Defence Science Journal, Vol. 64, No. 6, November 2014, pp. 557-563, DOI : 10.14429/dsj.64.5295

Intelligent UAV-Assisted Localisation to Conserve Battery Energy in Military Sensor Networks

P. Shunmuga Perumal*, V. Rhymend Uthariaraj, and V.R. Elgin Christo

Ramanujan Computing Centre, Anna University, Chennai, India
**E-mail: shunmugaperumal@auist.net,*

ABSTRACT

Wireless sensor networks (WSNs) are extensively used in military applications for border area monitoring, battle-field surveillance, tracking enemy troops, where the sensor nodes run on battery power. Localisation of sensor nodes is extremely important to identify the location of event in military applications for further actions. Existing localisation algorithms consume more energy by heavy computation and communication overheads. The objective of the proposed research is to increase the lifetime of the military sensor networks by reducing the power consumption in each sensor node during localisation. For the state-of-the-art, we propose a novel intelligent unmanned aerial vehicle anchor node (IUAN) with an intelligent arc selection (IAS)-based centralised localisation algorithm, which removes computation cost and reduces communication cost at every sensor node. The IUAN collects the signal strength, distance data from sensor nodes and the central control station (CCS) computes the position of sensor nodes using IAS algorithm. Our approach significantly removes computation cost and reduces communication cost at each sensor node during localisation, thereby radically extends the lifetime and localisation coverage of the military sensor networks.

Keywords: Military wireless sensor networks, localisation, localisation-computation cost, localisation-communication cost, localisation-coverage, WSN

1. INTRODUCTION

Wireless sensor networks (WSNs) are useful in different areas such as natural disaster management systems, forest fire detection, ocean navigation, industrial automation and control, etc. WSN provides many services to military and Air Force like border area monitoring, information collection, battlefield surveillance, detection of intrusions and attacks. Akyildiz[1], *et al.* explained the applications of sensor networks and factors influencing the WSN design. Yick[2], *et al.* explained the types of WSNs, their operating systems and platforms, standards, data storage methods, power management, task distribution, data aggregation, security mechanisms. WSN is a collection of sensor nodes and energy lack of a single sensor node causes severe impacts. Sensor node's battery energy is consumed in active mode, idle mode and sleep modes. Anastasi[3], *et al.* explained mobility-based energy conservation schemes using mobile sink-based and mobile relay-based approaches. Aguiar[4], *et al.* proposed an integer programming model for optimizing energy consumption.

The process of estimating the physical location of sensor nodes is localisation and it is important as the sensed data becomes meaningless without the location of the event. The detailed classification of the localisation algorithms is explained by Han[5], *et al.* Pal[6] explained the centralised and distributed localisation schemes. Existing localisation algorithms consume more energy by heavy computation and communication overheads, thereby deplete the sensor node's battery quickly. Ou and Ssu[7] proposed a localisation scheme with high communication and computation overhead. Ou[8] proposed a localisation scheme with beacon scheduling where each sensor node executes heavy computations to determine their locations. Mobile anchor node-based localisation proposed by Liao[9], *et al.* insists communication between sensor nodes during localisation, which causes delay, data loss and high energy consumption. High beacon overhead localisation scheme is given by Ssu[10], *et al.* Sheu[11], *et al.* proposed a localisation scheme which involves inter-sensor communications. Xiao[12], *et al.* proposed a localisation scheme which involves anchor classification phase, distance estimation phase and location estimation phase. Satyamurti[13], *et al.* proposed artificial neural network assisted node localisation to estimate the location errors using artificial neural models. Pescaru[14] and Curiac proposed anchor node-based localisation for WSN using video and compass information fusion. Han[15], *et al.* proposed a mobile anchor-assisted localisation algorithm. High communication and computational overhead reduces the overall lifetime of WSN. Any approach in WSN must conserve the battery energy of sensor nodes. The authers have proposed an IUAN approach with IAS algorithm to conserve energy in each sensor node to extend the life time of military sensor networks.

2. DESIGN OF INTELLIGENT UAV ANCHOR NODE

The UAV can fly and broadcast beacons on their own energy. Guerrero[16], *et al.* proposed an approach that uses

Received 15 October 2013, revised 29 August 2014, online published 13 November 2014

UAVs to carry positioning device and transmitter to localise the sensor nodes. Yadav[17], *et al.* explained the localisation using global positioning system (GPS)-enabled flying anchor nodes. Vincent[18], *et al.* employ the UAVs to distribute the energy burden across the WSN. Our main idea was to collect the location and distance data from the sensor nodes and compute their locations at (CCS). In our approach, UAV was empowered with intelligence and utilised as flying anchor node. This is referred as IUAN and its novel design is shown in Fig.1.

