TrustLP: A *Trust*-Based *L*ocalization *P*rotocol for Wireless Sensor Networks

Ash Mohammad Abbas
Department of Computer Engineering
Aligarh Muslim University
Aligarh - 202002, India
am.abbas.ce@amu.ac.in

***Abstract*—Devising a protocol for computing the location of a sensor in a wireless sensor network (WSN) is a challenging task. In this paper, we present a protocol for computing the location of a sensor that is part of a WSN. The proposed protocol is based on the extent of the trust contained in the reply messages sent by the neighboring sensors that are either anchors or already localized nodes against the request messages sent by the node intending to localize itself. The proposed protocol is distributed, localized, and asynchronous. It does not require any prior knowledge of the topology of the network. The computational and communication complexities of the proposed protocol are $O(E)$ and $O(2E + V)$, respectively, where E is the number of links and V is the number of nodes in the network. The proposed protocol provides a significant improvement in terms of the computational and communication overhead over the existing protocols.**

***Keywords*—Localization, trust, wireless sensor networks.**

I. Introduction

A wireless sensor network (WSN) is a collection of sensor nodes that are connected through wireless links. There are many applications of a WSN. One such application is to *raise an alarm* well before the spread of forest fires and helping the persons of fire-control department in controlling and confining the fire. Another application can be to *habitat monitoring* of wild animals in the forest. Other applications may include monitoring the movement across the international boundaries also called *detection of infiltration* across the line of control (LoC).

All these applications require that the sensors should collect the data about various environmental parameters. The data collected by sensors is to be processed at a command and control center (CCC) in order to infer decisions. Further, to decide about some event and actions, it is not simply the data gathered by sensors that is required but the locations of sensors sending the particular information are also required. In order to send data (to the CCC for further processing) together with their locations, sensors need to know their own location. For the sensors to be aware of their locations, a possible solution can be to equip them with Global Positioning System (GPS) devices. However, since the sensors are spread in large quantities, therefore, equipping all sensors with GPS devices may not be a cost-effective solution.

It may happen that some of the sensors might be equipped with GPS devices, the remaining sensors can compute their locations using the locations of these sensors. The sensors equipped with GPS devices are called *anchors* or *beacons*, and the remaining are called ordinary sensors. The problem of computing the locations of ordinary sensors using the locations of anchors or by some other means is called the localization problem.

Many researchers have proposed different solutions to address the problem of localization in WSNs from different perspectives. An anchor-based localization protocol called AnchLP is proposed in [1]. The problem of localization pertaining to robots is addressed in [6] and how one can minimize localization errors in a localization technique called trilateration in the presence of landmark positions is described in [2].

A Confidence-based Iterative Localization (CIL) protocol is proposed in [7] with an emphasis on Quality of Trilateration (QoT). The issue of outliers in localization is addressed in [8]. An algorithm for localizing an object in a collaborative manner in a WSN is proposed in [9]. A major issue addressed in [9] is localization impairments due to faulty sensor and orientations of cameras. The issue of distance measurement in the presence of noise is addressed in [10] and an algorithm for localization based on descent gradient is proposed.

In this paper, we propose a trust-based localization protocol that we call TrustLP for WSNs. A feature that is common between an existing protocol called CIL [7] and the proposed protocol, TrustLP, is that both of them are based on the trust or confidence of neighboring nodes. However, there are significant differences between CIL [7] and TrustLP, which are as follows.

- The first major difference is that CIL [7] makes use of a distributed algorithm, which is based on the Bellman-Ford shortest path algorithm. However, the proposed protocol, TrustLP, neither requires any such algorithm nor the path followed during the localization need to be the shortest path. Instead, our protocol is based on the *request-reply* cycle. In other words, a node intending to localize itself sends a *request* message to its neighbors. A neighboring node that is aware of its own location responds with a *reply* message. The *reply* message brings the location and confidence of the responder node to the requester node. Using the locations and confidences of responders, the requester computes its own location and the confidence of localization.
- The second difference is that the CIL [7] consists of

978-3-9018-8273-9/15/ ©2015 IFIP

several iterations to localize even a single sensor as it allows the nodes already localized to improve their levels of confidences. However, the proposed protocol, TrustLP, does not allow already localized nodes to improve their confidences so that the localization process is finished in a time-bound fashion.

- The third difference is that in CIL [7], there can be more than one candidate trilaterations and from these candidate trilaterations, the one that results in the maximum confidence is chosen by the node intending to localize itself. In the proposed protocol, TrustLP, the strategy is different. The node first collects the reply messages until a timeout. Out of the confidences contained in the gathered reply messages, the node chooses the first $d+1$ maximum confidences, and thereafter, it computes their minimum, and that becomes its own confidence of localization.
- The computational complexity of the proposed protocol, TrustLP, is $O(E)$. However, the computational complexity of CIL [7] is $O(VE)$, where E is the number of edges (or links) in the network. Therefore, there is a significant improvement in the computational complexity of the proposed protocol, TrustLP, as compared to CIL [7].

As opposed to the existing protocols where the trust is assumed to be an independent random variable, we formulated the notion of trust in such a fashion so that it is capable of incorporating dependencies among the trusts or confidences of individual responding nodes in a WSN. We analyze the resulting confidence of an ordinary sensor as a function of the confidences of its responders and compute the number of quantums and the total number of quantums required to exhaust the upper limits of the confidences. Further, we analyze the computational and communication complexities of the proposed protocol and compare them with those of the existing protocols.

