



Fault-Tolerant UAV Data Acquisition Schemes

Temesgen Seyoum Alemayehu¹ · Jai-Hoon Kim² · Wonsik Yoon³

Published online: 8 May 2020

© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Through the use of UAV, the functional lifetime of WSN can be elongated in exchange for higher data delivery latency as the UAV replaces the multi-hop communication among nodes during data acquisition. Due to the NP-hardness of the TSP whose computational complexity increases exponentially as an increment of number of nodes, heuristic algorithms, such as nearest neighbor heuristic TSP algorithm (NN), have been developed for reducing this data delivery latency in shortest possible time. In our previous research work we have published the directional NN algorithm directed to the next nearest node (DDNN) (Alemayehu and Kim in *Wirel Pers Commun* 95:3271–3285, 2017) which modifies the existing NN algorithm to gain a reduction in this data delivery latency. However, the DDNN algorithm does not consider the reliability of the system in case of node or link failures. To collect the sensing data rapidly and reliably, the DDNN algorithm should be able to react to node or link failures and manage the data transmissions effectively in the network. In this study, we propose an extension of the DDNN scheme, fault tolerable DDNN scheme for data gathering to gain a reduction in the data acquisition time with fault-tolerant capability. The performance analysis has demonstrated that our proposed algorithm tolerates fault in case of malfunctions of sensors due to node/link failures and improves the detection rate of the DDNN scheme up to 34.93% at the cost of a little bit distance.

Keywords UAV · Drone · Data acquisition · Monitoring · WSN · Fault-tolerance

✉ Jai-Hoon Kim
jaikim@ajou.ac.kr

Temesgen Seyoum Alemayehu
tomsymal@yahoo.com

Wonsik Yoon
wsyoon@ajou.ac.kr

¹ Department of Computer Engineering, Ajou University, Suwon, South Korea

² Department of Cyber Security, Ajou University, Suwon, South Korea

³ Department of Electrical and Computer Engineering, Ajou University, Suwon, South Korea

1 Introduction

WSNs are useful to monitor physical conditions by sensing events happening within specific area and then passing the data gathered to the sink. It has unlimited potential applications such as military, environmental control, habitat monitoring and acoustic data gathering, etc. Compared with the traditional computer networks, WSNs consist of many resource-constrained small smart nodes with limited processing power, memory space, data processing ability, and communication distance [1]. The traditional approach of relaying data in WSN diminishes the network's lifetime as sensor nodes are used as a relay and most of their power is consumed by radio transmission in the multi-hop routing [2]. As a solution of minimizing wastage of power, UAVs have been used for traversing the network and collecting the sensing data directly from the sensor nodes [3–6] to prolong network lifetime by reducing sensors power that might be wasted due to data transmission through multi-hop routing.

UAVs help to maximize the data communication capability of the network as relay of nodes [7]. They support data acquisition from a wide geographical area of interest that are harsh, remote and/or isolated due to lack of communication infrastructure [8]. For instance, they can be used in various military and civilian fields that include search and rescue missions [9], wildfire management, observation support [10], agricultural monitoring, border surveillance and monitoring, environmental and meteorological monitoring [11], etc.

While UAVs provide enormous benefits as a relay for WSNs, it also introduces some challenges that need to be considered. It spends more time to collect data, which in turn incurs a higher data delivery latency. In our previous research work we have published the directional NN algorithm directed to the next nearest node (DDNN) [12]. The DDNN algorithm modifies the existing NN algorithm to gain a reduction in the data delivery latency. Although node and/or link failures occur frequently in WSNs, fault tolerant aspects related to this algorithm were left open and are the focus of this study. As sensor nodes are prone to failure due to energy depletion, hardware failure, communication link errors, and so on, fault tolerance should be seriously considered in UAV relay WSNs. In this paper, we propose a fault tolerable DDNN scheme (FT-DDNN) as an extension of the DDNN algorithm to enable the network to persist in spite of the failed nodes and/or links as well as to reduce the data acquisition latency of UAV relay WSN. The objective of this study is to contribute an efficient method that produce shorter path that enable faster data delivery with fault tolerable capability.

The rest of the paper is organized as follows. Section 2 reviews related works and presents problem description. Section 3 addresses our proposed algorithms. Section 4 describes the Performance analysis of the FT-DDNN algorithm. Section 5 presents performance evaluation and results, and finally Sect. 6 concludes the paper.

2 Related Works and Problem Description

UAVs play a significant role in achieving the effectiveness of data gathering in WSNs. The use of UAV as a relay in wireless sensor networks (WSNs) significantly reduces sensor nodes' energy consumption as the UAV replaces the multi-hop communication among nodes. This is due to the large radio coverage and conveniently rechargeable batteries of UAVs [4]. The authors in [5] proposes the usage of UAVs as a relay to guarantee the

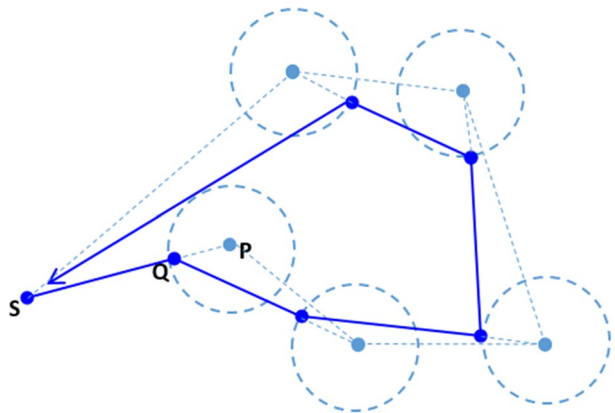
delivery of sensed data from WSN nodes to the base station. They pointed out how UAV resolves connectivity difficulties in WSNs due to failures that may lead to isolation of nodes. Due to the relatively lower speed of UAVs, when compared with electromagnetic or acoustic waves, collecting data through UAVs causes significant delay compared with multi-hop forwarding [6]. This delay of routing data through UAVs in turn suffers higher data delivery latency. Previous research works in the literature focused on TSP based solutions in which each node is visited exactly once in the tour [13, 14].