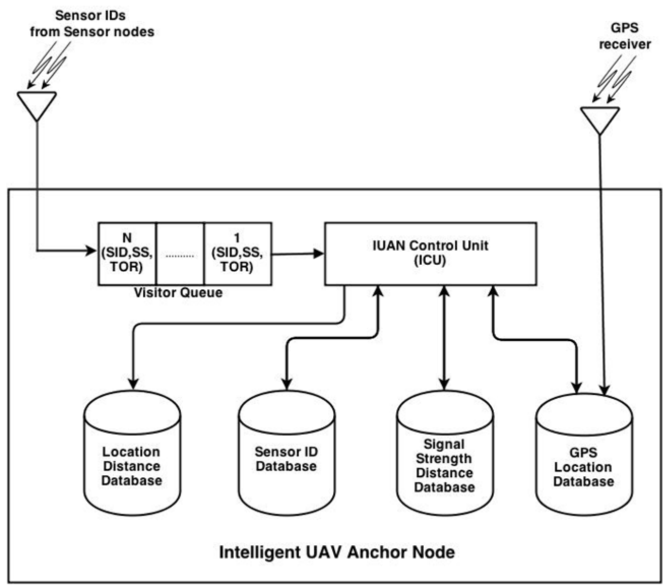


Figure 1. Architecture of the proposed IUAN.

The IUAN contains IUAN control unit (ICU), sensor ID database (DB_{SID}), visitor queue (VQ), signal strength-distance database (DB_{SSD}), IUAN location database (DB_{ILOC}) and location-distance database (DB_{LD}). The ICU controls the overall operation of the IUAN like receiving sensor identity (SID) from sensor nodes, activating GPS receivers, communication with CCS, guiding the IUAN in an optimistic trajectory. When an IUAN enters into the transmission range of a sensor node, the sensor node transmits its SID to that IUAN and the ICU executes the sequence of processes to construct the location-distance message (LDM). IUAN may receive the SIDs from multiple sensor nodes and the VQ stores the SIDs along with their signal strength (SSs) and corresponding time of reception SID_{TOR}. SID_{TOR} is the time at which the IUAN receives the SID. Whenever the IUAN receives the SID (at TOR_{SID}), it stores the SID, its own location (LOC_{IUAN}) along with TOR_{IUAN} to the DB_{ILOC} where TOR_{IUAN} = TOR_{SID}. The Table 1 gives the structure of the DB_{ILOC}.

Table 1. Structure of the IUAN location database (DB_{ILOC})

Time of IUAN location reception TOR_{IUAN} (where TOR_{IUAN}=TOR_{SID})	SID	Location of IUAN (LOC_{IUAN})
T_1	SID_1	(x_1, y_1)
T_2	SID_1	(x_2, y_2)
T_3	SID_1	(x_3, y_3)
T_n	SIN_n	(x_n, y_n)

The IUAN authenticates the SIDs in the VQ using the DB_{SID} as DB_{SID} contains valid sensor id numbers. The invalid SIDs are ignored and LD messages are constructed for only the valid sensor nodes. The DB_{SSD} is used to withstand with radio irregularity. The signal strength at the locations around the transmission range of a sensor node varies with the distance and environmental effects. The path loss effect causes the radio signal to attenuate variously in different directions. The proposed work assumes the radio irregularity model (RIM) explained by Zhou[19], *et al*. The Table 2 shows the structure of DB_{SSD} with sample values of the Signal Strengths (SSs) at 3.048 m away from the mica2 mote in four directions. The SS is calculated based on signal strength-distance relation using received signal strength indication (RSSI) measurements with the knowledge of degree of irregularity (DOI).

Table 2. RSSI value and distance of signal strength-distance database (DB_{SSD})

Signal strength (SS) (dBm)	Distance (D) (m)	Direction
-63	3.048	East
-59	3.048	North
-58	3.048	West
-57	3.048	South

For a valid SID, an IUAN fetches the distance (D) between itself and the sensor node using the corresponding SS_{SID}. In our approach, the computation cost for the signal strength and distance measurements were shifted from the sensor nodes to the powerful IUAN to conserve the energy at each sensor node. During the validation of SID and distance-fetching process, the IUAN moves to different locations. Hence after fetching the distance, the IUAN fetches the corresponding location information (LOC_{IUAN}) at which it received the SID, using the TOR_{IUAN} and TOR_{SID}. This mechanism ensures the collection of valid location and distance information. The IUAN constructs the LDM and stores it in the DB_{LD} database, where each LDM contains the location (L), distance (D) and the corresponding SID.

The Table 3 shows the list of symbols used in the LDM construction algorithm (Algorithm 1) and corresponding flow diagram is shown in Fig. 2. After constructing the LDM, the ICU removes the corresponding entries from the VQ and DB_{ILOC} to manage the storage efficiency. The IUANs constructs the LDMs and reaches the CCS, where CCS aggregates DB_{LD} database of all IUANs and selects the sufficient LDMs to calculate the locations of the sensor nodes. If an IUAN (say IUAN-100) is shot down by enemies, it immediately transmits its crashed code to the nearby sensor node (say SN-51). When another IUAN (say IUAN-200) reaches SN-51 later, it transmits the crashed code of IUAN-100 to IUAN-200. When IUAN-200 reaches CCS, it identifies that the IUAN-100 is crashed by the enemies.

