A Relative Ad-hoc Localization Scheme using Optical Wireless

Jayasri Akella, Murat Yuksel, and Shiv Kalyanaraman
Electrical, Computer and Systems Engineering,
Rensselaer Polytechnic Institute, Troy, New York.

***Abstract*— Scalable network localization is key for realizing ad-hoc networks. In this paper we propose a localization scheme where nodes form a relative coordinate system of the network in a distributed manner. Each node in the ad-hoc network is capable of estimating both the range and orientation of its 1-hop neighbors. The proposed localization scheme then achieves a relative coordinate system for any topology as long as the underlying graph is connected, irrespective of the node density. We evaluate the performace of the proposed scheme and show with simulations that it is more scalable than a similar localization scheme that uses triangulation. We also present propagation of localization error in the network due to estimation errors in both the range and the orientation. We also discuss how this scheme can be implemented using optical wireless technology.**

I. INTRODUCTION

Ad-hoc networks benefit from node localization as it enables stateless geographic routing within the network [8]. In sensor networks, localization makes information from the sensors more meaningful. The most important aspect of the localization algorithm is scalability, specially when applications with thousands of sensor nodes are envisioned for the future. Both ad-hoc and sensor networks ideally require localization be achieved with few or no anchor nodes, with low density deployment of the nodes in the network, and with minimally centralized infrastructure to support localization and mobile node tracking. Node density cannot always be counted on, specially when ad hoc nodes are sprinkled from an aeroplane onto a geographic location, as often described in literature. In addition the localization scheme should accomodate changes in the network topology with very small or no additional control messaging overhead and should be robust to mobility of the nodes in the network.

The problem of end-to-end wireless geographic routing using network localization can be broadly categorized into three layers as shown in Figure 1. The lowest layer addresses the localization scheme to obtain the node coordinates. And the second layer maps these coordinates to the node "Identifiers" like a name or a number [10], [15] . The third layer uses these identifiers to perform stateless geographic routing [8]. A successful network localization scheme addresses all the three layers, localization to routing in a distributed, and scalable manner. In this paper we focus on the first layer, to localize the nodes and obtain their coordinates in a distributed manner.

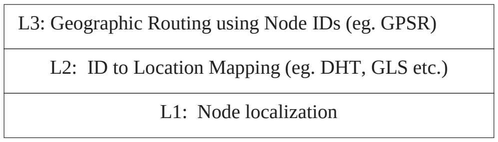


Fig. 1. Classification of research issues in distributed localization.

Typically in a geographic localization scheme an estimate for "distance" between the nodes is obtained either by the number of hops [11], [14] or an RTT [6], or an explicit range [4] or orientation [12] and then it is translated into virtual or (global or relative) physical coordinates using triangulation. In triangulation, each node needs to communicate with *three* already localized nodes to compute its own location. Therefore, in order to implement a distributed localization scheme using trangulation, a very high node density and a very high average node degree are needed to achieve acceptable node localization percentages (for example, localization for a ring topology is hard to acheive using triangulation). In the past literature, the average node degree ranged from 6 to 16 [9], [12], [4],[19], [16], [17], [5] to achieve a reasonable coverage (extent of node localization).

In this paper we propose an approach to obtain relative coordinate system in an ad-hoc network scenario where node localization can be achieved with a *single* localized neighbor. The method uses both range and orientation information between the adjacent nodes. The method achieves 100% node localization as long as the underlying graph is *connected*, irrespective of the average node degree and node density. The method does not require any anchor or landmark nodes. Any randomly placed node can become the origin of the relative coordinate system and nodes can obtain their coordinates in a distributed manner with respect to this origin. We evaluate our localization algorithm and show the improvement in performance in terms of the percentage of network localization, number of iterations needed to obtain the relative coordinate system and the number of control messages needed. We also study the error in localization due the range and orientation estimation errors and how it propagates with the number of hops away from the origin in the relative coordinate system.