The remaining portion of this paper is organized as follows. In Section II, we describe the proposed protocol. In Section III, we describe a model for the trust. Section IV contains results and discussion. Finally, the last section is for conclusion and future work.

II. Proposed Protocol

The proposed protocol is based on the request-reply cycle. In other words, a node that intends to compute its own location sends request for localization to the nodes in its neighborhood. The neighboring nodes knowing their own position respond with reply messages. We first describe the major actions taken at the sender and receiver sides, and then describe how the node can compute its confidence of localization.

A. Sender Side

The following actions are taken at the sender side.

- A node that is unaware and intends to compute its location generates a *LocalizationREQuest* (LREQ) message. The format of LREQ is as follows.
  $<$ *SourceAddress, SrcSequenceNo, LocationRequired, TimeStamp* $>$
  where, *SourceAddress* is the address of the source that generated the LREQ, *SrcSequenceNo* is the source sequence number contained in the LREQ, and *LocationRequired* is a flag indicating whether it is a LREQ message or not. Note that $<$ *SourceAddress, SrcSequenceNo* $>$ uniquely identifies the LREQ. The field *TimeStamp* contains the time of sending the LREQ.
- When the source of LREQ receives the LREP, it records the time at which the LREP is received, and the location together with confidence of node sending the LREP. It computes the difference of the times of sending an LREQ, t_q, and receiving an LREP, t_r, and subsequently computes the distance between the itself and the node sending LREP, which is $c\left(\frac{t_r-t_q}{2}\right)$, where c is the propagation speed of electromagnetic waves.
- The node waits for a timeout to collect the LREPs sent by other neighbors. After the timeout, it selects the first $d+1$ LREPs with the maximum confidences.
- It then computes its own location using the locations and distances of its $d+1$ neighbors. It computes the confidence about its own computed location, which is the minimum of the confidences of the selected $d+1$ neighbors.

These steps are summarized for sender side in Algorithm 1.

Algorithm 1 Sender Side of TrustLP.

```
if MyLocationFlag == 0 then
    Generate an LREQ with the following format
    < SourceAddress, SrcSequenceNo,
    LocationRequired, TimeStamp >
    Send the LREQ to neighbors
    Wait for LREPs from neighbors
end if
if ReceivedPacketType == LREP then
    Record the ArrivalTime of LREP in LREPCache
    together with LREP.SourceAddress and LREP.Trust.
    Compute the time difference of arrival of LREP
    and sending of LREQ, t_r − t_q
    Compute the distance from MySelf to the
    LREP.SourceAddress s = c((t_r − t_q)/2)
    Wait for a TimeOut
end if
if TimeOut expired && |LREP| ≥ (d + 1) then
    Select d + 1 LREPs with max confidences
    Compute MyLocation using trilateration
    Set MyConfidence to the minimum of
    d + 1 max confidences
    Set MyLocationFlag = 1
end if
```

B. Receiver Side

When an LREQ is received by a neighboring node, it acts as follows.

- It examines *LocationRequired* flag. If *LocationRequired* is set to 1, it means that the node sending the LREQ intends to compute its own location.
- If it either knows its own position (i.e it is an anchor node) or it has already computed its own position, then it sends a *LocalizationREPly* (LREP) packet to the source node of the LREQ. The format of LREP is as follows.
  $<$ *SourceAddress, DestAddress, SrcSequenceNo, MyLocation, MyConfidence* $>$,
  where, the *SourceAddress* is the address of the node replying with LREP. *DestAddress* is the address of the source that generated the LREQ, *SrcSequenceNo* is the source sequence number contained in the LREQ, and *MyLocation* is the location of the node sending the LREP. The field *MyConfidence* contains the confidence of the node generating the LREP.

These steps are summarized for receiver side in Algorithm 2.

Algorithm 2 Receiver Side of TrustLP.

```
1: if MeAnchor == True then
2:    MyLocationFlag == 1
3:    MyConfidence = 1
4: else
5:    MyConfidence = EstimatedConfidence
6: end if
7: if MyLocationFlag == 1 && ReceivedPacketType
   == LREQ then
8:    if LREQ.LocationRequired==1 then
9:       Generate an LREP with the following format
         < SourceAddress, DestAddress, SrcSequenceNo,
         MyLocation, MyConfidence >, where,
         LREP.DestAddress = LREQ.SourceAddress,
         LREP.SrcSequenceNo = LREQ.SrcSequenceNo
10:      Send the LREP to LREQ.SourceAddress
11:   end if
12: end if
```

C. Computing the Trust

The field *Trust* contains the confidence of the responder[1]. One can compute one's own confidence as follows. If one is an anchor then one's confidence is 1. If one is not an anchor, then one has to localize itself. During the localization, if one has received the LREPs from at least $d+1$ nodes in the neighborhood out of which some nodes might be anchors and others might be nonanchors, then one can compute its location using trilateration. The confidence is the minimum of the confidence of at least $d+1$ neighboring responders. To compute the confidence, one may adopt the following procedure. Suppose a node receives LREPs from more than $d+1$ responders[2]. It sorts the LREPs in descending order of their confidences, and selects the top $d+1$ LREPs. The confidence of localization of the node that intends to localize itself is the minimum of the *MyConfidence*s of the $d+1$ anchors so selected.

