



Performance evaluation of range-free localization algorithms for wireless sensor networks

Ibrahim Nemer¹ · Tarek Sheltami¹ · Elhadi Shakshuki² · Abdulrahman Abu Elkhail¹ · Mumin Adam¹

Received: 21 August 2019 / Accepted: 13 January 2020 / Published online: 19 May 2020
© Springer-Verlag London Ltd., part of Springer Nature 2020

Abstract

Wireless sensor network (WSN) consists of a number of nodes that are mostly distributed in a random way to monitor or control different phenomena, such as military operations, earthquake monitoring, environmental monitoring, factory automation and security. A sensor network is formed from a large number of tiny, low energy, limited processing capability and low-cost devices called sensor nodes (SNs) that communicate with each other in an ad-hoc fashion. SNs gather and forward data in order to achieve targeted missions. However, the manual configuration for any sensor network is difficult especially when they are distributed in such a harsh environment. The task of determining the exact position of SNs in WSNs is known as localization, which is an important factor in all WSNs applications that deal with monitoring or controlling phenomena. The localization accuracy varies from one application to another; localization techniques are deployed in different applications based on given requirements. Localization techniques are categorized into two groups: range free and range based. In range-free techniques, localization depends on the relationship between nodes and topological information of sensor nodes; however, in the range-based group, it is required to calculate the distance between sensor nodes. The scope of this paper is on the range-free localization techniques only. First, we survey different range-free localization techniques and discuss some of the localization-based applications where the location of the SNs is vital and sensitive. Then, we describe five localization algorithms: Centroid, Amorphous, approximate point in triangle, DV-Hop and DV-HopMax. After that, we simulate these algorithms using MATLAB based on different setups and topologies. Finally, we make a comparative study between aforementioned localization algorithms based on different performance metrics showing their pros and cons such as localization accuracy and energy consumption.

Keywords Localization · Wireless sensor network · Range-free · Range-based · Energy consumption · Anchor/Beacon nodes

1 Introduction

Wireless sensor networks (WSNs) are of major interest as they enable us to continuously monitor and control the environmental conditions. These devices are limited by resource constraints such low- power and limited processing. Due to the rapid growth and development of WSNs, localization becomes a primary concern. The reported data event in WSN is meaningful only if its position (and may be other data like time) is known. It is not practical to provide each SN with

Global Positioning System (GPS) for large network sensors because of power consumption and cost [1]. WSN is broadly used in different areas such as military applications, industrial applications, medical care, disaster relief and environmental monitoring [2]. For such applications, we need to find out the position of the gathered data in order to take the right decision. The method to determine the position of a node assigned in a network is often considered as localization problem [3].

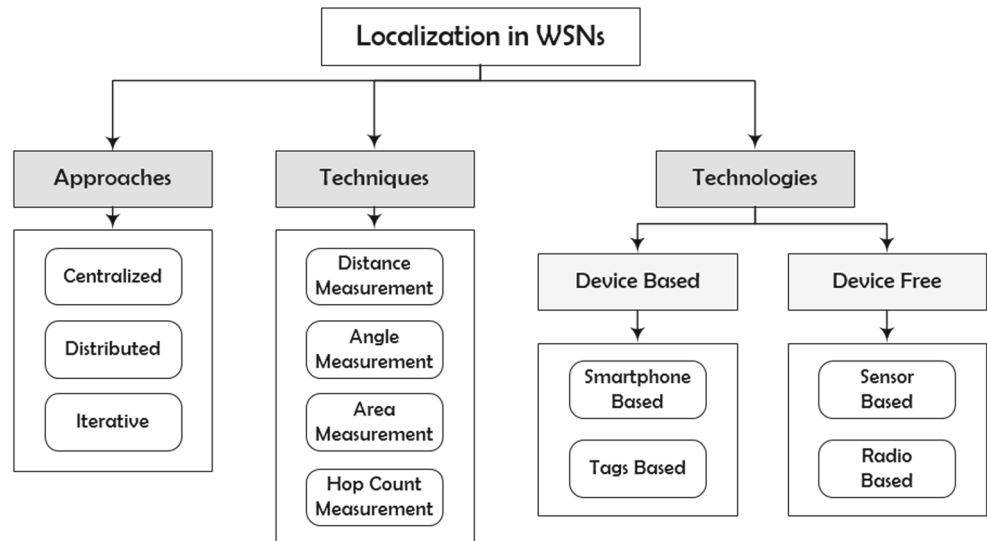
Researchers proposed various ways of how to classify the localization techniques. Localization techniques are widely divided into two categories, namely, centralized and distributed techniques based on their computational model as illustrated in Fig. 1. In centralized localization techniques, there is a base station that collects measurements and performs the computation, and then simply, the outcomes are returned back to the SNs. The virtue of this scheme is that all computations are taking place at the base station so there is no computation process in every node. The scalability remains the main

✉ Ibrahim Nemer
inemer@kfupm.edu.sa

¹ King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

² Acadia University, Nova Scotia, Canada

Fig. 1 Localization in WSNs



problem for this scheme. Multi-dimensional scaling-mobile-assisted programming (MDSMAP), semidefinite programming (SDP) and simulated annealing-based localization (LBSA) are the typical representations of the centralized localization techniques. On the other hand, in the distributed scheme, each node involves the computation and nodes collaborate with each other to specify their positions [4–6].

Distributed localization schemes can be classified into two groups: range-based and range-free localization techniques as described in Fig. 2. Range-based localization determines the distances between beacon nodes and sensor nodes based on inter devices angles or device to device distances. After that, it

estimates their location by using different geometric methods. The position of beacon nodes is known, either by using GPS or manual pre-programming during the deployment stage, while the sensor nodes can calculate their locations with reference to these beacon nodes. The benefit of using range-based techniques is that it has a high ranging accuracy compared to range-free techniques. However, the limitation of these techniques is that they need additional hardware, expensive for large systems, and also its deployment is very difficult. Some well known range-based techniques are as follows: received signal strength indicator (RSSI), time of arrival (ToA), angle of arrival (AoA) and others [5]. For range-free localization techniques, there is no need to determine distances directly; instead, it uses hop count between beacon nodes and SNs. Then, it computes location using geometric methods. Some benefits can be achieved by using these techniques that they do not need special hardware support; usually, they are cost efficient, mostly at the expense of the level of accuracy [7]. The popular range-free localization techniques are as follows: Centroid, Amorphous, approximate point in triangle (APIT), DV-Hop and DV-HopMax. The comparison between range based and range free is summarized in Table 1; the range based is compared to range free in terms of additional hardware support, level of accuracy, power consumption, deployment and robustness.

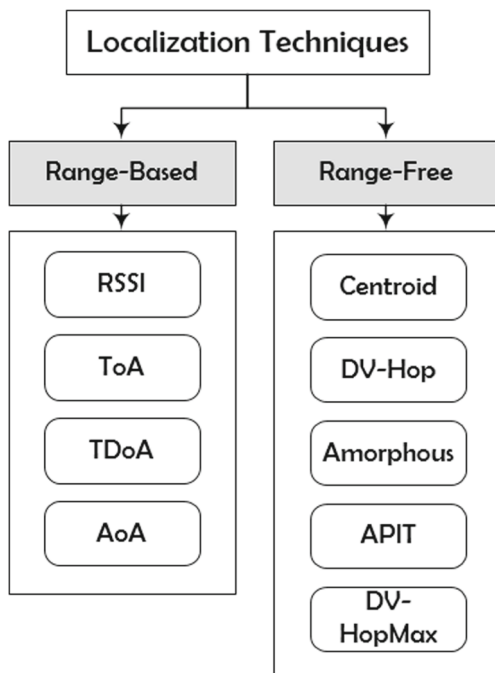


Fig. 2 Classification of localization techniques

Table 1 Comparison between range-based and range-free techniques

Parameters	Range-based technique	Range-free technique
Additional hardware	Needed	Not needed
Level of accuracy	High (85–90%)	Low (70–75%)
Power consumption	High	Low
Deployment	Hard	Easy
Robustness	High	Low

Early range-free techniques have deployed a set of nodes along with their GPS, which has the ability to know their exact location. Range-free localization techniques are highly relying on the communication and the cooperation between nodes in the wireless network. However, using the communication modules in these sensor nodes consumes a lot of energy while achieving certain task and this process has highly impacts on the node's battery. Nevertheless, not much interest is offered to investigate the influence of energy exhaustion on localization error.

The main contributions of this paper are as follows: it provides a detailed analysis of the various indoor position schemes, in particular for the recent tracking techniques, which were presented in the last 3 years. These techniques are analyzed using an assessment process to demonstrate the performance metrics for these techniques as well as show their advantages and disadvantages. In particular, we concentrate on more recent techniques that were published between 2016 and 2019 and have achieved a high accuracy. Then, we select five techniques (Centroid, Approximate Point in Triangle (APIT), Distance Vector Hop (DV-Hop), Amorphous, and DV-HopMax) to study and compare between them based on different performance metrics in square, C, O, L, U and W topologies. Moreover, we show the mathematical representation and the main steps that are needed to implement these techniques. After that, we evaluate the average consumption energy and the localization error of the selected techniques for different topologies with random distribution. However, we selected the techniques since we need to compare the DV-HopMax with the basic famous techniques in terms of the average consumption energy as well as the accuracy. Also, no one until now has compared the DV-HopMax with other techniques. At the end, we conclude our discussion section by comparing these five techniques and tabulating the simulation results based on our metrics. This work is mainly different from the previous studies by observing the energy level during the localization process. Moreover, we evaluate the nodes' energy consumption with respect to the transmitted and received packets between nodes in the same network, once the unknown node specifies its position it stays as the usual node only. The remaining parts of this paper are divided as follows. In Sect. 2, we discuss the main points of some previous literature surveys. In Sect. 3, we investigate five range-free algorithms. The performance analysis for the selected algorithms is presented in Sect. 4. Lastly, we conclude our work in Sect. 5.

2 Related work

Various localization algorithms are proposed to address the localization issues. There are several metrics that are considered to compare between these algorithms. In this paper, we

survey different approaches from recent studies in the literature and try to identify the research gaps that can be addressed in this study. Eva Tuba et al. [8] used a bat algorithm for range-based localization to estimate the sensor node positions. The algorithm has two stages: first stage, four mobile beacons have been located at the edges of the region where nodes are placed to estimate the coordinates of sensors. Then, in the second stage, the beacons go to their ideal positions with the lowest possible distance to sensor nodes. The proposed algorithm is more suitable for regular localization.

In [5], a hybrid extendable range-based distributed cooperative approach is proposed; the goal is to examine the impact of the MAC layer on the localization process. It is crucial to reduce collisions and re-transmissions of packets, which significantly participate in saving of energy consumption. To evaluate the performance, authors have varied parameters in order to investigate their effects on the MAC. These parameters include energy consumption, average localization accuracy, average localization time and the deployment strategy. Authors showed the importance of integration the MAC protocol along with the localization algorithm; even though this process minimizes the energy consumption and the localization time, it needs to come with a localization method with the capability of MAC optimizing energy consumption.

In [9], authors reduced the localization error in WSN, they presented a model called adaptive information estimation strategy time of arrival (AIES-TOA), and then they compared it against root mean square standard (RMS-STD), in terms of localization error, error tolerance and localization distance error. They showed that the AIEA-TOA outperforms the RMS-STD by 12.64%, 10.62% and 86.62%, for localization error, error tolerance and localization distance error respectively. Authors in [10] concentrated on enhancing the localization technique that basically relies on received signal strength (RSS). To bear some nugatory errors in the information of node location, a reference beacon node is used. Furthermore, Dixon approach is employed to eradicate the aberrations of the received signal strength indicator (RSSI) algorithm. The learning model here is used to minimize the ranging error of RSSI and boost the localizing precision adequately.

Authors in [11] carried out a comparative evaluation performance of various linear least-square (LS) approaches. To determine the location of the intended node for ToA parameters, linear or non-linear LS estimation approaches are applied. In their work, they compared the performance of linear least square techniques against the Cramer-Rao lower bound (CRLB), and the results that have been demonstrated showed the linear least-square based on ToA techniques yielded better results. While many researchers are doing researches on RSSI using various techniques, Moses A. Koledoye et al. [12] worked on improving RSSI readings by using filtering

techniques which are utilized to process unprocessed RSSI data in order to enhance the preciseness of range estimation. A number of filtering techniques have been selected as a candidate such as simple moving average (SMA), exponential moving average (EMA), moving median and moving mode. Moreover, the data are collected from commercial off-the-shelf (COTS) devices since they are more suitable with the selected filters, and then the COTS devices have been investigating within the presence of fixed nodes located in line of sight (LOS) environment. They showed that the filtering techniques were used with RSSI that enhance the preciseness of range computing and produce good results compared to direct computing from the raw data.

Researchers in [13] proposed a range-free localization approach that is used to estimate the positions of nodes. In their work, they improved a minimum mean square error technique that makes the computation of the distance between the source and destination nodes easier. Moreover, they considered the hop counts between source and destination nodes and their coverage radius. Some features of this approach have high accuracy in determining geographical coordinates, low traffic load and good performance for both homogeneous and heterogeneous environments. They showed that their approach has good location estimation and minimized traffic load at the same time compared to other existing localizing methods. Authors in [14] introduced a 3D-GAIDV range-free localization model for WSN. To modify the average hop size of beacon nodes, the correction factor has been used. Moreover, the authors used a linear search approach to optimize the modified hop size. The authors applied also the concept of the three-dimensional spatial geometry (coplanarity) to minimize the location errors generated by beacon nodes that are coplanar. In order to furthermore improve the location precision, genetic algorithm (GA) is applied. They evaluated their algorithm by comparing it with other existing algorithms, and the results showed that their algorithm outperforms the other existing algorithms since their algorithm performs the process and calculations at the target sensor level; this feature adds virtue to this algorithm by making it more energy efficient.

