Ad Hoc Positioning System (APS) *

Dragos Niculescu and Badri Nath
{dnicules, badri}@cs.rutgers.edu

April 12, 2001

Abstract

Many ad hoc network protocols and applications assume the knowledge of geographic location of nodes. The absolute location of each networked node is an assumed fact by most sensor networks which can then present the sensed information on a geographical map. Finding location without the aid of GPS in each node of an ad hoc network is important in cases where GPS is either not accessible, or not practical to use due to power, form factor or line of sight conditions. Location would also enable routing in sufficiently isotropic large networks, without the use of large routing tables. We are proposing APS – a distributed, hop by hop positioning algorithm, that works as an extension of both distance vector routing and GPS positioning in order to provide approximate location for all nodes in a network where only a limited fraction of nodes have self location capability.

1 Introduction

Ad hoc networks have mostly been studied in the context of high mobility, high power nodes, and moderate network sizes. Sensor networks, while typically having low powered nodes, low mobility and large sizes, classify as ad hoc networks in many cases, when deterministic placement of nodes is not possible. With recent advances in sensing device architectures [4], it can be foreseen that cheap, or even disposable nodes, will be available in the future, enabling an array of new agricultural, meteorological and military applications. These large networks of low power nodes face a number of challenges: routing without the use of large conventional routing tables, adaptability in front of intermittent functioning regime, network partitioning and survivability. In this paper, we address the problem of self locating the nodes in the field, which may provide a solution to the first challenge, and solve other practical problems as well. One scenario involving sensor networks frequently mentioned in literature is that of aircraft deployment of sensors followed by in flight collection of data by simply cruising the sensor field. This and other meteorological applications, are implicitly assuming that the data provided by the sensor is accompanied by the sensor's location, which makes it possible to attach this information to a geographical map of the monitored region. If this is an absolute necessity in order to make sense of the observed data, accurate location might also be useful for routing and coordination purposes. Algorithms such as GEDIR[1], or geocast[2], enable routing with reduced or no routing tables at all, which are appropriate for devices like the Rene mote[4], with only half a kilobyte of RAM. An improvement that can be applied to some ad hoc routing schemes, Location Aided Routing [11] limits the search for a new route to a smaler request

---

*This research work was supported in part by DARPA under contract number N-666001-00-1-8953 and a grant from CISCO systems.

zone. Also, APS is appropriate for indoor location aware applications, when the network's main feature is not the unpredictable, highly mobile topology, but rather deployment that is temporary, and ad hoc. These networks would not justify the cost of setting up an infrastructure to support positioning, like proposed in [7], [8], or [9].

GPS, which is a public service, can satisfy some of the above requirements. However, attaching a GPS receiver to each node is not always the preferred solution for several reasons: *cost* – if we are envisioning networks of thousands, or tens of thousands of nodes, (this factor might be of diminished importance in the future); *limited power* – battery capacities are increasing much slower than, say Moore's law; *inaccessibility* – nodes may be deployed indoors, or GPS reception might be obstructed by climatic conditions; *imprecision* – even with the selective availability recently turned off (May 2000), the location error might still be of 10-20m, which might be larger the hop size of some networks; *form factor* – a Rene board [4] is currently the size of a small coin.

