Chapter 4

An Energy Efficient Routing Algorithm for Target Tracking in Directional WSNs

4.1 Introduction

The process of selecting a path for establishing a communication link from source node to the destination node in a network termed as routing. The main focus of the existing routing protocols in WSNs is to reduce the energy consumption and provide the desired level of connectivity in WSNs.

Fuzzy logic is a form of many-valued logic in which the truth values of variables may be any real number between zero and one. The authors in Ahvar et al. (2013); Al-Kiyumi et al. (2018); AlShawi et al. (2012); Brante et al. (2013); Dima et al. (2014); Dutta (2015); Jaradat et al. (2013); Manjate et al. (2018); Sran et al. (2018); Wang et al. (2010) use the fuzzy logic to solve the various problem of routing in WSNs, such as energy-efficient, prolong the network lifetime, network load balancing, and reduce the network delay. The authors used the residual energy of the sensor nodes, total power consumption for communicating the tracking information, network delay, hop count, and packet loss as the variables in designing a many-valued logic with fuzzy logic. A route selection metric in a WSN is used to select a routing path out of available multiple paths based on the requirement of the users. The input variables in fuzzy logic help to estimate the route selection metric based on the requirement of the users. The examples of the requirements of the users are energy-efficiency, fault tolerance, low packet loss, etc.

In this chapter, we consider the regular sensor nodes deployment patterns for tracking of moving targets. *We address the problem: How to estimate the route selection metric which provides an energy-efficient routing path for relaying the moving target tracking information in a directional WSN?*

Major Contributions: To the best of our knowledge, this is the first work to address the problem of estimating the route selection metric, determining a regular node deployment pattern, and locations for tracking the moving targets in an energy-efficient directional WSN. Along with this, the major contributions of our work are as follows:

- We formulate a direction sensor node deployment problem for tracking of moving targets inside the FoI. The object of the problem is to estimate the locations and select the regular deployment pattern which provides the tracking and connectivity of the network.

- We use Fuzzy Logic System (FLS) for estimating the route selection metric Al-Kiyumi et al. (2018). The transmission energy, neighbor energy consumption, hop count, target tracking percent, and residual energy are the input variables of FLS. The estimated location of the sensor nodes uses to calculate the inputs of FLS.

- We use multiclass logistic regression for predicting the quality of tracking by supplying distance from traget and duration of tracking as input to the regression model. The outcome of this system is three classes '0', '1' and '2'. Similar to that of fuzzy logic 'low', 'medium' and 'high',class '0' indicates the quality of tracking is lowest and class '2' indicates the quality of tracking is highest.

- We propose an energy-efficient routing algorithm by using the estimated route selection metric. The algorithm selects the shortest path for reducing the energy consumption, consider the residual energy for providing the stability of the selected path, and using tracking percent for the maintain the quality of the tracking. The propose routing algorithm can be integrated with any existing routing protocols.

- We demonstrate an application of the propose energy-efficient routing algorithm in the deployment of a **M**oving Target **T**racking **S**ystem based on directional wireless sensor Network called MTS. We consider indoor and outdoor Scenarios for validation of our work. Using the directional tracking devices regularly deployed in such Scenarios for estimating the energy consumption, lifetime of the network, and stability of the network. We compare the proposed work with the existing works Dutta (2015); Guo et al. (2013); Gupta and Rao (2016). We also demonstrate the application of the MTS for counting the moving objects pass from the deployed region.

Motivation: Table 4.1 summarized the features considered in the exiting work for finding an efficient routing path. The important observations of this table are as follows. First, it shows that none of the existing work considered deterministic deployment of sensor nodes for the energy-efficient routing protocol. Next, the exiting work also not considered all five fuzzy logic important for estimating the route selection metric. Finally, very limited work used hardware implemented for validating the proposed work. Such limitations in the existing work motivate us. In this work, we propose an energy efficient routing algorithm which considers all important inputs for estimating the route selection metric. We use deterministic deployment of a network, which requires less number of sensor nodes and therefore reduces the cost of the WSNs. We use a hardware implementation for validation of our work and comparison with existing work Dutta (2015); Guo et al. (2013); Gupta and Rao (2016).

Table 4.1 Fuzzy based approaches in routing protocol of WSN

Paper	Addressing issues	Routing protocol	Validated	Tool	Deployment Type	Fuzzy System Input				
						Energy Consumption	Hop Count	Residual Energy	Transmission Load	Routing Delay
Ahvar et al. (2013)	Identification of fuzzy rule and fuzzy set for WSN protocol.	MHR	Simulated	GloMoSim	Random	√	√	—	—	—
Messaoudi et al. (2016)	Cluster head Election	LEACH	Simulated	Matlab	Random	—	√	√	—	—
Al-Shawi et al. (2012)	Optimal path selection	LEACH	Simulated	Matlab	Random	√	—	√	√	—
Guo et al. (2013)	Reliable end-to-end delivery	CTP	Simulated	CTP debug tool	Deterministic	—	—	—	√	√
Wang et al. (2010)	Energy optimization for target tracking	—	Simulated	—	Random	√	√	—	—	—
Dutta (2015)	Effective maintenance of connectivity	AODV	Simulated	Matlab and NS-2	Random	√	—	√	—	—
Al-Kiyumi et al. (2018)	Energy balancing and efficiency	—	Simulated	Matlab	Random	√	—	√	—	—
Jaradat et al. (2013)	Energy efficient clustering	LEACH	Simulated	NS-2	Random	√	—	√	—	—
Dima et al. (2014)	Congestion reduction in network	—	Simulated and Hardware implemented	Matlab, MSP430 CPU	—	√	√	—	√	—
Hoang et al. (2013)	To enhance the network lifetime	mLEACH	Hardware Test-bed	TinyOS	Random	—	—	—	—	√
Gharajeh and Khanmohammadi (2016)	Increasing network lifetime		Simulated	—	Random	—	√	—	—	√
Hassani et al. (2012)	Detection of rotating object		Simulated	—	Random	—	—	—	—	√
Feyzi and Sattari-Naeini (2015)	Efficient Routing through fuzzy logic	AODV	Simulated	NS-2	Random	—	√	—	√	√
Brante et al. (2013)	Increasing communication performance	—	Simulated	—	—	√	√	—	√	√

The rest of the chapter is organized as follows: In the next section, we state the assumptions and Preliminaries and models that we use in the proposed target tracking system. We propose an energy-efficient routing algorithm using FLS in Section 4.3. Section 4.4 presents an application of the proposed energy-efficient routing algorithm, *i.e.*, MTS and conclude the chapter in Section 4.5.

4.2 Preliminaries and Models

This section introduces fuzzy logic, features of directional sensor nodes, and regular deployment patterns of directional sensor nodes for tracking the moving targets in the FoI. We discuss the target tracking model, communication model, and energy consumption model, that we use in this work. We also define the lifetime of the WSNs.

4.2.1 Fuzzy Logic

Fuzzy Logic (FL) finds their applications in various domains in WSNs due to its ability of combining and evaluating the diverse parameters. Examples of such applications are routing protocols, target tracking, indoor navigation system Brante et al. (2013); Feyzi and Sattari-Naeini (2015); Gharajeh and Khanmohammadi (2016); Guo et al. (2013); Hassani et al. (2012); Hoang et al. (2013). Easier implementation, the robustness of decision making, and the capability of processing a non-linear system make FL more efficient Kulkarni et al. (2011).

Fuzzy logic analyses the information using a definite set called fuzzy set. The fuzzy set also known as uncertain set, holds the linguistic terms. Examples of such linguistic terms are *Low*, *Medium*, and *High*. Usually low and high values of linguistic terms are equal to zero and one, respectively. The value of other linguistic terms lies between zero and one. Membership function represents the degree of truth in fuzzy logic. It characterizes input and output of the fuzzy logic. The membership function in fuzzy logic consists core, support, and boundary features as shown in Fig. 4.1. The core, support, and boundary features of a membership function are the regions of the universe that are characterized by full, nonzero, and nonzero but incomplete membership in the set, respectively. It is shown that the core feature provides the highest degree of trust in fuzzy logic.