In methods where triangulation is used, either range or orientation estimates are obtained to come up with the coordinate system for the network. Therefore, nodes are assumed to have the hardware capability to measure either the range or orientation of the neighboring nodes. Though this is a simple requirement from the hardware capability point of view, triangulation itself puts a very high demand on the network topology in terms of node density and average node degree. In addition, to achieve the network coordinate system in a distributed manner, the method may require a few beacon/landmark nodes. On the other hand, our method does not demand high node density or degree from the network but needs that the node be able to measure both the range and orientation of the neighboring nodes. The benefit is that node localization can be achieved with a *single* neighbor.

Our scheme results in a relative coordinate system of the network without any anchor nodes and network wide floods in a distributed manner. An additional benefit of the proposed method is that it can be easily extended for mobile tracking. Due to space limitation we limit the scope of our paper to static network localization.

The method described in this paper can be implemented with any physical layer technology, provided that a node capable of measuring both the range and the orientation of its 1-hop neighbors. We propose to implement the present scheme with Optical wireless, which we refer to as Free-Space Optics (FSO) communication techology in the paper. FSO uses light for communication between two nodes with air as the medium [1]. FSO is known for its high bandwidth, low power per bit and easy deployment. We propose to use the “directionality” of the light beams to measure the orientation between the two nodes and time-of-flight between two nodes to measure the range, thus obtaining the position “vector” of any node relative to another.

The paper is organized as follows: Section II we describe the principle of localization, and illustrate with simulations the algorithm for the proposed localization scheme and its evaluation. Section III-A we discuss how the error is propagated. Section IV We briefly discuss the details on how the nodes can be implemented using FSO technology to have the hardware capabilities to measure the orientation and the range. In Section V we discuss the prior work on network localization techniques. Section VI concludes the paper.

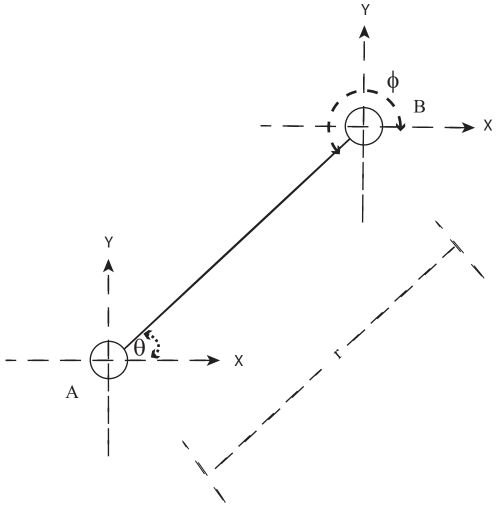


Fig. 2. Illustration of the principle of an FSO based location system.

II. FSO LOCALIZATION SCHEME

A. Principle

Figure 2 illustrates the principle of our localization scheme. Two nodes A and B are such that the perpendicular axes through each of them are aligned with each other. Then, any node, in this case, node A measures the range r and the orientation θ of its 1-hop neighbor, node B and computes the coordinates of the node B, with itself at the origin as following:

$$x_b = rcos\theta$$

$$y_b = rsin\theta$$

If node A is already localized with coordinates (x_a, y_a) then the coordinates of node B can be obtained by simple vector addition:

$$x_b = x_a + rcos\theta$$

$$y_b = y_a + rsin\theta$$

Thus each node can compute its 1-hop neighbors coordinates relative to itself. A leader is selected to be at the origin and a relative coordinate system of the entire network can be obtained in a distributed manner. Thus this scheme requires only *one* already localized node for any given node to localize. When the underlying graph is connected, we can have all the nodes in the network (100% coverage or extent) localized.

In contrast, triangulation needs atleast three localized nodes to obtain node localization. Typically these nodes are the landmark nodes and their location in the network plays a significant role on the extent of localization. And the anchors need to know that they are indeed the anchor nodes. Moreover, triangulation needs a high average node degree and high node density to achieve a reasonable percentage (coverage) of node localization.