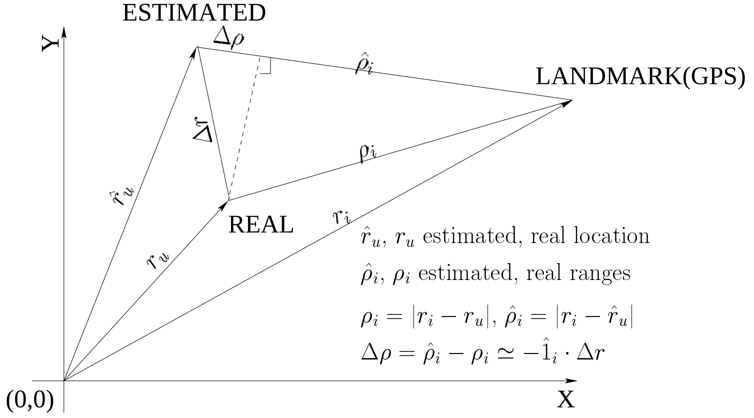
There are several requirements a positioning algorithm has to satisfy. First, it has to be distributed: in a very large network of low memory and low bandwidth nodes, designed for intermittent operation, even shuttling the entire topology to a server in a hop by hop manner would put too high a strain on the nodes close to the basestation/server. Partitioned areas would make centralization impossible, and anisotropic networks would put more strain on some nodes that have to support more forwarding traffic than others. Changing topologies would also make the centralized solution undesirable. Second, it has to minimize the amount of node to node communication and computation power, as the radio and the processor are the main sources of draining battery life. Also, it is desirable to have a low signaling complexity in the event a part of the network changes topology. Third, the positioning system should work even if the network becomes disconnected - in the context of sensor networks, the data can be later collected by a fly-over basestation. Finally, our aim is to provide absolute positioning, in the global coordinate system of the GPS, as opposed to relative coordinates, for the following reasons: relative positioning might incur a higher signaling cost in the case the network topology changes, and absolute positioning enables a unique namespace, that of GPS coordinates.

The rest of the paper is organized as follows: the next section summarizes similar efforts in current research, section III presents a short GPS review, as its principles are central to our approach. Section IV explains the APS approach, with the proposed propagation methods, section V presents simulation results and we conclude with some considerations about node mobility effects on APS.

2 Related Work

Reference [3] is proposing a positioning scheme that works in a centralized manner by collecting the entire topology in a server and then solving a large system that will minimize positioning errors for each node. Reference [5] presents a relative positioning system, without the use of GPS, in which the origin of the coordinate system is voted by a collection of nodes called reference group. The disadvantages, besides the ones stemming from the relative positioning versus absolute, are that when the reference moves, positions have to be recomputed for nodes that have not moved, and if intermediate nodes move, fixed nodes depending on them also have to recompute position (not knowing if the reference has moved). However, the coordinate system propagation is appropriate for hop by hop dissemination of distances to landmarks, and is applicable with our distance based scheme. In [8] a location system based on an uniform grid of powerful (compared to the nodes) basestations, serves as landmark mesh. A random node in the network will be able to localize itself by estimating its distance to the well known positions of closest basestations. RADAR [9] is a scheme in which the entire map is in advance measured for its radio propagation properties,

Figure 1: GPS, simplified problem



and positioning is achieved by recognizing fingerprints of previously mapped locations. The cricket location system [7] uses radio and ultrasound signals to estimate euclidean distances to well known beacons, which are then used to perform triangulation. The key features of our proposed approach, in contrast with the ones mentioned above, are that it is decentralized, it does not need special infrastructure, and provides absolute positioning.

3 GPS review

In Global Positioning System (GPS) [6], triangulation uses ranges to at least four known satellites to find the coordinates of the receiver, and the clock bias of the receiver. For our node location purposes, we are using a simplified version of the GPS triangulation, as we only deal with distances, and there is no need for clock synchronization.

The triangulation procedure starts with an apriori estimated location that is later corrected towards the true location. In figure 1, let $\hat{r}_u$ be the estimated location, r_u the real location, $\hat{\rho}_i = |r_i - r_u| + \epsilon_i$ and $\rho_i = |r_i - \hat{r}_u| + \hat{\epsilon}_i$ the respective ranges to the GPS i. The correction of the range, $\Delta\rho$ is approximated linearly to accommodate a linear system solving(as opposed to quadratic). If $\hat{1}_i$ is the unit vector of $\hat{\rho}_i$, $\hat{1}_i = -\frac{r_i - \hat{r}_u}{|r_i - \hat{r}_u|}$ and $\Delta r = \hat{r}_u - r_u$, then the approximate of the correction is: $\Delta\rho = \hat{\rho}_i - \rho_i \simeq -\hat{1}_i \cdot \Delta r + \Delta\epsilon$. Performing the above approximation for each satellite independently leads to a linear system in which the unknown is the location correction $\Delta r = [\Delta x \ \Delta y]$.