Since TSP belongs to a family of NP complete problems [15] whose computational complexity increases exponentially with the increment of number of nodes, heuristic algorithms, such as nearest neighbor heuristic TSP algorithm (NN), have been developed for reducing this data delivery latency in shortest possible time. NN is the most straightforward TSP heuristic algorithm which is nearly close to the optimal route with relatively short running time. The key concept of this algorithm is to start with a randomly chosen node and to add the nearest but not yet visited node to the last node in the tour until all the nodes are visited, and then return to the initial node [16]. The NN algorithm can be employed for reducing the length of the tour path of UAV to gather and deliver sensor data in WSN. Although NN runs fast, producing the optimal tour highly depends on the layout of the nodes within the problem.

In our previous research work, we have proposed and published the directional NN algorithm directed to the next nearest node (DDNN) that modify the existing NN algorithm to gain a reduction in the data delivery latency [12]. The basic concept of the DDNN algorithm is to amend the previous NN algorithm to let the UAV initially move directly to the first nearest node of the base station until it reaches to the boundary of the transmission range of the node. Then, it directs to the nearest node of the current node until it arrives to the boundary of the transmission range of the node. The algorithm follows the same procedure until all nodes are visited as shown in Fig. 1. In the figure the dotted line represents the path in the existing NN-TSP algorithm, whereas the solid line indicates the path based on the modified algorithm (DDNN).

The DDNN algorithm requires location of the point where the drone turns its direction to the next move. Suppose that every node corresponds to a point in Euclidean space and the points $P(P_x, P_y)$, $S(S_x, S_y)$ and $Q(Q_x, Q_y)$ are locations of the first node, the drone initial point and the point where the normal path of the previous NN-TSP algorithm meets the transmission range boundary of the first node respectively as shown in Fig. 1.

Fig. 1 UAV path based on the DDNN approach



Consider $(\vec{S} - \vec{P})$ to be a vector in the direction of \vec{S} . We know that the unit vector $\frac{(\vec{S}-P)}{\|\vec{S}-\vec{P}\|}$ is in the same direction with $(\vec{S} - \vec{P})$ but with a length of one unit. If we multiply the transmission range r of the node with the unit vector, we reach the circle (the transmission boundary of the node) at point Q in the direction of the unit vector a distance of r units from node P .

$$\text{Thus, } \vec{Q} = \vec{AP} + r \cdot \frac{(\vec{S} - \vec{P})}{\|\vec{S} - \vec{P}\|}$$

Therefore, the point Q with its X and Y coordinates (the point where the normal path of the previous NN-TSP algorithm meets the transmission range boundary of the node) is thus:

$$Q(Q_x, Q_y) = \left(P_x + r \cdot \frac{S_x - P_x}{\sqrt{(S_x - P_x)^2 + (S_y - P_y)^2}}, \quad P_y + r \cdot \frac{S_y - P_y}{\sqrt{(S_x - P_x)^2 + (S_y - P_y)^2}} \right)$$

The UAV uses the point $Q(Q_x, Q_y)$ as turning point to the next move in order to shorten the tour length for reducing the data delivery time. The detail of the algorithm is provided in our published paper [12]. Although the DDNN algorithm reduces the data delivery latency by modifying the existing nearest neighbor heuristic TSP algorithm (NN), it does not consider the reliability of the system in case of node or link failures. To collect the sensing data rapidly and reliably, the DDNN algorithm should be able to react to node or link failures effectively and manage the data transmissions in the network.

The situation of WSNs is unpredictable due to various reasons including electromagnetic interferences, system noise, the running out of battery, loss of transmission signal and other faults in the networks [17, 18]. Such kind of faults degrade the data transmission quality and reliability of the WSNs. Specifically, WSNs are prone to failures when nodes are deployed in harsh environments. Atmospheric phenomena, such as changes in weather modify signal propagation, which may in turn cause communication errors as signal strength decreases [19]. Several environmental conditions such as humidity and temperature can modify signal propagation. In WSN, even if the hardware condition of nodes is good, communication between sensor nodes may get affected by signal strength, interferences, and obstacles [20]. Radio interference can cause link failure among nodes. Movement of objects near the nodes, may block the signal. For instance, in agricultural fields [21], pointed out that link range is considerably reduced when plants start growing. As link failure degrades quality in WSNs, a fault tolerance mechanism that tolerates this kind of failure need to be developed.

Fault tolerance is the ability of a system to immediately detect and recover from faults and deliver the desired level of functionality in the presence of faults [22, 23]. It is one of the requirements of a dependable system [24]. The authors [25, 26] designed protocols that are to be resilient against communication faults caused by failed nodes. Those protocols mitigate failures by sending multiple copies of data among different routes, thus increasing the probability of correct reception at the access point. However, the cost of sending multiple copies of data among different routes is significant as communication among nodes consume much energy.

The authors in [27] suggested an algorithm to automatically select redundant nodes in WSNs deployed on areas with obstruction, based on availability requirements of an application and network configurations. The redundancy enhances the availability as redundant nodes replace faulty nodes. Although the redundant sensors form a fault tolerant network, it may incur cost of deployment of redundant nodes.

Suggested work in [28] presented a fault tolerance and energy-efficient clustering (FTEC) algorithm to detect and recover the faults of cluster heads and cluster member nodes. In FTEC, a node as a backup cluster head is selected to detect and recover the faults of cluster heads and cluster member nodes. A weighted median is also utilized to detect and recover the faults of cluster nodes. Faulty node is isolated and replaced by a nearby node switched from sleep mode to wake up mode to recover a fault. The existing fault tolerant mechanisms need to use extra hardware and software resources due to faulty situations.

The authors in [29] have proposed dual separate path (DSP) algorithm which is used to determine redundant paths to provide fault-tolerant communication for WSN applications, such as fault-tolerant cyber physical systems. Although increasing the number of disjoint paths or redundant routes in the WSN can significantly enhance fault-tolerant against link failures, finding path weights and sorting paths incur higher computational overhead.

The authors in [30] have introduced a management framework that can provide fault tolerance capability in network nodes and in the communications between them. In the proposed management framework ECRAFT (Energy-Check Point/Recovery/Based Scheme for Fault), each cluster head node should select a node as its check point node. In ECRAFT, cluster heads send aggregated information towards the sink and backup copy of aggregated information to the Check Point node. However, due to the frequent exchange of data between the cluster head and non-cluster head members, energy consumption would increase.