Comparative evaluation of different range-free techniques in [15] was carried out. They studied and compared each of these techniques in terms of the localization error. The results showed the presence of inverse proportional relation between the number of nodes and the range value. They also demonstrated the MDS-MAP technique that produces better results compared to other existing techniques, even if both number of nodes and range value are increased. On other hand, the error is less changed for both distance vector hop (DV-Hop) and Amorphous techniques. Moreover, in an effort to solve the localization problem, Najeh et al. [16] developed a novel range-free technique using genetic algorithms. The authors

calculated the position of the unknown node by using genetic algorithms (GAs). They claimed that there is no need for extra hardware in the network localization. In their work, they concentrated on two parts: first, they demonstrated the traditional DV hop technique, and then they explained the proposed technique in both the theoretical and experimental analysis and they showed that the proposed technique enhances the positioning accuracy of the traditional DV hop technique. To compute the unknown node position, they placed a node on an estimated position that has less connectivity difference compared to the unknown node and they found that the most accurate estimated position by just reducing the connectivity difference using the GAs. Also, they showed that their technique outperforms some other existing techniques.

In [17], authors compared between two range-free localization algorithms: Fingerprint and Centroid algorithms using the Tmote sky devices. The authors compared the algorithms in terms of localization accuracy and average error. They showed that the fingerprint algorithm has better performance compared to the Centroid algorithm. Paper [18] presented an improved version of Distance Vector Hop (DV-Hop) technique; the main goal is to save the energy by allowing only one beacon node to broadcast his position information. When the coordinates are estimated, the location error is reduced by using the ratio position; consequently, location accuracy is improved. The proposed algorithm was compared against two other DV-Hop techniques: the first one is based on least squares and the second relies on the improved particle swarm. The proposed technique performs better in terms of localization accuracy and energy saving.

Authors in [19] presented a multi-objective DV-Hop localization technique called NSGA-II-DV-Hop based on the Non-dominated Sorting Genetic Algorithm II (NSGAI). The main target is to reduce the localization error, since the old version from the DV-Hop localization technique depends only on the single-objective optimization; however, it is converted into a multi-objective DV-Hop localization technique. They deployed the multi-objective constraint in order to minimize the convergence domain of the unknown nodes that are distributed in the network as well as achieve better localization accuracy. Moreover, they built a better search approach by changing the population constraint based only on three beacon nodes to the constraint of all beacon nodes. They showed that the NSGA-II-DV-Hop technique outperforms the old DV-Hop and the previous techniques in all topologies; also, it supports the reliability and practicability of the multi-objective model in general. Another interesting research [20] published recently to serve the localization field based on the fuzzy decision for the localization techniques in WSN. Firstly, the received signal strength database of the reference points and the unknown nodes is formed; after that, the distance relationships between these

nodes are quantified, tested and saved through the fusion of the main database. Last step, the positions of the unknown nodes are calculated based on the process of different attribute decision making at the same time. They showed that the localization error of this technique is stayed in the low level values for different regions with different path loss values and noise interferences; this gives an indication that the technique has a high reliability and good environmental adaptability.

Authors in [21] proposed a localization technique based on fingerprint localization system for indoor and outdoor real time applications. This system mainly depends on four improved techniques: AFS technology (it used to achieve better robustness and survivability for the whole system by identifying the main problems), praxeological tracking (it used to improve the localization accuracy as well as the user experience of the targeted system), traffic regulation-based radio map establishing and full-overlapping clustering (it was deployed to provide the assistance to the tracking stage). This system showed a very high performance under low- and high-speed tests, and it supported the reliability and validity of the system with low localization error values. Paper [22] presented a new tool that gives the researchers and designers a fast method to evaluate the performance of the selected localization technique for specific topologies. The Cramér-Rao lower bound (CRLB) has been used as a lower bound in the TOA or RSS models. This bound represents as a benchmark for testing the localization technique for different channel characteristics and WSN topologies in order to define the accuracy value. Moreover, the channel characteristic for any node can be varied from one to another node in the WSN by using this tool and you can test the cooperative localization techniques in same WSN. Furthermore, the tool allows to consider independent characteristics for each node in the WSN and it has ability to run Monte-Carlo simulations as well produce the statistical reports.

In paper [23], authors proposed a lightweight localization technique for low resources and small WSNs, which compromise and minimize the gap between the high accuracy demands and the low resources that are available in WSNs. This technique is fully distributed and fulfils a high localization performance for the targeted sensor network by looking at the uncertainty values of the distance measurements in order to reduce the positioning errors that might occur from the range measurements and the interconnections between all neighbouring nodes. The proposed technique validated the performance, robustness and feasibility of the whole system. Authors in [24] presented a scheme to improve the Monte-Carlo localization technique by deploying a modified-genetic algorithm based on the coefficients of the Least-Mean-Square (LMS). This enabled to reach the maximum sample sets and minimize the hop counts. Based on that, the overhead communication, network traffic, and the localization inaccuracy are decreased

accordingly. According to the Maximum Likelihood Estimator (MLE), authors in [25] proposed an adaptive iterative localization approach based on the steepest gradient descent. This approach considers the cost function to be the requested function, and within the context of the gradient error, the required location can be identified. Moreover, they improved the localization accuracy and the convergence speed by using the Sigmoid function in the searching technique with a variable step. In three-dimensional WSNs [26], authors presented a new approach which is combination between the RSS and AoA localization techniques, where sensor nodes are randomly distributed with unknown transmission power and path loss exponents. Also, the authors derived a weighted least squares approximations to evaluate jointly the position of the unknown nodes and the channel components. This sub-optimal solution achieved high accuracy and save the energy of the SNs. In researches [27, 28], authors studied an event-driven received signal strength based localization for underwater WSNs. Based on a centralized system, they designed a fast localization approach using a linear fitting model for the received signal strength and distance relationship between sensor nodes. Moreover, they have also developed a technique for calculating the relative distance and optimizing the effects of the error in distance estimation. The suggested localization approach achieved better performance comparing with other approaches and the accuracy almost reached 90%. Authors in [29] derived the MLE for estimating the node position, where the transmission power can be considered as either a known or unknown variable, the generalized trust area technique has been applied with different approximations in order to change the MLE into an easy solvent form. The main drawback of this approach is that the overall performance may be degraded due to the approximations stage, and the output solution will be a sub-optimal solution. In paper [30], authors studied the problem of localization and synchronization of a SN in WSNs at same time. Hence, they created a bi-iterative estimator that evaluates the position and time of the source node alternately. Moreover, the convergence of the bi-iterative is proved by using the invariance theory of LaSalle. This joint approach achieved better accuracy with less number of anchor nodes and less communication overload comparing with other approaches in terms of the structure size and the network resources requirements. Authors in [31] proposed an intelligent approach called DeepML; it is a deep long short-term memory-based model for indoor localization requirements using light sensors and smartphone magnetic. They developed the DeepML approach to create bimodal pictures through data pre-processing stage and then train the deep network to visualize the local position features. After that, newly acquired magnetic field and light intensity data are used in an advanced probabilistic approach to determine the orientation of the mobile node. A fingerprint

localization technique based on the improved weighted k-Nearest Neighbor (KNN) approach and the RSS distance was proposed in [32]. The data base of this technique is constructed in the offline phase, and real-time measurements of the RSS are collected in the online phase. Moreover, the two-stage received signal strength distance is evaluated using the Euclidean formula. Eventually, they used an optimized weighted KNN approach to determine the last coordinates of the measuring point to resolve the issue of the non-Gaussian distribution noise. This approach can reduce the impact and improve the accuracy of signal strength fluctuations. Authors in [33] proposed a Cooperative Localization and Tracking Approach (CLTA) for complex and crowded indoor environment based on grid. The CLTA approach is split into two phases: offline and online. A collaborative fingerprint data base based on reliable nodes is constructed during the offline phase. On the other phase, a region overlapping technique is deployed to tight positions in the multi-network environment. To tight more the position area, they used a predictive technique to predict the mobile target position. Finally, a cross-grid technique is applied to increase the accuracy, if necessary, by upgrading the related information in the fingerprint database. This approach shows good results for the multi-network over the single network comparing with other approaches. A soft-clustering approach for the fingerprint-based localization method was proposed in [34]. The received signal strength is used as the main signal parameter in this approach. A perfect spatial resolution of a database corresponds to the time needed to match the patterns is required for the high precision of the localization. Clustering can be used to decrease the evaluation time by grouping similar data in the same cluster since it basically reduces the database sizes. The main consideration is the exactness of the algorithm for clustering the target and fingerprint positions. Eventually, the target position can be determined more easily by using soft clustering with the fingerprint strategy than by using traditional fingerprint strategies since the target position is calculated within a small set of cluster fingerprints rather than in the database using all fingerprints. In paper [35], authors presented an optimal Weighted K-Nearest Neighbour (WKNN) approach in order to acquire the optimized node location estimate for fingerprint-based localization technique in a noisy environment for WSNs. This approach is based on two algorithms: Adaptive Kalman Filter (AKF) and Memetic Algorithm (MA). The AKF is used to minimize the RSSI measuring noise between the WSN nodes. Next, MA is used to maximize the reference point weight in order to approximate the location of a node according to the filtered RSSI and the measured radiomap. At the end and based on the optimized weight, the optimal location of the node is achieved. Extensive results show that this method has obtained high accuracy regardless of the location of the requested

node, the number of anchor nodes and the calibration cell. Authors in [36] proposed a theoretical framework to determine how the temporal similarity of the received signal strength will affect the efficiency of the position prediction, which depends on a newly radio propagation model, taking into account the time-varying properties of Wi-Fi access point signals. Then, the method used to evaluate the localization in one-dimensional space and show the basic reason why localization efficiency can be enhanced by exploiting the temporal similarity of the received signal strength. Next, they expand the research to high-dimensional situations and mathematically represent the limits of the signal strength space that separate one physical position from another. Paper [37] presented a diversity-leveraging architecture for indoor localization named by Wireless Locator (Wi-LO) approach. Wi-LO is a client-server model that achieves a high accuracy by comparing the obtained signal samples with a previous reference for each sender, thereby allowing an stand-alone decision for these sensing states. In this paper [38], authors updated the basic DV-Hop approach and used a metaheuristic (Particle Swarm Optimization (PSO)) strategy to solve a positioning mistake without raising the additional hardware and the expense of computing. During the upgrading stage, average hop size, hop count value, location analysis estimates and error analysis were determined and a mathematical structure for the implementation of the metaheuristic method was developed. They studied the efficiency of the proposed approach in terms of accuracy, location error, error variance, and coverage. Authors in [39] invented a cheap solution based on the support vector machine (SVM) and twin SVM (TWSVM) approaches. The network is trained to indicate the closed nodes to the source or the destination; thus the event region is detected. Next, the central location of the target area is an approximation of the source position. However, the SVM and TWSVM learning approaches are used to show the region of event detection. Consequently, the SVM and TWSVM approaches were referred to as “Red-S” and “Red-T,” respectively. It has been demonstrated by simulating various scenarios that Red-S and Red-T operate adequately in target location, especially in dense networks. A RSS based localization approach was proposed in [40] for energy harvesting underwater optical WSNs. The sensor nodes with inadequate battery collect ambient resources and start communicating when they have suitable energy storage. However, the localization of the network is achieved by calculating the received signal strengths of the active SNs. Moreover, a block kernel matrices are then determined for measurements of the range based on the received signal strength values. When the full block kernel matrices are usable, a closed-form localization method will be built to locate each optical sensor node in the network. As a result, the proposed approach reduces the error

values by introducing a novel matrix completion technique in estimating the shortest paths, in block kernel matrices. In this research [41], authors presented an iterative positioning approach that deploys mobile beacon nodes and transmit location information to its neighbors to determine where they are located roughly. A progressive approach also used to identify low beacon sensor nodes. The proposed approach depends on the multilateration technique; it is applied to determine the location by using the distance between nodes. It is observed that the error of localization is further decreased throughout their mobility by receiving multiple beacons from the mobile anchor nodes from the different location. The main benefit of the mobile beacon node over the fixed beacon node is that with fewer mobile beacon nodes, localization is achieved across the entire network, which is preferable for energy-restricted WSN. In paper [42], authors suggested a target location approach focused on the FuzzyInference System (FIS) in three-dimensional WSNs. The suggested solution measures the distance between beacon and target nodes by adding a correction factor with a beacon node hop dimension. In order to enhance position precision for this approach, the principle of fuzzy logic-based edge weight evaluation is implemented. Based on this addition, the proposed approach when compared to the existing tracking technique, it achieved less location error and greater accuracy. Table 5 and Table 6 in the Appendix compare between these techniques with respect to the used localization technique, performance metrics, advantages, and disadvantages/limitations.