$$\begin{bmatrix} \Delta\rho_1 \\ \Delta\rho_2 \\ \Delta\rho_3 \\ \dots \\ \Delta\rho_n \end{bmatrix} = \begin{bmatrix} \hat{1}_{1x} & \hat{1}_{1y} \\ \hat{1}_{2x} & \hat{1}_{2y} \\ \hat{1}_{3x} & \hat{1}_{3y} \\ \dots & \dots \\ \hat{1}_{nx} & \hat{1}_{ny} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

After each iteration, the corrections Δx and Δy are applied to the current position estimate. The iteration process stops when the corrections are below a chosen threshold. Solving the linear system can be done using any least square method (we used the Householder method).

4 Ad Hoc Positioning System(APS)

If a graph is sufficiently connected, and the lengths of its edges are all known, then its plane topology may be reconstructed. But what is a *sufficient* degree of connectivity? If we assimilate the graph with a wire frame, where nodes act as hinges, our goal is to determine which part of the graph has nonmoving parts, and those will be the nodes which can determine their location. Once such a wire-frame is fixed, it will have a reference system of its own, that eventually has to be aligned to the global coordinate system of the GPS. In order to fix this wire frame somewhere on the global plane, at least three nodes(called landmarks), that are GPS enhanced, or know their position by some other means, have to be present in the connected graph.

Devices as simple as the Rene motes [4] have software access to the signal strength of the radio signal, thus offering a way to estimate distance to immediate neighbors. This measurements however, are affected by errors. One of the aims of our positioning system is to enhance position accuracy as the fraction of landmarks of the entire population increases. Even if it is theoretically sufficient to have three landmarks, the presence of measurement errors will demand higher fractions of landmarks, depending on the requirements of the application.

4.1 APS Algorithm

It is not desirable to have the landmarks emit with large power to cover the entire network for several reasons: collisions in local communication, high power usage, coverage problems when moving. Also, it is not acceptable to assume some fixed positions for the landmarks, as the applications we envision are either in flight deployments over inaccessible areas, or possibly involving movement and reconfiguration of the network. In this case, one option is to use hop by hop propagation capability of the network to forward distances to landmarks. In general, we aim for the same principle as GPS, with the difference that the landmarks are contacted in a hop by hop fashion, rather than directly, as ephemerides are. Once an arbitrary node has estimates to a number(≥ 3) of landmarks, it can compute its own position in the plane, using a similar procedure with the one used in GPS position calculation described in the previous section. The estimate we start with is the centroid of the landmarks collected by a node.

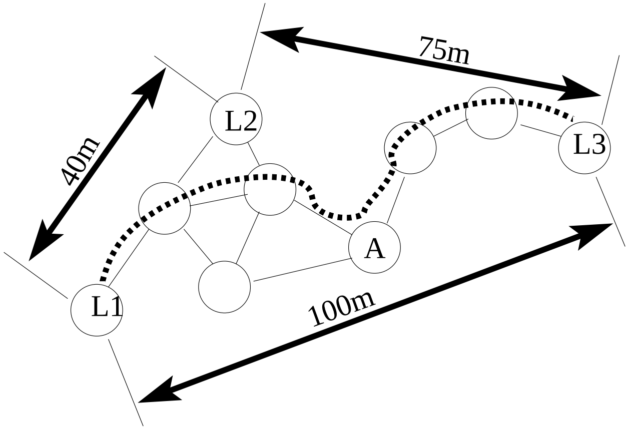
In what follows we will refer to one landmark only, as the algorithm behaves identically and independently for all the landmarks in the network. It is clear that the immediate neighbors of the landmark can estimate the distance to the landmark by direct signal strength measurement. Using some propagation method, the second hop neighbors then are able to infer their distance to the landmark, and the rest of the network follows, in a controlled flood manner, initiated at the landmark. Complexity of signaling is therefore driven by the total number of landmarks, and by the average degree of each node.