A fault-tolerant cluster based multipath algorithm (FTCM) that utilizes a hybrid energy-efficient distributed clustering approach to cluster nodes is proposed for WSNs in [31]. In FTCM, backup node that stores a copy of the data of cluster head is selected to increase the fault tolerance of cluster head. FTCM also adopts three routing paths between cluster head and base station to increase fault-tolerant routing and improve path reliability. The different routing paths are implemented based on effective routing parameters like residual energy of nodes, number of hops, reliability and propagation speed. However, selecting the backup node for cluster head and discovering an alternative path for reliable data transmission may cause extra computational complexity and resource utilization.

Various mechanisms have been discussed in the literature to tolerate faults in WSNs. However, those mechanisms consume extra costs such as cost of deployment of redundant nodes, use of extra resource utilization, higher computational overhead, high node power consumption, etc. to tolerate faults. The traditional approach of relaying data in WSNs diminishes the network's lifetime as sensor nodes are involved as a relay and most of their power is consumed by radio transmission. As a resolution of minimizing wastage of power, UAVs have been used for traversing and collecting data directly from the sensor nodes to the sink to prolong network lifetime by reducing sensors power that might be wasted due to data transmission through multi-hop routing. However, during data collection the UAV may lose communications with the faulty or blocked sensors due to link failures related to surrounding environments and radio interference of sensor nodes. To mitigate the problem, we propose a fault tolerable DDNN scheme (FT-DDNN) as an extension of the DDNN algorithm to enable the network to persist in spite of the failed nodes and/or link failures while reducing the data acquisition latency of UAV relay WSNs. The main motivation of our algorithm is driven by the need to provide support for reliability and low latency in

the presence of errors. In other words, the algorithm should allow the UAV to visit each node with the objective of increasing the data detection rate and reducing the data delivery latency.

3 The Proposed Algorithm: The Fault Tolerable DDNN Algorithm (FT-DDNN)

A fault in the communication link can cause the UAV not to receive the required data from the sensors as well as relaying the data to the sink. In the DDNN scheme, the UAV may lose communications with the faulty or blocked sensors at a distance from its turning point. As a solution, the UAV must have an effective route even in the presence of failures. To make our proposed algorithm more descriptive, we outline the main issues involved in the data acquisition process of the scheme.

The FT-DDNN scheme assumes the UAV knows a priori the positions of sensor nodes. For convenience, we will refer to the data acquisition scenarios of the DDNN and the FT-DDNN schemes as shown in Figs. 2 and 3, respectively. We call the UAV is in contact with a node when it is within the radio transmission range r of the node. Since the UAV and the node cannot communicate unless they are in contact, we define discovery as the process which allows the UAV to detect a contact with the node and receive data from it.

Reducing the tour length of the UAV using the DDNN scheme reduces the data acquisition latency [12]. However, it affects the system performance as the UAV may not discover a nearby node at its turning point (on the transmission boundary of the node) during its visit due to link or node faults. Data Acquisition will not be performed unless the UAV detects (recovers) the underlying sensor nodes. The probability of discovering a node by the UAV can be improved by increasing the contact time until the node is detected (recovered from failure) as shown in Fig. 3. The basic idea is that if the UAV doesn't detect the node at its turning point of the DDNN scheme due to some error, it crosses the transmission range boundary directly towards the center of the node (instead of turning its direction to the next node) to get a longer effective contact time thereby allowing the UAV to have the opportunity to recover the error and discover

Fig. 2 Representation for UAV relay WSN data acquisition scenario in the DDNN scheme

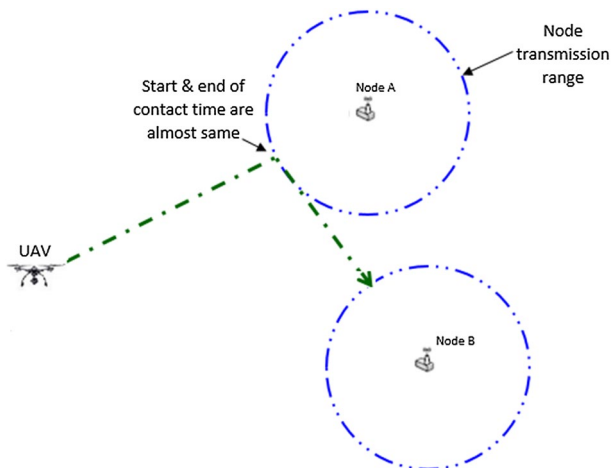
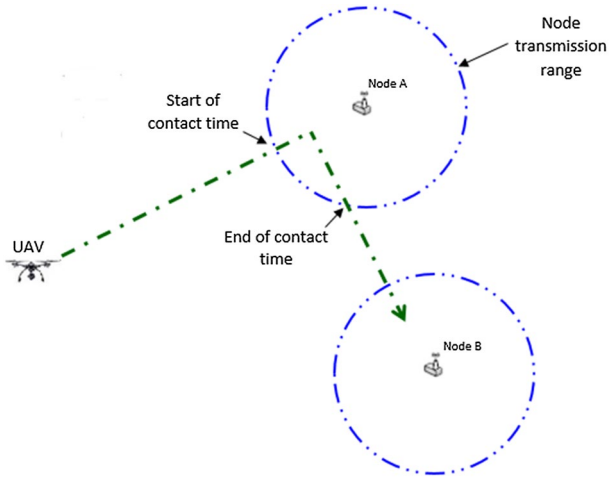


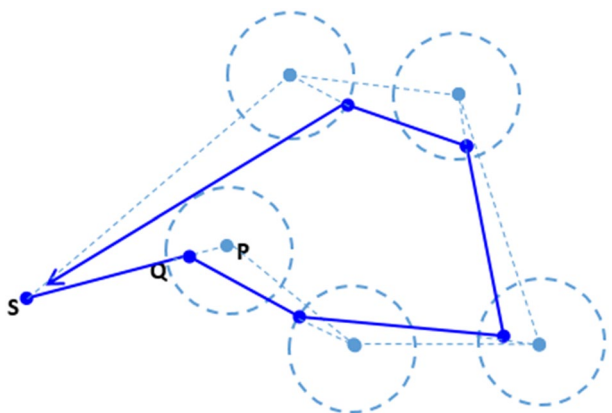
Fig. 3 Representation for UAV relay WSN data acquisition scenario in the FT-DDNN scheme



(successfully gather data) from the node. We assume that transient faults occur in sensor nodes or wireless links between UAV and sensor nodes. Once the data is collected from the current node, the UAV turns its direction on its recovery point and moves towards the next nearest node of the current node. The UAV follows the same procedure until it visits all the nodes in the network. By doing so, data detection rate could be increased at the cost of minimum tour length. Figure 4 shows an instance of UAV route based on the FT-DDNN scheme where the dotted and the solid lines represent the paths in the NN and FT-DDNN algorithms, respectively.