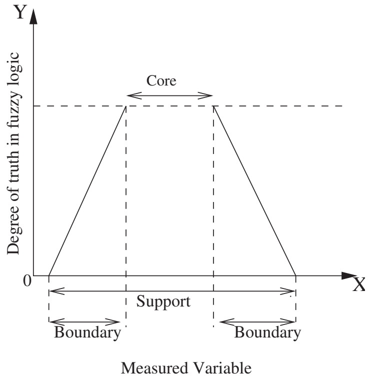


Figure 4.1 Illustration of a fuzzy logic with features.

A Fuzzy Logic System (FLS) consists of fuzzification, interface engine, defuzzification, and rule base components as shown in Fig. 4.2. The fuzzification is a process of transforming a crisp set to a fuzzy set. The support fuzzification and grade fuzzification are the methods used for transforming the crisp input values into linguistic variables. The defuzzification is a process of reducing a fuzzy set into a crisp set or to convert a fuzzy member into a crisp member. Max-membership, centroid, weighted average, and mean-max membership are the methods used for defuzzification in the FLS. The interface engine is a process of mapping from a given input to an output using fuzzy logic. The fuzzy rules are the basis of the fuzzy system; it illustrates the dynamic nature of a system. This judgment of membership value is not at random; they are obtained through numerical data analysis or some human expert. The rule base in an FLS, is a collection of IF-Then rules associated statement. The statement after IF is called the antecedent whereas the statement after THEN is called consequent. Let $x_1, x_2, \cdots, x_m$ and y are the linguistic terms. The FLS uses the following IF-THEN rule: **IF** $input_1 \rightarrow x_1$ **and** $input_2 \rightarrow x_2$ **and** $\ldots input_m \rightarrow x_m$, **THEN** $output \rightarrow y$.

Regular directional node placement patterns in the FoI.

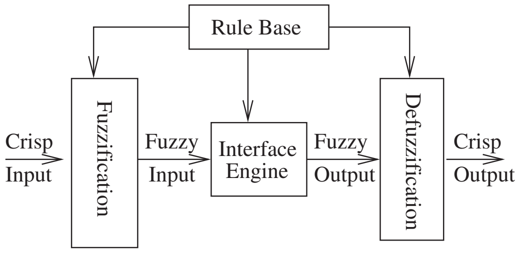


Figure 4.2 Illustration of the components of a Fuzzy Logic System (FLS).

4.2.2 Logistic Regression

The logistic model is a type of statistical model that is applicable for the binary variable. The logistic regression does not take any prior knowledge in as input whereas the fuzzy logic model is first based on the apriori knowledge; it may be from some expert or human activity recognition. Secondly, the fuzzy logic approach uses some pre-train dataset Honzik et al. (2010). In calculation and interpretation, fuzzy models are more complicated than logistic regression. Logistic regression is used to acquire knowledge and this in term use as the base for the fuzzy interface system. So, in-spite of going one step ahead after knowledge acquisition logistic regression is preferable in case of new classification to be made. In this work we are using multiclass logistic regression, the distance and duration of tracking are two parameters which we can not predict prior as the target arrival pattern is not following any distribution Fernandez-Peralta et al. (2017). The procedure for stochastic gradient descent is given in Procedure 1. The logistic regression comprises of logistic function $\sigma(t)$, which is a sigmoidal function. The logistic function is defined as:

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$$

$$t = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

$$p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}} \tag{4.1}$$

where, $p(x)$ is probability of depenedent variable, β_0 is coefficeint independent on x, β_1 is coefficient of parameter x_1 and β_2 is coefficient of parameter x_2. Let distance and duration are denoted by x_1 and x_2, respectively.

$$Output = \begin{cases} class\ 2 & \text{if } 0.666 < p(x) \leq 1 \\ class\ 1 & \text{if } 0.333 < p(x) \leq 0.666 \\ class\ 0 & \text{otherwise} \end{cases}$$

4.2.3 Target Tracking Model

A moving object is said to be tracked, if the location of the moving object is inside the given region of at least one deployed sensor node in the WSNs. Such sensor node and the region are called the tracking device and tracking region, respectively. In this work, we consider the directional WSNs that provide better accuracy and consume less power as compare with omnidirectional WSNs. The main characteristics of the directional WSNs are the tracking angle, range, and the orientation of the nodes as shown in Fig. 4.3. We consider the homogeneous WSN in this work, where the characteristics of all deployed tracking devices are the same.

- The *tracking angle* describes the angular range that a sensor node can track the moving target. A directional sensor node has a fixed angle range, denoted by ϕ, in a given direction $\phi \leq 360°$. The value of the tracking angle of a given directional sensor node regulates the cost of the sensor node and the energy consumption of the node. A sensor node consumes more energy and become expensive if the tracking angle increases. However, it also provides more tracking region in the given FoI.

- The *tracking range* of a directional sensor node, denoted by R_s, is the maximum range of the node within which the node can track a target in the FoI.

- The *orientation* of the tracking region is defined as the orientation to which a directional node heading in the FoI. Fig. 4.3 shows that the orientation of a sensor s, located at (x, y), is denoted by $\theta_{x,y}$, where $0 \leq \theta_{x,y} \leq 2\pi$.

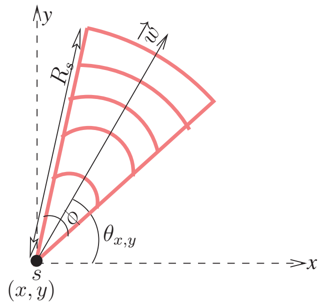


Figure 4.3 A directional sensor node with its characteristics.

4.2.4 Deployment of Directional Sensor Nodes

In this chapter, we consider deterministic deployment of directional sensor nodes, where the sensor nodes are manually placed in a given FoI. In continuation of work in previous chapter, we use three possible regular patterns for placement of sensors, *i.e.*, equilateral triangle, square, and hexagon. A triangular deployment pattern consists of sensor nodes deployed on the vertices of the triangle as shown in part (a) of Fig. 4.4. Next, the vertices of the square are the deployment points in case of square deployment of the sensor. Finally, part (c) of Fig. 4.4 depicts the hexagonal deployment of the sensor nodes.

Let two sensor nodes in a given pattern has d minimum distance. For tracking a moving target in a given FoI, the point $v \in \mathbb{V}$ need to lie in the tracking region of at least one sensor node. If the communication range is more than the tracking range of the deployed sensor nodes, point v needs to consider only the communication neighbors. They themself is provide the tracking of the FoI. Similarly, if the tracking range is more than the communication range of the deployed sensor nodes, point v needs to consider only the tracking neighbors. We consider these conditions for developing a connected directional WSNs.