[1]One may ask a question: What is the difference between confidence and trust? In our view, the term *confidence* can be used for the first person and the term *trust* can be used for second or third persons. Although, the extents of confidence and trust might not be equal. However, for the process of locating sensors in a WSN, one may assume the extents of confidence and trust to be equal.

[2]One may ask a question: What should the node do, if it receives less than $d+1$ LREPs? The answer is that if the number of LREPs is less than $d+1$, the node cannot localize itself.

TABLE I. COMPUTING THE CONFIDENCE FOR TWO-DIMENSIONAL LOCALIZATION.

Confidences c_1	c_2	c_3	*Resulting Trust*	*Number of Anchor Responders*
0	0	0	$\min(c_1, c_2, c_3)$	0
0	0	1	$\min(c_1, c_2)$	1
0	1	0	$\min(c_1, c_3)$	1
1	0	0	$\min(c_2, c_3)$	1
0	1	1	c_1	2
1	0	1	c_2	2
1	1	0	c_3	2
1	1	1	1	3

III. A MODEL FOR TRUST

Note that at the responding side of LREP, the extent of estimation can be termed as the *confidence* and at the receiving side, it is called the *trust*. The rationale behind this terminology is that one is responding with a specific confidence about its own location, however, at the side of the node that intends to localize itself, the node has to worry about to what extent it can trust the LREPs sent by the responders.

A. Confidence of Localization

Depending on whether the LREP received by the sensor is sent by an anchor or a nonanchor node, one can compute the confidence of localization. If an LREP is received by an anchor, the confidence of the anchor is 1.0, otherwise, the confidence of the responder node is taken to be the minimum of the confidences of its own responders, and so on.

Let a node other than an anchor received k LREPs from the neighboring nodes. Let c_1, c_2, ..., c_k be the values of trusts contained in the *MyConfidence* field of LREPs. After sorting these trusts (or confidences), let the sequence be $\bar{c}_1$, $\bar{c}_2$, ..., $\bar{c}_k$. The confidence of the node that intends to localize itself is as follows.

$$c_f = \min\left\{\bar{c}_1, \bar{c}_2, ..., \bar{c}_{d+1}\right\}. \quad (1)$$

In other words, the confidence of the node to be localized is the minimum of the $d+1$ maximum confidences of its neighbors responding with LREPs.

Lemma 1: Let the number of LREPs received by a node be ν, the expected number of neighbors of a node. The computational complexity for computing the confidence of localization by first sorting ν confidences is $O(\nu \log \nu)$ for a single node. For the whole network with V nodes is $O(V\nu \log \nu)$.

Proof: For any comparison based sorting algorithm, the time complexity to sort ν confidences is $O(\nu \log \nu)$. Therefore, selecting $d+1$ maximum confidences out of ν confidences contained in the LREPs, requires a time[3] of $O(\nu \log \nu)$.

[3]This is the time complexity when a comparison based sorting algorithm is used to sort the confidences. However, one can reduce it to $O(\nu)$ using an algorithm such as Bucket Sort for restricted input. Here, since the values of confidences are lying in the interval $[0, 1]$, therefore, one may apply Bucket Sort algorithm.

Note that the number of nodes replying in response to an LREP is less than or equal to ν because all neighbors of a node might not be localized. On an average, the expected number of localized neighbors is $\frac{\nu}{2}$. The computational complexity in the average case is $O\left(\frac{\nu}{2}\log\left(\frac{\nu}{2}\right)\right)$. This also comes out to be $O(\nu\log\nu)$. For a network with V, the number of nodes to be located is at most V. Therefore, the complexity of locating all nodes in the network is $O(V\nu\log\nu)$. ■

Instead of sorting the confidences, one can successively compute the maximum of the confidences of the received LREPs. Out of the LREPs received by the node that sent LREQs, the node needs to select only $d+1$ LREPs with the maximum confidences. We use the following notation.

$$\mathcal{C} = \max_{i=1}^{d+1}\{c_j|_{j=1}^{k}\}. \tag{2}$$

Here, $\mathcal{C}$ is the set of confidence with cardinality $d+1$.

Lemma 2: The computational complexity for computing the confidence of localization by computing the successive maximums is $O(\nu)$ for a single sensor to be localized. For all nodes to be localized in a WSN, the computational complexity is $O(\nu V)$, where V is the number of nodes in the sensor network.

Proof: For ν confidences, the time taken to compute the first maximum is $O(\nu)$, for second maximum it is $O(\nu)$, and so on. For computing $d+1$ successive maximums is $O\left((d+1)\nu\right)$. Ultimately the time complexity in this case is $O(\nu)$ because $d+1$ is a constant factor for a d-dimensional space. It is for a single sensor to be localized. For all sensors to be localized in a WSN, the computational complexity is $O(\nu V)$, where V is the number of nodes in the WSN. ■

Note that the computational complexity of $O(\nu V)$ in case of TrustLP is a significant improvement over CIL [7], which uses Bellman-Ford algorithm, whose time complexity is $O(VE)$ where V is the number of vertices or nodes and E is the number of edges or links. The reason is that $\nu \ll E$.

One may ask a question: What is the average number of neighbors of a node? The answer is as follows. Assume that a WSN with n nodes is deployed in a region of area A in such a fashion that the sensors are randomly distributed with uniform density. Then, the node density is $\rho = \frac{n}{A}$. Let the range of each sensor be r, then the average number of neighbors of a sensor is $\nu = \rho\pi r^2 - 1$.