For transient error and recovery times, we use a random number generated by an exponential distribution. Regardless of the fault type, the rate at which faults occur can be quantified by using probability distributions, and more specifically, the exponential probability distributions. For reliability modeling purposes, electronic components are generally assumed to fail according to the exponential distribution [32]. If a random node or link failure occurs, a recovery mechanism has to be performed by the UAV in order to transmit data from the sensor to the sinker with near the minimum tour cost. To achieve this goal and verify by analysis, the FT-DDNN scheme uses a random number

Fig. 4 The FT-DDNN approach



generated by an exponential distribution to let the UAV move directly towards the center of the node until it recovers the error so as to receive the data collected by the node.

The probability density function for the length of time between events with parameter λ in an exponential process is:

$$\Pr(X = x) = f(x) = \lambda e^{-\lambda x}, \quad \text{for } x > 0$$

Continuous failure rate depends on a failure distribution, which is a cumulative distribution function that describes the probability of failure prior to time t . The cumulative distribution function for an exponential process is the integral of the failure density function, $f(x)$.

$$\Pr(X \leq x) = F(t) = \int_0^t f(x) dx, \quad \text{for } x > 0$$

This exponential failure distribution function with parameter λ , which is based on the exponential density function, can be used to model the failure distribution.

$$F(t) = \int_0^t \lambda e^{-\lambda x} dx = 1 - e^{-\lambda t}, \quad \text{for } x > 0$$

The inverse transformation method can be used to generate random variables from this exponential failure distribution function $F(t)$ as follows.

Suppose $y = F(t) = 1 - e^{-\lambda t}$. Then, we can solve for t as follows:

$$\begin{aligned} y &= 1 - e^{-\lambda t} \\ e^{-\lambda t} &= 1 - y \\ -\lambda t &= \log(1 - y) \\ t &= -\frac{1}{\lambda} \log(1 - y) \end{aligned}$$

Since $1 - y$ is also uniformly distributed over the interval $(0, 1)$, it is possible to generate exponential random variables with parameter λ using the transformation as follows:

$$T = -\frac{1}{\lambda} \log(Y)$$

The error recovery times are also assumed exponentially distributed, although the parameters of the distributions for both the error and recovery times are different. Therefore, we use the random numbers generated by an exponential distribution for error and recovery times for our proposed algorithm (FT-DDNN) described in Table 1.

4 Performance Analysis of FT-DDNN

In this section we analyze the performance of the FT-DDNN in terms of the distance covered by the UAV and the detection rate. To make the analysis more descriptive, we outline the main issues involved in the data acquisition process of the FT-DDNN scheme using Fig. 5.

Assuming that the origin of the current node with transmission range of r units lays on the X-axis, we say that the relative position of the next nearest node is at an angle θ of the current node from the X-axis, where θ is an angle formed by the line joining

Table 1 The FT-DDNN algorithm

Input: - Base station(**S**) and nodes location information
 - Nodes transmission range(**r**)
 - Error rate λ
 - Recovery rate μ
 - Miss=0

Output: -Tour length
 -Miss ratio

1. Begin from **S**.
 2. Assign **S** to **B**.
 3. Locate the nearest unvisited node from **B** (Let this node be **P**).
 4. Compute the point of intersection of **PB** and the transmission range boundary of node **P** in the direction of vector $(\vec{B} - \vec{P})$. (Let this point be **Q**).
 5. Generate an exponentially distributed random number with parameter λ that represent the point between **B** and **P**. (Let this point be **E**, the point where error happens)
 6. Generate an exponentially distributed random number with parameter μ that represent the point between **E** and **P**. (Let this point be **R**, the point where recovery happens)
 7. If **R** is between **B** and **Q**, assign **R** to **B** and go to step 5
 8. Else if **R** is between **Q** and **P**,
 - 8.1. Move to **R**
 - 8.2. Mark node **P** as visited
 - 8.3. If unvisited node exists,
 - 8.3.1. Locate the nearest unvisited node from **P** (Let this node be **A**).
 - 8.3.2. Assign **R** to **B**
 - 8.3.3. Assign **A** to **P** and go to step 4
 - 8.4. Else return to **S**
 9. Else
 - 9.1. Move to **P**
 - 9.2. Mark node **P** as visited
 - 9.3. If unvisited node exists,
 - 9.3.1. Locate the nearest unvisited node from **P** (Let this node be **A**).
 - 9.3.2. Assign **P** to **B**
 - 9.3.3. Assign **A** to **P** and do steps 4, 5 and 6
 - 9.3.4. If **R** is outside of the boundary of node **B**, Miss=Miss+1
 - 9.3.5. Go to step 5
 - 9.4. Else return to **S**
 10. Compute the tour length and Miss ratio
-

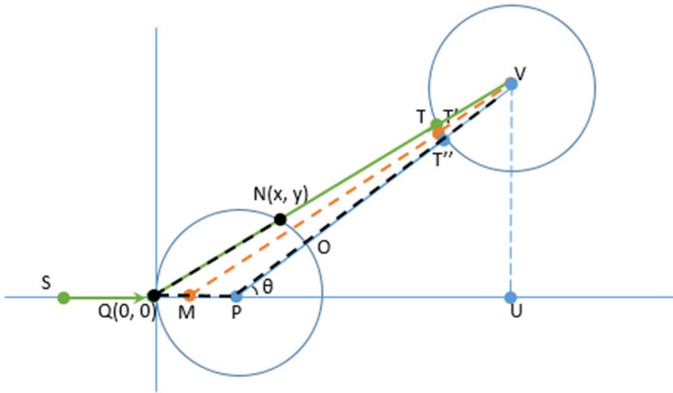


Fig. 5 Performance analysis of FT-DDNN

the origin of the two nodes and the X-axis. If the distance between the current node to be visited and the next nearest node, $d(P, V)$ is g units, then, the distances $d(P, U)$ and $d(U, V)$ will be $g \cos \theta$ and $g \sin \theta$ units respectively.