What makes this method similar with the distance vector routing, is that at any time, each node only communicates with its immediate neighbors, and in each message exchange it communicates its available estimates to landmarks acquired so far. This is appropriate for nodes with limited capabilities, which do not need, and cannot handle the image of the entire, possible moving, network. We are exploring three methods of hop to hop distance propagation and examine advantages and drawbacks for each of them. Each propagation method is appropriate for a certain class of problems as it influences the amount of signaling, power consumption, and position accuracy achieved.

4.2 "DV-Hop" propagation method

This is the most basic scheme, and it first employs a classical distance vector exchange so that all nodes in the network get distances, in hops, to the landmarks. Each node maintains a table

Figure 2: *"DV-hop"* correction example



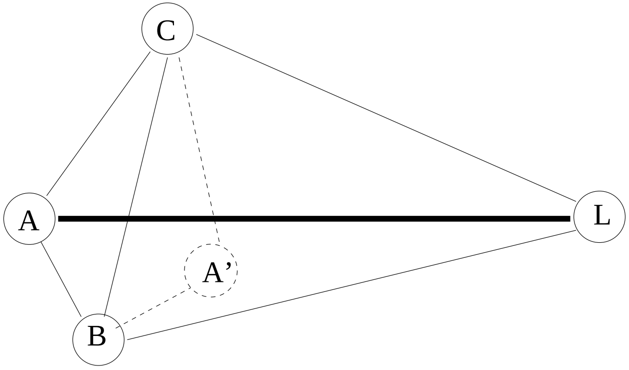
$\{X_i, Y_i, h_i\}$ and exchanges updates only with its neighbors. Once a landmark gets distances to other landmarks, it estimates an average size for one hop, which is then deployed as a correction to the entire network. When receiving the correction, an arbitrary node may then have estimate distances to landmarks, in meters, which can be used to perform the triangulation. The correction a landmark (X_i, Y_i) computes is

$c_i = \frac{\sum \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}}{\sum h_i}$, $i \neq j$, $all\, landmarks\, j$

In the example in figure 2, nodes L_1, L_2 and L_3 are landmarks, and node L_1 has both the euclidean distance to L_2 and L_3 , and the path length of 2 hops and 6 hops respectively. L_1 computes the correction $\frac{100+40}{6+2} = 17.5$, which is in fact the estimated average size of one hop, in meters. L_1 has then the choice of either computing a single correction to be broadcasted into the network, or preferentially send different corrections along different directions. In our experiments we are using the first option. In a similar manner, L_2 computes a correction of $\frac{40+75}{2+5} = 16.42$ and L_3 a correction of $\frac{75+100}{6+5} = 15.90$. A regular node gets an update from one of the landmarks, and it is usually the closest one, depending on the deployment policy and the time the correction phase of APS starts at each landmark. Corrections are distributed by controlled flooding, meaning that once a node gets and forwards a correction, it will drop all the subsequent ones. This policy ensures that most nodes will receive only one correction, from the closest landmark. When networks are large, a method to reduce signaling would be to set a TTL field for propagation packets, which would limit the number of landmarks acquired by a node. Here, controlled flooding helps keeping the corrections localized in the neighborhood of the landmarks they were generated from, thus accounting for nonisotropies across the network. In the above example, assume A gets its correction from L_2 – its estimate distances to the three landmarks would be: to $L_1, 3 \times 16.42$, to $L_2, 2 \times 16.42$, and to $L3, 3 \times 16.42$. This values are then plugged into the triangulation procedure described in the previous section, for A to get an estimate location.

The advantages of the *"DV-hop"* propagation scheme are its simplicity and the fact that it does not depend on measurement error. The drawbacks are that it will only work for isotropic networks, that is, when the properties of the graph are the same in all directions, so that the corrections that are deployed reasonably estimate the distances between hops.

Figure 3: *Euclidean* propagation method



4.3 "DV-distance" propagation method

This method is similar with the previous one with the difference that distance between neighboring nodes is measured using radio signal strength and is propagated in meters rather than in hops. As a metric, the distance vector algorithm is now using the cumulative traveling distance, in meters. On one hand the method is less coarse than *"DV-hop"*, because not all hops have the same size, but, on the other hand it is sensitive to measurement errors.