3 Range-free algorithms

In this section, we are going to describe five popular range-free algorithms (Centroid, DV-Hop, Amorphous, APIT and DV-HopMax) in details since they are mostly used in the literature reviews:

3.1 Centroid algorithm

The centroid localization algorithm was proposed by Bulusu in [43]. It is one of the simplest range-free algorithms since it requires only a minimum number of computations and little communication expenses comparing with other algorithms. Simply, all unknown nodes calculate their locations as the centroid of all received packets from the beacon nodes within the communication range. Generally, this algorithm is like a binary information whether the unknown node is within the communication range or not in order to consider it as the estimated value. However, each beacon node has a circular shape and nodes who are located inside this circle can communicate with them as described in Fig. 3. Here, there are four beacons in the circular range and there is only one unknown node; in this figure, the estimated location for the beacons is the centroid value.

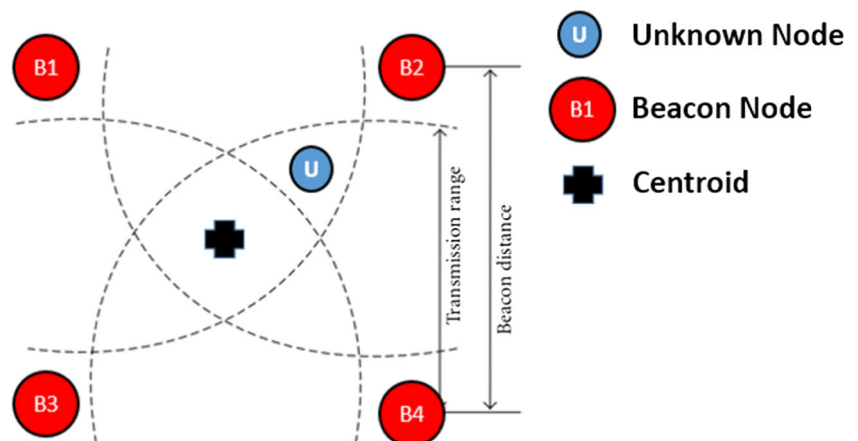
This algorithm consists of three stages:

- Stage one: all beacon nodes send their locations to all sensor nodes within their communication range. At the same time, each unknown node listens to the medium for a specific period of time and then collects the received packets from the beacon nodes.
- Stage two: these unknown nodes evaluate their locations by the centroid determination from all locations of the beacons.
- Stage three: this centroid is a simple operation that uses only the beacon nodes locations; this value can be evaluated by the following:

$$C_i(x,y) = \frac{1}{N} \sum_{j=1}^N B_j(x,y) \tag{1}$$

where, $C_i(x,y)$ represents the location of unknown node i and this gives us the coordinates of this node. $B_j(x,y)$ is the location of the beacon j , where this includes all beacons that are in range with this node and it starts from beacon 1 up to last beacon N , where $N \geq 3$.

Fig. 3 Nodes representation in the Centroid Algorithm



The pseudocode of this algorithm can be described as follows:

ALGORITHM 1 Centroid Localization Algorithm

- 1: **First**, Receive the location from N neighbor beacon nodes $B_j(x;y)$
- 2: **Next**, Evaluate the location of unknown nodes $C_i(x;y)$ using Centroid Formula
- 3: **if** $N \geq 3$ **then**
- 4: $C_i(x\text{-coordinate}) \leftarrow \frac{1}{N} \sum_{i=1}^N x_i$ and $C_i(y\text{-coordinate}) \leftarrow \frac{1}{N} \sum_{i=1}^N y_i$
- 5: **end if**
- 6: **Last**, Same procedure will be repeated for all unknown nodes.

As we mentioned in the beginning, Centroid algorithm is a simple algorithm but the achieved accuracy is high compared with other algorithms and this is due to the centroid formula. However, the accuracy of the estimated location depends on the beacon nodes density and the type of distribution; usually, the uniform distribution topology increases the localization accuracy.

3.2 DV-Hop algorithm

Another well-known algorithm from the range-free localization group is called DV-Hop algorithm. This algorithm was proposed by Niculescu et al. [44] in 2003; it is a distributed hop-by-hop localization algorithm. Mainly, it is based on the distance vector as in the traditional routing algorithms. However, this gives an approximate value for the estimated location of any unknown node in the network by using a few numbers of known location nodes, which are most probably equipped with GPS.

DV-Hop algorithm is not using the usual ranging techniques to localize and find the distances of the unknown nodes. Simply, each sensor node calculates its distance based on the minimum hop number and average distance with respect to the beacon node. After that, the distance can be computed between the usual node and the beacon node by multiplying the minimum hops with the average distance of each hop. Finally, each node estimates its position coordinates using different estimators, such as triangulation, trilateration, maximum likelihood estimators and so on. DV-Hop algorithm consists of three stages as described in the following detailed manner:

- Stage one: Finding the minimum number of hop counts for each node; each beacon node broadcasts a beacon message including his position coordinates and the hop count, which is set to 0 at the beginning of the transmission. This value is incremented by other neighbour nodes when they receive it and then they re-broadcast again. Therefore, if the beacon or normal node receives a beacon message, it stores the coordinates of the sender node and increment the hop count by one. In the meantime, it

initializes a new field called hop size, where this value represents the minimum number of hops between the sender and the node itself. Actually, if the receiver node receives a message from the same beacon node, it first checks the hop number and increment it directly and then compare it with the stored one if it is less, it updates its value and rebroadcast it with the new hop value. Otherwise, it drops the message. At the end of this stage, all nodes, beacon and normal, will have the minimum hop counts with respect to all beacon node in the network.

- Stage two: Evaluating the average hop distance; each beacon node evaluates the average distance per hop by using the received coordinates from other beacon nodes along with the minimum number of hops that is needed to reach this beacon; this value can be estimated by using the following:

$$HopSize_i = \frac{\sum_{j=1}^n \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j=1}^n HopCount_{ij}} \quad (2)$$

where (x_i, y_i) and (x_j, y_j) are the coordinates of the beacon nodes i and j respectively, $HopCount_{ij}$ is the number of hops between i and j and n is the total number of beacon nodes. Then, each beacon node needs to flood this value to other nodes. Once, this value is received by the unknown node; it stores the first received packet only and then sends it again to its neighbours. This process assures that most of the nodes receive the value of the nearest beacon node. In the meantime, once the unknown node receives this value and stores it, it evaluates the distance between itself and the beacon node; this can be done by using the following:

$$Distance_{ub} = HopSize_i \times HopValue_{ub} \quad (3)$$

where $HopSize_i$ is the hop value received by this unknown node from the nearest beacon node i , and $HopValue$ is the minimum hops between the beacon node and this unknown node.

– Stage three: Estimating the location of the unknown node; the unknown node can estimate its location by using the triangulation estimator [45] along with the evaluated values from stage two. First, let the coordinate of the unknown node be (x, y) , the coordinate of the beacon node is (x_i, y_i) and the distance between this beacon node and the unknown node is d_i . Then, we have n beacon and we would like to estimate the position of this unknown node; this can be done by using this equation:

$$\begin{cases} (x-x_1)^2 + (y-y_1)^2 = d_1^2 \\ (x-x_2)^2 + (y-y_2)^2 = d_2^2 \\ \vdots \\ (x-x_n)^2 + (y-y_n)^2 = d_n^2 \end{cases} \tag{4}$$

After that, in order to simplify our system and make these equations linear, just subtract the last equation from the previous $n - 1$ equations and rearrange them. In the meantime, we will get the following set of equations:

$$\begin{aligned} x_1^2 + y_1^2 - x_n^2 - y_n^2 - 2(x_1 - x_n)x - 2(y_1 - y_n)y &= d_1^2 - d_n^2 \\ x_2^2 + y_2^2 - x_n^2 - y_n^2 - 2(x_2 - x_n)x - 2(y_2 - y_n)y &= d_2^2 - d_n^2 \\ &\vdots \\ x_{n-1}^2 + y_{n-1}^2 - x_n^2 - y_n^2 - 2(x_{n-1} - x_n)x - 2(y_{n-1} - y_n)y &= d_{n-1}^2 - d_n^2 \end{aligned} \tag{5}$$

These equations can be represented in a matrix form as follows:

$$AX = B \tag{6}$$

where A, B and X are a matrices and equal to the following:

$$A = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ 2(x_2 - x_n) & 2(y_2 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix} \tag{7}$$

$$B = \begin{bmatrix} x_1^2 + y_1^2 - x_n^2 - y_n^2 - d_1^2 + d_n^2 \\ x_2^2 + y_2^2 - x_n^2 - y_n^2 - d_2^2 + d_n^2 \\ \vdots \\ x_{n-1}^2 + y_{n-1}^2 - x_n^2 - y_n^2 - d_{n-1}^2 + d_n^2 \end{bmatrix} \tag{8}$$

$$X = \begin{bmatrix} x \\ y \end{bmatrix} \tag{9}$$

Finally, in order to solve the targeted equation, we need to do some simple operations and then we will get the following value for the position:

$$X = (A'A)^{-1}A'B \tag{10}$$

where A' is the transpose of matrix A .

However, to simplify and summarize the procedure of the DV-Hop algorithm, the following pseudocode describes all steps that mentioned before:

ALGORITHM 2 DV-Hop Localization Algorithm

- 1: **First**, For each Beacon node i
 - 2: (i) Set HopCount $h_{ij} = 0$.
 - 3: (ii) Broadcast packet including (i, x_i, y_i, h_{ij}) to all neighbours.
 - 4: **Then**, The receiving neighbor:
 - 5: (i) Let $h_{i,r} = h_{ij}$ received in the packet, and $h_{i,s} = hopcount$ stored in its HopCount table
 - 6: (ii) Will defined whether to consider or drop the packet:

 - 7: **if** $h_{i,s} = 0$ OR $h_{i,r} \leq h_{i,s} - 1$ **then**
 - 8: Set $h_{i,r} = h_{i,r} + 1$
 - 9: Set $h_{i,s} = h_{i,r}$
 - 10: Broadcast packet $(i, x_i, y_i, h_{i,r})$ to all neighbours.
 - 11: **else**
 - 12: Discard the packet.
 - 13: **end if**

 - 14: **After that**, Each Beacon node i :
 - 15: (i) Calculate the $HopSize_i$.
 - 16: (ii) Broadcast $HopSize_i$ to all neighbours.
 - 17: **Finally**, Each receiving node i :
 - 18: (i) Calculate the distance between itself and the Beacon node.
 - 19: (ii) Calculate its location using least square method.
-

The main advantages of the DV-Hop algorithms are being simple, less complexity and low-cost (i.e. no need for ranging techniques). On the other hand, it may suffer from low accuracy, especially when we have a small network. If we have a pair of nodes with the same hop distance value from all beacon nodes, we will get the same estimated location. But this is not the case, since the physical distance might be different. Therefore, most of the researches after 2003 tried to improve the localization accuracy regardless how many beacons and unknown nodes are in the network and the distances between them.

3.3 Amorphous algorithm

This algorithm [46] was proposed separately from DV-Hop algorithm; it has some similar specification of the DV-Hop algorithm; however, the main idea is to evaluate the hop distance between any two nodes rather than finding the linear distance between them. It consists from the following steps [47]:

- Step one: evaluate the minimum hop number from the unknown node to the beacon node, where every beacon node transmits packets to the unknown nodes using the flooding technique. The minimum hop number from the unknown node i to the beacon node k is evaluated by using this formula:

$$S_{(i,k)} = \frac{\sum_{j \in nbrs(i)} h_{(j,k)} + h_{(j,k)}}{|nbrs(i)| + 1} - 0.5 \quad (11)$$

where $S(i,k)$ is the minimum hop value from the unknown node i to the beacon node k ; $h(j,k)$ represents the hop value from the unknown node j to the beacon node k ; $h(i,k)$ represents the hop value from the unknown node i to the beacon node k ; $nbr(i)$ represents the number of neighbour nodes around the unknown node i ; $|nbr(i)|$ represents the number of neighbour nodes around the unknown node i .

- Step two: evaluate the distance value from the unknown node to the beacon node; however, the average distance value of one hop is evaluated by using this formula:

$$HopSize = r \left(1 + e^{-n_{local}} - \int_{-1}^1 e^{-(n_{local}/\pi)(\arccos t - \sqrt{1-t^2})} dt \right) \quad (12)$$

where r represents the wireless range of the node and n_{local} represents the average connectivity of the network. Formula (13) is used to evaluate the distance value d from the unknown node to the beacon node based on the average distance value for one hop and the minimum hop number from the unknown node to the beacon node.

$$d = HopSize_i \times S_{(i,k)} \quad (13)$$

- Step three: use the least squares technique to locate the unknown nodes; when the estimated distances from the unknown node to three or more beacon nodes have been gained, the position value of the unknown node can be evaluated by using same calculations deeply discussed in the DV-Hop algorithm for the least square estimation.

3.4 APIT algorithm

Approximate point in triangle (APIT) is a free range algorithm [48]. It needs a heterogeneous network of sensing equipment where a small number of these devices should have a high-powered transmitters and location data gained via a GPS device or any other mechanism. We called these location-equipped devices as the beacon nodes. In this algorithm as described in Fig. 4, the network area is split into a triangular shape between beacon nodes, where inside or outside every triangle there is a node that allows to decrease down the main area in which it can potentially reside. By utilizing the combinations of the beacon locations, the diameter of the evaluated area in which a node resides can be decreased; this accordingly will give a good position estimate with satisfactory accuracy.