Suppose the UAV travels directly towards the current node and reaches the boundary of the current node at point Q . There are two cases at this point depending on whether the current node is discovered (recovered) by the UAV or not. The UAV may discover (recover), or may fail to detect the underlying sensor node. The first case happens with a probability of $\frac{\mu}{\lambda+\mu}$ if the UAV discovers (recovers) the underlying node at point Q and receives data from it. If the first case happens, the underlined node is marked as detected and the UAV turns its direction at point Q and moves directly towards the transmission boundary of the next nearest node to point T a distance of $d(Q, T)$ which is given by

$$\sqrt{(r + g \cos \theta)^2 + (g \sin \theta)^2} - r$$

Otherwise, the second case happens with a probability of $\frac{\lambda}{\lambda+\mu}$ if the UAV fails to detect the underlying node at point Q . In this case, the UAV crosses the transmission range boundary directly towards the center of the node (instead of turning its direction to the next node) to get a longer effective contact time thereby allowing the UAV to have the opportunity to recover the error and discover (successfully gather data) from the current node. The second case by itself has three subcases. Each of the three subcases of the second case is described as follows. The first subcase happens with a probability of $\frac{\lambda}{\lambda+\mu}(1 - e^{-\mu r})$ if the UAV discovers (recovers) the underlying node at a certain random point (say at point M) along the path in between the transmission boundary of the current node at point Q and the origin of the current node at point P . If this first subcase happens, the current node is marked as detected and the UAV turns its direction at the point of detection (point M) and moves directly towards the transmission boundary of the next nearest node to point T' . Following this path, the UAV covers a total distance of $d(Q, M) + d(M, T')$ which is given by:

$$E(r) + \sqrt{(r - E(r) + g \cos \theta)^2 + (g \sin \theta)^2} - r$$

where

$$E(r) = \frac{1}{\mu} - \frac{re^{-\mu r}}{1 - e^{-\mu r}}$$

If the UAV fails to detect the current node until it reaches to its origin, either of the second or the third subcases will follow. The second subcase happens with a probability of $\frac{\lambda}{\lambda + \mu}(e^{-\mu r}(1 - e^{-\mu r}))$ if the UAV discovers (recovers) the underlying node at certain random point in between the origin of the current node (point P) and the point of departure of the UAV from the current node (point O). If the second subcase happens, the current node is marked as detected, and the total distance covered by the UAV from Q to P, and then to T'' will be:

$$d(Q, P) + d(P, T'') = r + g - r = g$$

The third subcase happens with a probability of: $\frac{\lambda}{\lambda + \mu}e^{-\mu r}(e^{-\mu r}) = \frac{\lambda}{\lambda + \mu}(e^{-2\mu r})$ if the UAV fails to detect the underlying sensor node along its path until its point of departure (point O). If the third subcase happens, the current node is marked as missed, and the total distance covered by the UAV from point Q to the boundary of the next nearest node T'' across the origin (point P) is the same as subcase 2. The performance of the NN, DDNN and

Table 2 Distance covered by the UAV and detection rate of the schemes

Schemes	Distance covered by the UAV	Detection rate
NN	$r + g - r = g$	$\frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu}(1 - e^{-2\mu r})$
DDNN	$\sqrt{(r + g \cos \theta)^2 + (g \sin \theta)^2} - r$	If $(0^\circ \leq \theta < 90^\circ)$ $\frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu}(1 - e^{-\mu\sqrt{x^2 + y^2}})$ Else if $(90^\circ \leq \theta \leq 180^\circ)$ $\frac{\mu}{\lambda + \mu}$ where $x = \frac{2r}{A + 1}$ $y = \frac{2r}{A + 1} \left(\frac{g \sin \theta}{r + g \cos \theta} \right)$ $A = \left(\frac{g \sin \theta}{r + g \cos \theta} \right)^2$
FT-DDNN	$\frac{\mu}{\lambda + \mu} \left(\sqrt{(r + g \cos \theta)^2 + (g \sin \theta)^2} - r \right)$ $+ \frac{\lambda}{\lambda + \mu}(1 - e^{-\mu r})$ $\left(E(r) + \sqrt{(r - E(r) + g \cos \theta)^2 + (g \sin \theta)^2} - r \right)$ $+ \frac{\lambda}{\lambda + \mu}(e^{-\mu r}(1 - e^{-\mu r}))g$ $+ \frac{\lambda}{\lambda + \mu}(e^{-2\mu r})g$ where, $E(r) = \frac{1}{\mu} - \frac{re^{-\mu r}}{1 - e^{-\mu r}}$	$\frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu}(1 - e^{-2\mu r})$

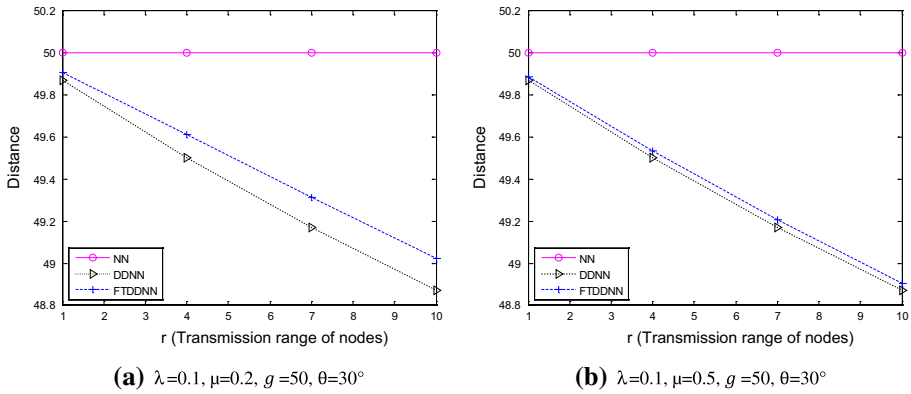


Fig. 6 Distance covered versus transmission range of nodes (r)