The attractive part of the technique is that the final coordinate system can be achieved even when the network is sparse, as long as the graph is connected. Our scheme needs additional hardware capability for a node to measure both the range and orientation of the neighboring nodes. At the end of the algorithm, we obtain a relative coordinate system with an elected leader at the origin. Once the initial relative coordinate system is obtained, the origin is independent of the position of the leader node. And all the nodes are free to move, and the location of the origin is preserved. Our approach does not need any network wide flooding or anchor nodes for synchronization and does not depend on the knowledge of the network topology.

B. Assumptions and Problem Definition

We assume that each node has a set of perpendicular axes passing through it as shown in Figure 2. We assume that the FSO nodes have the capability to measure the range of the 1-hop neighbors and the orientation of the neighbor. In addition, each node is also capable of re-orienting the axes passing through it. We will explain how these capabilities can be achieved using nodes with Free-space Optical transceivers. Further, the nodes in the network have unique IDs, which are used to elect a leader. We assume that network is connected and all the nodes at bootstrap have $(0, 0)$ as coordinates.

Then, the network localization problem is defined as follows: At bootstrap, the nodes are randomly located. At bootstrap, the axes of different nodes are oriented randomly with respect to each other. All the nodes in the network graph are as shown in Figure 3.a . The objective of the FSO localization algorithm (FLA) is to orient the axes of all the nodes such that they are parallel to each other as shown in Figure 3.b. This is achieved between any two nodes by measuring the orientations of each other and exchanging that information. This procedure is explained in Section II-D. After that step, each node then estimates the "direction" at which the neighbor is located by measuring the angle with respect to its X-axis. Then, the nodes jointly obtain a relative coordinate system in a distributed manner.

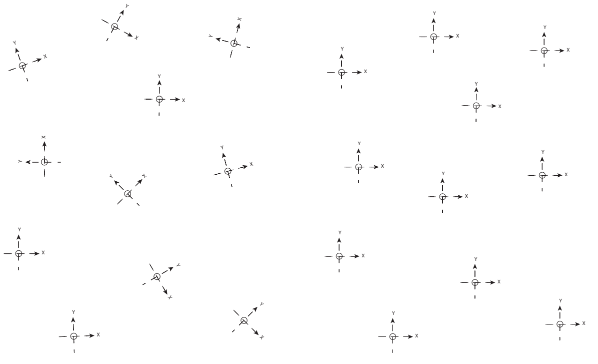

(a) (b)

Fig. 3. (a) Nodes before localization. (b) Nodes after localization.

C. FSO Localization Algorithm

The FSO localization algorithm has three phases. First, the node with the "highest ID" is elected as the leader in a distributed manner. Then all the nodes align their axes with the leader node's axes. Then each node computes the coordinates of its neigbors with lower IDs.

At bootstrap each node communicates with all its 1-hop neighbors and the IDs of the neighbors are exchanged. Each node becomes aware of the 1-hop neighbor with the highest ID and axes orientation and saves that information. Whenever a node updates to a new higher neighbor ID, it broadcasts the same to its 1-hop neighbors. This process of exchanging the highest ID happens until there are no updates at any node. At that time, all the nodes in the network are aware of the leader node's ID and its axes orientation information. Each node waits for a pre-assigned time duration and when it does not hear any more broadcasts from its neighbors, it aligns its axes according to the leader node's orientatio information. The actual alignment procedure is explained under Section II-D. This completes the leader selection and alignment phase.

Once aligned, each node can measure the range and orientation of its neighbor with a lower ID. When a node computes the coordinates of the nodes with lower IDs it sets the nodes "Highest-CoOrd-ID" to the leader ID. A node becomes eligible to compute the coordinates of the neighboring nodes when it receives its coordinates from a node whose Highest-CoOrd-ID is equal to the leader ID. By default, the leader with the highest ID has this condition satisfied, so it starts to compute the coordinates of its 1-hop neighbors by measuring their range and orientation. The leader thus establishes itself as the origin. The 1-hop neighbors of the leader node receives their coordinates from the leader and update their coordinates. These 1-hop neighbors of the leader, inturn become eligible to calculate the coordinates of their 1-hop neighbors who have not already received the coordinates from the leader. The relative coordinates with respect to the leader, are calculated using the vector addition described in Section II-A. The pseudo-code of the algorithm is shown in Algorithm 1.