The hypothetical technique that is used to decrease down the area in such way that the targeted node resides is the point-in-triangulation (PIT) test. In this test, a node selects three beacon nodes from all noticeable beacons and tests if this node is inside the triangle that formed by connecting these three beacons. APIT then repeats this PIT test with different noticeable beacon groups until all groups are exhausted or the needed accuracy is achieved. After that, APIT evaluates the centre of gravity (COG) for the intersection between all triangles in which a way that the node resides to

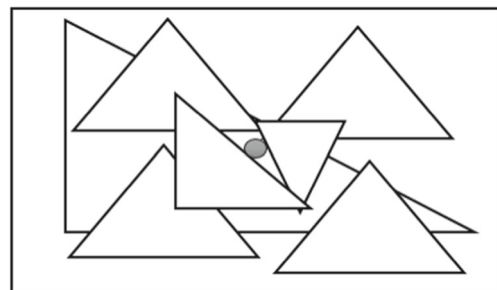


Fig. 4 Area-based APIT Algorithm Overview

defined its estimated location. APIT technique mainly consists from three steps and these steps are completed at separate nodes in a distributed way. Before starting with

the detailed description of these steps, we present the pseudocode of this algorithm.

ALGORITHM 3 APIT Algorithm

```

1: Receive location beacons  $(X_i, Y_i)$  from  $N$  beacon nodes.
2: InsideSet =  $\Phi$  // the set of triangles in which I reside
3: For (each triangle  $T_i \in \binom{N}{3}$ ) triangles)
   {
4: If (Point-In-Triangle-Test ( $T_i$ ) == TRUE)
5: InsideSet = InsideSet  $\cup$  {  $T_i$  }
6: If (accuracy(InsideSet) > enough ) break;
   }
   /* Center of gravity (COG ) calculation */
7: Estimated Location = COG (  $\cap T_i \in$  InsideSet);

```

APIT algorithm consists of the following three steps:

- Step one: gather data such as info of the beacon node near the unknown node: identification number, location and signal strength.
- Step two: test whether the unknown node is inside the triangle area that constructed with different beacon nodes or not. Figure 5 shows an example of the APIT testing. For the three given beacons, $A(ax, ay)$, $B(bx, by)$ and $C(cx, cy)$, we will specify whether a point M with an unknown location is inside the triangle DABC or not. Proposition 1: If M is inside the triangle area DABC, when M is shifted in any direction, the new location must be closer to at least one beacon A , B or C as we can see in Fig. 5a. Proposition 2: If M is outside the triangle area DABC, when M is shifted, there must a direction in which the location of M is further from or nearer to all the three beacons A , B and C as we can see in Fig. 5b.

- Step three: calculate the overlapping area between all triangles that cover the unknown nodes. Once the APIT tests are completed, the APIT collects the results through a grid SCAN technique as in Fig. 6. In this technique, a grid array is deployed to find the maximum area in which a node will reside.

For each APIT, inside decision, the collected values of the grid regions over which the triangle resides are incremented. For the outside decision, the grid area will be similarly decremented. Once all triangular areas are found and computed, the resulting data usually is used to specify the maximum overlapping area (e.g. the value 2 in Fig. 6 is the maximum) and then calculate the centroid of the overlapping area and the location of the unknown node. The pseudocode of the APIT aggregation/collection technique is as follows:

ALGORITHM 4 APIT Aggregation

```

1: For (each triangle  $T_i \in \binom{N}{3}$ ) triangles) {
2: If (APIT( $T_i$ ) == Out )
3: AddNegativeTriangle( $T_i$ );
4: If (APIT( $T_i$ ) == In )
5: AddPositiveTriangle( $T_i$ ); }

```

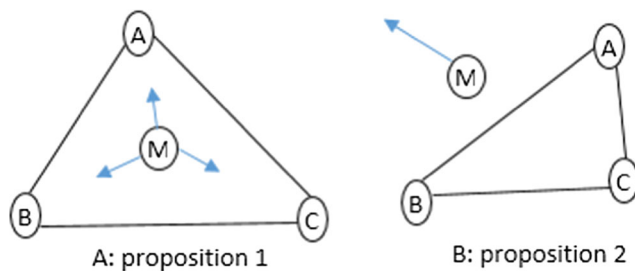


Fig. 5 APIT testing example

3.5 DV-HopMax algorithm

Shahzad et al. [49] proposed an extended technique for the DV-Hop called DV-Hop-Max for both anisotropic and isotropic networks. Authors in this paper introduced a new parameter called MaxHop; this parameter value is related and depending on the characteristics of the network topology in a way that may improve the accuracy of the whole network. The concept of this algorithm is very simple; it follows the same steps of the DV-Hop with minor modifications. The main difference is that the hop count value is restricted by a threshold value and nodes will not forward the received packet if the value is larger than this value. If the hop count is more than MaxHop, the unknown node neglects the information from the beacon nodes. Since the distance estimation of some nodes

0	0	0	0	0	0	1	0	0	0
0	0	1	0	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	1	2	2	1	1	0	-1	0	0
1	1	2	2	1	1	0	-1	-1	0
0	0	2	2	2	1	0	-1	-1	-1
0	0	1	1	1	0	0	-1	-1	-1

Fig. 6 SCAN approach

is inaccurate, especially that are not helping or even deforming the accuracy; this produces some errors in the estimation and increases the delay factor, which is not suitable for most of the Internet of things (IoT) applications. Authors found that MaxHop value is between 5 and 12 hop count, and this gives a good result for different topologies and anisotropic factors. This value can be adjusted based on the needs such as desired accuracy, power consumption and convergence time. The following algorithms (5) and (6) can be used to find the optimal MaxHop value based on the required accuracy or the desired convergence time.

ALGORITHM 5 Finding maxHop for the required accuracy

- 1: **First**, Initialize maxHop to 5
 - 2: **Do** {
 - 3: Run the DV-maxHop Algorithm
 - 4: Compute **AvgLocError**
 - 5: **If** ($AvgLocError \leq DesiredAvgError$) break
 - 6: **Else** maxHop++
 - 7: **While** ($maxHop \geq 5 \ \& \ maxHop \leq 20$)
-

ALGORITHM 6 Finding maxHop for the required convergence time

- 1: **First**, Initialize maxHop to 15
 - 2: **Do** {
 - 3: Run the DV-maxHop Algorithm
 - 4: Compute **AlgoRuntime**
 - 5: **If** ($AlgoRuntime \leq DesiredRuntime$) break
 - 6: **Else** maxHop–
 - 7: **While** ($maxHop \geq 5 \ \& \ maxHop \leq 20$)
-

It is clear from their results that the convergence time, localization accuracy, computational overheads and energy consumption in both isotropic and anisotropic networks are improved compared with other similar algorithms. However, these metrics depend on the MaxHop value and the network topology.

4 Performance analysis

There are many constraints in WSNs such as node size, energy and cost. Many localization algorithms were proposed during the last decade, but most of them were deployed to serve a specific application and they were not necessarily suitable for other applications. For example, some of these algorithms are specifically designed only for mobile nodes and it is not applicable for fixed nodes. Based on that, evaluating the performance of any localization algorithms in WSNs is highly needed, either to validate the algorithm and compare it with the previous algorithms or to select a specific localization algorithm that is more suitable than others for a specific application. Indeed, it is important for the researchers to define the performance evaluation metrics that are used in comparison with other algorithms and to deeply understand the application requirements. Some of the famous evaluation metrics are as follows: localization accuracy, coverage, scalability, topology, cost and robustness. Consequently, these metrics have highly impact on the WSN constrains such as complexity, power consumption and scalability. In this study, we focus only on the following factors:

- Topology model: Random distribution strategies will be used, where the unknown and beacon nodes are distributed in a random way with different shapes like square, C-shape, W-shape, U-shape, L-shape and O-shape.
- Localization error: The commonly used metric for any comparison between the localization algorithms is the location estimation error (accuracy), where this metric is defined as in the following equation:

$$Error = \frac{\sum_{i=1}^n \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2}}{NR} \tag{14}$$

where (x_i, x'_i) represents the exact and estimated x -coordinates, and (y_i, y'_i) represents the exact and estimated y -coordinates, respectively. N is the number of unknown nodes, and R is the radio range for the unknown node.

- Energy consumption: The biggest challenge in sensor nodes is how to keep them working as long as possible with respect to their remaining energy. This amount of energy is defined as the lifetime of the sensor node, and the energy consumption in the sensor nodes is mostly

viewed in three parts: computation part, communication part and the sensor/actuator part. Computation part represents the microprocessor section where the node carries out different functions and tasks in one direction or another to achieve a certain purpose in an efficient manner. Generally, the microcontroller’s functions may contain the following: data acquisition coming from the sensor/actuator part in the sensor node. In the WSN, in order to conserve the energy, it is more suitable for a node to be inactive and change its mode to active when an event happens. The microcontroller usually manages the power usage by changing to different modes based on different operating conditions or conditions differ from one application to another or from one manufacturer to another. The communication part is related to the physical layer and the medium access control (MAC) layer, where a node uses part of its energy in sending, listening and receiving data. Sensor network uses the same wireless channel for all nodes. This communication channel stays idle most of its time; it mainly generates a low traffic value. On other hand, keeping nodes awake during all time consumes a lot of energy, and accordingly, the lifetime of the sensor network is reduced. Nodes should go to the sleep mode when there is nothing to send or receive and wake up if they require to send information or if there is another node has transmitted information to them.

In order to find the approximate consumed energy for our targeted techniques, let n neighbours to the unknown nodes, E_s is the consumed energy in sending packets, E_r is the consumed energy in receiving packets and lastly and E_c is the energy consumption in the computation part. Table 2 shows

Table 2 Consumption energy in the five algorithms

Technique	Step	Consumed Energy	
		Beacon Node	Unknown Node
Centroid	One	$E = E_s$	$E = n_-(E_r + E_c)$
DV-Hop	One	$E = E_s$	$E = n_-(E_r + E_s)$
	Two	$E = E_c + E_s$	$E = E_r + E_s + E_c$
	Three	-	$E = E_c$
Amorphous	One	$E = E_s$	$E = n_-(E_r + E_c)$
	Two	$E = E_s$	$E = E_r + E_c$
	Three	-	$E = E_c$
APIT	One	$E = E_s + E_r$	-
	Two	-	$E = n_- E_r$
	Three	-	$E = E_c$
DV-HopMax	One	$E = E_s$	$E = n_-(E_r + E_s)$
	Two	$E = E_c + E_s$	$E = E_r + E_s + E_c$
	Three	-	$E = E_c$

the approximate consumed energy for the selected localization techniques [50]:

From Table 3, in DV-Hop and DV-HopMax, the Hop size value is evaluated by the beacon nodes then flooded in the network. Therefore, DV-Hop algorithm has high overhead value compared to Amorphous, APIT and DV-HopMax due to the flooding process and the number of hops. The energy-saving factor in these techniques has been studied with respect to the consumption value resulted from the execution of each localization technique according to the number of nodes and beacon nodes. The estimated energy can be evaluated as follows:

$$E_{cons} = \frac{\sum_{i=1}^n E_{b_i} + \sum_{i=1}^N E_{u_i}}{n_b + N} \quad (15)$$

where n_b represents the number of the beacon nodes, E_{b_i} indicates the average consumption energy by the beacon nodes, N represents the number of the unknown nodes and E_{u_i} is the average consumption energy by the unknown nodes.

We have simulated different algorithms under different topologies and parameters like radio range configuration, beacon node density and node density. Figure 7 shows these topologies for only one single run. However, in order to achieve 95% confidence interval, we run the simulation for this experiment fifty times and reported the average results for varying beacon and total nodes. The performance metrics for our comparison are the location estimation error and the energy consumption. We conduct a variety of experiments to cover a wide range of system configurations including varying (1) beacon density, (2) node density and (3) degree of Integrity. Simulation parameters are described in Table 3.

Table 3 Parameters Setting

Type	Value
Covered Area	2D- 250m x 250m
Nodes	150 nodes
Beacon Nodes	20% of Nodes
GPS Error	Zero
Radio Range	100m
Distribution/Topology	Random/[Square, C, L, U, O, W]-Shape
Simulation Runs	Repeated 50 times
MaxHop	6 hops
Initial Cons. Energy	0.02 mJ
Transmission Energy	0.15 mJ
Receiving Energy	0.115 mJ
Computation Energy	0.05 mJ

Then, we studied the behaviour of the localization error and the energy consumption under changing the following parameters:

- Impact of anchor/beacon density: It represents the number of beacons that are distributed in our network and used to estimate the location of the unknown nodes. Consequently, more beacon nodes will improve accuracy but will increase the overall cost of the whole system. In our simulation, the number of beacons will be varied from 10 to 45 nodes and 20% of total nodes when the node density is set to be varied from 100 to 500 nodes.

Figure 8 shows the localization error curve for the five selected algorithms for all topologies when the number of beacons is varied from 10 to 45 beacon nodes. Overall, it shows that the average localization error decreases as the number of beacon increases. It can be observed also that the APIT and DV-HopMax algorithms perform the best in term of localization error when the number of beacons is varied for (Square, O-Shape and W-Shape) topologies compared to the DV-Hop, Centroid and Amorphous algorithms. For APIT and DV-HopMax algorithms, it can be notified that the localization errors of are between 0.28 and 0.35 when the number of beacons is only 10 for all topologies. However, for most topologies, the localization errors of are between 0.27 and 0.3 when the number of beacons is 45. It is very important to note that different algorithms show different behaviours based on the topology type. For example, the centroid scheme performs worse than the DV-Hop and Amorphous algorithms in square and O-shape random topologies. On the contrary, the centroid scheme outperforms both of the DV-Hop and Amorphous algorithms in C-shape, L-shape, U-shape and W-shape random topologies.