One may raise a question: Can one express the average number of neighbors of a node in terms of the number of edges (or links) and the number of vertices (or nodes) in a network? The answer is yes and is used in the following lemma.

Lemma 3: Let the network be represented by an undirected graph $G = (V, E)$ and all links are assumed to be bidirectional. The computational complexity of TrustLP is $O(E)$.

Proof: In terms of the number of edges and the number of vertices, the average degree (equivalently average number of neighbors of a node) is [3], [4], $\nu = \frac{2E}{V}$. Using Lemma 2, the computational complexity of the proposed protocol, TrustLP, is $O\left(V \times (2E/V)\right)$, which comes out to be $O(E)$. ■

Expressed this way, the proposed protocol, TrustLP, whose computational complexity is $O(E)$, provides an improvement over CIL [7], whose computational complexity is $O(VE)$. Note that the computational complexity of a protocol shows up in the delay incurred during the localization of nodes in a WSN.

TABLE II. COMPUTING THE CONFIDENCE FOR TWO-DIMENSIONAL LOCALIZATION.

Confidences				*Resulting*	*Number of Anchor*
c_1	c_2	c_3	c_4	*Trust*	*Responders*
0	0	0	0	$\min(c_1, c_2, c_3, c_4)$	0
0	0	0	1	$\min(c_1, c_2, c_3)$	1
0	0	1	0	$\min(c_1, c_2, c_4)$	1
0	1	0	0	$\min(c_1, c_3, c_4)$	1
1	0	0	0	$\min(c_2, c_3, c_4)$	1
0	0	1	1	$\min(c_1, c_2)$	2
0	1	0	1	$\min(c_1, c_3)$	2
0	1	1	0	$\min(c_1, c_4)$	2
1	0	0	1	$\min(c_2, c_3)$	2
1	0	1	0	$\min(c_2, c_4)$	2
1	1	0	0	$\min(c_1, c_2)$	2
0	1	1	1	c_1	3
1	0	1	1	c_2	3
1	1	0	1	c_3	3
1	1	1	0	c_4	3
1	1	1	1	1	4

B. Computation of Confidence: Examples

We now illustrate how one can compute the confidence of localization in a d-dimensional space, where $d = 2$.

1) Two-Dimensional Space: Table I lists the possibilities for computing the confidence of one's own localization using the extent of confidences of the responders of the LREPs that one receives from either the anchor nodes or already localized nodes in a two-dimensional region of deployment. Let c_f be the confidence of the node to be localized. Then, an expression for computing c_f as a function of the confidences of the responders can be written as follows.

$$\begin{aligned} c_f &= \{\min(c_1, c_2, c_3)\} \vee \{c_1 \vee c_2 \vee c_3\} \vee \{1\} \\ &\vee \{\min(c_1, c_2) \vee \min(c_1, c_3) \vee \min(c_2, c_3)\} \end{aligned} \tag{3}$$

Here, $\vee$ represents the logical OR operator.

Note that (3) can be recursively applied to compute the confidence of responders. The confidence of localization of a nonanchor node depends on the confidences of the nodes that respond against the LREQ sent by it, and similarly, the confidence of a responding nonanchor node depends on the confidences of its own responders while it computed its own location, and so on. A recursive expression incorporating the possibilities of the number of nonanchor nodes in stage i can be written as follows.

$$\begin{aligned} c_j^i|_{j=1}^3 = \{\min(c_j^{i-1}|_{j=1}^3\} \vee \{\vee_{j=1}^k c_j^{i-1}\} \vee \{1\} \\ \vee\{\min(c_j^{i-1}, c_k^{i-1})|_{j=1,k=1,j\neq k}^{j=3,k=3}\}, \end{aligned} \tag{4}$$

where the subscript j represents the anchor and superscript i represents the iteration.

Let us now illustrate how one can compute the confidence of localization in a d-dimensional space, where $d = 3$.

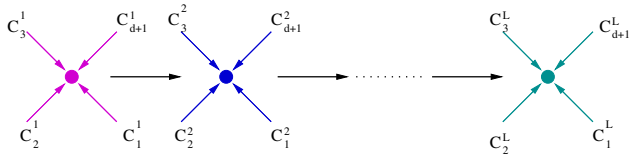

Fig. 1. Levels of localization: Successive localization of unlocalized sensors using the confidences of nodes responding with LREPs.

2) Three-Dimensional Space: For three-dimensional space, the number of nodes sending a response after receiving an LREQ, has to be at least 4 to enable a node to compute its own location. These responders might be either anchors or nodes knowing their own locations. As mentioned earlier, an anchor is assumed to possess a confidence of 1.0, and nodes other than anchors are assumed to possess a confidence less than or equal to 1.0. Ideally, if a sensor intending to localize itself receives at least $d+1$ LREPs from the neighboring anchors, its confidence of localization is 1.0 provided the distance measurement is not erroneous.

Depending on the number of anchors as the responders, the expression for computing the resulting confidence might be different. In other words, the expression for computing the resulting confidence has to take into account all possibilities pertaining to the number of anchors sending responses against the LREQ sent by the sensor intending to locate itself.