Let the angle between two tracking regions in the triangle, square, and hexagon patterns, denoted by θ_p, are $\{\pi/3, \pi/2, \pi/3\}$, respectively. Here $p = 0$, 1, and 2 denote the triangle, square, and hexagon patterns, respectively. The width of a pattern is denoted by d as shown in Fig 4.4. The height of the pattern is therefore width × angle between tracking regions, *i.e.*, $d \sin(\theta_p)$. The coordinates of the sensor nodes in a triangle pattern, denoted by (x, y), are given by

$$y \in g \times d \sin(\theta_0) \textbf{ and } x \in g \times d + \frac{d}{2} \cos\left(\frac{\theta_0}{2}\right) mod\left(\frac{y}{d \sin(\theta_0)}, 2\right), \tag{4.2}$$

where $g = \{0, 1, 2, \dots\}$. Here, we use $mod(y/d \sin(\theta_0), 2)$ for shifting the alternate nodes by $0.5d \cos(\theta_0/2)$ in x axis. Part (b) of Fig. 4.4 shows that the coordinates of the nodes in square pattern are given by

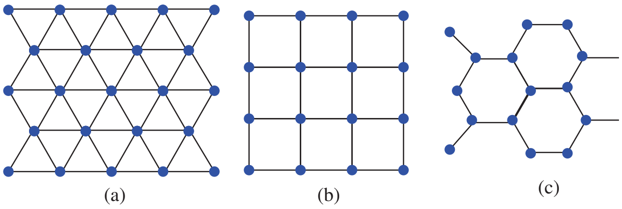
$$x \in g \times d \textbf{ and } y \in g \times d \sin(\theta_1). \tag{4.3}$$

Part (c) of Fig. 4.4 shows that the nodes are not located at the center of the hexagon pattern. We use Eq. 4.2 for estimating the coordinates of the nodes in a hexagon pattern with the following condition: **if** $mod(y, 2) = mod(x, 3) + 1$ **then** skip the deployment of the node. From Eqs. 4.2 and 4.3, the coordinates of a node at $v \in \mathbb{V}$ in a pattern p is given by

$$\left(gd + d \cos\left(\frac{\theta_p}{2}\right) \cos(\theta_p)\, mod\left(\frac{y}{d \sin(\theta_p)}, 2\right), gd \sin(\theta_p)\right). \tag{4.4}$$

4.2.5 Communication Model

We consider a connected directional WSNs where a sensor node can be communicated with other sensor nodes in the network. Communication range of a sensor node, denoted as R_c, is the maximum distance between two sensor nodes such that they can communicate with each others. The communication region of a sensor s, denoted by $A(s, R_c)$, is a region



(a) (b) (c)

Figure 4.4 Regular directional node placement patterns in the FoI. Parts (a), (b), and (c) illustrate the equilateral triangle, square, and hexagon patterns, respectively.

such that s can communicate with any sensor node located in this region. We assume that all sensor nodes in the directional WSNs have equal communication range.

4.2.6 Energy Consumption Model

The energy consumption model is used to estimate the route selection metric in the propose energy-efficient routing algorithm. When a sensor transmits one bit data directly to a receiver sensor over a distance d, the energy consumed for reception (E_r) and transmission $(E_t(d))$ can be calculated as follows:

$$E_r = k_1\ , and E_t(d) = k_1 + k_2 d^{\alpha}$$

where k_1 is the energy consumed by the transceiver electronics per-bit and k_2 is the energy consumed in the transmitter unit per-bit. Based on, the typical values of these parameters are: $k_1 = 3.32\ \times 10^{-7}$ J/bit and $k_2 = 8\ \times 10^{-11} J/bit/m^2$ for α= 2, where α is the power index of the propagation path loss, which is typically between 2 and 4 Gupta and Rao (2016).

4.2.7 Lifetime of WSNs

One of the inherent problem in WSNs is of unbalanced energy consumption, which mainly arises due to the many-to-one traffic pattern. This traffic pattern will reduce the lifetime of WSNs AlShawi et al. (2012). The rate of battery depletion will be faster in the case of sensor node becoming the disconnection point between two network partitions. Entire traffic from one part of the network reaches via one node to another part of the network. Thus, it can be said that the lifetime of WSNs is the necessary parameter for evaluating the performance of any routing protocol. The time up to which the first sensor node in the WSN retains its energy or depletes its energy till power off or non-functional is called the **lifetime of WSNs**. Thus, for enhancing the lifetime of WSNs, the routing algorithm must play a role in the fairway so that network stays longer.

4.3 An Energy-Efficient Routing Algorithm

In this section, we propose a routing algorithm that enhances the lifetime of directional WSNs. We use FLS for reducing the power consumption and finding a suitable path between the source device and destination. A directional sensor node work as a source if the node tracked a moving object in the tracking region. This section first presents the crisp inputs that use to design the FLS for energy efficient routing algorithm. Next, we use the inputs for estimating the route selection metric. Finally, we use the metric to propose a fuzzy logic based energy-efficient routing algorithm for directional WSNs.

4.3.1 Inputs of the FLS

The proposed routing algorithm considers the power consumption, quality of target tracking, and stability of the selected path metrics as inputs of the FLS.

- **Reducing the power consumption:** Low power consumption prolongs the lifetime of the directional WSNs. We use transmission energy, neighbor energy consumption, and hop count for reducing the power consumption.

- **Improve the quality of target tracking:** The main objective of the deployment of the sensor nodes is to track the moving targets. We use the percentage of target tracking for proving the better tracking quality. The percentage of target tracking is equal to unit(1) if it is very closed to the sensor node. The percentage is equal to null(0) if the distance between the target and the sensor node is almost equal to the tracking range of the sensor.

- **Enhance the stability of the path:** The stability of the selected path provides the low packet loss and overhead of the WSNs. In this section, we use the residual energy of the sensor nodes in the selected routing path for maintaining the stability of the path.

Reduce the Power Consumption (RPC)

The value of hop count in a given path indicates the distance between the source and a given sensor node. Fig. 4.5 shows that the value of hop count between source node a and sensor m is three. It helps to estimate the required power for routing the tracking information. The higher value of hop counts require more number of relay nodes to transmit the tracking information and therefore increases the power consumption of the WSNs.

A sensor node in WSNs consumes the power during tracking a target, processing the information, and transmitting the tracking information to the destination. The transmission energy is the sum of the total required energy for relaying or transmitting the tracking information from source node to the destination node. Let h_i denoted the required hop count for relaying the information between a source node a and given node m in a path i. Let $\|d_{ab}\|$ is the distance between directly connected sensor nodes a and b. From Dutta (2015), the required transmission energy for relaying a L bit size tracking information is given as

$$E_{mL}^{am} = \sum_{n=1}^{h_i} (L(k_1 + k_2 \left\| d_{ab} \right\|^{\alpha}) + Lk_1). \tag{4.5}$$

The neighbor energy consumption is the required power for forwarding a data packet between two neighbor sensor nodes. From Eq. 4.5, the neighbor energy consumption that requires to transmit L bits between sensor node a and sensor node b, is given by

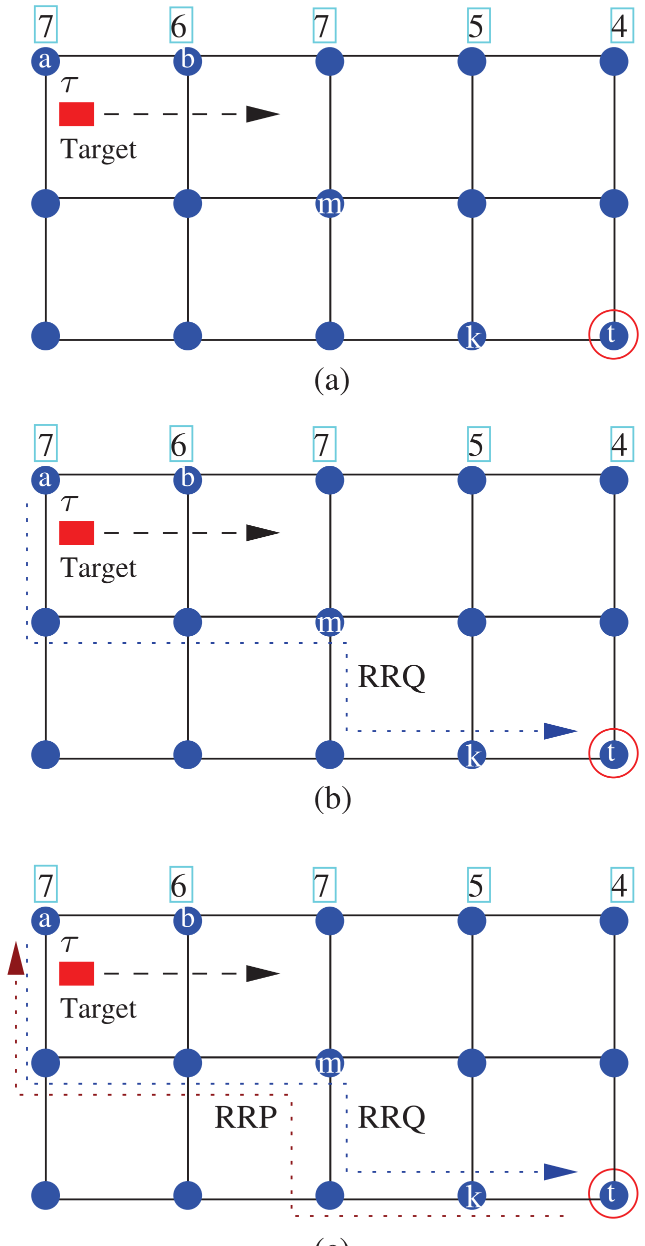
$$E_L^{ab} = L(k_1 + k_2 \left\| d_{ab} \right\|^{\alpha}) + Lk_1. \tag{4.6}$$