Figure 9 shows the energy consumption behaviour of the beacon and normal nodes in the network. We can observe that with a small number of nodes, the localization error for all algorithms is the highest compared to other points. DV-Hop algorithms are based on multi-hop communications; thus, the beacon nodes that are unreachable by other beacons are connected in a multi-hop fashion. Moreover, the unknown nodes are required to cooperate with the beacon nodes by broadcasting the hop size to its neighbours. As a result, the number of transmitted and received packets is the highest for the DV-Hop algorithms compared to other algorithms; this increases the consumption energy in the network. Moreover, in random topology, more beacon nodes are required due to the distance estimation in the hop count computation which is mainly affected by a wrong shortest path. Centroid algorithm has the least number of received packets compared to other localization algorithms since the node depends only on the position of the beacon nodes which broadcast directly to the unknown nodes without sending any packets to their neighbours.

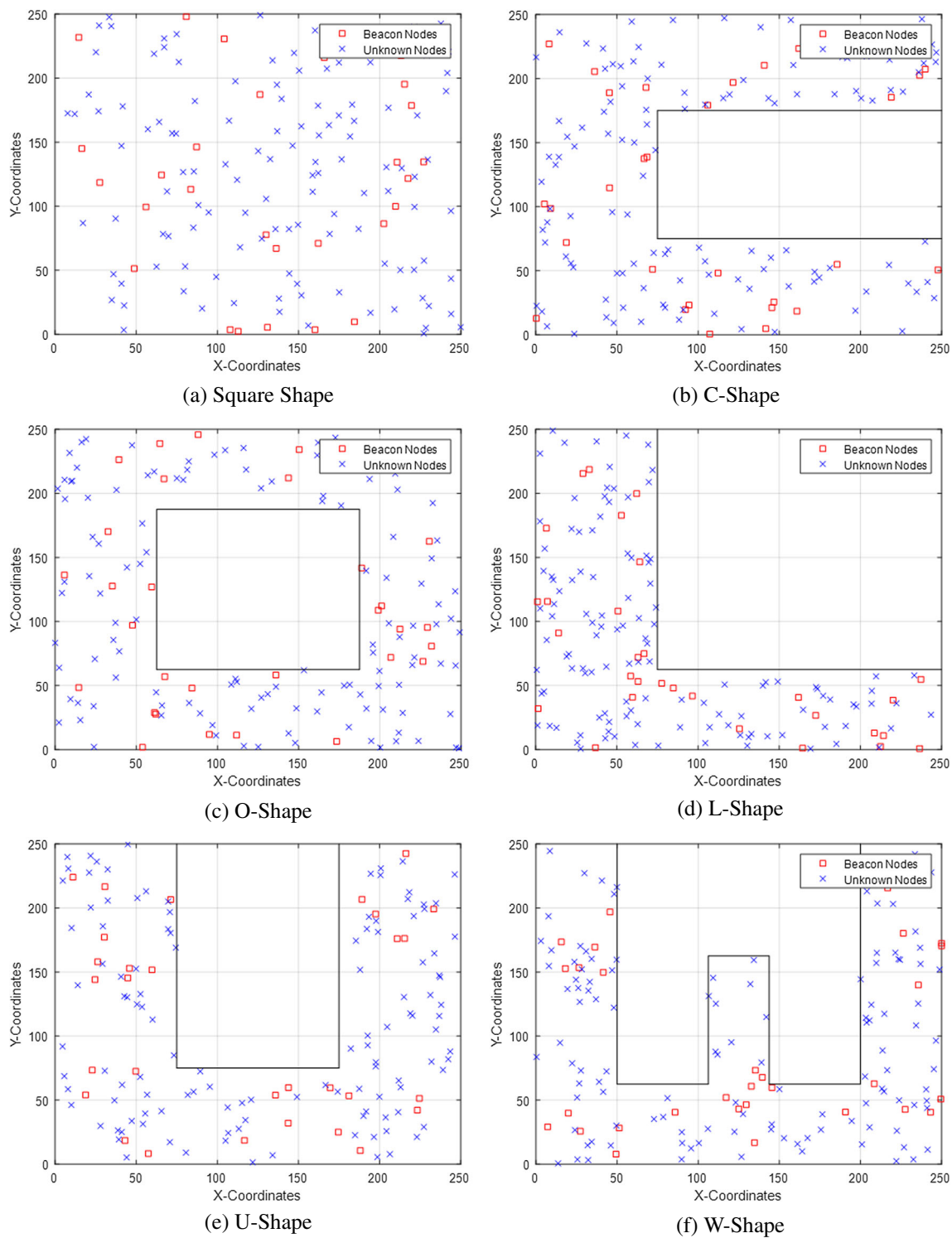


Fig. 7 Topologies (Square Shape, C-Shape, O-Shape, L-Shape, U-Shape, and WShape). **a** Square Shape. **b** C-Shape. **c** O-Shape. **d** L-Shape. **e** U-Shape. **f** W-Shape

However, this requires more beacons to achieve the needed accuracy, especially in random topology.

The received packets in the APIT algorithm increased when we added more beacon nodes in the network. This is due that in APIT process, beacon nodes are required to broadcast their

positions to all unknown nodes so that the three beacons in the triangle will save their positions and the needed RSS values. On the other hand, the number of transmitted packets decreased when the beacon density increased. These transmitted packets are representing the step when the unknown nodes received

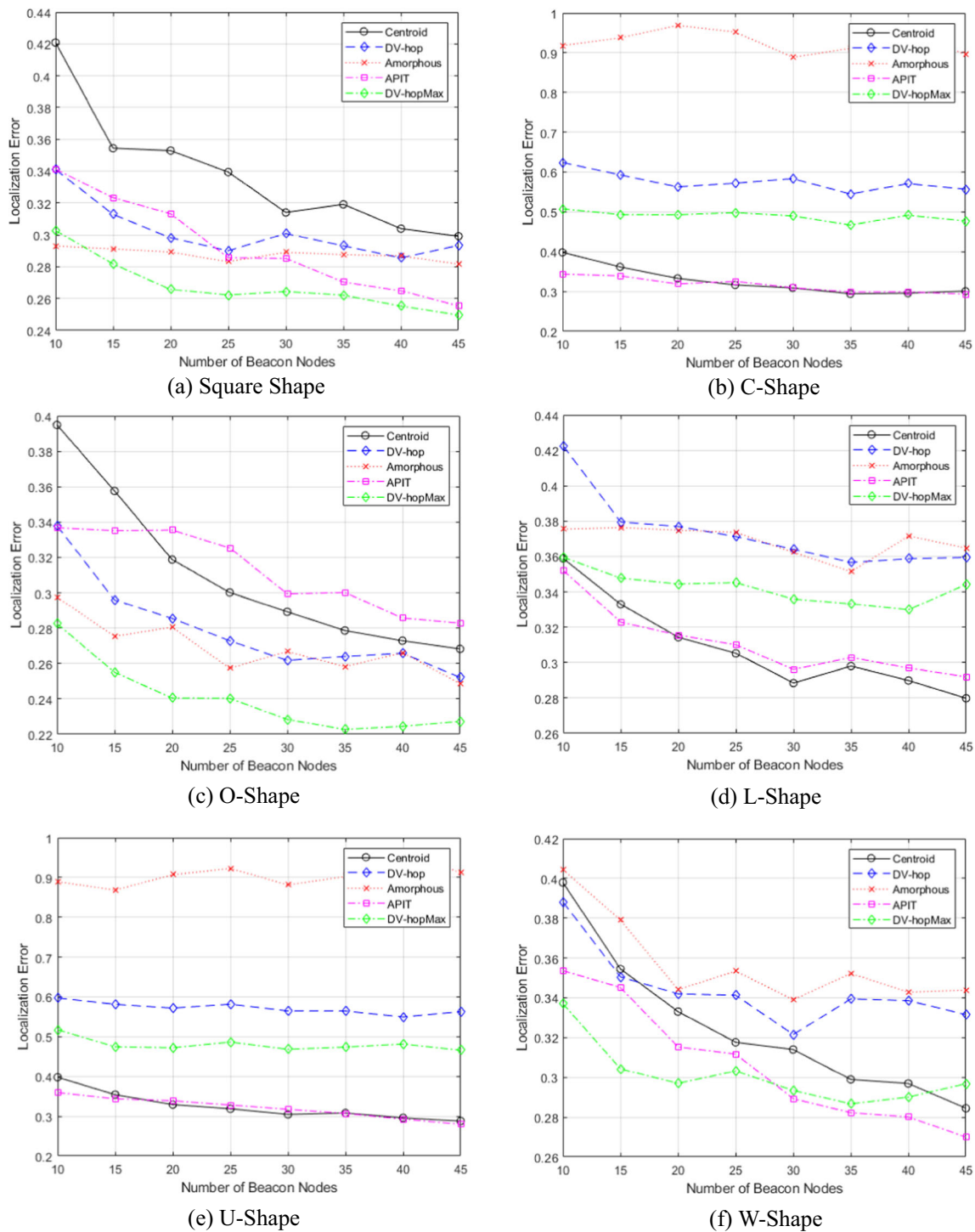


Fig. 8 Localization error for different numbers of beacon nodes. **a** Square Shape. **b** C-Shape. **c** O-Shape. **d** L-Shape. **e** U-Shape. **f** W-Shape

packets from the beacon nodes; it would directly send the packet that contains the information to its neighbours within its communication range. Nevertheless, the APIT has same shape for the energy consumption for different topologies. Lastly, DV-Hop algorithms need a high communication cost but they can achieve a good accuracy by using less number of beacons comparing with APIT and centroid.

- Impact of node density: It represents the number of beacon and unknown nodes who are in communication range with a specific node. We changed this number between 100 to 500 nodes to study the behaviour of the localization error and energy consumption.

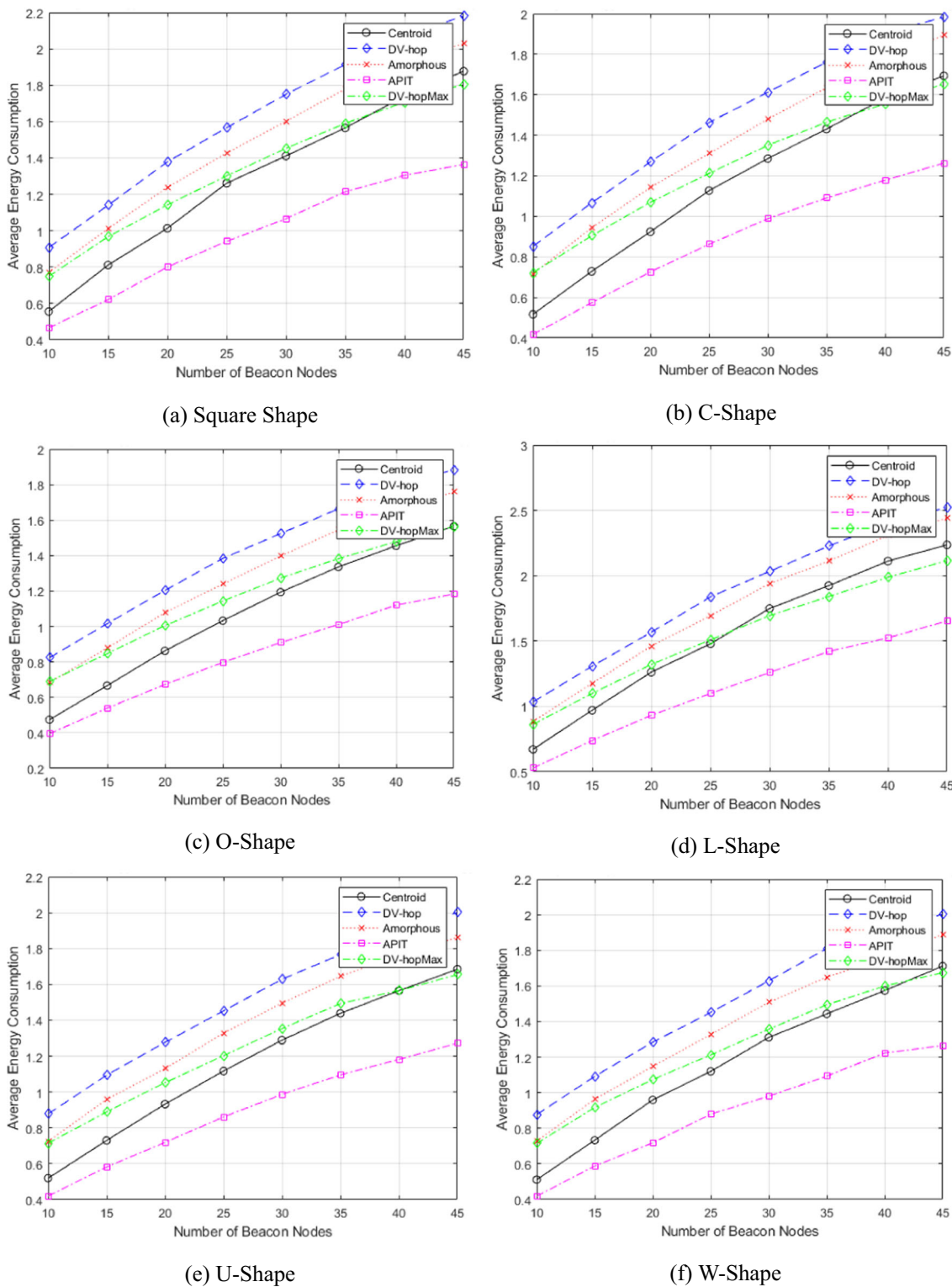


Fig. 9 Average energy consumption for different numbers of beacon nodes. **a** Square Shape. **b** C-Shape. **c** O-Shape. **d** L-Shape. **e** U-Shape. **f** W-Shape