4.4 "Euclidean" propagation method

The third scheme works by propagating the true *euclidean* distance to the landmark, so this method is the closest to the nature of GPS. An arbitrary node A needs to have at least two neighbors B and C which have estimates for the landmark L (figure 3). A also has measured estimates of distances for AB, AC, and BC, so there is the condition that: either B and C, besides being neighbors of A, are neighbors of each other, or A knows distance BC, from being able to map all its neighbors in a local coordinate system.

In any case, for the quadrilateral $ABCL$, all the sides are known, and one of the diagonals, BC is also known. This allows node A to compute the second diagonal AL, which in fact is the euclidean distance from A to the landmark L. It is possible that A is on the same side of BC as L – shown as A' in the figure – case in which the distance to L is different. The choice between the two possibilities is made locally by A either by voting, when A has several pairs of immediate neighbors with estimates for L, or by examining relation with other common neighbors of B and C. If it cannot be chosen clearly between A and A', an estimate distance to L won't be available for A until either more neighbors have estimates for L that will suit voting, or more second hop neighbors have estimates for L, so a clear choice can be made. Once the proper choice for A is available, the actual estimate is obtained by applying Pithagora's generalized theorem in triangles ACB, BCL, and ACL, to find the length of AL. An error reduction improvement applicable for the "*Euclidean*" propagation, but not for the *"DV based"* methods is for a landmark to correct all the estimates it forwards. It uses the true, GPS obtained coordinates, instead of relying on the measurement based received values.

5 Simulation results

We simulated APS with the proposed propagation methods in ns-2, with randomly generated topologies of 100 nodes. The two main goals of ad hoc positioning are to get location for mapping