Note that the higher values of hop count, transmission energy, and neighbor energy consumption, do not prolong the lifetime of the given WSNs. Therefore, all these inputs of the FLS are negative linguistic variables. Fig. 4.6 illustrates the linguistic variables of the above inputs of the FLS. It shows that the lower values of the inputs given higher linguistic variable for prolonging the lifetime of the WSNs.

Improve the Quality of Target Tracking (IQT)

A directional sensor node tracks a target only if the location of the moving target is within the tracking range of the sensor node. In some Scenario, a target is tracked by more than one sensor node, *i.e.*, the location of the target lies in the tracking region of multiple sensor nodes. The proposed routing algorithm selects only one sensor node for forwarding the tracking information to the destination. The logistic regression helps to select a sensor node which provides more meaningful tracking information. It is true that a sensor node can better track a mobile object if the distance between the sensor node and the object is less than the other sensor nodes. Fig. 4.5 shows that a sensor node tracks the target with more accuracy than b sensor node.

We consider the distance between the sensor nodes and the duration of stay of the mobile object lying in the tracking region of the sensor node as the inputs of logistic



(a)

(b)

(c)

Figure 4.5 An example of the route selection technique using squared deterministic deployment. The sensor nodes are directional in nature. Value at the sensor nodes represent the residual energy. Part(a) of figure illustrates a WSNs with a target τ entering in the FoI. Part(b) and Part (c) show the selected paths using fuzzy logic for RRQ and RRP, respectively.

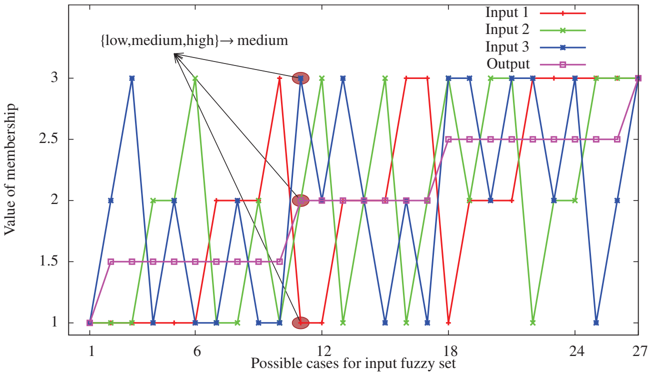


Figure 4.6 Illustration of three inputs *i.e.,* hop count, neighbor energy consumption, and transmission energy to fuzzy system and its corresponding output.

regression. To estimate the distance, we use three received signal strength levels '0', '1', and '2' as three different classes of the above inputs of the logistic regression. The value '2' indicates that it is belong to class '2' indicates that its probability is more than 0.666 and hence quality of tracking is higher than other two classes *i.e,* '0' and '1'. Similarly for '1' probability is between 0.333 and 0.666 and its quality of tracking is medium. For '0' probability is less than 0.333 and its quality of tracking is lowest. The high value indicates that the target is very close to the sensor node. Similarly, more duration of staying the object in the tracking region of a sensor node improves the quality of the tracking system. Such inputs (distance and duration) are therefore the input parameter in logistic regression.

Remaining Energy of the Sensor Nodes (RES)

Considering the Remaining Energy of the Sensor nodes (RES) in a selected routing path, is an important metric for proving the stability of the WSNs. Such stable path reduces the cost of path selection and control packet overhead. Let e_{ij}^{ab} denotes the energy of the j^{th} sensor in i^{th} path between sensor nodes a and b, where $i \leq t$, $j \leq h_i$, t is total possible paths between a and b, and h_i is the hop count in i^{th} path. The residual energy in the i^{th} path is therefore given by

$$e_i^{ab} = min\{e_{i1}^{ab}, e_{i2}^{ab}, e_{i3}^{ab} \dots e_{ih_i+1}^{ab}\} \tag{4.7}$$

The higher value of the residual energy increases the stability of the network. We therefore state that the residual energy works as a positive input in the route selection technique.

4.3.2 Route Selection Metric for Routing Algorithm

The main objective of the propose algorithm is to find a routing path which prolongs the lifetime of the WSNs, enhances the stability of selected path, and provides the desired quality of the target tracking. We use RPC, IQT, and RES for estimating the value of *route selection metric* for selecting a path between the source sensor node and the destination. The FLS uses the *if-then rule* for estimating the value of route selection metric by using RPC, IQT, and RES. Let φ_{ab} denotes the route selection metric of a path between a source a and destination b. As we have RPC is the negative input, therefore $\varphi_{ab} \propto \frac{1}{RPC}$. The IQT and RES are the positive inputs in the propose algorithm, *i.e.*, $\varphi_{ab} \propto IQT$ and $\varphi_{ab} \propto RES$. We can rewrite the route selection metric as

$$\varphi_{ab} \propto \frac{\{IQT, RES\}}{RPC}. \tag{4.8}$$

Fig. 4.7 illustrates the route selection metric using RPC, IQT, and RES. An example Scenario of the route selection metric is given as:
IF ($RPC \rightarrow low$ and $IQT \rightarrow high$ and $RTS \rightarrow high$),
THEN (route_selection_metric → *low*).

4.3.3 Routing Algorithm

In this section, we propose an energy efficient routing algorithm for tracking the moving targets in directional WSNs. We use the estimated route selection metric for finding a routing path which prolongs the lifetime of the network with maintaining the quality of the tracking of the moving targets.

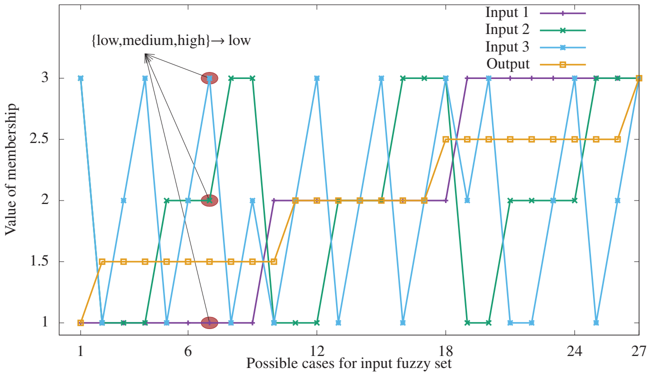


Figure 4.7 Illustration of three inputs *i.e.,* power consumption, quality of tracking, and stability of path to fuzzy system and its corresponding output.