Algorithm 1: LDM Construction

Input: SID from Sensor Node
Output: LD Message (LDM)
1: **Begin**

Table 3. List of symbols used in LD message construction algorithm

Symbol	Definition
IUAN	Intelligent UAV anchor node
Δ(f)	Flight period of IUAN
SID	Sensor ID sent by the sensor node
TOR_{SID}	Time of SID reception
TOR_{IUAN}	Time at which the IUAN location is received from GPS
VQ	Visitor queue
DB_{SID}	Sensor ID database
DB_{SSD}	Signal strength-distance database
DB_{ILOC}	IUAN location database
DB_{LD}	Location-distance database
SS_{SID}	Signal strength of SID
D	Distance between sensor node and IUAN at TOR_{SID}
LOC_{IUAN}	Location of the IUAN when ir receives the SID
LDM	Location-distance message
ConstructLDMPacket (LOC_{IUAN}, D, SID)	Procedure to construct LDM with 'D', 'LOC_{IUAN}' and 'SID'

Interrupt Process 1: Whenever SIDi is received from sensor node
measure SS_{SIDi}
VQ ← enqueue (SIDi, SS_{SIDi}, TOR_{SIDi})
DB_{GLOC} ← insert(LOC_{IUAN}, SIDi, TOR_{IUAN}=TOR_{SIDi})
end Interrupt Process 1
while(VQ != NULL)
if SIDi ∉ DB_{SID} then
VQ ←dequeue (SIDi, SS_{SIDi}, TOR_{SIDi})
else
D←select D from DB_{SSD} where SS=SS_{SIDi}
L←select LOC_{IUAN} from DB_{ILOC} where TOR_{IUAN} = TOR_{SIDi}
LDM.x ←LOC_{IUAN}.x
LDM.y←LOC_{IUAN}.y
LDM.d←D
LDM.id←SIDi
LDM←ConstructLDMPacket (LDM.x, LDM.y, LDM.d, LDM.id)
insert LDM into DB_{LD}
VQ←dequeue (SIDi, SS_{SIDi}, TOR_{SIDi})
delete LOC_{IUAN}, SIDi, TOR_{IUAN} from DB_{ILOC} where TOR_{IUAN}=TOR_{SIDi} and SID = SID_i
end if
end while
End

3. IAS BASED CENTRALISED LOCALISATION ALGORITHM

For the state-of-the-art, the authers have proposed a novel approach to remove the computation cost at each sensor node during localisation process. Our approach computes the location of sensor nodes at CCS where the computing and communication resources are not constraint. The proposed algorithm aggregates the LDMs from different IUANs and computes the location of a sensor node with three relevant LDMs. Each LDM has the LOC_{IUAN} and distance between the LOC_{IUAN} and corresponding sensor node location (LOC_S). The proposed algorithm is an optimum version of the trilateration method. Our algorithm selects the boundary points and arc angles to construct the intelligent arc segments, where the intersection of these arcs gives the location of the sensor node. The algorithm starts with three LDMs, where these messages are collected within the range of a sensor node, as shown in Fig. 3. The proposed algorithm has 6 phases.

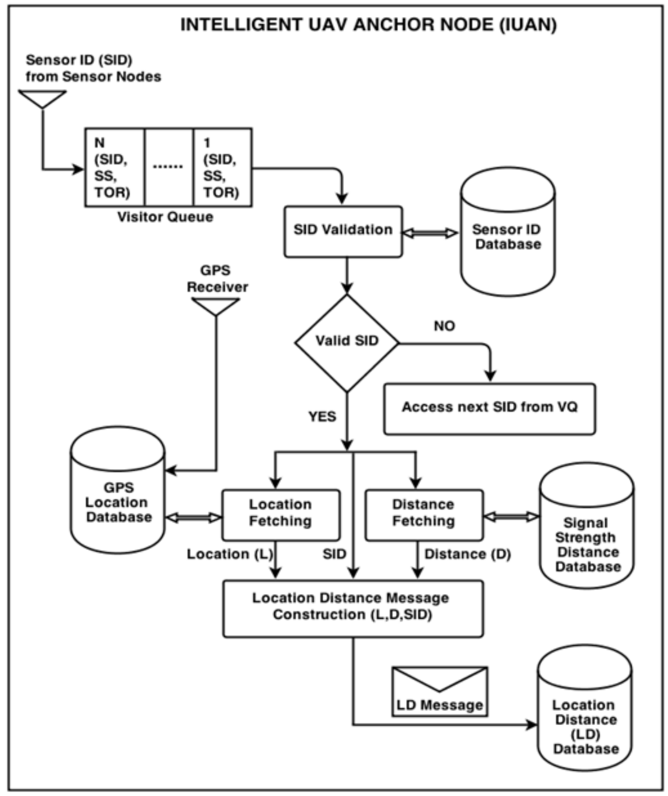


Figure 2. Flow diagram of the LD message construction algorithm.

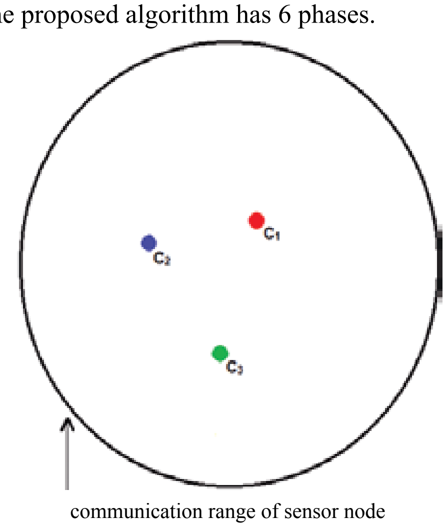


Figure 3. Three LD messages collected within the range of a sensor node.