Table II lists the possibilities for computing the confidence of one's own localization using the extent of confidences of the responders of the LREPs that one receives from either the anchor nodes or already localized nodes in a three-dimensional region of deployment. One may write the expressions similar to (3) and (4) for computing the confidence of the node intending to localize itself using the confidences of responder nodes and that are contained in the LREPs.

In what follows, we analyze the communication complexity of the proposed protocol.

C. Communication Complexity

By the term *communication complexity*, we mean the overhead in terms of the number of transmissions incurred during the localization of nodes in the network. For that purpose, we have the following lemma.

Lemma 4: Let the communication overhead be represented in terms of the number of LREQ and LREP messages. The communication overhead to locate the sensors that are not aware of their locations in a WSN is $O(2E+V)$, where E be the number of links and V be the number of nodes in a WSN.

Proof: A node intending to compute its own location transmits an LREQ message to its neighbors. In response to the LREQ, the neighbors aware of their locations transmit the LREP messages. Note that these LREP messages are upper bounded by the number of neighbors of a node. In other words, on an average the number of LREPs transmitted to localize a node is less than or equal to the average number of neighbors of a node is ν, which is equal to $\rho\pi r^2 - 1$. Therefore, total of transmissions required to locate a sensor is $\nu + 1 = \rho\pi r^2$.

Let there be V nodes and E links in the network. The total number of transmissions required to locate all nodes in the network that are not already located is upper bounded by $V(\nu+1)$. Using $\nu = \frac{2E}{V}$, the total number of transmissions to locate all nodes that are not aware of their own locations is $O(2E+V)$. ■

Note that a localization protocol with less number of LREQ and LREP transmissions is expected to consume less energy as compared to the one with more number of LREQ and LREP transmissions. Also, the amount of delays incurred during the localization of nodes in the WSN is expected to be smaller for a protocol with less number of such transmissions because each transmission is to be scheduled. As a result, the proposed protocol is expected to consume less energy and smaller delays as compared to the existing protocols.

D. Incorporating Measurement Errors

Note that even if a node that intends to compute its location receives LREPs from neighbors that are anchors and are supposed to be confident about their own location, however, the node might not be able to compute its own location with 100% confidence because of several reasons such as inaccuracies introduced during the measurement of distances. Also, the errors may propagate during the process of localization. Let us model the errors and the extent of their propagation during the process of localization.

Let there be a sensor that receives an LREP from a node which is already localized and whose confidence is c. However, due to errors in measurement of distance between the anchor and the node, let the resulting confidence be ξc. Here $1-\xi$ is assumed to be very small, e.g. $1-\xi \approx 0.01$, for relatively an accurate distance measurement method.

Let an unlocalized sensor receives LREPs from $d+1$ neighbors that are already localized, where d equals to either 2 (for 2-dimensional space) or 3 (for a 3-dimensional space). Let the confidence brought in the LREPs be c_i, $1 \le i \le (d+1)$. Let the corresponding factors incorporating the errors due to measurement be ξ_i, $1 \le i \le (d+1)$. The resulting confidence of localization is as follows.

$$c = \min\left(\xi_i c_i\right), \quad 1 \le i \le (d+1). \tag{5}$$

As mentioned earlier, for an LREP received from an anchor node, $c_i = 1$, in ideal conditions (e.g. no errors in measurement of distances, etc). Equation (5) provides an expression for computing the confidence of an unlocalized sensor node making use of confidences of responding nodes that are already localized. It may happen that responding nodes might have been previously localized using LREPs from their responders, and so on. This situation is shown in Figure 1. Let there be ℓ such localization stages with confidences c_i^j and factors ξ_i^j for $1 \le i \le (d+1)$ and $1 \le j \le \ell$. The resulting confidence is as follows.

$$c = \min(\xi_i^j c_i^j), \quad \text{for} \quad 1 \le i \le (d+1) \quad \text{and} \quad 1 \le j \le \ell, \tag{6}$$

where, subscript i represents the ith responder and superscript j represents the jth stage of localization.

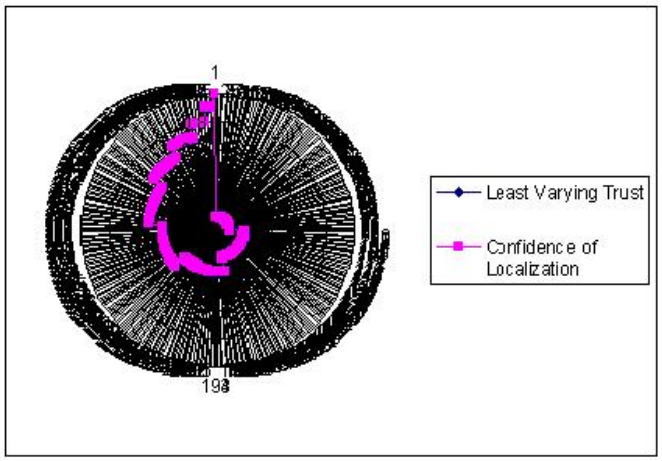

Fig. 2. Confidence of localization together with the least varying trust among a set of anchors.