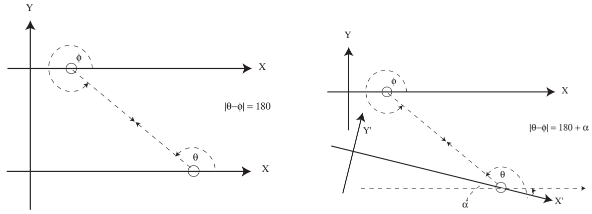
Algorithm 1 Localization

```
if $MyLeaderIDChangeFlag = 1$ then
    UPDATE and broadcast to neighbors of this new highest ID
end if
repeat
    Listen for more updates from the neighbors
    if Received a broadcast from the neighbor then
        if $ReceivedID > MyLeaderID$ then
            $MyLeaderID = ReceivedID$ and $MyLeaderIDChangeFlag = 1$
        else if $MyLeaderIDChangeFlag = 0$ then
        end if
    end if
until No broadcast from the neighbors for time T
ALIGN axes with the highestID neighbor
if $HighestCoOrdID = LeaderID$ then
    COMPUTE coordinates of neighbors with lower ID
end if
```

After the localization is complete, the location of the leader node can then be considered as "Virtual Origin Node" O. Since at the time of initial localization, the positions of each of the nodes are determined by this location, the localization does not get affected even if the leader node is changed or moved. This feature makes this localization scheme robust to node movements. If a new node joins the network it simply communicates with the nearest neighbor and calculates its co-odinates from its position with respect to the neighbor and the neighbor's coordinates with respect to O, irrespective of its ID. A node that either goes into sleep or dies will not have any affect on the coordinate system.

D. Alignment

The alignment procedure for the nodes needed in our localization scheme is explained here. Consider two FSO nodes as shown in Figure 2. A sees B at (θ, r) and B sees A at (ϕ, r). The two nodes exchange this information while aligning. When the axes of A and B are aligned, as shown in Figure 4(a), $|\theta - \phi| = 180$. When the axes are not aligned, say by an angle $\pm\alpha$, then the equation becomes $|\theta - \phi| = 180 \pm \alpha$. The method is illustrated in Figure 4(b). Depending on who the leader is, for example if node A has a higher ID than node B, node B aligns itself with node A. When the nodes are aligned with each other, then the node with the higher ID becomes the reference and the node with the lower ID simply accepts the coordinates given by the node of the higher ID.



(a) (b)

Fig. 4. (a) Aligned nodes with parallel axes. (b) Non-aligned nodes.

III. PERFORMANCE OF THE LOCALIZATION ALGORITHM

We evaluated the performance of our localization algorithm for scalability using the following metrics,

- Extent of node localization
- Convergence time
- Number of messages per node to localize in the relative coordinate system.

We will discuss each of them below. We simulated for random networks ina area of 200X200 Sq. units for two node densities, 100 nodes and 400 nodes. We compared the metrics against a simple distributed triangulation scheme with three landmark nodes. Bear in mind that the triangulation scheme *does not* give a relative coordinate system, but just localizes the nodes relative to three landmark nodes. Whereas with our scheme, we obtain a coordinate system, with an origin and the co-ordiantes of the nodes with respect to the origin. We observe that even the simple version of triangulation performs worse than our scheme.

As mentioned in the previous section, with our scheme, all the nodes in the network are localized if the underlying graph is connected. Thus, the extent of localization is always 100%, irrespective of the average node degree of the graph. Figure 5 illustrates the 100%

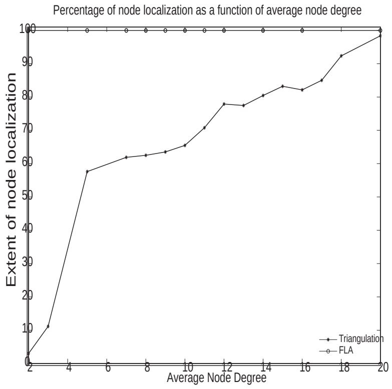


Fig. 5. Extent of localization as a function of average node degree.

localization achieved using this algorithm. The figure also illustrates how triangulation needs a high average node degree to achieve a resonable extent of localization.