In phase 1, the LD messages are processed to extract the $IUAN_{Centres}$ $\{C_1,C_2,C_3\}$ and the corresponding $IUAN_{Distances}$ $\{r_1,r_2,r_3\}$, where r_i is the distance between $IUAN_{centre}$ C_1 and the sensor node. In phase 2, the boundary points (BPoints) around each $IUAN_{Centre}$ are calculated using the 2D translation, as shown in Fig. 4. All Bpoints and $IUAN_{Centres}$ contain the associated '*x*' and '*y*' coordinate values and based on these both the $IUAN_{Centres}$ and BPoints are classified as the '*x*' axis centre points (XCPoints), '*x*' axis boundary points (XBPoints), '*y*' axis centre points (YCPoints), '*y*' axis boundary points (YBPoints). Each $IUAN_{Centre}$ has four associated bounday points. Eg: $\{P_1,P_2,P_3,P_4\}$ is the set of boundary points around the IUAN centre (C_1). A complete circle around this centre can be constructed using these 4 boundary points and corresponding distance/radius r_1. Similarly, three circles can be constructed around three centres. The intersection of these three circles gives the location of sensor node. Our approach optimises the trilateration method by constructing the intelligent arcs using intelligent boundary points instead of considering the complete circles, which reduces the computation cost significantly.

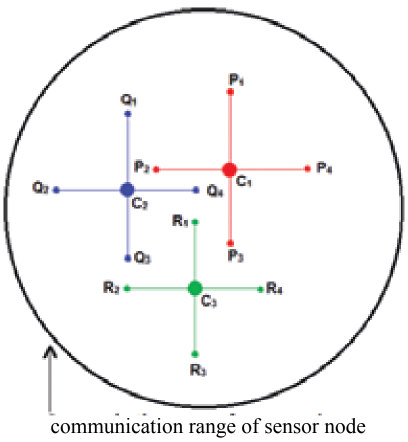


Figure 4. Computation of boundary points.

Phase 3 selects the intelligent boundary points on 'x' axis and '*y*' axis, which contribute to the construction of the intelligent arcs and these are referred as IXB Points, and IYB Points ,respectively. Among 6 boundary points $\{P_2,P_4,Q_2,Q_4,R_2,R_4\}$ on '*x*' axis, only 3 points contribute to the intelligent arcs. Similarly among 6 boundary points $\{P_1,P_3,Q_1,Q_3,R_1,R_3\}$ on '*y*' axis, only three points contribute to the intelligent arcs. The XCPoints and XBPoints are grouped (Points1) and sorted (Points 2). The intelligent points IXBPoints are selected from (Points 2), where IXBPoints lies between the range of first and last XCPoints in Points 2. This selection decision is taken based on the fact that, the boundary points lie out of this range are not contributing to the intelligent arcs. Similarly IYBPoints are selected. The Fig. 5 shows the intelligent boundary points IXBPoints and IYBPoints selected in phase 3.

Definition of an arc: An arc can be constructed with a centre point *C*(*x,y*) and radius (*r*). The points on the arc are defined as the set of points *P*(*h,k*) using the parametric Eqns (1) and (2), $\forall\ \Theta$, $\Theta_1 \le \Theta \le \Theta_2$ where $0 \le \Theta_1 \le 2\pi$ and $\Theta_1 < \Theta_2 < (\Theta_1+2\pi)$. The point $P_1(x_1,y_1)$ with $\Theta = \Theta_1$ is called the start point of the arc, and the point $P_2(x_2,y_2)$ with $\Theta = \Theta_2$ is called the end point of the arc.

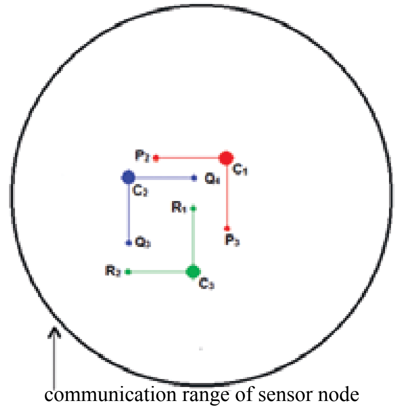


Figure 5. Selection of intelligent boundary points.

$$h = x + r\cos(\Theta) \quad (1)$$

$$k = y + r\sin(\Theta) \quad (2)$$

After selecting the intelligent boundary points, phase 4 identifies the starting and ending angles of each intelligent arc. The sensor node is positioned, where the three intelligent arcs intersect. So any 2 arbitrary arcs can be constructed, as the intersection of any two intelligent arcs gives the sensor node location. Phase 5 constructs any two arbitrary intelligent arcs using the parametric Eqns (1) and (2). Phase 6 compares the points on the arbitrarily selected intelligent arcs and identifies the intersection point, which is the location of the sensor node $S(LOC_S.x, LOC_S.y)$ as shown in the Figs. 6(a) and 6(b). The Algorithm 2 explains the mathematical logic behind the proposed IAS algorithm.