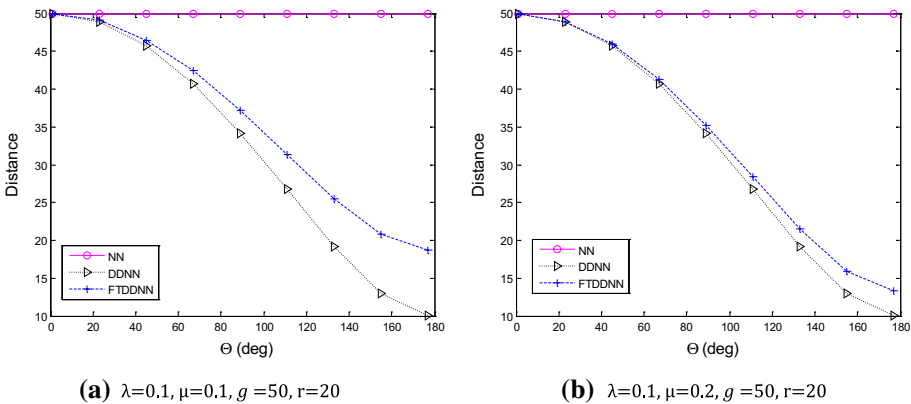


Fig. 7 Distance covered versus θ

FT-DDNN schemes in terms of the distance covered by the UAV and the detection rate are stated in Table 2.

5 Performance Evaluation and Results

In this section, we evaluate the performance of the NN, DDNN and FT-DDNN algorithms in terms of the distance covered by the UAV and the detection rate by varying the different parametric values of r, g, θ, μ and λ .

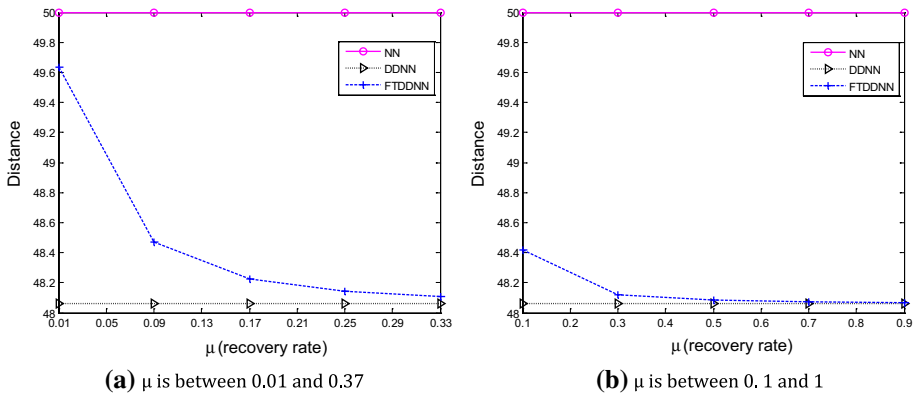


Fig. 8 Distance covered versus recovery rate (μ): (with $\lambda=0.1, g=50, r=20, \theta=30$)

5.1 Comparison of Distance Covered by the UAV Among Each Scheme

Figures 6, 7 and 8 illustrate the performance comparisons in terms the distance covered by the UAV to detect nodes based on the NN, DDNN and FT-DDNN algorithms. Both figures suggest that the proposed scheme shortens the tour length found by the previous NN algorithm, and thus improves the data acquisition latency. The reason for this is that, in the FT-DDNN scheme, once the data is collected from the current node, the UAV turns its direction and moves towards the next nearest node instead of going to the center of the current node. Keeping the error rate $\lambda=0.1$ constant, the impacts of the recovery rate μ on the tour length is also reflected when the value of the recovery rate is incremented from each figure of (a)–(b). It is clearly observed that the tour length based on the FT-DDNN algorithm becomes closer to that of the DDNN algorithm with relatively higher recovery rate. This indicates that the tour length of our proposed algorithm can be improved farther by incrementing the recovery rate.

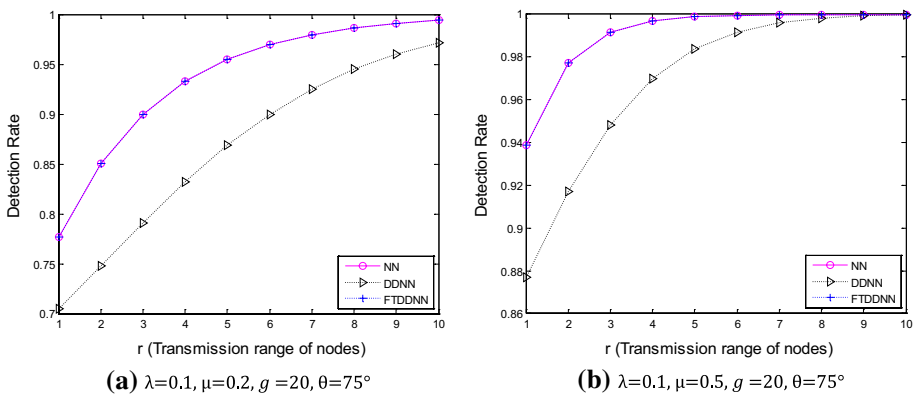


Fig. 9 Detection rate versus transmission range of nodes (r)