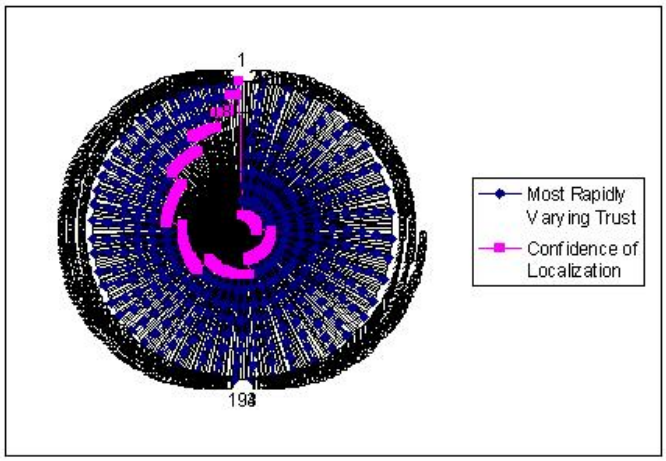

Fig. 4. Confidence of localization together with the most varying trust among a set of anchors.

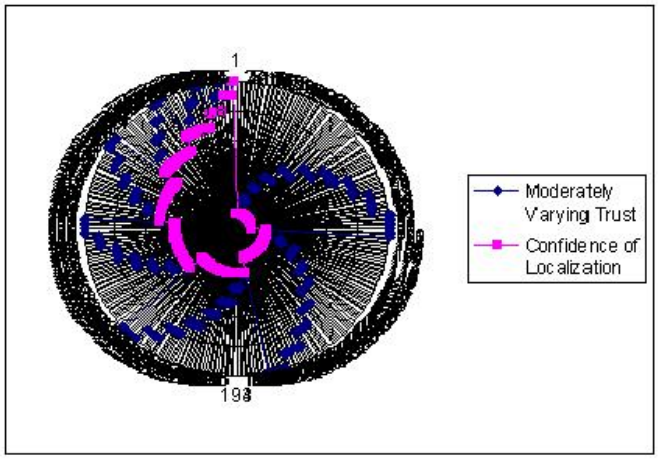

Fig. 3. Confidence of localization together with moderately varying trust among a set of anchors.

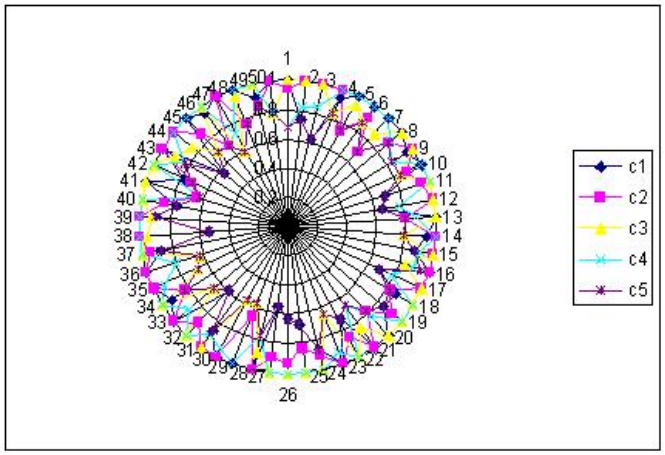

Fig. 5. Confidence of localization for random values of confidences of responding nodes.

IV. Results and Discussion

We first describe the setting for generating samples of trusts and then discuss some results.

A. Generation of Samples of Trusts

We varied the confidences of localized nodes for 2-dimensional and 3-dimensional spaces. A description of the setting is as follows. For 2-dimensional space max three confidences were chosen and assumed to be the confidences contained in the best three LREPs. The resulting confidence of localization of the node to be localized is the minimum of these three confidences (treated as trusts for the node to be localized). For the purpose of illustrations, the trusts are arranged and the samples of trusts are generated as follows. The trust of the third localized node is varied first in steps of an increment keeping the trusts of the second and first localized nodes to be constant. When the upper limit of the trust for third node is exhausted, the trust of second node is incremented and again the trust of third node is varied and so on. When the upper limit of the trust for second node is exhausted, the trust of the first node is incremented and the lower limits of the trusts of second and third node are now set to be equal to the trusts of the first node. This process is repeated until the upper limit on the trust of first localized node is exhausted.

Let c_i and c_f be the initial and final values of trusts and Δ be the step of increment. We define a parameter that we call the *number of quantums*, q, as follows.

$$q = \left(\frac{c_f - c_i}{\Delta}\right) + 1. \tag{7}$$

Obviously, the number of quantums is the number steps required to exhaust the upper limit of the trust of the third localized node starting from the lower limit of the trust where the lower limit is set to be equal to the trust of the first localized node.

For a particular value of the trust of the first localized node and starting from thereon, the number of steps needed to exhaust the upper limits of the trusts of the third and second localized nodes, in steps of given increment, is defined to be the *total number of quantum cases*. For a d-dimensional space, the total number of quantum cases, S_q can be written

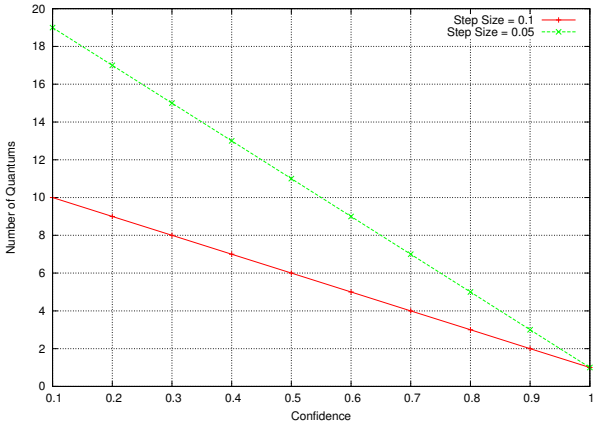

Fig. 6. Number of quantums versus confidence of localized nodes for different values of step size.