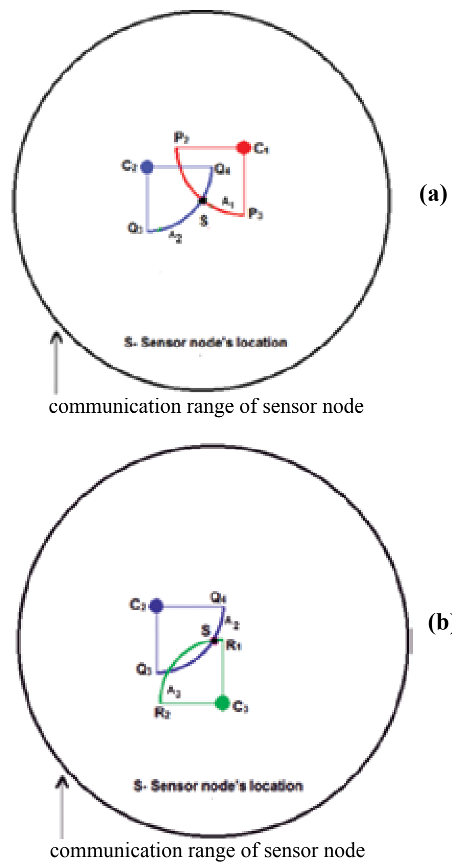


Figure 6. Intersection of arcs:(a) A1 and A2, and (b) A2 and A3.

Algorithm 2: Intelligent Arc Selection-based Centralised Localisation Algotithm

Input: LDM_1, LDM_2, LDM_3, where LDM_1.id= LDM_2.id= LDM_3.id
Output: Sensor Node's Location S (LOC_S.x, LOC_S.y)
Begin
($IUAN_{Centres}$, $IUAN_{Distances}$) ←ExtractLDMessage (LDM_1, LDM_2, LDM_3)
(XCPoints, XBPoints, YCPoints, YBPoints) ←BPoints ($IUAN_{Centres}$,$IUAN_{Distances}$)
IXBPoints←IXBPoints (XCPoints, XBPoints)
IYBPoints←IYBPoints (YCPoints, YBPoints)
St_End_Angles←Start_End_ArcAngles (IXBPoints, IYBPoints)
(A_1, A_2) ←Arc_Segments (XCPoints, YCPoints, St_End_Angles, $IUAN_{Distances}$)
(LOC_S.x, LOC_S.y) ←Sensor_Location (A_1, A_2)
End

Phase 1: Extracting Location Distance (LD) Messages

Procedure ExtractLDMessage
C(1).x←LDM_1.x
C(1).y←LDM_1.y
C(2).x←LDM_2.x
C(2).y←LDM_2.y
C(3).x←LDM_3.x
C(3).y←LDM_3.y
r_1←LDM_1.d
r_2←LDM_2.d
r_3←LDM_3.d
$IUAN_{Centres}$ ←{C(1).x , C(1).y, C(2).x , C(2).y, C(3).x, C(3).y}
$IUAN_{Distances}$ ←{r_1, r_2, r_3}
return ($IUAN_{Centres,}$ $IUAN_{Distances}$)
End

Phase 2: Computation of Boundary Points

Procudre BPoints($IUAN_{Centres,}$ $IUAN_{Distances}$)
P(1).x←C(1).x
P(1).y←C(1).y+r_1
P(2).x←C(1).x-r_1
P(2).y←C(1).y
P(3).x←C(1).x
P(3).y←C(1).y-r_1
P(4).x←C(1).x+r_1
P(4).y←C(1).y
Q(1).x←C(2).x
Q(1).y←C(2).y+r_2
Q(2).x←C(2).x-r_2
Q(2).y←C(2).y
Q(3).x←C(2).x
Q(3).y←C(2).y-r_2
Q(4).x←C(2).x+r_2
Q(4).y←C(2).y
R(1).x←C(3).x
R(1).y←C(3).y+r_3
R(2).x←C(3).x-r_3
R(2).y←C(3).y
R(3).x←C(3).x
R(3).y←C(3).y-r_3
R(4).x←C(3).x+r_3
R(4).y←C(3).y
XCPoints←{C(1).x , C(2).x , C(3).x}
YCPoints←{C(1).y , C(2).y , C(3).y}
XBPoints←{P(2).x , P(4).x , Q(2).x, Q(4).x , R(2).x, R(4).x}
YBPoints←{P(1).y , P(3).y , Q(1).y, Q(3).y , R(1).y, R(3).y}
return (XCPoints, XBPoints, YCPoints, YBPoints)
End