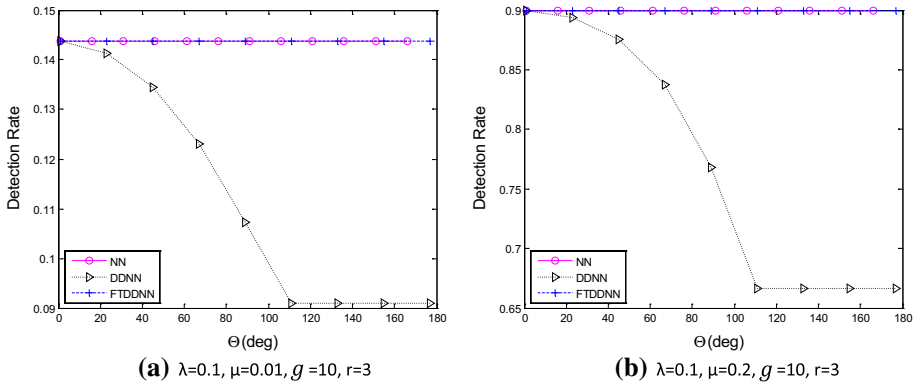


Fig. 10 Detection rate versus θ

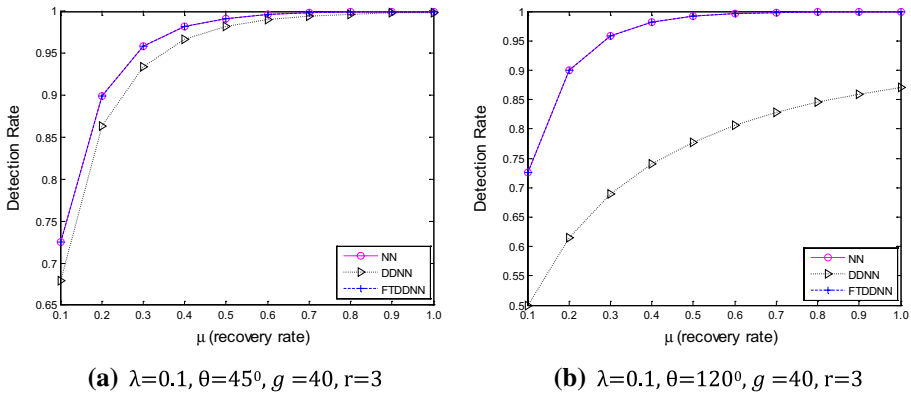


Fig. 11 Detection rate versus recovery rate (μ)

5.2 Comparison of Detection Rate Among Each Scheme

In Figs. 9, 10 and 11, we analyzed the detection rate of the proposed algorithm with respect to the NN and the DDNN algorithms in terms of r , θ and μ . Each figure suggests that the detection rate of the proposed scheme (FT-DDNN) is better than that of the DDNN scheme. In addition to that, the detection rate found by the FD-DDNN and the NN approach do not vary with regard to r , θ and μ as the FD-DDNN allows the UAV to go to the center of the node (just like the NN scheme does) if a node is not detected at the boundary of its transmission range. Therefore, as compared to the DDNN scheme, the FT-DDNN scheme increases the number of detected nodes as much as the NN scheme does and maintains the continuity of the network with minimum tour length costs. The reason for this is that, when there is a node which is unreachable by the UAV at the boundary of its transmission range due to fault device or link error, the UAV crosses the transmission range boundary directly towards the center of the node to have the opportunity to recover the error and successfully gather the data from the node before it gets turned to the next nearest node. Hence, the performance analysis demonstrates that the proposed algorithm enhances the DDNN approach in terms of fault tolerant ability. Furthermore, the effects of the recovery rate μ in

each figure indicates that, increasing the recovery rate (For instance, from $\mu=0.2$ of Fig. 9a to $\mu=0.5$ of Fig. 9b) increases the detection rate in each scheme.

The FT-DDNN algorithm increases the detection rate at the cost of a little bit more route length as compared to the DDNN algorithm. It should be noted that there is a trade-off between fault tolerance and route length in the DDNN and FT-DDNN algorithms.

6 Conclusion

The FT-DDNN algorithm aims to enhance the fault tolerance capability of the network while considering the reduction of data acquisition latency of the UAV. In the DDNN scheme, the UAV may lose communications with the faulty or blocked sensors at a distance from its traveling path. In this study, we proposed efficient FT-DDNN scheme that produces shorter paths which enables faster data delivery with fault tolerant capability. The proposed algorithm enables the UAV to visit each node with the objective of increasing the detection rate and reducing the data delivery latency. Compared with the DDNN algorithm, the proposed approach increases detection rate at the cost of a little bit more tour length. It also outperforms the NN algorithm in terms of reducing the data delivery latency. The reason for this is that, in the FT-DDNN scheme, once the data is collected from the current node, the UAV turns its direction and moves towards the next nearest node instead of going to the center of the current node. The performance analyses have demonstrated that our proposed scheme improves the detection rate of the DDNN scheme up to 34.93% at the cost of a little bit distance.

Acknowledgements This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. NRF-2018R1D1A1B07040573).

References

1. Lin, C., Xiong, N., Park, J. H., & Kim, T. (2009). Dynamic power management in new architecture of wireless sensor networks. *International Journal of Communication Systems*, 22, 671–693.
2. Wu, X., Chen, G., & Das, S. K. (2008). Avoiding energy holes in wireless sensor networks with non-uniform node distribution. *IEEE Transactions on Parallel and Distributed Systems*, 19, 710–720.
3. Lai, C., Hwang, D. D., Kim, S. P., & Verbauwhede, I. (2004). Reducing radio energy consumption of key management protocols for wireless sensor networks. In *Proceedings of the ACM/IEEE International symposium on low power electronics and design (ISLPED)* (pp. 351–356).
4. Cobano, J. A., Martínez-de Dios, J. R., Conde, R., Sánchez-Matamoros, J. M., & Ollero, A. (2010). Data retrieving from heterogeneous wireless sensor network nodes using UAVs. *Journal of Intelligent and Robotic Systems*, 60, 133–151.
5. de Freitas, E. P., Heimfarth, T., et al. (2010). UAV relay network to support WSN connectivity. In *Proceedings of IEEE international congress on ultramodern telecommunications and control systems (ICUMT'10)* (pp. 309–314).
6. Sugihara, R., & Gupta, R. K. (2009). Optimizing energy-latency trade-off in sensor networks with controlled mobility. In *IEEE INFOCOM* (pp. 2566–2570).
7. Ho, D. T., Grötl, E. I., Shimamoto, S., & Johansen, T. A. (2012). Optimal relay path selection and cooperative communication protocol for a swarm of UAVs. In *Proceedings of the 3rd international workshop on wireless networking & control for unmanned autonomous vehicles: Architectures, protocols and applications* (pp. 1585–1590).