Figure 10 demonstrates the localization error of DV-HopMax, APIT, DV-Hop, Centroid and Amorphous algorithms for all topologies by setting the beacon nodes to be 20% of the total nodes and changing the number of nodes

from 100 to 500 nodes. In general, it shows that the average localization error decreases as the number of neighbours increases. As we can see from Fig. the DV-HopMax and APIT algorithms are robust to varying node density and produce

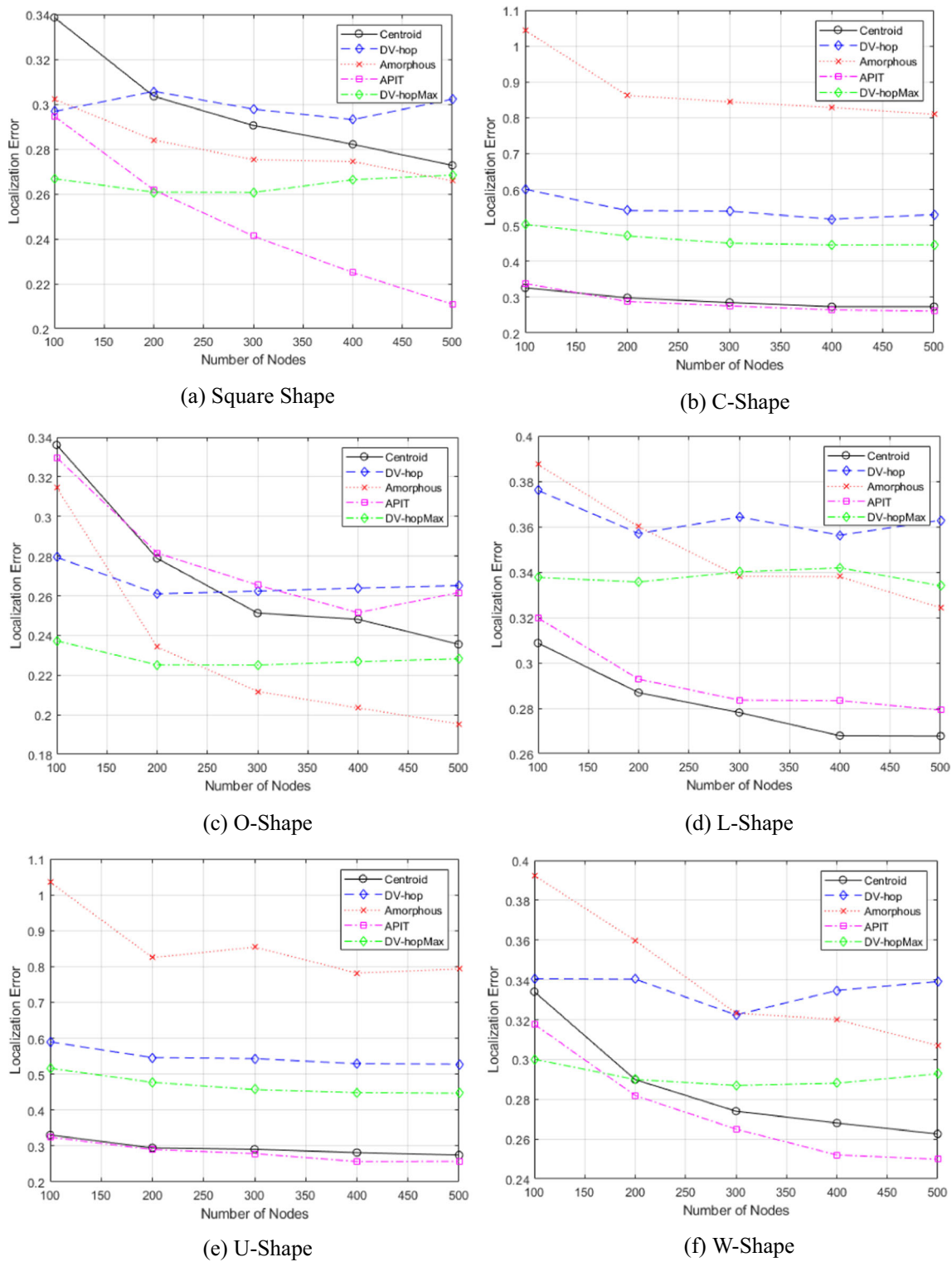


Fig. 10 Localization error for different numbers of nodes. **a** Square Shape. **b** C-Shape. **c** O-Shape. **d** L-Shape. **e** U-Shape. **f** W-Shape

good results as long as the neighbour density increases and it achieves the best performance in terms of localization error when the number of nodes is changing in square topology compared to other localization algorithms. It also shows that the localization errors of the DV-HopMax and APIT

algorithms for most topologies except C and U shapes are 0.26–0.34 for 100 nodes and 0.25–0.29 for 500 nodes. Moreover, Centroid algorithm produces better results than the DV-Hop, DV-HopMax and Amorphous algorithms in C-shape, L-shape, U-shape and W-shape random topologies as

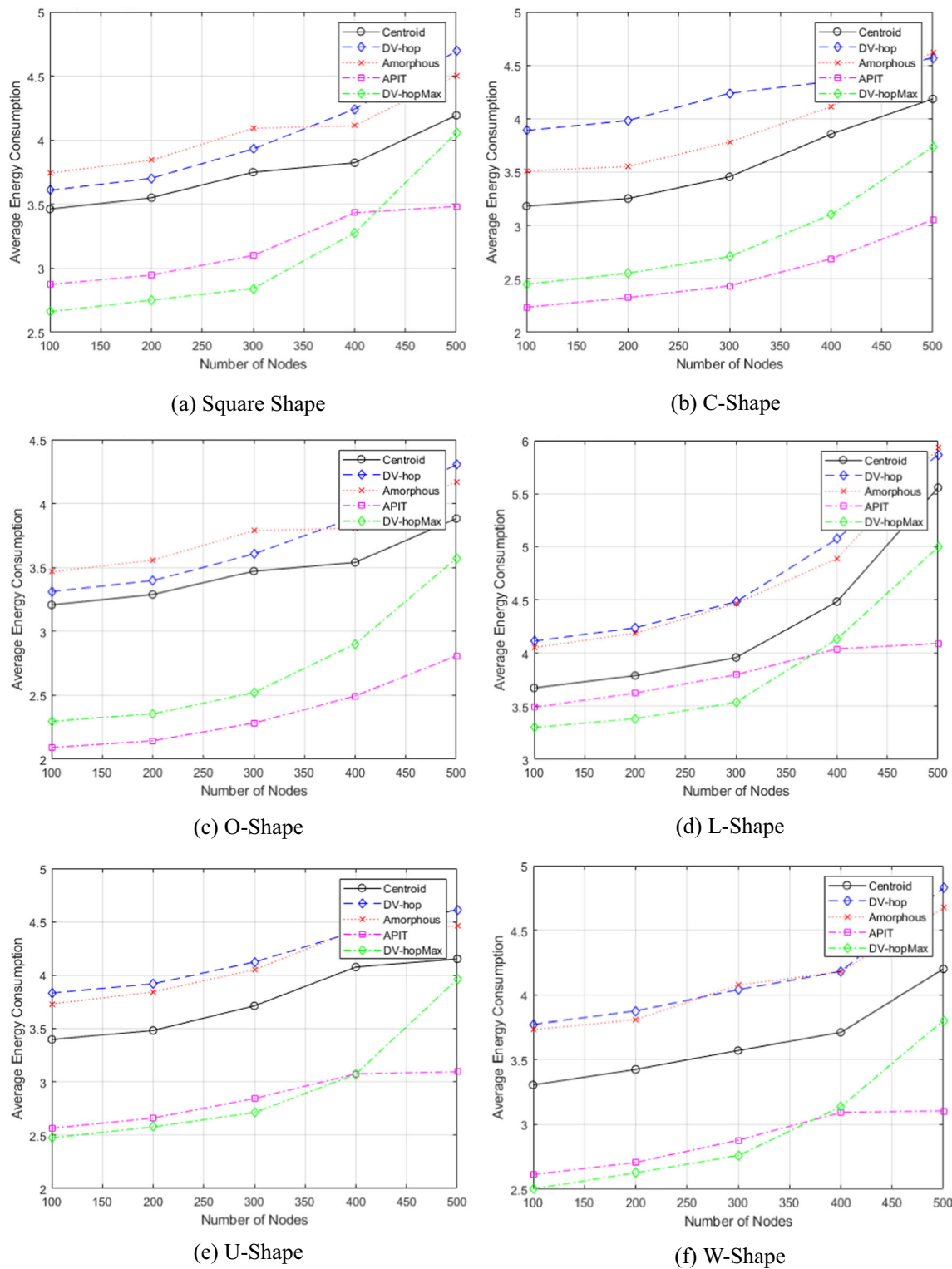


Fig. 11 Average energy consumption for different numbers of nodes. **a** Square Shape. **b** C-Shape. **c** O-Shape. **d** L-Shape. **e** U-Shape. **f** W-Shape

long as the neighbour density increases, whereas DV-Hop and Amorphous algorithms perform better than the Centroid algorithm in square and O-shape random topologies.

In Figure 11, the number of nodes is varied between 100 and 500 nodes, the number of beacons is 20% of the total

nodes and the communication range is the same as in the previous parts. Moreover, the area of this network is 250 m × 250 m. For APIT, DV-Hop and DV-HopMax, the transmitted packets are directly proportional to the number of nodes. However, in APIT, the number of transmitted

packets is less compared to DV-Hop. Dv-hop mainly needs to transmit packets to all beacon nodes in the network, but APIT transmits only to the neighbouring nodes with only one single hop. The energy consumption is less due to that reason.

In the Centroid algorithm, the number of transmitted packets received by the node in the networks is directly proportional to the node density due to its process in finding the unknown node location by using the centroid equation. Hence, the energy consumption depends on the neighbouring beacons for any unknown node. Additionally, in the APIT algorithm, the area is divided into sub-areas and each sub-area contains three beacons. The energy consumption increases when one beacon from one area transmits a packet to the next area, especially in the random topologies. In DV-Hop, as explained before, SNs that are closed to the beacons receive more packets than others when the number of nodes increases based on the shortest path algorithm. Hence, this increases the traffic in the network as well as increases transmitting, receiving and computation energy that is needed for the network in order to achieve the localization process.

- Impact of communication range: most of the previous localization algorithms assume that the node has a circular radio range. Nevertheless, we are going to change the degree of irregularity (DOI) in this section. It is defined as the maximum path loss percentage variation per unit degree change in the direction of radio propagation. When the DOI is 0, there is no range variation, and the communication range is a perfect sphere. Otherwise, the communication range becomes irregular as shown in Fig. 12. We change this parameter from 0.1 to 0.5 in order to study the new behaviour for these metrics.

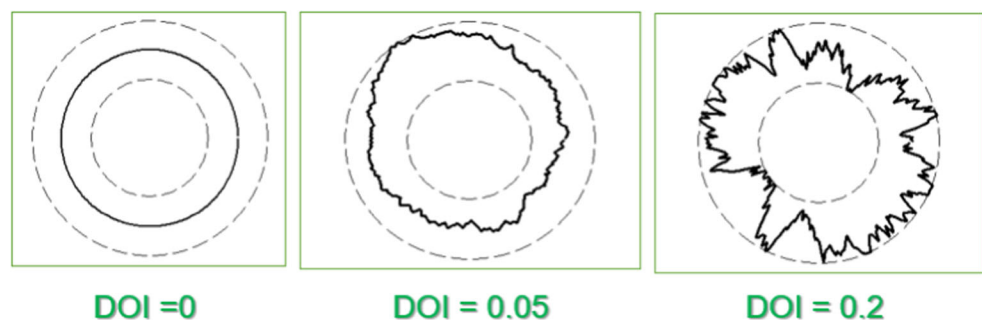
Figure 13 demonstrates the localization error for the five algorithms for all topologies as the degree of integrity values are changed. Overall, Fig. 13 shows that the average localization error increases as the DOI value

increases. We notify that APIT and Centroid algorithms are robust to varying DOI value and they attain better results than DV-HopMax, DV-Hop and Amorphous, in terms of localization error, when the DOI value is changing, for all topologies. This is obvious from Fig. 13, which shows that the DV-Based algorithms especially the Amorphous algorithm are more sensitive to irregular radio patterns than the APIT and Centroid algorithms as well. This is mainly due to the fact that the HopSize values in DV-Hop and Amorphous algorithms are less precise in the presence of irregular radio patterns.