Phase 3: Selecting Intelligent Boundary Points on 'X' and 'Y' axis

Procedure IXBPoints (XCPoints, XBPoints)
Points1←{C(1).x , C(2).x , C(3).x, P(2).x , P(4).x , Q(2).x, Q(4).x , R(2).x, R(4).x}
Points2←Sort (Points1), where C(i).x < C(j).x < C(k).x; ∀ i,j,k = 1,2,3; $i \neq j \neq k$
IXBPoints ⊂ Points2, where C(i).x < {IXBPoints} < C(k).x
IXBPoints←{P(m).x , Q(n).x , R(o).x}, wher m,n,o are either 2 or 4
return (IXBPoints)
End
Procedure IYBPoints (YCPoints, YBPoints)
Points1←{C(1).y , C(2).y , C(3).y, P(1).y , P(3).y , Q(1).y, Q(3).y , R(1).y, R(3).y}
Points2←Sort (Points1), where C(i).y < C(j).y < C(k).y; ∀ i,j,k = 1,2,3; $i \neq j \neq k$
IYBPoints ⊂ Points2, where C(i).y < {IYBPoints} < C(k).y
IYBPoints←{P(m).y , Q(n).y , R(o).y}, wher m,n,o are either 1 or 3
return (IYBPoints)
End

Phase 4:

Procedure Start_End_ArcAngles (IXBPoints, IYBPoints)
if P(m).y = P(1).y then Θ_1 ←π/2
else Θ_1 ←3π/2
end if
if P(m).x = P(4).x then Θ_2 ←0π
else Θ_2 ←π
end if
if Q(m).y = Q(1).y then Θ_3 ←π/2
else Θ_3 ←3π/2
end if
if Q(m).x = Q(4).x then Θ_4 ←0π
else Θ_4 ←π
end if
return (Θ_1, Θ_2, Θ_3,Θ_4)
End

Phase 5:

Procedure Arc_Segments (XCPoints, YCPoints, St_End_Angles, $IUAN_{Distances}$)
A_1(i).x ←C(1).x+r_1.cos (Θ), where $\Theta_1 \leq \Theta_i \leq \Theta_2$ ∀ i=1,2,…n
A_1(i).y ←C(1).y+r_1.sin (Θ), where $\Theta_1 \leq \Theta_i \leq \Theta_2$ ∀ i=1,2,…n
A_2(j).x ←C(2).x+r_2.cos (Θ), where $\Theta_3 \leq \Theta_j \leq \Theta_4$ ∀ j=1,2,…n
A_2(j).y ←C(2).y+r_2.sin (Θ), where $\Theta_3 \leq \Theta_j \leq \Theta_4$ ∀ j=1,2,…n
A_1 ←{(A_1(i).x, A_1(i).y) ∀ i=1,2,…n }: set of points on arc A_1
A_2 ←{(A_2(j).x, A_2(j).y) ∀ j=1,2,…n }: set of points on arc A_2
return (A_1, A_2)
End

Phase 6:

Procedure Sensor_Location (A_1, A_2)
Compare {(A_1(1).x, A_1(1).y), ... (A_1(n).x, A_1(n).y)} and {{(A_2(1).x, A_2(1).y), ... (A_2(n).x, A_2(n).y)}
if ((A_1(i).x, A_1(i).y) = (A_2(j).x, A_2(j).y)) then
LOC_s.x←A_1(i).x
LOC_s.y←A_1(i).y
end if
return S(LOC_S.x, LOC_S.y)
End

The central control station stores the location of sensor nodes in the sensor position database (DB_{SP}) along with the SID. This database is used to identify the location of event as each sensor node sends the sensed information along with its identification.

4. RESULTS AND DISCUSSIONS

The authers analysis assumes the stationary sensor nodes in two dimensions, where the latitude and longitude of the sensor nodes are considered. The power consumed by the IUAN is not taken into account for this analysis. The RIM model is assumed to withstand the radio irregularity. The performance of the proposed IUAN-assisted intelligent arc selection-based localisation algorithm is compared with the existing schemes in terms of its communication and computation cost to prove its efficiency. For numerical analysis of the communication cost, the first-order radio model explained by Heinzelman[20], *et al.* is considered. The proposed approach calculates the sensor node position by three LD messages. The analysis assumes 60nJ energy consumption for transmission/reception of one byte of data. Each message is assumed to be of 34 byte size. From the survey, it was observed that the localisation schemes Ou[7] , *et al.*, Sheu[11], *et al.* cause each sensor node to use 14 and 16 messages, respectively to calculate their position. Figure 7 shows the comparison of localisation-communication cost of sensor nodes.

Our approach removes the computation overhead involved in localisation at individual sensor nodes. After transmitting three SIDs, a sensor node will not bother about the localisation process as the IUAN and CCS computes the sensor node location. But existing algorithms insist each sensor node to execute heavy calculations to find their locations. This consumes huge energy from sensor node's battery. Our intelligent arc selection approach further reduces the energy consumption and localisation time in the CCS by minimising the computation cost. Our approach executes less computations to find the location of sensor nodes than Ou[7], *et al.* and Sheu[11], *et al.* Figure 8 shows the localisation-computation cost at sensor nodes.