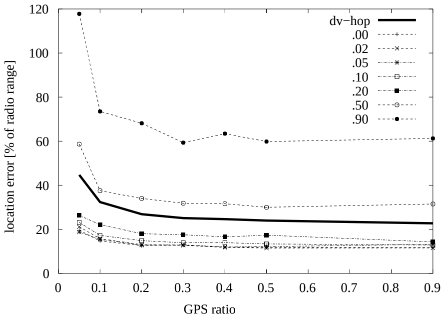


Figure 4: Location error - isotropic topology, *"DV-distance"*

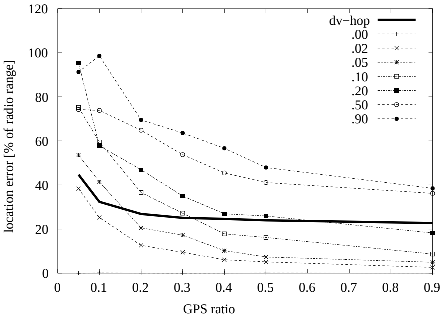


Figure 6: Location error - isotropic topology, *"Euclidean"*

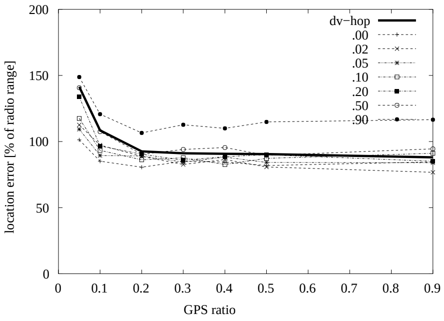


Figure 5: Location error - anisotropic topology, *"DV-distance"*

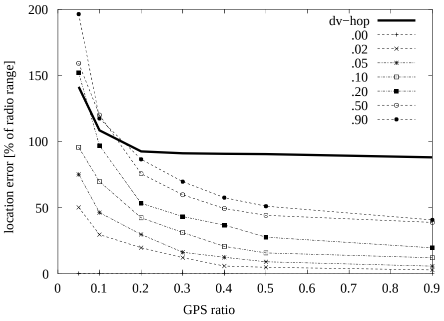


Figure 7: Location error - anisotropic topology, *"Euclidean"*

purposes, and to route using geodesic routing. The simulations evaluate the three possible propagation methods with respect to these goals. Two topologies are considered – an isotropic topology of 100 nodes, average node degree of 7.6, diameter 10, where nodes are placed in a random uniform manner, so that density, connectivity and communication range are approximately the same throughout the network. The second topology we examine is anisotropic in connectivity - it has the shape of letter "C", so that number of hops between the north and south branches is not a correct indication of geometric distance. This network has 100 nodes, maximum and minimum sections are 24 respectively 1 hop. All the performance graphs presented have the ratio of GPS enabled nodes on the x axis and several curves corresponding to error in signal strength evaluation of distance. This measurement error is considered to be in the range $2\% - 90\%$ of the nominal value, uniformly distributed. The *"DV-hop"* propagation method, being immune to measurement error, is represented as a thick line on both *"DV-distance"* and *"Euclidean"* graphs, for easier comparison of the three methods.

Figures 4 and 6 show location error in percents, relative to the hop size (100% error means one maximum sized hop away). While *"Euclidean"* has the advantage of increasing accuracy with GPS ratio, *"DV based"* algorithms are better suited for lower GPS ratios. Figures 5 and 7 show location error for the anisotropic topology. There are two things to notice: first, the corrections of the *"DV based"* methods are off because of the "C" shaped network, and this is reflected on the lower performance for this category. Second, for *"Euclidean"* measurement error does not make much difference compared to the anisotropy caused error. *"Euclidean"* performance has the advantage of small variation across different topologies, thus offering predictable performance across