The WSNs consist table-driven and on-demand routing protocols for relaying the tracking information to the destination. In this work, we use table driven approach for designing an energy efficient routing algorithm in directional WSNs. However, the estimated route selection metric can be used with both types of routing protocols. The label on a given sensor node shows the residual energy of the sensor node. We use the square deployment pattern in this example Scenario. Each sensor node is connected with four other neighbor nodes and therefore it consists multiple paths for routing the tracking information. If we use triangle and hexagon regular deployment pattern then each sensor node has three and six neighbor sensor nodes, respectively.

The propose routing algorithm is similar to the on-demand routing protocol. The main difference is the route selection metric. The propose algorithm uses proposed route selection metric for selection routing path. The routing algorithm consists of the following steps.

- Let a and b are the sensor nodes which detect the moving targets in the FoI. The algorithm first calculates the quality of the tracking. We use the distance between a target and the location of the sensor node as the quality of the tracking. Such value we can easily calculate by the signal strength of the reflected signal from the moving object. The quality of the tracking work as IQT input in FLS. The quality of the tracking of a and b

are denoted as IQT_a and IQT_b, respectively.

• Next, sensor nodes that detect the targets broadcast route request message (RRQ). This message consists the source sensor node identifier, destination identifier, residual energy, and the quality of the tracking. We also add the travel distance metric which is nothing but the hop counts from the source node to the current sensor node. Let t is the destination of the routing algorithm. The sensor node uses the FLS to estimate the route selection metric as proposed in Section 4.3.2 and Eq. 4.8. The RRQ message generated by a and b are $\{a, t, \phi_a\}$ and $\{b, t, \phi_b\}$, respectively.

• When an intermediate sensor node m receives a RRQ message, first check whether the sensor node is a destination node. If the sensor node is not a destination node, m uses Eq. 4.5 and Eq. 4.7 for estimating the updated hop count and residual energy metrics. m updates its routing table. If m receives multiple RRQ, it selects a sensor node which consists high value of fuzzy output. Here the fuzzy output is nothing but route selection metric. We use the FLS to estimate the fuzzy output as proposed in Section 4.3.2 and Eq. 4.8. Let p is sensor node which consists maximum value of route selection metric. The node p acts as the neighbor of m. The routing table of m appends the following information source identifier, destination identifier, route selection metric, identifier of message sender, *i.e.*, $\{a, t, \phi_m, p\}$. Finally, the sensor node broadcasts updated RRQ message $\{a, t, \phi_m\}$ to its neighbors.

• If the RRQ message is received by destination node t, it repeats the same procedure for estimating the route select metric. t selects a sender node which provides maximum route select metric. Let a sensor node k gives maximum route select metric. Sensor node t generates route reply message (RRP) and sends the message to k. The RRP message format is given as $\{t, a, k\}$.

• When a sensor node received RRP message from a sensor node k, it checks the routing table and sends the RRP message to its neighbor node. The sensor node updates its routing table and appends k for forwarding the data packets. If the sensor node is the source node, it starts routing the data packets to the next selected node.

The complete procedure shown in Algorithm 2.

Algorithm 2: Energy Efficient Routing using Fuzzy Logic

```
/* Let sensor nodes $a$ and $b$ first detect the target $\tau$ when it comes in their sensing range.*/
Sensor nodes $a$ and $b$ broadcast the RRQ messages $\{a, t, \phi_a\}$ and $\{a, t, \phi_b\}$, respectively.
/* Let $m$ and $t$ are intermediate and destination node, respectively.*/
/* $\phi_a$ and $\phi_b$ are the residual energy of node $a$ and $b$ and $IQT_a > IQT_b$*/
while $m$ is not $t$ do
    if $m$ receives single RRQ then
        $m$ estimates hop count and residual energy using Eq. 4.5 and Eq. 4.7.
        $m$ updates its routing table.
    if $m$ receives multiple RRQ then
        $m$ computes route selection metric using Fig. 4.6, 4.7 for each RRQ.
        /* Let $p$ is any intermediate node*/
        if $p$ has maximum value of route selection metric then
            $p$ acts as neighbor of $m$.
            $m$ appends $\{a, t, \phi_m, p\}$ in its routing table.
            $m$ broadcasts RRQ message $\{a, t, \phi_m\}$.
if $m$ is $t$ then
    /*Let $k$ is an interference node at one hop distance from $t$*/
    $t$ unicasts RRP message $\{t, a, k\}$ to node $k$ and $k$ further forwards the RRP till source $a$ is reached.
```

4.4 Moving Target Tracking System

In this section, we developed a prototype to evaluate the performance of the energy efficient routing algorithm. We developed a Moving target Tracking System (MTS) using directional WSNs for tracking the targets, identifying the moving direction of the targets, and counting the targets passed from the FoI in a given time period. We discussed hardware components, communication protocols between devices, networking parameters, and the deployment Scenarios used in the prototype. We also discussed the performance metrics to evaluate the performance of the proposed work.

4.4.1 Specification of Hardware

We considered indoor and outdoor Scenarios for conducting the experiments. We used the following devices for tracking the targets and routing the tracked information to the destination. Wireless Sensor Networks (WSNs) is one of the promising techniques comes into existence due to advancement in microcontrollers technology. WSNs find their applications in various domains. Examples of such domains are monitoring the given Field of Interest (FoI), tracking of the moving objects, and the capture of the events. The deployment of sensor nodes in WSNs can be divided into deterministic and random based on the placement strategy of the sensor nodes. The sensor nodes in random deployment scatter uniformly at random in the FoI. The deterministic deployment uses some given patterns of placement of sensor nodes in the given FoI. The authors in Zhang and Hou (2006) proved that the ratio of the required number of sensor nodes for the desired level of tracking of the targets in random deployment and deterministic deployment is equal to six.

A sensor node in WSNs consists of sensing, communication, processing, and storage units. The sensing unit consists of the sensors, which detects and responds to some input from the physical environment. Such sensors are directional and omnidirectional based on the architecture of the device. Directional sensors detect the events in a given direction, whereas omnidirectional doesn't have any fixed direction. The authors in Tripathi et al. (2018) shown that the directional sensor nodes consume less power than omnidirectional sensor nodes.

Target tracking is an important application of WSNs. The main objective of the target tracking is to detect a moving target in the FoI as soon as possible. A sensor node in WSNs works as a tracking device if the device detects the moving targets inside the given FoI. Triangles, squares, and hexagons are the only possible regular shapes which tessellate by themselves. The deterministic deployment of sensor nodes usually uses the regular shapes for placement of sensor nodes. It helps to reduce the cost of the network and provides the desired quality of the monitoring of the moving target. The authors in Ammari (2009) illustrates that the deterministic deployment of directional sensor nodes reduces the interference problem, cost of the sensing unit, and the power consumption.

Communication between the sensor nodes in a WSN is an important metric to measure the quality of connectivity of the WSN. A device in WSNs known as a routing device if the device relying the tracking information from source node to the destination node.

Target Tracking Device

The tracking device is used for tracking the moving target in the FoI. In this work, we used the directional tracking devices which has fixed tracking angle and tracking range. We used proximity sensor and camera for tracking the targets. The CC2650 SensorTag is a wireless MCU targeting Bluetooth Low Energy (BLE) remote control applications. We used *CC2650 SensorTag* for tracking the moving targets. The device is a member of the CC26xx family of cost-effective, ultralow power, 2.4-GHz RF devices. Table 4.2 illustrates the hardware Specification of target tracking device used in the prototype. Fig. 4.8 illustrates CC2650 SensorTag with the sensory data on smartphone. The Sensortag tracks moving targets if it passes near the SensorTag. The main limitation of CC2650 SensorTag is its tracking range. It has very limited tracking range and also does not support the optimization of the existing BLE communication framework. We therefore use Raspberry Pi Camera Board v1.3 for tracking the moving targets. We used 5MP Omnivision 5647 camera module which supports 15-pin MIPI camera serial interface to plug directly into the Raspberry Pi Board. The tracking angle and tracking range are 48.8 degrees and 10 meter, respectively. The camera tracks the moving target and transmits the tracking information to the attached routing device.