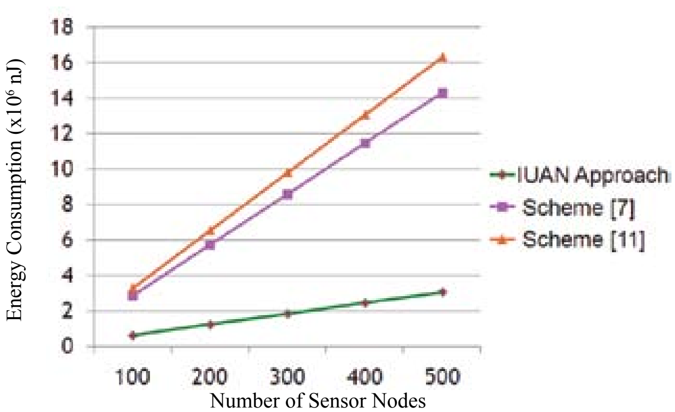


Figure 7. Comparison of localisation-communication cost.

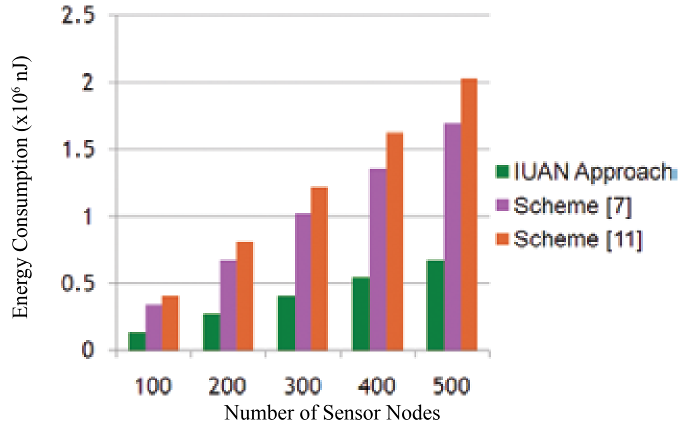


Figure 8. Comparison of localisation-computation cost.

Our approach increases the localisation coverage. In existing schemes, the sensor nodes are constrained to receive the radio beacons from same flying anchor, which is not at all possible as the trajectory of the flying anchors are not very efficient. With the basic anchor-guiding mechanisms, our method provides better localisation coverage. If an IUAN is not able to receive three SIDs from a same sensor node, the next IUAN can collect the balance as the CCS aggregates the LD messages of all IUAN. This ensures the better location coverage. Figure 9 shows the comparision of the localisation coverage effieciency of IUAN approach, Ou[7], *et al.* and Sheu[11], *et al.*

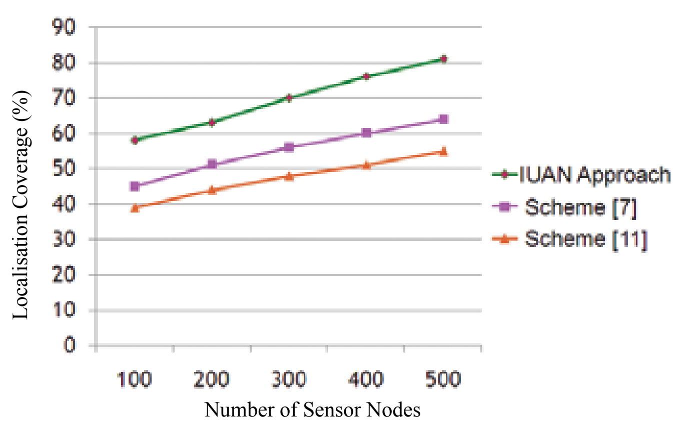


Figure 9. Comparison of localisation coverage.

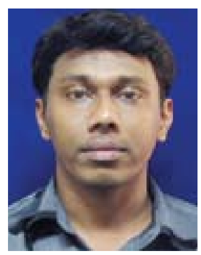
5. CONCLUSION

For the state-of-the-art, a novel IUAN-based localisation approach is proposed to conserve the energy in military sensor networks. The proposed localisation approach reduces the communication overhead and removes the computation cost at each sensor node, thereby reduces the overall energy consumption. Our approach used RIM model to withstand the radio irregularity problem, thereby the location accuracy is guaranteed. The IUAN approach shifts the localisation overheads from sensor nodes to the powerful CCS, where the CCS calculates the location of sensor nodes using IAS algorithm with low computation costs. The authors approach significantly reduces overall energy consumption at each sensor node, and thereby increases the overall lifetime of military sensor networks with high localisation coverage.