The second metric is the convergence time on the algorithm, which we measured as the number of iterations needed to achieve 100% localization. Each iteration is defined as a new update of the highest ID at a node and the broadcast associated with it. We count the maximum number of iterations needed for all the nodes to localize. Since leader election and identification is implemented in a hop-by-hop manner, the maximum number of iterations taken by the algorithm is a function of how many hops away a node is from the leader node being selected. In Figure 6 the number of iterations taken by the algorithm to achieve 100% localization is shown as a function of the average node degree. As the node degree increases, the number of iterations needed to localize decreases, since the information about the leader node spreads more quickly. Whereas as the node density in the network increases, the number of iterations increase because then hop length becomes smaller and the number of hops from the leader node increases.

Figure 7 compares the number of iterations taken by triangulation and our scheme. Our scheme out performs triangulation for all node degrees.

Figure 8 shows the average number of messages each node needs to localize. The number of messages for higher node density is higher because of higher number of iterations needed. As we observed, the number of messages increase linearly with node degree. Figure 9 shows that the number of messages needed for localization is independent of the node density, making the algorithm more scalable.

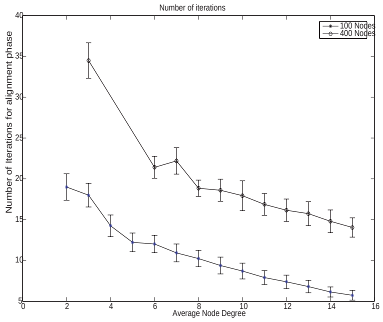


Fig. 6. Number of iterations needed to localize as a function of average node degree.

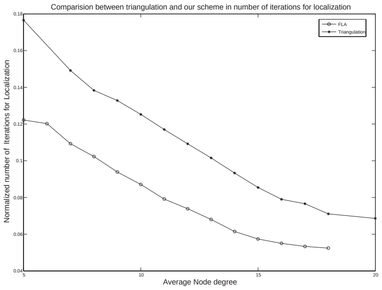


Fig. 7. Comparison of the number of iterations for localization for FLA and triangulation.

A. *Measurement Errors and Accuracy of Localization*

In this section we evaluated the robustness of our localization scheme in the presence of measurement errors in both the range and the angle. The error creeps into the location system from the following sources:

- Finite field of view of the photo-detecors: This effects the alignment angle.
- Finite package density of the transceivers. This too effects the alignment angle.
- Measurement error of the range r.

As shown in Figure 12, the transceiver a has a finite field of view, a magnitude denoted by the angle ϕ. Consequently, the transceiver, when trying to measure the orientation at which it "sees" another node, the angle becomes $\theta \pm \phi/2$. A similar error results when the number of transceivers on the FSO node are few, thereby reducing the resolution of the angle with which a neighbor is perceived. In our simulation we introduced an error of $\pm 20\%$ in the measurement of both range and orientation. Figure 11 shows how percent error in X, Y co-ordinates due to measurement errors in range

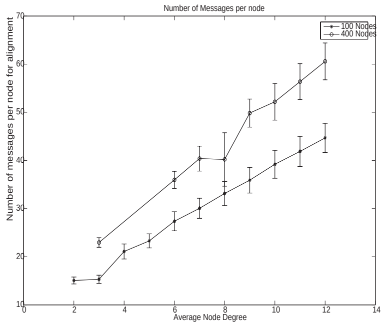


Fig. 8. Number of messages per node for alignment and leader selection as a function of average node degree.

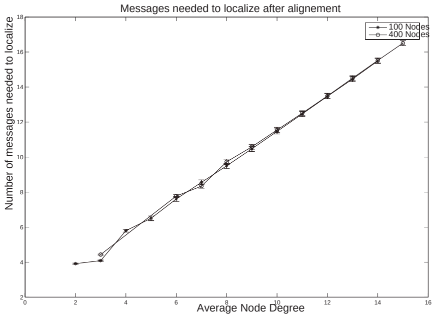


Fig. 9. Number of messages per node for localization as a function of average node degree.