Figure 4.8 Devices used in MTS.

Table 4.2 Target tracking devices used in the prototype.

Devices	**Specification**
Target Tracking Device	**CC2650 T1 SensorTag**: Sensors: accelerometer, gyroscope, magnetometer, • Sampling range: $0.1sec - 2.55sec$, • Idle power: $0.88mW$, • Sensing power: $20.35mW$, • Communication power (BLE): $0.15mW$, • BLE communication with sleep mode support. **Camera**: • Raspberry Pi Camera Board v1.3, • Vertical field of view: 48.8 degrees, • Sensor image area: 4.6 mm diagonal, • 15-pin MIPI serial interface with Raspberry Pi.

Routing Device

The target tracking device transfers the tracking information to the routing device in MTS. The main role of the routing device is to communicate the tracking information from the source sensor node to the destination. It also collects the neighbors information and estimates the route selection metric. We used Raspberry PI 3 for communicating the tracking information from source to the destination. The Raspberry PI can collect the tracking information from the CC2650 SensorTag and camera that captured the tracking images. The Raspberry PI can also communicates with other neighbor Raspberry PI using BLE for estimating the route selection metric and communicating the tracking information. The Raspberry PI 3 is equipped with a 1.2GHz 64-bit quad-core ARMv8 CPU, 802.11n Wireless LAN, Bluetooth 4.1, Bluetooth Low Energy (BLE), camera interface, display interface, Micro SD card slot, and VideoCore IV 3D graphics core.

4.4.2 Overview of MTS

The MTS used deterministic deployment of the target tracking devices and the routing devices in a given FoI. MTS uses three types of devices: target tracking devices (18 in number), routing devices (18 in number), and one communication device to the destination device. Fig. 4.9 depicts the configuration of MTS with the devices. It shows that the target tracking devices and routing devices use BLE protocol for communication with each others. The routing devices use the proposed routing algorithm for finding an energy efficient routing path. Since BLE protocol has limited communication therefore the routing devices communicates the tracking information to the destination using communication device. Such connectivity uses Wi-Fi (802.11b) protocols to form a connected directional WSNs.

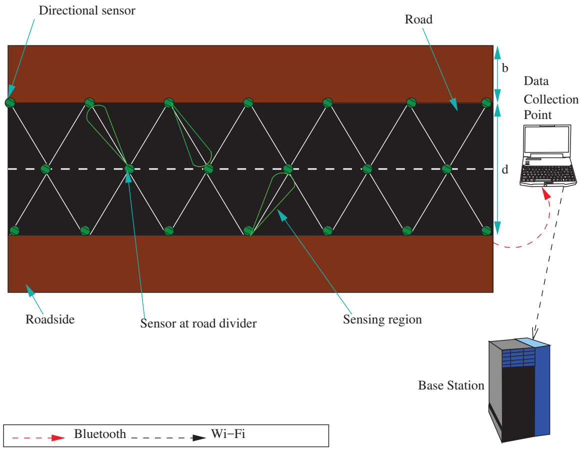


Figure 4.9 Illustration of MTS. The tracking, relay, and communication devices are indicated by rectangle, triangle, and hexagon, respectively.

Deployment of MTS

The experiments were conducted in three different Scenarios:

- **Inside the laboratory:** Part (a) of Fig. 4.10 illustrates a Scenario of deployment of MTS inside the laboratory. This supports the indoor environment with no placement constraints. Such Scenario helps to monitored the activities of the moving targets.
- **Corridor of the department:** Next deployment space of the MTS is corridor inside the department as shown in part (b) of Fig. 4.10. The corridor inside the department supports the slow speed moving targets. Here, the students, faculties, or staffs are only the moving targets. This Scenario is used to count the number of moving targets available inside the department.
- **Campus Road:** Finally, the road in the institute campus is an outdoor Scenario, where the vehicles and the persons are working as moving targets. The Scenario is shown in part (c) of Fig. 4.10, it helps to count the number of targets pass through the road, moving directions of the targets, and the speed of the targets. This is the only outside Scenario that we considered for the experiment.

A CC2650 SensorTag with a proximity sensor and a Raspberry PI, connected to a laptop serve as the tracking sensor. It also relays the tracking information in the experiment. When we use CC2650 SensorTag, the values of tracking range, tracking angle, and tracking direction are $21m$, $2m$, $10m$, $7m$ and $30m$, respectively. We use 20 tracking directional sensor nodes. The FoI is assumed to be rectangle. We substitute these values in Eq. 4.4, and calculate the locations of the tracking sensor nodes inside the FoI in all Scenarios. To validate the tracking information collected by the WSNs, we used a video camera to record the motion of the moving targets and the accuracy was computed by comparing the data from these two sources.

(a) (b)

(c)

Figure 4.10 Experiment setup at IIT (BHU) Varanasi. Part (a) and part (b) illustrate the indoor Scenarios where tracking devices are deployed inside the department. Part (c) illustrates the outdoor Scenario. The devices are connected with each others using BLE and WiFi protocols.

Working of MTS

Fig. 4.10 illustrates the Scenarios where tracking devices and routing devices are placed inside the FoI. The MTS performs the following steps for tracking the moving targets:
Step 1: A tracking device placed inside the deployment region (camera or CC2650 SensorTag) for tracking the movement of the moving object. In part (c) of Fig. 4.10, the device tracked a moving person passes on the road. The CC2650 SensorTag works as a tracking device which transfers the tracking information to the routing device by using BLE protocol. The Raspberry PI camera works as a tracking device directly connected with Raspberry PI by using 15-pin MIPI serial interface connection.
Step 2: The routing device (Raspberry PI) first collects the tracking information from the tracking device. Next, processes the tracking information and extracts the meaningful information. The routing device shares this information with its neighbor routing devices and estimates the *route selection metric*. The routing devices use BLE protocol for sharing the information between them. Finally, the routing device uses the proposed energy efficient routing algorithm for relaying the information to the communicating device. Here we use laptop as communication device.
Step 3: The communicating device forwards the information to the destination by using Wi-Fi (802.11b) protocol.

4.4.3 Experimental Results

The MTS tracks the moving targets inside the FoI. We consider indoor and outdoor Scenarios based on the movement of the tracking objects as shown in Fig. 4.10. Our experimental results consider the following metrics:

1. *Network Lifetime:* The time up to which the first sensor node in the WSN retains its energy or depletes its energy till the power off or non-functional is called the lifetime of WSNs.