REFERENCES

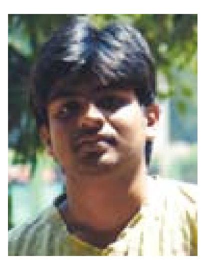
1. Akyildiz, IF, Su, W, Sankarasubramaniam, Y & Cayirci, E. Wireless sensor networks: A survey. *Comput. Networks,* 2002, **38**, 393-422. doi: 10.1016/S1389-1286(01)00302-4
2. Yick, J, Mukherjee, B & Ghosal, D. Wireless sensor network survey. *Comput. Networks*, 2008, **52**, 2292-2330. doi: 10.1016/j.comnet.2008.04.002
3. Anastasi, G, Conti, M, Francesco, MD & Passarella, A. Energy conservation in wireless sensor networks: A survey. Ad Hoc Networks, 2009, **7**, 537-568. doi: 10.1016/j.adhoc.2008.06.003
4. Aguiar, ABD, Neto, ADMS, Pinheiro, PR & Coelho, ALV. Applicability of a novel integer programming model for wireless sensor networks. *Int. J. Comput. Sci. Info. Security,* 2009, **3**(1),1-7.
5. Han, G, Xu, H, Duong, TQ, Jiang, J & Hara, T. Localization algorithms of Wireless Sensor Networks: A survey. *Telecommun. Sys.,* 2013, **52**(4), 2419-2436. doi: 10.1007/s11235-011-9564-7
6. Pal, A. Localization Algorithms in Wireless Sensor Networks: Current Approaches and Future Challenges. *J. Network Protoc. Algorithms*, 2010, **2**(1), 45-74.
7. Ou, CH & Ssu, KF. Sensor position determination with flying anchor in three dimensional wireless sensor networks. *IEEE Trans. Mobile Comput*., 2008, **7**(9), 1084-1097. doi: 10.1109/TMC.2008.39
8. Ou, CH. A localization scheme for wireless sensor networks using mobile anchors with directional antennas. *IEEE Sensors J.,* 2011, **11**(7), 1607-1616. doi: 10.1109/JSEN.2010.2102748
9. Liao, WH, Lee, YC & Kedia, SP. Mobile anchor positioning for wireless sensor networks. *IET Commun.*, 2011, **5**(7), 914- 921. doi: 10.1049/iet-com.2010.0336
10. Kuo-Feng Ssu, Chia-Ho Ou & Hewijin Christine Jiau. Localization with mobile anchor points in wireless sensor networks. *IEEE Trans. Vehicular Techno*., 2005, **54**(3), 1187-1197. doi: 10.1109/TVT.2005.844642
11. Sheu, JP, Hu, WK & Lin, JC. Distributed localization scheme for mobile sensor networks. *IEEE Trans. Mobile Comput*., 2010, **9**(4), 516-526. doi: 10.1109/TMC.2009.149
12. Xiao, Q.; Xiao, B.; Cao, J. & Wang, J. Multihop range-free localization in anisotropic wireless sensor networks: A pattern-driven scheme. *IEEE Trans.Mobile Comput.*, 2010, **9**(11), 1592-1607. doi: 10.1109/TMC.2010.129
13. Satyamurti, S & Joshi, R. ANN assisted node localization in WSN using TDOA. *Int. J. Innovative Research in Comput. Comm. Eng.,* 2014, **2**(4), 3871-3877.
14. Pescaru, D. & Curiac, D.I. Anchor node localization for wireless sensor networks using video and compass information fusion. *Sensors,* 2014, **14**, 4211-4224. doi: 10.3390/s140304211
15. Han, G.; Zhang, C.; Lloret, J.; Shu, L. & Rodrigues, J.J.P.C. A mobile anchor assisted localization algorithm based on regular hexagon in wireless sensor networks. *The Sci. World J.*, 2014, (2014), 1-13. doi: 10.1155/2014/919518
16. Guerrero, E.; Gao, Q. & Xiong, H.G. Employing Unmanned Aerial Vehicles for localization of sensor nodes in wireless sensor networks. *J. Networking Techno.*, 2010, **1**(1), 44-57.
17. Yadav, V.; Mishra, M.K.; Singh, A.K. & Gore, M.M. Localization Scheme for three dimensional wireless sensor networks using GPS enabled mobile sensor nodes. *Int. J. Next Generation Networks*, 2009, **1**(1), 60-72
18. Vincent, P.; Tummala, M. & McEachen, J. A new method for distributing power usage across a sensor network. *Ad Hoc Networks*, 2008, **6**, 1258-1280. doi: 10.1016/j.adhoc.2007.11.014
19. Zhou, G, Tian, HE, Krishnamurthy, S & Stankovic, J.A. Models and solutions for radio irregularity in wireless sensor networks. *ACM Trans. Sensor Networks*, 2005, **2**(2), 221–262. doi: 10.1145/1149283.1149287
20. Heinzelman, W.R.; Chandrakasan, A. & Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. *In* Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000. doi: 10.1109/HICSS.2000.926982

CONTRIBUTORS



Dr P. Shunmuga Perumal received ME in VLSI Design from Anna University, India. He is received his PhD from Anna University, Chennai, India. He is currently working as a Teaching Fellow in the Department of Information Science and Technology, College of Engineering, Anna University, India. His research interests include : Wireless sensor networks, power generation, wireless networks and location based services.

Dr V. Rhymend Uthariaraj received his PhD from Anna University, Chennai, India. He is currently a Professor and Director at the Ramanujan Computing Centre, College of Engineering, Anna University, Chennai, India. His research interests include wireless sensor networks, power generation, network security, pervasive computing, distributed computing and computer algorithms.



Mr V.R. Elgin Christo received BTech in Information Technology from Anna University, Chennai, India. He is pursuing his MTech in Information Technology in Anna University, Chennai, India. His area of interests include : Wireless sensor networks, embedded systems, power generation, wireless networks, Algorithm designs and location based services.