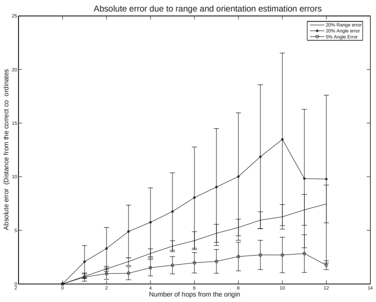


Fig. 10. Absolute error in terms of distance from correct coordinates.

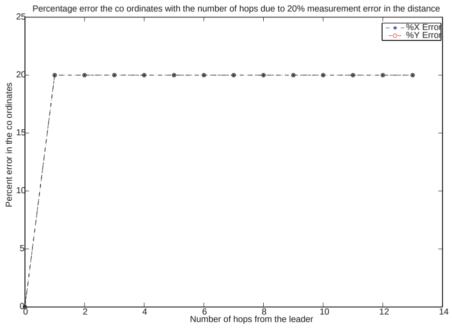


Fig. 11. Percent Error in X, Y coordinates as a result of measurement error in range.

behave with the number of hops from the leader node. The simulations show the worst case error results. The error stays constant at 20% for all the hops. Figure 10 shows how the "absolute" error due to an estimation error in both the angle and range propagates with the number of hops from the origin (leader node). The plot shows an linear increase due to the range error as expected. The error due to an error in angle is much more pronounced than that of the range. Figure 10 also shows an improvement in the error behaviour when the number of transceivers on the FSO node is increased. For the lower error, we are increasing the number of transceivers on the FSO node and also decreasing each transceivers field of view. This will decrease the value of ϕ as shown in Figure 12.

IV. FSO SYSTEM

In this section we discuss how to realize a practical scheme to implement the measurement of both the angle and the range between two communicating nodes. Typically in RF technology, range is measured using TDOA or signal strength of the received signal [13], [18], [3], [11]. There are currently techniques available in RF to obtain orientation [12] and the range information of a neighbor. In this section we will describe how a practical system can be implemented using free-space-optical technology. Since the primary focus of the paper is to introduce our localization scheme, but not the hardware implementation, we describe the implementation briefly. More details on this implemetation are in [2].

We propose to use the "directionality" of the optical signals to measure the orientation of the neighbors. In our implementation, each node is equipped with multiple optical transceivers as shown in Figure 12. Each trasceiver on the node has a direction defined by its line of sight, in this scheme it coincides with the X-axis of the node. Each transceiver also has a finite field of view denoted by ϕ, the X-axis being right in the middle of the field of view. And the orientation of the neighbor is measured with respect to this axis. Since FSO communication is directional, there is no interference as experienced in RF.

Each transceiver can both receive and send signals to and from it's 1-hop neighbors it is directly in view with. These systems can be implemented using off-the-shelf

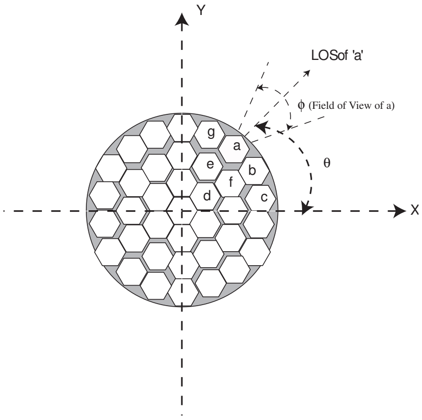


Fig. 12. FSO antenna for localization.

components. The density with which the transceivers are tessellated on the node and the field of view of these transceivers decides the accuracy that can be obtained while measuring the orientation of the 1-hop neighbor. On the otherhand, the range error depends on the electronics used to compute the time-of-flight information between two nodes. The time-of-flight can be streched artificially so as to be able to measure using off-the-shelf electronics.