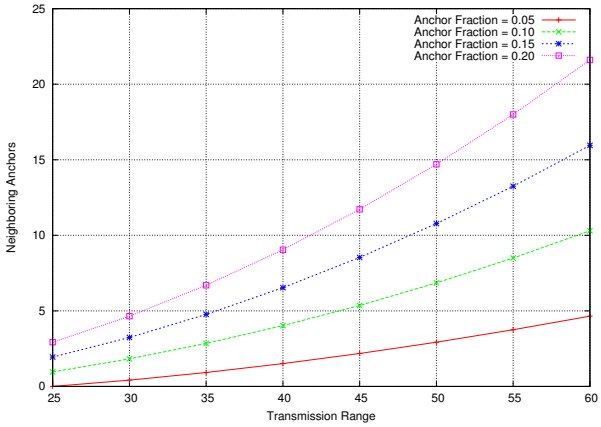

Fig. 8. The average number of neighboring anchors of a node in a WSN versus the transmission range of nodes.

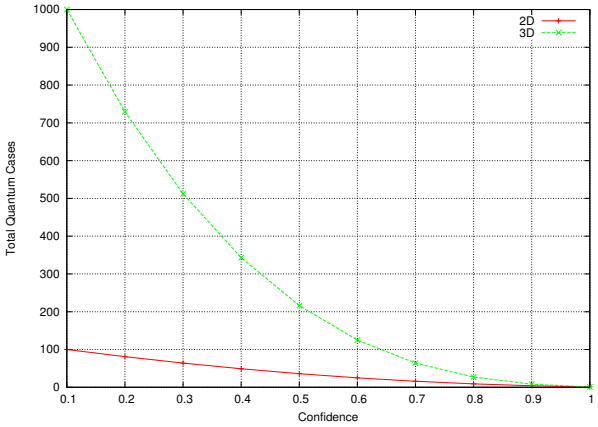

Fig. 7. Total number of quantum cases versus confidence of localized nodes for $d = 2$ and $d = 3$.

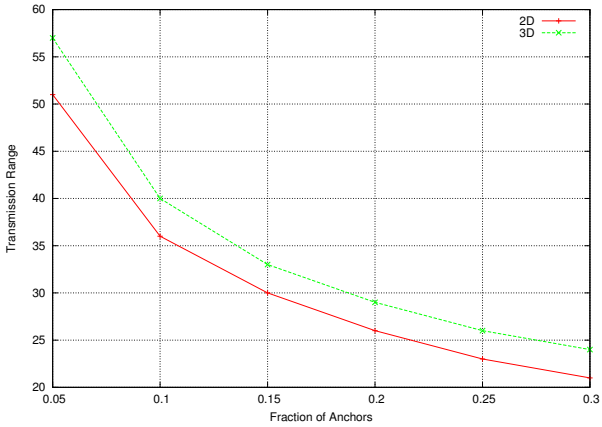

Fig. 9. The minimum transmission range of nodes in a WSN versus the fraction of anchors.

as follows.

$$S_q = q^d. \tag{8}$$

Recall that the minimum number of anchors, a, or already localized nodes required to locate a sensor is $d + 1$ for a d-dimensional space. Therefore, (8) can also be written as follows.

$$S_q = q^{a-1}. \tag{9}$$

B. Performance Results

We first present results incorporating confidence (or trust) of responding neighbors of a sensor intending to localize itself, and then we present results related to deployment of a sensor network in a d-dimensional space.

1) Confidence Results: Figure 2 shows confidence of localization together with the least varying trust among a set of localized nodes. It shows that confidence of localization varies linearly with the least varying trust among the set of localized nodes. Actually, it shows that the confidence of localization is equal to the least varying trust. This is due the properties of *min* function which results in the minimum confidence among a set of localized nodes.

Figure 3 shows confidence of localization together with the moderately varying trust among a set of localized nodes. A closer look reveals that the confidence of localization for one set of trust, $0.1 - 1.0$ is 0.1; for the next set of trust, $0.2 - 1.0$ is 0.2; for the next set of trust, $0.3 - 1.0$ is 0.3; and so on. It implies that for moderately varying trust, the confidence of localization is the lower limit of the successive sets of trust from 0.1 to 1.0.

Figure 4 shows confidence of localization together with the most varying trust among a set of anchors. A closer look reveals that the value of confidence of localization for the first 100 trust values varying between $[0.1, 1.0]$ is 0.1. For the next 81 values trust varying between $[0.2, 1.0]$, the confidence of localization is 0.2. For the next 64 trust values varying between

$[0.3, 1.0]$, and so on. We call them total number of quantum cases, which are squares of the number of quantums.

Figure 5 shows confidence of localization for random values of confidences of responding nodes. For nodes that are not anchors, the confidence is generated in such a fashion so that it lies in the interval $[0.5, 1.0]$. It means that all nonanchor responders need to possess a confidence of 0.5 so as to participate in the localization process. This assumption seems to be logical because allowing a responder whose confidence about its location falls below 0.5 is of no use for locating a node. As mentioned earlier, the confidence of each anchor is assumed to be 1. We observe that the resulting confidence is minimum of the max confidences of the respondersi.e. $c_f = \min\{c_1, c_2, c_3, c_4\}$.