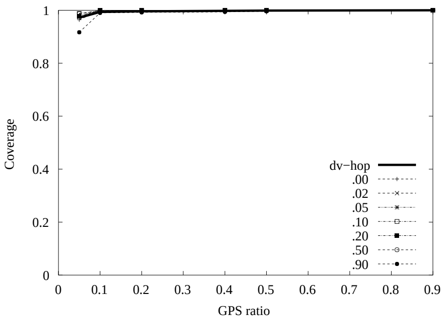


Figure 8: Coverage - isotropic topology, *“DV-distance”*

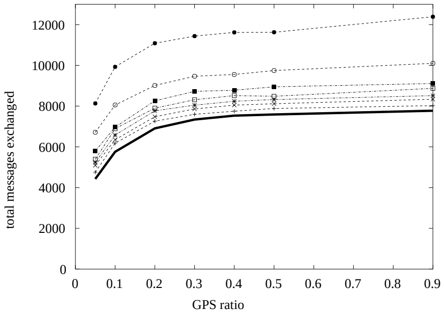


Figure 9: Messages exchanged - isotropic topology, *“DV-distance”*

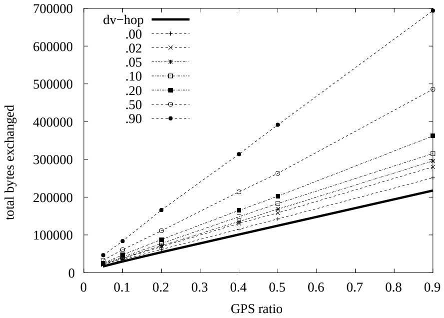


Figure 12: Bytes exchanged - isotropic topology, *“DV-distance”*

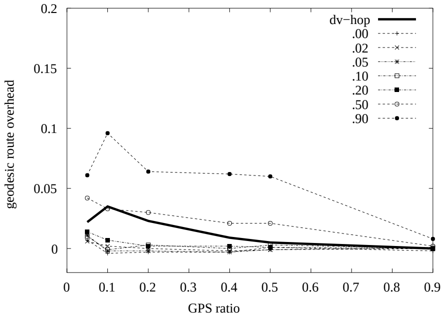


Figure 13: Routing overhead - isotropic topology, *“DV-distance”*

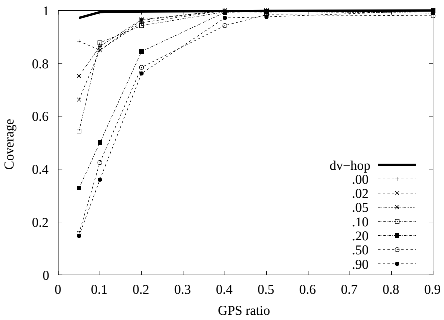


Figure 10: Coverage - isotropic topology, *“Euclidean”*

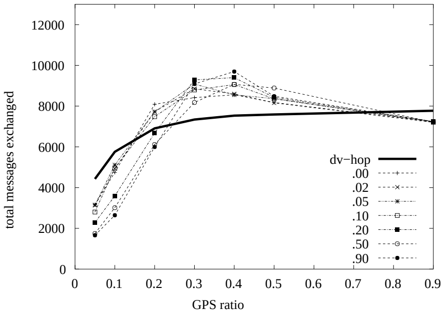


Figure 11: Messages exchanged - isotropic topology, *“Euclidean”*

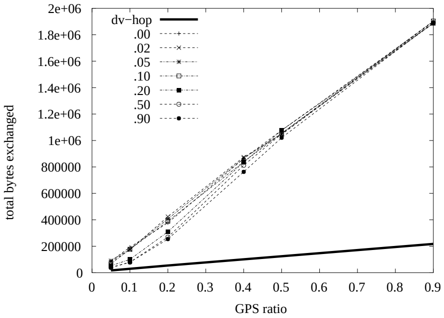


Figure 14: Bytes exchanged - isotropic topology, *“Euclidean”*

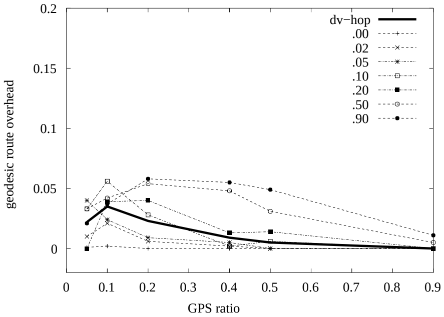


Figure 15: Routing overhead - isotropic topology, *“Euclidean”*

unpredictable conditions.

The way in which errors are propagating is the factor which determines which nodes can successfully estimate their location. Some nodes may not have an estimate due to not having at least three estimates to three noncolinear landmarks, or not attaining convergence during the iterative system solving. As seen from figures 8 and 10, when using *"DV-based"* algorithms, almost all nodes get estimates, even at low GPS ratios, whereas *"Euclidean's"* error build-up will produce some unreachable nodes. In practice, nodes uncovered by APS, can be approximated as the centroid of their neighbors, producing a location that can be used for both reporting and geodesic routing.

Message complexity is relevant because usually nodes communicate over a shared medium, and a high density of nodes, coupled with a high messaging complexity, leads to a high collision rate and ultimately to lower throughput and higher power consumption. Figures 9 and 11 show the number of messages exchanged under the three propagation policies. *"DV-distance"* is the only one spending more messages as the measurement precision decreases, and this is justified by the existence of several paths with similar metric in distance, which triggers more shorter paths updates. This does not happen for "*Euclidean*" because what is propagated is the straight line distance to the landmark, here there is no shorter path to be updated. A maximum number of messages is reached around the ratio of 40% GPS enabled nodes – because at higher densities, messages become larger and propagate more updates at once. At lower densities, there are more waves of smaller updates to be sent. Number of bytes exchanged is higher for *"Euclidean"* than for the *"DV-based"* algorithms by a factor depending the degree of a node, which can be seen in figures 12 and 14. This is due to the fact that *"Euclidean"* forwards second hop information, which increases the size of the average message.

To evaluate how effective the APS estimated locations are for purposes of routing, we implemented a simple, greedy version of geodesic routing. Having the coordinates (X, Y) of the packet destination, a forwarding node will choose as the next hop the neighbor that estimates the least euclidean distance to (X, Y). There are no routing loops because when all neighbors declare a larger distance than the forwarding node, the packet is dropped. This obviously works better for isotropic networks and this is the case that we simulated. The algorithm does not guarantee delivery, such algorithms are described elsewhere in the literature[10]. Figures 13 and 15 show the overhead in route length measured as the difference in the length of geodesic routes between using the true coordinates and the ones estimated by APS. The path overhead for all three proposed propagation methods is less than 6% and may be as low as 0.5% when using more precise measurements.

