visiting nodes in the target neighborhood while finding an energy saving route. The FC node located in the center of the region particularly possesses a rectangular shape to differentiate from the regular leaf nodes. The minimum acceptable amount of signal
energy detected by an individual sensor node is five and inactive nodes in the sleep state do not detect any signal energy. A single target is arbitrarily placed in the network region. The area of near the target represents the sensing zone within which the sensor nodes usually detect higher signal energy than others far away from the target. The PSO successfully discovered a clean route that goes through every node within the sensing zone of the target while avoiding the nodes in sleep state. The global random optimization strategy makes the proposed technique capable of exploring any potential sensing zones and visiting nodes in the target neighborhoods with the highest priority for an optimal detection performance in addition to preserving an energy-saving route.

The simulation results show that the multiobjective approach is successful for solving the mobile agent routing problem as compared to the weighted genetic algorithm. The performance is compared with combinatorial optimization approach that produces a single solution. Figures 4.5 and 4.6 show that, the number of hops in the route and the detection accuracy increases with the network size. In the proposed algorithm the no of hops is less when the network size increases. As the network size increases the detection of accuracy also gets increased for the proposed algorithm. As the number of node increases the mobile agent visit more number of sensors in the given network.

Figure 4.5 No of hop Comparison
Figure 4.6 Detection accuracy Comparison

As the number of hops increases the energy consumption and the path loss also increases, which is shown in Figures 4.7 and 4.8. The combinatorial algorithm is simple but suffers poor performance in terms of path loss. The path loss and the energy consumption depend on physical distance between the sensors and the number of hops in the route. As the network size increases the path loss gets decreased for the proposed algorithm when compared to the combinatorial algorithm. The energy consumption by the proposed algorithm is less as the network size increases.

Figure 4.7 Path loss Comparison
Figure 4.8 Comparison of energy consumption

4.6 SUMMARY

A hybrid optimization technique for routing of sensor nodes is proposed in this chapter. The Chapter proposes a multiobjective approach for mobile agent routing in distributed sensor networks. This technique is able to discover satisfying routes with high detected signal energy while minimizing the energy consumption and path loss. It obtains routes with fewer numbers of hops even when the network size is large. The forth coming Chapter discusses on collision avoidance in wireless sensor networks.