2. *Energy Consumption:* The energy consumption of the WSN at time instance t is the difference between the initial energy of all deployed devices and the residual energy at time t (Gupta and Rao (2016)). Let the initial and residual energy at time instance t of a device i are denoted by $E_i(0)$ and $E_i(t)$, respectively. If network consists of n devices, the average energy consumption of the network in percentage is given by

$$\begin{aligned}\text{\% Avg. Energy Consumption} &= \frac{\sum_{i=1}^{n}(\text{ Initial Energy of } i - \text{ Remaining Energy of } i \text{ at } t)}{\sum_{i=1}^{n} \text{ Initial Energy of } i} \times 100, \\ &= \frac{\sum\limits_{i=1}^{n} (E_i(0) - E_i(t))}{\sum\limits_{i=1}^{n} E_i(0)} \times 100.\end{aligned}$$

3. *Residual Energy:* Let $E_i(t)$ denotes the residual energy of a device i at time instance t as given in Gupta and Rao (2016). The average of the residual energy of n deployed devices in the network at t is given by

$$E(t) = \frac{\sum\limits_{i=1}^{n} E_i(t)}{n}.$$

The standard deviation of the residual energy of n deployed devices in the network at t is given by

$$\sigma_t = \sqrt{\frac{\sum\limits_{i=1}^{n} (E_i(t) - E(t))^2}{n-1}}. \tag{4.9}$$

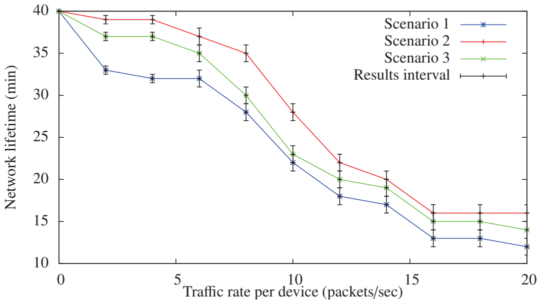
Lifetime of the Network

We study the impact of the proposed algorithm on the lifetime of the WSNs. Fig. 4.11 shows the network lifetime for different values of traffic rate. We increased the traffic rate by increasing the number of moving targets inside the FoI. The results also show the network lifetime when other existing routing algorithms are used Dutta (2015); Guo et al. (2013); Gupta and Rao (2016). The experiment are done in all three Scenarios. Parts (a), (b), and (c) illustrate the results for different deployment patterns.

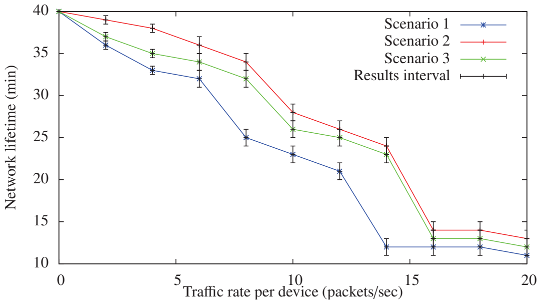
Fig. 4.11 illustrates that the lifetime of the Scenario 2 is maximum and Scenario 1 is minimum in all the deployed patterns. This is because, most of the moving targets (students and faculties) in Scenario 1 are stopped moving once entered inside the laboratory. The selected tracking devices and routing devices are always used for tracking and routing the information. Therefore, the energy drain of such devices are very fast as compare to other devices in Scenario 1. In Scenario 2 and Scenario 3, the moving objects are crosses the deployment region from starting to ending. Therefore, most of the deployed routing devices work as the source of the information and partition in the routing and lifetime of both Scenarios (Scenario 2 and Scenario 3) are almost equal.

Fig. 4.11 also illustrates that the triangle and hexagon deployment patterns provide the minimum and maximum lifetime of the network, respectively. The tracking range of all the devices are equal. A moving target in the triangle always lies in the tracking region of at least one tracking device. Therefore, the tracking and routing devices of the source of the tracking information are same in longer duration of time as compared with hexagons. The residual energy of the same tracking and routing devices and its neighbor devices reduces very rapidly as compare with the changes of the source devices.

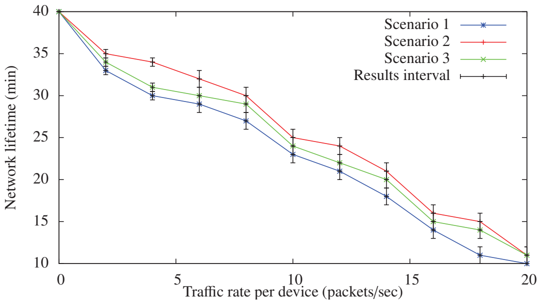
The comparison of the existing works with the proposed work is shown in Fig. 4.12. The results also show the network lifetime when the energy efficient technique is used. The work in Guo et al. (2013), Gupta and Rao (2016), and Dutta (2015) consider only “transmission delay”, “energy consumption and hop count”, and “energy consumption and residual energy” for estimating the route selection metric, respectively. The proposed work consider transmission energy, neighbor energy consumption, hop count, and residual energy. Table 4.1 illustrates none of the existing work consider all these metric. The FLS uses five required metrics for estimating the route selection metric. Therefore, the proposed routing algorithm finds a balanced route selection metric and enhances the lifetime of the network.



Part (a) Triangle deployment pattern.



Part (b) Square deployment pattern.



Part (c) Hexagon deployment pattern.

Figure 4.11 Relationship between network lifetime and traffic load.

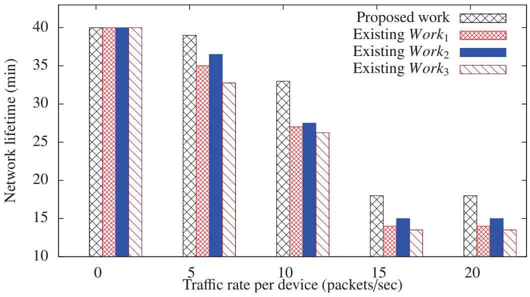


Figure 4.12 Comparison of network lifetime by proposed work with existing works, where Exiting Work_1, Exiting Work_2, and Exiting Work_3 are Gupta and Rao (2016), Guo et al. (2013), and Dutta (2015), respectively.

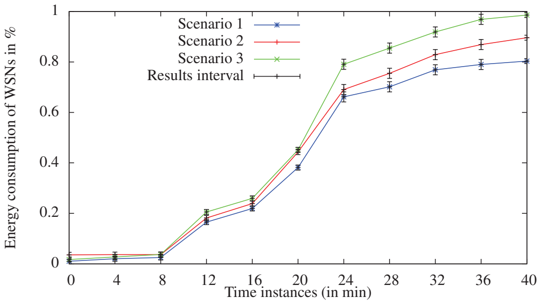
Energy Consumption of the Directional WSN

Next, we study the impact of the proposed work on the energy consumption of the WSN. We consider the same Scenarios as discussed in the previous results. The energy consumption of the WSN at a given time instant is the difference between the initial energy of the all deployed devices and the residual energy of the all deployed devices. Fig. 4.13 shows the total energy consumption for different time instances. It is true that the number of packets transfered on the network is increased with time and therefore total residual energy of the devices goes down.

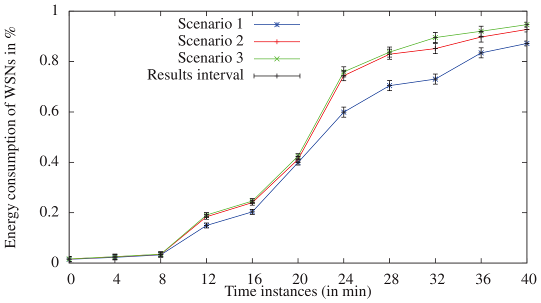
Fig. 4.13 illustrates that the Scenario 1 consumes minimum energy in all the experiments. This is because the movement of targets in Scenario 1 is very limited and therefore the routing paths do not change very frequently. Such low route discovery in Scenario 1 reduces the control packet overhead and the energy consumption. The results also illustrate that the Scenario 2 consumes less energy than Scenario 3. Speed of moving targets in Scenario 2 are slower than Scenario 3. Slow speed reduces the frequency of changes in the path and MTS does not require path finding process again and again, and therefore it saves the energy of the devices. As a result, the Scenario 2 consumes less energy.

Fig. 4.13 also illustrates that the triangle pattern consumes maximum energy in MTS, for all three Scenarios. The reason is same as given in previous result. As we have seen in the previous result that triangle pattern and hexagon pattern require minimum and maximum number of tracking devices for providing the desired quality of tracking in a fixed size FoI, respectively. The tracking range of all the devices are equal. A moving target in the triangle always lies in the tracking region of at least one tracking device. However, it is not true in other type of deployments. They always collect more tracking information and consumes more energy for routing the tracking information to the destination.