The number of nodes, including the beacon nodes, is the same as described in the simulation setup. We vary the communication range to study the impact on the localization error and the consumption energy for the whole network based on the needed number of transmitted/received packets. In Fig. 14, there are no changes in the Centroid algorithm behaviour when the communication range value is changed to achieve better accuracy in spite of using different topologies. This is because the algorithm does not require more connectivity between nodes in finding their positions. On other hand, for Amorphous, DV-Hop/Max and APIT, the communication range is an important parameter for specifying the exact estimated position. In random topology, finding the shortest path in the DV-Hop algorithms is hard if the communication range increases as more nodes are closed to each other. Nevertheless, this figure shows that the consumption values are increased when the communication range value increased. This is due to the increment in the needed number of transmitted and received packets. In APIT algorithm, the localization accuracy is improved when the communication range increased since more information is received from the new neighbouring nodes. Accordingly, this changes the energy consumption for the whole network based on the transmitted/received packets.

In order to summarize our simulation results, we compare five techniques in terms of accuracy, network type, hardware size, overhead, communication cost, scalability, localization error, and consumption energy as illustrated

Fig. 12 Degree of irregularity [51]



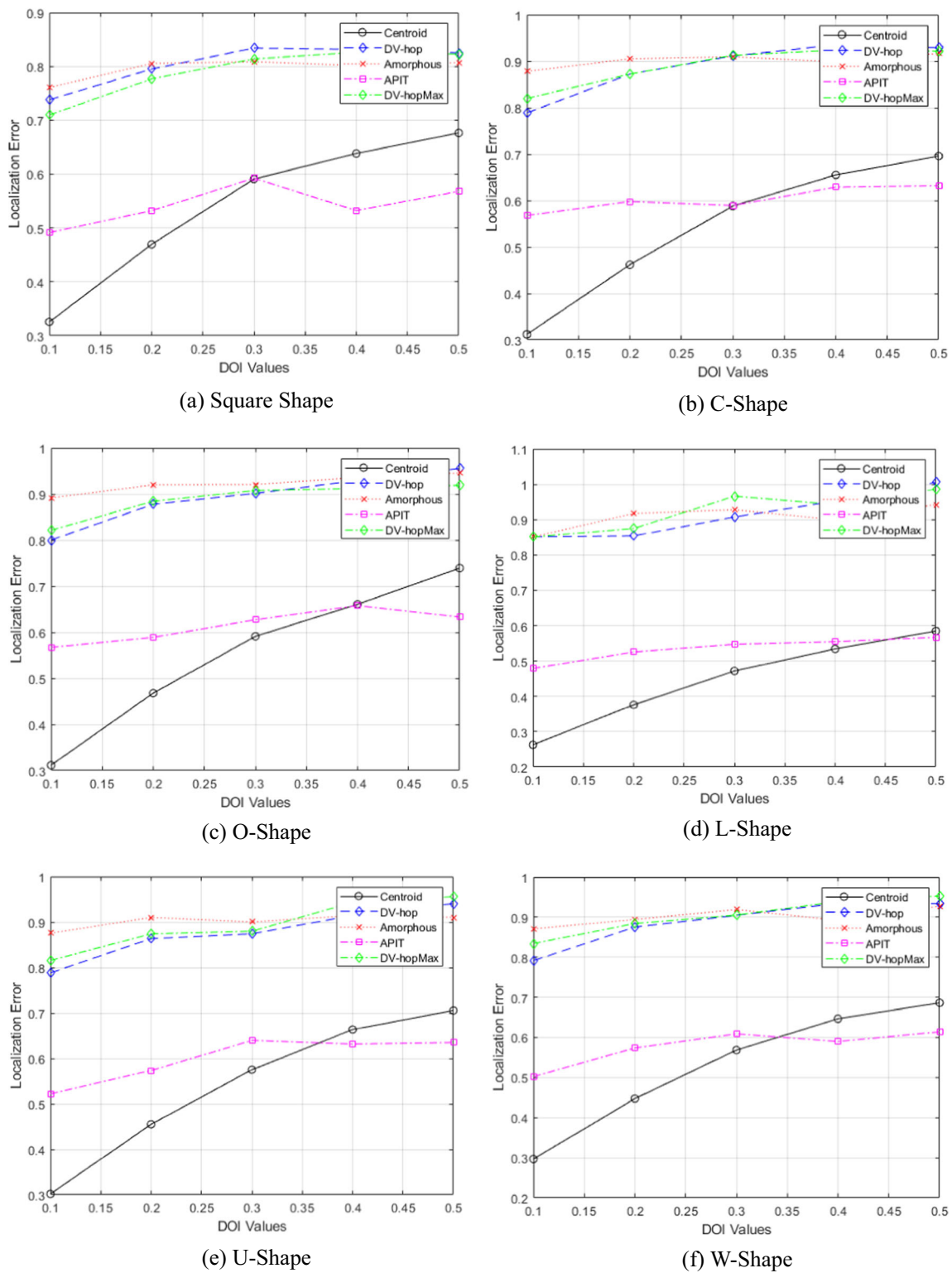


Fig. 13 Localization Error with different DOI values. a Square Shape. b C-Shape. c O-Shape. d L-Shape. e U-Shape. f W-Shape

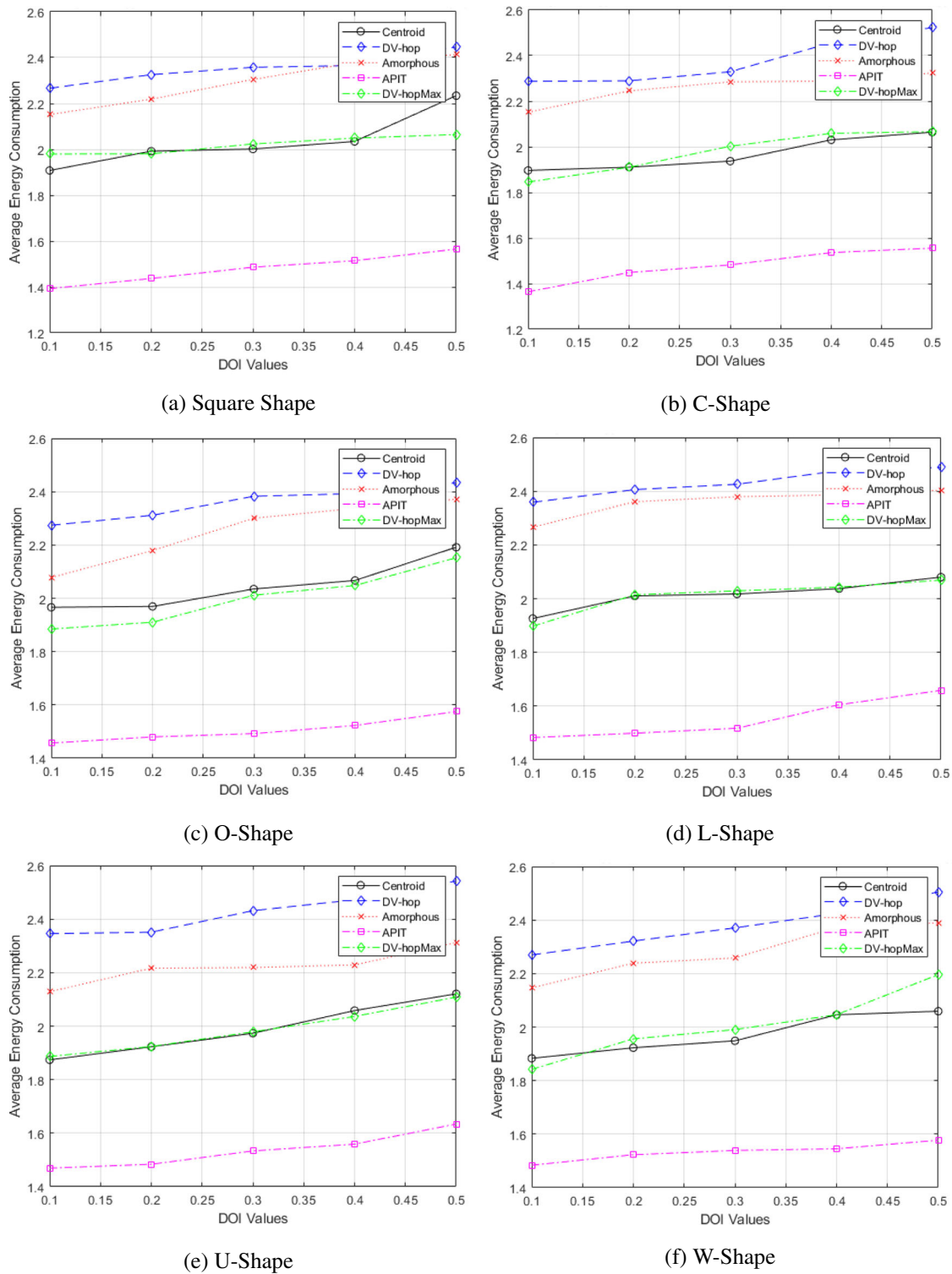


Fig. 14 Average Energy Consumption for different DOI values

Table 4 Comparison between five localization algorithms

Metric		Centroid	DV-Hop	Amorphous	APIT	DV-HopMax
Accuracy		Fair	Average	Average	Good	Good
Classified as		Proximity Based	Network-Connectivity Based	Network-Connectivity Based	Proximity Based	Network-Connectivity Based
Hardware Size		Low	Low	Low	Medium	Low
Overhead		Low	High	High	Medium	Lowest
Communication Cost		Low	High	High	Low	Low
Scalability		Yes	No	No	Yes	Yes
Localization Error	Beacons (+)	High	Medium	Medium	Low	Low
	Nodes (+)	High Change	Small Change	Small Change	High Change	Small Change
	DOI (+)	Low	High	High	Low	High
Consumption Energy	Beacons (+)	Medium	High	High	Low	Medium
	Nodes (+)	Medium	High	High	Low	Low
	DOI (+)	Medium	High	High	Medium	Low

in Table 4. We select the square random topology for our comparison, in order to show the impact of these techniques on the network behavior under the popular topology. Both APIT and DV-HopMax are the best comparing with other techniques in term of error and consumption energy. On the other hand, Centroid considers as the worst technique in term of the localization error. Whereas, DV-Hop and Amorphous have similar results but with less performance comparing with DV-HopMax and APIT and better than Centroid in term of the localization error. At same time, APIT has higher complexity comparing with other four techniques due to the needed stages during the APIT setup. In term of the communication cost, DV-Hop and Amorphous techniques have the highest values due to the increment in the number of hops and the network will be more connected when the number of beacons and nodes are increased. Therefore, the consumption energy will be increased directly in both techniques. However, DV-HopMax consumes less energy since the number of hops is limited during the update stage. Moreover, when the DOI values are increased during the simulation, the localization error and consumption energy increased dramatically in DV-Hop and Amorphous comparing with other techniques. The comparison between these localization algorithms for the previous section is summarized as in Table 5. DV-HopMax is the best in terms of accuracy and has less error value compared to other techniques.

5 Conclusion

In this paper, we have reviewed and compared different range-free localization algorithms, namely Centroid,

Amorphous, APIT, DV-Hop and DV-HopMax algorithms in different topologies (Square, C, O, L, U and W) shapes. Based on topology, localization algorithms have different accuracy values. For instance, Centroid algorithm performs worse than the DV-Hop and Amorphous algorithms in square and O-shape random topologies. In contrast, the Centroid algorithm outperforms both of the DV-Hop and Amorphous algorithms in C-shape, L-shape, U-shape and W-shape random topologies. However, DV-HopMax technique reduces the computational overhead and the overall cost of the network compared with all algorithms. Moreover, the communication range parameter for the beacon nodes and unknown nodes in executing the localization techniques absolutely has a direct impact on their energy consumption values experienced by the sensor node within the localization process. Distances between sensor nodes have an impact on the energy consumption of the whole sensor network, especially in the flooding step. Future work on WSNs localization techniques should address the current tracking challenges and the problem of low accuracy in realistic applications. Indeed, we need to extend the comparative study to include more recent techniques that adopt in the new technologies in term of the popular performance metrics. Moreover, we need to study the behaviour of the WSNs when the anchor nodes have predefined positions not randomly distributed as in this comparative study.

Acknowledgements We are thankful to King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia for providing the necessary lab equipment and setup for experimentation.

Funding information Acadia University and Natural Sciences and Engineering Research Council (NSERC) of Canada funded this research.