8. Sujit, P. B., Lucani, D. E., & Sousa, J. B. (2012). Bridging cooperative sensing and route planning of autonomous vehicles. *IEEE Journal on Selected Areas in Communications*, *30*, 912–922.
9. Flushing, E. F., Gambardella, L., & Di Caro, G. (2012). Search and rescue using mixed swarms of heterogeneous agents: Modeling, simulation, and planning. *IDSIA/USI-SUPSI*, Technical Report.
10. Frew, E. W., & Brown, T. X. (2008). Networking issues for small unmanned aircraft systems. *Journal of Intelligent and Robotic Systems*, *54*, 21–37.
11. Ruz, J. J., Arevalo, O., Pajares, G., & De la Cruz, J. M. (2009). UAV trajectory planning for static and dynamic environments. *InTech*, *27*, 581–600.
12. Alemayehu, T. S., & Kim, J. H. (2017). Efficient nearest neighbor heuristic TSP algorithms for reducing data acquisition latency of UAV relay WSN. *Wireless Personal Communications*, *95*, 3271–3285.
13. Tekdas, O., Isler, V., Lim, J. H., & Terzis, A. (2009). Using mobile robots to harvest data from sensor fields. *IEEE Wireless Communications in Robotic Networks*, *16*, 22–28.
14. Yuan, B., Orlowska, M., & Sadiq, S. (2007). On the optimal robot routing problem in wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, *19*, 1252–1261.
15. Papadimitriou, C. H. (1978). Euclidean traveling salesman problem is NP complete. In *Theoretical computer science* (pp. 237–244).
16. Lawler, E. L., Lenstra, J. K., Kan, A. H. G., & Shmoys, D. B. (1985). *The traveling salesman problem: A guided tour of combinatorial optimization*. New York: Wiley.
17. Wang, J., Cao, Y., Li, B., Kim, H.-J., & Lee, S. (2017). Particle swarm optimization based clustering algorithm with mobile sink for WSNs. *Future Generation Computer Systems*, *76*, 452–457.
18. Yue, Y.-G., & He, P. (2018). A comprehensive survey on the reliability of mobile wireless sensor networks: Taxonomy, challenges, and future directions. *Information Fusion*, *44*, 188–204.
19. Szewczyk, R., Polastre, J., Mainwaring, A., & Culler, D. (2004). Lessons from a sensor network expedition. In *Proceedings of the first European workshop on sensor networks (EWSN)* (pp. 307–322).
20. Liu, H., Nayak, A., & Stojmenovic, I. (2009). Fault tolerant algorithms/protocols in wireless sensor networks. In *Guide to wireless sensor networks* (pp. 261–291). London: Springer.
21. Thelen, J., Goense, D., & Langendoen, K. (2005). Radio wave propagation in potato fields. In *First workshop on wireless network measurement* (2005).
22. Afsar, M. (2015). A comprehensive fault-tolerant framework for wireless sensor networks. *Security and Communication Networks*, *8*(17), 3247–3261.
23. Salayma, M., Al-Dubai, A. Y., Romdhani, I., & Nasser, Y. (2017). Wireless body area network (WBAN): A survey on reliability, fault tolerance, and technologies coexistence. *ACM Computing Surveys*, *50*, 1–38.
24. Lussier, B., Chatila, R., Ingrand, F., Killijian, M.-O., & Powell, D. (2004). On fault tolerance and robustness in autonomous systems. In *3rd IARP-IEEE/RAS-EURON joint workshop on technical challenges for dependable robots in human environments*.
25. Intanagonwivat, C., Govindan, R., Estrin, D., Heidemann, J., & Silva, F. (2002). Directed diffusion for wireless sensor networking. *ACM/IEEE Transactions on Networking*, *11*, 2–16.
26. Karlof, C., Li, Y., & Polastre, J. (2003). *ARRIVE: Algorithm for robust routing in volatile environments*. Technical Report UCBC/CS-03-1233, University of California, Berkeley, CA.
27. Costa, D. G., Vasques, F., & Portugal, P. (2017). Enhancing the availability of wireless visual sensor networks: Selecting redundant nodes in networks with occlusion. *Applied Mathematical Modelling*, *42*(1), 223–243.
28. Jafarali Jassbi, S., & Moridi, E. (2019). Fault tolerance and energy efficient clustering algorithm in wireless sensor networks: FTEC. *Wireless Personal Communications*, *107*(1), 373–391.
29. Tien, N. X., Kim, S., Rhee, J. M., & Park, S. Y. (2017). A novel dual separate paths (DSP) algorithm providing fault-tolerant communication for wireless sensor networks. *Sensors*, *17*, 1699.
30. Cheraghlou, M. N., Khadem-Zadeh, A., & Haghparast, M. (2017). Increasing lifetime and fault tolerance capability in wireless sensor networks by providing a novel management framework. *Wireless Personal Communications*, *92*, 603–622.
31. Moridi, E., Haghparast, M., Hosseinzadeh, M., & Jassbi, S. J. (2020). Novel fault-tolerant clustering-based multipath algorithm (FTCM) for wireless sensor networks. *Telecommunication Systems*. <https://doi.org/10.1007/s11235-020-00663-z>.
32. McGough, J., Smotherman, M., & Trivedi, K. S. (1985). The conservativeness of reliability estimates based on instantaneous coverage. *IEEE Transactions on Computers*, *C-34*(7), 602–608.

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



Temesgen Seyoum Alemayehu received his M.Sc. degree in Computer Science from Addis Ababa University, Ethiopia in 2008, and his Ph.D. degree in Computer Science and Engineering from Ajou University, South Korea in 2018. He is currently a research assistant professor in the Ubiquitous Convergence Research Institute, Ajou University, South Korea. My research interests include cyber physical systems, dependable systems, mobile computing, multiuser MIMO Systems, energy efficient communications and artificial intelligence.



Jai-Hoon Kim received the B.S. degree in Control and Instrumentation Engineering, Seoul National University, Seoul, South Korea, in 1984, M.S. degree in Computer Science, Indiana University, USA, in 1993, and his Ph.D. degree in Computer Science, Texas A&M University, USA, in 1997. He is currently a professor of Department of Cyber Security at Ajou University, South Korea. His research interests include distributed systems, cyber-physical systems, and mobile computing.



Wonsik Yoon graduated with his B.S. degree in Control and Instrumentation Engineering from Seoul National University in 1984. He received his M.S. and Ph.D. degrees from KAIST in 1986 and 1991, respectively. From 1986 to 1994, he worked with Goldstar Electrical Company and LG Innotek, South Korea. From 2000 to 2001, he worked as the CTO at Contela Inc., South Korea. Since 1994, he has been with the Department of Electrical Engineering at Ajou University, South Korea, where he is a Professor. His research interests include wireless communications and networks, massive MIMO, and network coding. He is a senior member of IEEE.