Each node has a set of perpendicular axes going through it. By idenifying the location of individual transceivers, a node can recognize the orientation of the axes with respect to itself. To re-orient the axes, the node just needs to shift its reference. Thus each node equipped with optical trasceivers and a processing capability can be used for network localization using our scheme.

V. PRIOR WORK

Depending on the application and the context for which location information is used, there are several types of location systems that exist. [7] reviews a host of location systems, that work with centralized infrastructure or in a distributed manner.

The most popular method of obtaining location information is using GPS (Global Positioning System). GPS is a *absolute physical* positioning technology, providing absolute global position of the objects. Because of the high cost and need for infrastructure, GPS is not entirely suitable for positioning in ad-hoc / sensor network environments.

Typically in a geographic localization scheme an estimate of “distance” is obtained either by the number of *hops* or an *RTT*, or an explicit *range* or *orientation* to compute the virtual or physical (absolute or relative) coordinates respectively. In literature, three kinds of node coordinates are proposed and are discussed below.

The first one, as described in [6] ”virtual coordinates” for the nodes are obtained based on the underlying connectivity of the network but not true geographic distances. The primary objective of these coordinates are to find servers which are located closer to the client, for example, in a peer to peer application. The method piggybacks on the existing traffic to get RTT data to another node which is used to compute the coordinates. The authors proposed a “height vector” which represents the access delays experienced by the nodes so as the coordinates accurately represent the total RTT between two nodes. The goal is to accurately predict RTT under changes in the network and use that information for server selection, rather than “geographic routing”. [14] proposes another virtual coordinate localization scheme used for geographic routing. This method identifies perimeter nodes using beacons placed in the middle of the ad-hoc network. The beacons and the identified perimeter nodes perform broadcast operations so triangulation for the number of hops can happen at regular nodes and within the perimenter nodes. The power of such systems is that geographic routing is achievabale without the “actual” location information.

The second type of coordinates are “global geographic” coordinates consistent with GPS when only a small subset of the nodes in the network has GPS information. These systems rely on range or orientation estimate with the 1-hop neighbors and hence are completely distributed. In [12] a distance-vector based technique that uses “orientation forwarding” to obtain localization is proposed to use with mapping and Geodesic routing. This technique uses angle of arrival to triangulate. With this method, even when only a fraction of nodes have global positioning information, location information is propagated hop-by-hop and network localization is achieved. This system can handle mobility with the mobile node communicating with it’s 1-hop neighbors to triangulate and compute its new position. This is possible because of the presence of a few GPS aware nodes in the network. Another similar technique is proposed by [17] by cooperative ranging between nodes used with TERRAIN (Triangulation via Extended Range and Redundant Association of lntermediate Nodes) approach to localize and reduce localization errors due to range measurement errors.

The third type “relative geographic” coordinates in GPS-free networks for location aided routing or Geodesic forwarding. These techniques typically result in a coordinate systems with respect to the network topology, and hence are relative, similar to our approach. For example [4] provides a relative coordinate system by each node from the knowledge of the distance from their 1-hop neighbors. Each node builds its local coordinate system with itself as the origin and the first hop neighbors. And in the second stage each node broadcasts to build a network coordinate system by aligning the axes of all the nodes.

Our work is very closely related to [5]. In [5], the authors showed that triangulation schemes that use either range or orientation only require very high node densities. They propose to make use of both the range and orientation to improve on both the density and placement of the anchor requirements of the localization schemes. We also propose to use two paramters for localization, range, r and orientation, θ. The novelty of our scheme is to align the nodes axes with an elected leader. This simplifies achieving a distributed relative coordinate system. And moreover easily extends itself to support localization of mobile nodes. In addition, we present the distributed algorithm and evaluate the performance of the algorithm in terms of the number of iterations needed to obain the relative coordinate system and the number of control messages needed. We briefly discuss the details on how the nodes can be implemented using FSO technology.