Appendix

Table 6 Part 2 - Comparison between group of localization techniques

Ref.	Year	Localization Technique	Performance Metric	Advantages	Limitations/Disadvantages
[27, 28]	2019, 2018	Event-driven RSS based localization for underwater WSNs.	Accuracy.	It is a fast localization approach.	It needs a linear fitting and optimization stages.
[29]	2018	MLE and the generalized trust area technique.	Localization Error.	It achieves good accuracy values.	The performance may be degraded due to the approximations stage, and the output solution will be sub-optimal solution.
[30]	2019	Bi-iterative estimator.	Structure size and network resources requirements.	It studies the localization and synchronization at same time. It achieves better convergence. Less anchor nodes and resources.	It needs to use the invariance theory of LaSalle.
[31]	2018	Deep long short-term memory based model for indoor localization (DeepML).	Localization Error.	It achieves a high accuracy.	Training time is high.
[32]	2018	Fingerprint, weighted KNN approach, RSS.	Accuracy.	It improves accuracy when there is signal strength fluctuations.	Data base is required. Online and Offline phases.
[33]	2018	Collaborative Fingerprint.	Localization Error.	It is for complex and crowded indoor environment based on grid. It reduces the error values.	Predictive technique. Data base is required. Online and Offline phases.
[34]	2018	Fingerprint-based localization.	Localization Time, Localization Error.	It achieves high precision.	RSS is the main parameter. It needs a spatial resolution of database.
[35]	2018	Fingerprint-based localization.	Accuracy.	It minimizes RSS measuring noise between WSN nodes. It achieves high accuracy regardless the location of the target node, the number of anchor nodes and the calibration cell.	Complexity is high.
[36]	2018	Temporal correlation of the RSS.	Reliability and localization error.	It achieves high accuracy.	It assumes some parameters in the model rather than to optimize them.
[37]	2018	Diversity-leveraging architecture.	Accuracy and localization error.	It achieves high accuracy by comparing the obtained signal samples with a previous reference for each sender.	It is a client-server model.
[38]	2018	DV-Hop and PSO.	Accuracy, location error, error variance, and coverage.	No additional hardware and less computational process. It decreases the localization error. It outperforms the basic DV-Hop.	It is time consuming technique.
[39]	2019	SVM and twin SVM approaches.	Accuracy and localization error.	SVM and TWSVM are used to show the region of event detection. It is suitable for dense networks.	The central location of the target area is an approximation of the source position. Training is time consuming stage.
[40]	2019	RSS in underwater WSNs.	Energy harvesting and localization error.	It reduces the error by introducing a novel matrix completion technique in estimating the shortest paths.	Nodes start communicating when they have only the suitable energy storage. The block kernel matrices is used.
[42]	2019	FIS three-dimensional WSNs.	Accuracy and localization error.	It achieves less location error and high accuracy.	Complexity is high.

Table 5 Part 1 - Comparison between group of localization techniques

Ref.	Year	Localization Technique	Performance Metric	Advantages	Limitations/Disadvantages
[8]	2016	Swarm Intelligence Bat Algorithm.	Localization Error.	It reduces the error rate.	Only for regular localization. It needs semi-mobile anchor nodes.
[5]	2016	Cooperative Range Based.	Energy, Accuracy, Localization Time, and Deployment Strategy.	It reduces energy wastage and localization time by using MAC and routing together.	It is affected any external environment.
[9]	2017	AIES-TOA, RMS-STD.	Error, Error tolerance, and Distance Error.	It outperforms other techniques in terms of error.	It depends on the TOA only.
[12]	2018	RSS indicator with filtering techniques.	Accuracy.	It achieves minimal computational overhead.	Filtering stages causes delays.
[13]	2016	DV-Hop.	Position Estimation Error.	High precision, less traffic load and good performance in the homogenous and non-homogenous environment.	Extra Hardware.
[14]	2017	3D genetic algorithm based Improved Distance Vector Hop.	Accuracy, Computational Efficiency, and Positioning Coverage.	It finds the optimal hop size. It increases the localization coverage and accuracy.	Not applicable for terrains environment.
[16]	2017	Range-free with genetic algorithms.	Optimal path and accuracy.	It is an energy-efficient technique. Good accuracy and it is flexible.	Error is still high.
[17]	2016	Fingerprint, Centroid.	Accuracy and Average Error.	No extra hardware is needed. Comparative study using Cooja.	It is limited to two techniques.
[19]	2019	DV-Hop, NSGA-II.	Localization Error and Convergence.	It reduces the positioning error. It studies DV-Hop as multi-objective localization algorithm. It reduces the convergence domain of unknown nodes.	Suitable for network with no obstacles. Complexity is high.
[20]	2019	RSS indicator with Fuzzy.	Localization Error.	It is a decision making technique (intelligent). Low localization error. It has high reliability and good environmental adaptability.	Complexity is high.
[21]	2019	Fingerprint	Localization Error.	It shows very high performance under low and high speed tests. It support the reliability and validity of the system with low localization error values.	It depends on four technologies. Complexity and delay are high.
[2]	2019	TOA, RSS indicator.	Accuracy.	It improves accuracy.	Complexity is high due to variable channel characteristic.
[23]	2019	RSS indicator.	Accuracy and resources.	It is a lightweight technique. It is a fully distributed technique. It achieves high performance.	It tracks neighbouring nodes. It is for small, low resources WSNs.
[24]	2019	Genetic algorithm with LMS.	Overhead communication, network traffic, and accuracy.	It minimizes the hop counts. Overhead, network traffic, and localization inaccuracy are decreased.	Complexity is high.
[25]	2019	Adaptive iterative localization based on steepest gradient descent.	Accuracy and convergence speed.	It improves the accuracy and the convergence speed.	Sigmoid function in the searching technique is used with a variable step (complexity and delays).
[26]	2019	TOA, RSS indicator.	Accuracy and remaining energy.	It achieves high accuracy and it is an energy efficient algorithm.	It depends on the unknown nodes location and the channel components.

References

- Desai J, Tureli U (2017) Evaluating performance of various localization algorithms in wireless and sensor networks. In IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications. IEEE, pp. 1–5
- Arampatzis T, Lygeros J, Manesis S (2005) A survey of applications of wireless sensors and wireless sensor networks. In Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control 2005. IEEE, pp. 719–724
- Mekelleche F, Haffaf H (2017) Classification and comparison of range-based localization techniques in wireless sensor networks. *J Commun* 12(4):221–227
- Hu L, Evans D (2004) Localization for mobile sensor networks. Proceedings of the 10th annual International Conference on Mobile Computing and Networking, ACM:45–57
- Pandey S, Varma S (2016) A range based localization system in multihop wireless sensor networks: a distributed cooperative approach. *Wirel Pers Commun* 86(2):615–634
- Alippi C, Vanini G (2006) A RSSI-based and calibrated centralized localization technique for wireless sensor networks. In Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06). IEEE, pp. 5.
- Singh SP, Sharma S (2015) Range free localization techniques in wireless sensor networks: a review. *Procedia Computer Science* 57: 7–16
- Tuba E, Tuba M, Simian D (2016) Range based wireless sensor node localization using bat algorithm. Proceedings of the 13th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, ACM:41–44
- Thimmaiah SH, Mahadevan G (2017) A range based localization error minimization technique for wireless sensor network. *Indonesian Journal of Electrical Engineering and Computer Science* 7:395–398
- Feng W-J, Bi X-W, Jiang R (2012) A novel adaptive cooperative location algorithm for wireless sensor networks. *Int J Autom Comput* 9(5):539–544
- Gezici S, Guvenc I, Sahinoglu Z (2008) On the performance of linear least-squares estimation in wireless positioning systems. In IEEE International Conference on Communications. IEEE, pp. 4203–4208
- Koledoye MA, De Martini D, Rigoni S, Facchinetti T (2018) A comparison of RSSI filtering techniques for range-based localization. In IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), vol. 1. IEEE, pp. 761–767
- Ahmadi Y, Neda N, Ghazizadeh R (2016) Range free localization in wireless sensor networks for homogeneous and non-homogeneous environment. *IEEE Sensors J* 16(22):8018–8026
- Sharma G, Kumar A (2017) Improved range-free localization for three-dimensional wireless sensor networks using genetic algorithm. *Computers & Electrical Engineering*
- Prashar D, Jyoti K, Kumar D (2016) A comparison of distributed range free localization algorithms in wireless sensor networks. *Indian J Sci Technol* 9(28):1–5
- Najeh T, Sassi H, Liouane N (2018) A novel range free localization algorithm in wireless sensor networks based on connectivity and genetic algorithms. *Int J Wireless Inf Networks* 25(1):88–97
- Shahra EQ, Sheltami TR, Shakshuki EM (2016) Comparative study of fingerprint and centroid localization protocol using cooja. *Procedia Computer Science* 98:16–23
- Yamei L (2014) An improved dv-hop localization algorithm based on energy-saving non-ranging optimization. *J Networks* 9(11): 3182–3188
- Wang P, Xue F, Li H, Cui Z, Chen J (2019) A multi-objective DV-hop localization algorithm based on NSGA-II in internet of things. *Mathematics* 7(2):184
- Yan J, Qiao R, Tang L, Zheng C, Fan B (2019) A fuzzy decision based WSN localization algorithm for wise healthcare. *China Communications* 16(4):208–218
- Zou D, Chen S, Han S, Meng W, An D, Li J, Zhao W (2019) Design of a practical WSN based fingerprint localization system. *Mobile Networks and Applications*:1–13
- Ruz ML, Garrido J, Jim'enez J, Virrankoski R, V'azquez F (2019) Simulation tool for the analysis of cooperative localization algorithms for wireless sensor networks. *Sensors* 19(13):2866
- Xiong H, Sichitiu ML (2019) A lightweight localization solution for small, low resources WSNS. *J Sens Actuator Netw* 8(2):26
- Sivasakthiselvan S, Nagarajan V (2019) A new localization technique for node positioning in wireless sensor networks. *Clust Comput* 22(2):4027–4034
- Chen H, Tan G (2019) Adaptive iteration localization algorithm based on RSSI in wireless sensor networks. *Clust Comput* 22(2): 3059–3067
- Nguyen TLN, Vy TD, Shin Y (2019) An efficient hybrid RSS-AOA localization for 3d wireless sensor networks. *Sensors* 19(9): 2121
- Nguyen TLN, Shin Y (2019) An efficient RSS localization for underwater wireless sensor networks. *Sensors* 19(14):3105
- Nguyen TL, Shin Y (2018) Improved RSS-based localization using linear regression approach in UWSNS. International Conference on Information and Communication Technology Convergence (ICTC), IEEE, 1208–1213
- Chang S, Li Y, He Y, Wang H (2018) Target localization in underwater acoustic sensor networks using rss measurements. *Appl Sci* 8(2):225
- Tian Q, Feng D-Z, Hu H-S, Yang F, Wei L (2019) Bi-iterative algorithm for joint localization and time synchronization in wireless sensor networks. *Signal Process* 154:304–313
- Wang X, Yu Z, Mao S (2018) Deepml: Deep lstm for indoor localization with smartphone magnetic and light sensors. In IEEE International Conference on Communications (ICC). IEEE, pp. 1–6
- Cheng L, Li Y, Zhang M, Wang C (2018) A fingerprint localization method based on weighted KNN algorithm. In IEEE 18th International Conference on Communication Technology (ICCT). IEEE, pp. 1271–1275
- Luo J, Zhang Z, Liu C, Luo H (2018) Reliable and cooperative target tracking based on wsn and wifi in indoor wireless networks. *IEEE Access* 6:24 846–24 855
- Chemtanomwong P, Sooraksa P (2018) Soft-clustering technique for fingerprint-based localization. *Sensors and Materials* 30(10): 2221–2233
- Fang X, Jiang Z, Nan L, Chen L (2018) Optimal weighted k-nearest neighbour algorithm for wireless sensor network fingerprint localisation in noisy environment. *IET Commun* 12(10):1171–1177
- Tian X, Wang M, Li W, Jiang B, Xu D, Wang X, Xu J (2017) Improve accuracy of finger printing localization with temporal correlation of the RSS. *IEEE Trans Mob Comput* 17(1):113–126
- Di Felice M, Bocanegra C, Chowdhury KR (2018) Wi-lo: wireless indoor localization through multi-source radio fingerprinting. In 10th International Conference on Communication Systems & Networks (COMSNETS). IEEE, pp. 305–311
- Singh SP, Sharma SC (2019) Implementation of a pso based improved localization algorithm for wireless sensor networks. *IETE J Res* 65(4):502–514
- Javadi SH, Moosaei H, Ciuonzo D (2019) Learning wireless sensor networks for source localization. *Sensors* 19(3):635
- Saeed N, Celik A, Al-Naffouri TY, Alouini M-S (2019) Localization of energy harvesting empowered underwater optical

- wireless sensor networks. *IEEE Trans Wirel Commun* 18(5):2652–2663
41. A. Kumar A, Paul RK (2019) Progressive localization using mobile anchor in wireless sensor network
 42. Liang J, Yu X, Liu X, Mao C, Ren J (2019) Target detection, localization, and tracking in wireless sensor networks. *Mission-Oriented Sensor Networks and Systems: Art and Science*, Springer 309–361
 43. Bulusu N, Heidemann J, Estrin D et al (2000) GPS-less low-cost outdoor localization for very small devices. *IEEE Pers Commun* 7(5):28–34
 44. Niculescu D, Nath B (2003) Dv based positioning in ad hoc networks. *Telecommun Syst* 22(1–4):267–280
 45. Hartley RI, Sturm P (1997) Triangulation. *Comput Vis Image Underst* 68(2):146–157
 46. Nagpal R, Shrobe H, Bachrach J (2003) Organizing a global coordinate system from local information on an ad hoc sensor network. *Information Processing in Sensor Networks*, Springer 333–348
 47. Ying L (2011) An adaptive multi-hop distance localization algorithm in WSN. *Manufacturing Automation* 33:161–163
 48. He T, Huang C, Blum BM, Stankovic JA, Abdelzaher T (2003) Range-free localization schemes for large scale sensor networks. *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, ACM 81–95
 49. Shahzad F, Sheltami TR, Shakshuki EM (2016) DV-maxhop: a fast and accurate range-free localization algorithm for anisotropic wireless networks. *IEEE Trans Mob Comput* 16(9):2494–2505
 50. Khelifi F, Bradai A, Benslimane A, Kaddachi ML, Atri M (2017) Energy-saving performance of an improved DV-hop localization algorithm for wireless sensor networks. *GLOBECOM 2017–2017 IEEE Global Communications Conference*, IEEE 1–6
 51. Zhou G, He T, Krishnamurthy S, Stankovic JA (2006) Models and solutions for radio irregularity in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)* 2(2):221–262

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.