6 Node mobility

Although we have not explicitly modeled mobility, APS aims to keep a low signaling complexity in the event network topology changes. While highly mobile topologies, usually associated with ad hoc networks, would require a great deal of communication to maintain up to date location, we envision ad hoc topologies that do not change often, such as sensor networks, indoor or outdoor temporary infrastructures. When a node moves, it will be able to get *"DV-based"* or "*Euclidean*" updates from its new neighbors and triangulate to get its new position, therefore communication remains localized to nodes that are actually mobile. This is in contrast with previously proposed solutions [5], which rely on a reference group that would prompt reevaluations in the entire network in case of movement of the reference group. Not even moving landmarks would cause a communication surge in our approach because the only things that identify a landmark are its coordinates. In fact, a moving landmark would provide more information to the APS algorithm, as the new position of the landmark acts as a new landmark for both mobile and fixed nodes. To refer again to the sensor network example, we can envision a case when a single, fly-over GPS enabled node is in fact enough for an entire network. Later mobility of the network is supported as long as a sufficient fraction of nodes remains fixed at any one time to serve updates for the mobile nodes.

7 Conclusion

We presented APS(Ad hoc Positioning System), a method to extend the capabilities of GPS to non-GPS enabled nodes in a hop by hop fashion in an ad hoc network. Positioning is based on a hybrid method combining distance vector like propagation and GPS triangulation to estimate location in presence of signal strength measurement errors. APS has the following properties: is distributed, does not require special infrastructure or setup, provides global coordinates and requires recomputation only for moving nodes. Three propagation methods were investigated, each providing a different tradeoff between accuracy, signaling complexity, coverage and the isotropy of the network. *"DV-based"* algorithms behave well for most purposes and have a low signaling complexity. "*Euclidean*" provides better accuracy for nonisotropic topologies, and is generally more predictable in performance, at the cost of more communication. Actual locations obtained by APS are on average less than one radio hop from the true location. Positions produced by APS are usable by geodesic and geographic routing algorithms, producing paths within 6% of the paths produced with the real locations.

References

[1] I. Stojmenovic and X. Lin, "GEDIR: Loop-free location based routing in wireless networks," *Proc. IASTED Int. Conf. on Parallel and Distributed Computing and Systems*, Nov. 3-6, 1999, Boston, MA, USA, 1025-1028.

[2] J. C. Navas and Tomasz Imielinski, "Geographic addressing and routing," *MobiCom'97*, Budapest, Hungary, September 26-30 1997.

[3] L. Doherty, "Algorithms for Position and Data Recovery in Wireless Sensor Networks," Master's Report, University of California, Berkeley, June, 2000.

[4] J. Hill et al., "System architecture directions for networked sensors," *ASPLOS-IX*, November 2000, Cambridge, MA

[5] S. Capkun, M. Hamdi and J.P. Hubaux, "GPS-free positioning in mobile Ad-Hoc networks," *Hawaii International Conference On System Sciences*, HICSS-34 January 3-6, 2001 Outrigger Wailea Resort.

[6] B. Parkinson et al., "Global Positioning System: Theory and Application," Volume I, Progress in Astronautics and Aeronautics, Volume 163, 1996.

[7] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan, "The Cricket Location-Support System", *Proc. 6th ACM MOBICOM*, Boston, MA, August 2000.

[8] Nirupama Bulusu, John Heidemann, Deborah Estrin, "GPS-less Low Cost Outdoor Localization For Very Small Devices", *IEEE Personal Communications Magazine*, Special Issue on Smart Spaces and Environments, October 2000.

[9] Paramvir Bahl and Venkata N. Padmanabhan, "RADAR: An In-Building RF-based User Location and Tracking System," *Proceedings of IEEE Infocom 2000*, Tel-Aviv, Israel March 2000.

[10] P. Bose, P. Morin, I. Stojmenovic and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks", *3rd int. Workshop on Discrete Algorithms and methods for mobile computing and communications*, Seattle, August 20, 1999

[11] Y.-B. Ko and N. H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks", In *MobiCom'98*, October 1998