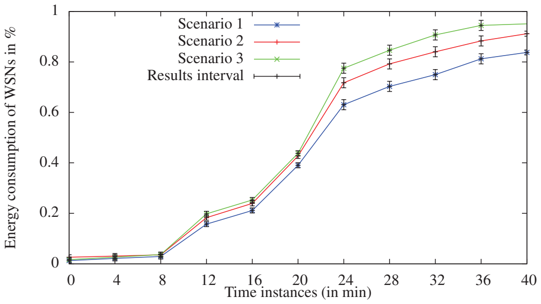
The comparison of the existing work Dutta (2015); Guo et al. (2013); Gupta and Rao (2016) with the proposed work is shown in Fig. 4.14. The results show that the



Part (a) Triangle deployment pattern.



Part (b) Square deployment pattern.



Part (c) Hexagon deployment pattern.

Figure 4.13 Relationship between time instances (in min) and energy consumption (in percentage).

proposed algorithm required least energy for routing the tracking information from source to destination in a given deployment pattern. This is because the propose work considers transmission energy, neighbor energy consumption, hop count, and residual energy.

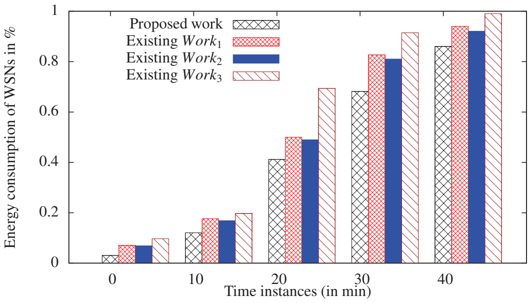
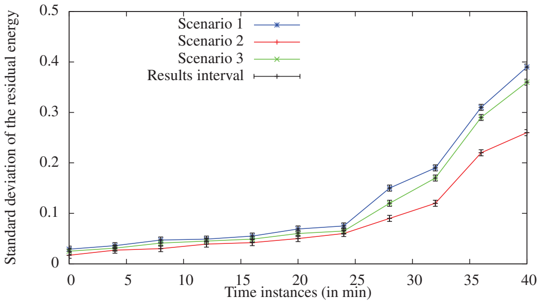


Figure 4.14 Comparison of energy consumption in percentage of the proposed work with existing work, where Exiting Work$_1$, Exiting Work$_2$, and Exiting Work$_3$ are Gupta and Rao (2016), Guo et al. (2013), and Dutta (2015), respectively.

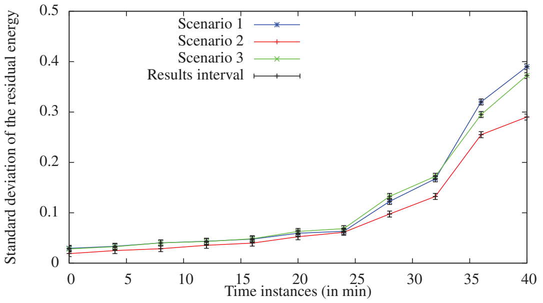
Residual Energy of the Devices

As we have seen in the previous two results that equal power consumption of the devices in MTS is necessary to obtain longer network lifetime. This experiment illustrates the residual energy of the all deployed devices at a given time instant. A network is said to be a stable and has long network lifetime if all devices have almost equal residual energy. In this experiment, we use Eq. 4.9 for estimating the variance of the residual energy of the deployed devices.

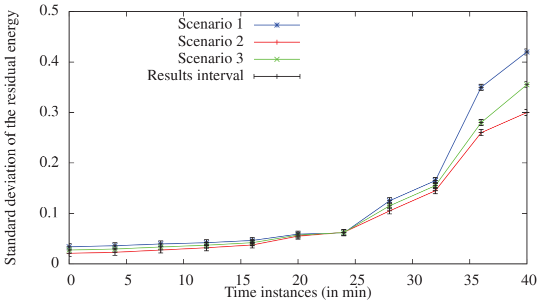
We estimate the standard deviation of the residual energy of n deployed devices in the network at time instance t, *i.e.* σ_t, by using Eq. 4.9. Fig. 4.15 illustrates the relationship between time instances (in min) and standard deviation of the residual energy. It illustrates that the Scenario 1 consists maximum variance as compare with other Scenarios for all types of regular deployments. It is because the mobility of the tracking objects near the entry gate of the Scenario 1 is very high as compared to the other location of the indoor space. The devices near the entry gate consumes huge power for routing the tracking information and therefore the difference of the residual energy is very high in Scenario 1.



Part (a) Triangle deployment pattern.



Part (b) Square deployment pattern.



Part (c) Hexagon deployment pattern.

Figure 4.15 Relationship between time instances (in min) and standard deviation of the residual energy.

Fig. 4.16 shows the comparison of the existing work Dutta (2015); Guo et al. (2013); Gupta and Rao (2016) with the proposed work for the residual energy of the deployed devices. It shows that the proposed work provides almost balanced energy consumption of the deployed devices as compared with others. It is possible only because the proposed work considers the residual energy as an input for estimating the route cost metric. However, it is not true for other existing works Dutta (2015); Guo et al. (2013); Gupta and Rao (2016).

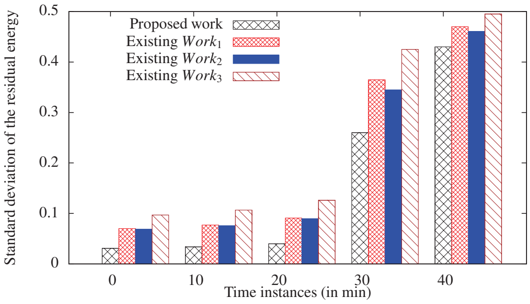


Figure 4.16 Comparison of standard deviation of the residual energy of the proposed work with existing works, where Exiting Work_1, Exiting Work_2, and Exiting Work_3 are Gupta and Rao (2016), Guo et al. (2013), and Dutta (2015), respectively.

Quality of Target Tracking

Finally, we illustrate the impact of the deployment patterns on the quality of target tracking of the proposed MTS. Scenario 3 consists slow and fast speed moving target tracers. We therefore used the Scenario 3 for this experiment. We count the number of moving targets passed through the given FoI. We used the camera to identify the true target movement in the proposed MTS. Table 4.3 illustrates the results of moving targets count in the experiments. The tracking range of all the devices are equal. A moving target in triangle pattern always lies in the tracking region of at least one tracking device. Therefore, the triangle pattern provides the highest accuracy as compare with hexagon and square.

4.5 Conclusion

In this chapter, we analyzed the tracking of moving targets inside the FoI using deterministic deployed of directional sensor nodes. This solution is relevant for civil emergency and military applications where the objective is to reduce the power consumption and

Table 4.3 Results of counting the moving targets experiments.

Experiment No.:		1	2	3	4	5	Total	**Accuracy**
Triangle	Camera Count	20	22	18	25	27	112	94%
	MTS Count	19	21	16	24	25	105	
Square	Camera Count	21	24	19	23	28	115	91%
	MTS Count	20	21	16	21	26	104	
Hexagon	Camera Count	18	20	15	29	24	106	90%
	MTS Count	18	19	12	24	23	96	

cost of the WSNs. We estimated the route selection metric that provides the desired level of tracking in the connected directional WSNs. The route selection metric were used to propose an energy-efficient routing algorithm for relaying the tracking information from source node to the destination.

We demonstrated an application of the proposed work to design a TTS based on directional WSNs called MTS. MTS was deployed along the road-side, inside the laboratory, and the corridor of the department for tracking the moving targets. We believe that our work can be used in planning and design of large scale energy-efficient directional WSNs with deterministic deployment of sensor nodes to trace the moving target.