VI. CONCLUSION

In this paper we presented a localization scheme that achieves a relative coordinate system in an ad-hoc network in a distributed manner. The scheme achieves 100% node localization when the underlying graph is connected, irrespective of the average node degree or node density. We evaluated the performance of the algorithm in terms of the coverage (extent of localization), number of iterations, and control messages needed to achieve the relative coordinate system. We also compared these metrics for a scheme that uses triangulation for localization and showed that our scheme performs better. We simulated the error in localization due to measurement errors in range and orientation and its propagation with the number of hops from the origin.

In the future, it will be interesting to investigate scenarios that minimize the error due to estimation errors in range and orientation. Our localization scheme can be extended easily to handle mobility of the nodes. Another interesting research direction is to develop a scheme wherein nodes in a neighborhood localize while moving, in a distributed manner without relying on a central infrastructure.

ACKNOWLEDGMENTS

This research was supported by NSF-STI 0230787.

REFERENCES

[1] http://www.freespaceoptics.org/freespaceoptics/default.cfm/.

[2] J. Akella, C. Liu, D. Partyka, M. Yuksel, S. Kalyanaraman, and P. Dutta. Building blocks for mobile free-space-optical networks. In *Proceedings of IFIP/IEEE International Conference on Wireless and Optical Communications Networks (WOCN),*, pages 164–168, Dubai, United Arab Emirates, March 2005.

[3] P. Bahl and V. N. Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *IEEE INFOCOM*, Israel, 2000.

[4] S. Capkun, M. Hamdi, and J. Hubaux. Gps-free positioning in mobile ad-hoc networks. In *Proceedings of the 34th Annual System Sciences, 2001*, Hawaii, Jan. 2001.

[5] K. Chintalapudi, R. Govindan, G. Sukhatme, and A. Dhariwal. Ad-hoc localization using ranging and sectoring. In *INFOCOM*, Hong Kong, China, March 2004.

[6] F. Dabek, R. Cox, F. Kaashoek, and R. Morris. Vivaldi: a decentralized network coordinate system. In *SIGCOMM'04*, Portland, Oregon, USA, Aug. 2004.

[7] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 34(8):57–66, Aug. 2001.

[8] B. Karp and H. Kung. Greedy perimeter stateless routing for wireless networks. In *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*, pages 243 – 254, Boston, MA, August 2000.

[9] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparision. *Elsevier Computer Networks*, 43(2003):499–518, 2003.

[10] J. Li, J. Jannotti, D. S. J. D. Couto, D. R. Karger, and R. Morris. A scalable location service for geographic ad hoc routing. In *ACM MOBICOM*, pages 120–130, Boston, MA, 2000.

[11] D. Niculescu and B. Nath. Ad hoc positioning system (aps). In *GLOBECOM 2001*, San Antonio, Nov. 2001.

[12] D. Niculescu and B. Nath. Ad hoc positioning system (aps) using aoa. In *INFOCOM 2003*, pages 1734–1743, Mar. 2003.

[13] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *Proc. of the Sixth Annual ACM International Conference on Mobile Computing and Networking (MOBICOM)*, aug 2000.

[14] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica. Geographic routing without location information. In *MOBICOM'03*, San Diego, CA, Sept. 2003.

[15] S. Ratnasamy, B. Karp, S. Shenker, D. Estrin, R. Govindan, L. Yin, and F. Yu. Data-centric storage in sensornets with ght, a geographic hash table. *MONET*, 8(4):427–442, 2003.

[16] C. Savarese, K. Langendoen, and J. Rabaey. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *USENIX Technical Annual Conference*, pages 317–328, Monterey, CA, 2002.

[17] C. Savarese, J. Rabaey, and J. Beutel. Location in distributed ad-hoc wireless sensor networks. In *Proceedings of Acoustics, Speech, and Signal Processing, 2001.*, pages 2037–2040, May 2001.

[18] A. Savvides, C.-C. Han, and M. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *ACM MOBICOM*, Rome, Italy, 2001.

[19] A. Savvides, H. Park, and M. Srivastava. The bits ans flops of the n-hop multilateration primitive for node localization problems. In *First ACM International Workshop on Wireless Sensor Networks and Applications*, pages 112–121, Atlanta, GA, 2002.