Figure 6 shows the number of quantums versus confidence of localized nodes for step size of 0.1 and 0.05. We observe that the number of quantums linearly decreases with an increase in the confidence of localized nodes. The decrease in the number of quantums is more significant when the step size is decreased from 0.1 to 0.05. This is in accordance with the expression given by (7). For a confidence of 0.5, the number of quantums for the step size 0.1 is 6, and for the step size 0.05, it is 11. It implies that decreasing the step size by a factor of 50% approximately doubles the number of quantums.

Figure 7 shows the total number of quantum cases versus confidence of localized nodes for $d = 2$ and $d = 3$. We observe that the total number of quantum cases decreases with an increase in the confidence. The total number of quantum cases decreases more rapidly in case of 3-dimensional space as compared to that in 2-dimensional space. The decrease is proportional to the square of the number of quantums in case of 2-dimensional space, and is proportional to the cube of the number of quantums in a 3-dimensional space.

2) Deployment Related Results: Let there be a WSN with 10000 sensors randomly spread in a region of area of $1000m \times 1000m$. Out of them, let a particular percentage of the sensors be anchors. The remaining are ordinary sensors that are not aware about their own locations.

Figure 8 shows the average number of neighboring anchors of a node in a WSN versus the transmission range of nodes. We observe that the average number of neighboring anchors of a node increases with an increase in the transmission range of nodes that are part of WSN. For example, when 10% of the sensors are anchors, if the transmission range increases from $35m$ to $40m$, the average number of neighboring anchors increases from 2.84 to 4.024. The reason is that with an increase in the transmission range, the number of neighbors of a node increases and so does the number of neighboring anchors.

Figure 9 shows the minimum transmission range of nodes in a WSN versus the fraction of anchors. Actually, it is the transmission range for which the average number of neighboring anchors exceeds $d + 1$ in a d dimensional space, where $d = 2$ for 2-dimensional space and $d = 3$ for 3-dimensional space. We pointed out earlier that in a d-dimensional space, the number of anchors needed is $d + 1$. We observe that the transmission range decreases with an increase in the fraction of anchor nodes. For example, for $d = 2$, for 5% anchor nodes, the minimum transmission range required is $51m$ and for 10% anchors the minimum transmission range needed is $36m$. For, $d = 3$ the transmission ranges decreases from $57m$ to $40m$ when the percentage of anchors is increased from 5% to 10%.

V. Conclusion

Addressing the problem of locating a sensor in a wireless sensor network is a challenging task. The contributions made in the paper are as follows.

- We proposed a localization protocol for locating a sensor in a WSN. Using the protocol, a node can localize itself and can compute its confidence of localization by trusting a required number of nodes in its vicinity with the maximum confidences.
- We proposed a model for evaluating the performance of the proposed protocol. The model consists of the parameters such as confidence of localization as a function of the maximum trusts of the d responders from a set of responders.
- The computational complexity of the proposed protocol, TrustLP, is $O(E)$, where E is the number of links in the network. Further, the communication complexity of TrustLP is $O(V + 2E)$, where V is the number of nodes in the network.
- The proposed protocol provides an improvement over existing protocols in terms of the computational and communication overhead. Further, the proposed protocol is distributed, scalable, and asynchronous.

One may propose further optimizations for enhancing the performance of the protocol and it forms the future work.

References

[1] A.M. Abbas, H.A.A. Qasem, "AnchLP: An Anchor-Based Localization Protocol for Wireless Sensor Networks", *Proceedings of IEEE International Conference on Advanced Computing, Communication and Informatics (ICACCI)*, pp. 2122-2128, September 2014.

[2] A. Bahr, J.J. Leonard, "Minimizing Trilateration Errors in the Presence of Uncertain Landmark Positions", *Proceedings of 3rd European Conference on Mobile Robots*, pp. 1-6, September 2007.

[3] C. Bettstetter, "On the Minimum Node Degree and Connectivity of a Wireless Multihop Network", *Proceedings of ACM Conference on Mobile and Ad Hoc Networks (MobiHoc)*, pp. 80-91, June 2002.

[4] R. Diestel, "Graph Theory", Springer, Second Edition, 2000.

[5] G. Mao, B. Fidan, B.D.O. Anderson, "Wireless Sensor Network Localization Techniques", *Computer Networks*, vol. 51, pp. 2529–2523, 2007.

[6] F. Thomas, L. Ros, "Revisiting Trilateration for Robot Localization", *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 93–101, February 2005.

[7] Z. Yang, Y. Liu, "Quality of Trilateration: Confidence-Based Iterative Localization", *IEEE Transactions on Parallel and Distributed Systems*, pp. 631–640, vol. 21, no. 5, May 2010.

[8] Q. Xiao, K. Bu, Z. Wang, B. Xiao, "Robust Localization Against Outliers in Wireless Sensor Networks", *ACM Transactions on Sensor Networks*, vol. 9, no. 2, article 24, March 2013.

[9] M. Karakaya, H. Qi, "Collaborative Localization in Visual Sensor Networks", *ACM Transactions on Sensor Networks*, vol. 10, no. 2, article 18, January 2014.

[10] M.N. Pour, G.C. Rojas, "A Novel Algorithm for Distributed Localization in Wireless Sensor Networks", *ACM Transactions on Sensor Networks*, vol. 11, no. 1, article 1, November 2014.