Position Estimation Error in Edge Detection for Wireless Sensor Networks using Local Convex View

Marwan Fayed
SITE, University of Ottawa
Ottawa, Canada
Email: mmf@site.uottawa.ca

Hussein T. Mouftah
SITE, University of Ottawa
Ottawa, Canada
Email: mouftah@site.uottawa.ca

***Abstract*—Intuitively, identification of nodes close to the network edge is key to the successful setup, and continued operation, of many sensor network protocols and applications. In a previous study [1] we introduced local convex view (*lcv*) as a means to identify nodes close to the network edge by computing the convex hull of nodes within range. In this paper we evaluate *lcv* in the presence of position estimation error. Extensive simulations with networks of varying size and topology reveal the surprising observation that *lcv* seems unaffected by estimation error. Motivated by this observation we enumerate a complete set of base node configurations seen by *lcv*. An analysis reveals that *lcv* is immune to two of these configurations. Further simulations show the frequency of false-positives and false-negatives imposed by a third, ambiguous, configuration to be low. The frequency of the ambiguous case is 10% in the worst case, for all networks tested. We conclude that the geometric properties underlying *lcv* are responsible for its resilience to error.**

I. Introduction

An appropriate level of context-awareness is necessary for the success of many wireless networks. Intuitively, identification of nodes close to the network edge is key to the successful setup, and continued operation, of many sensor network protocols and applications [2], [3], [4], [5], [6], [7], [8]. Currently, identification of nodes along the network edge remains a challenging problem.

In previous work we presented the local convex view (*lcv*) as a means to identify nodes close to the network edge [1]. The local convex view is defined as the convex hull of the nodes within view. (Thus, as many local convex views exist as nodes in the network.) In the simplest terms a node decides it is close to the network boundary by asking the following question: Am I in my own local convex view? It is a heuristic inspired by a geometric construct, the $\alpha-$hull, which is able to capture the 'shape' of a set of nodes [9]. Furthermore, it is motivated by the hypothesis that within view of many nodes there exists structural information relevant to the network.

If node position information is unavailable *lcv* assigns local coordinates using a node's 1-hop distance measurements and the 1-hop measurements of its neighbours. Each node constructs a Cartesian space by placing itself at the origin and its furthest neighbour along the horizontal axis. A third, witness, node in range of the origin node and its furthest neighbour is then placed appropriately. (Convex hull computations are unaffected by translations and rotations of a set of node locations.) *lcv* relies on triangulation to position remaining nodes, though any localisation scheme suffices. In our previous study we found *lcv* performance to be consistent, and resilient to some of the qualitative drawbacks of other similar methods.

In this paper we evaluate *lcv* under the assumption that position estimation is erroneous. A growing number of projects suggest error in position estimation is currently unavoidable [10], [11], [12]. Our work is propelled by these studies. We expected *lcv* to be adversely affected by position estimation errors. Observations and analysis show this may largely be untrue.

In our evaluation we simulate networks of varying size and density. We vary topology by generating networks with node locations selected from uniform, normal, and skewed (Pareto) distributions. Position error is added to the system by blurring neighbour positions stored at each node. In doing so, a pair of neighbouring nodes is unlikely to agree on the position of any shared node in common view. (Recall that each node composes its own coordinate system independent of its neighbouring coordinate systems.) We add to each node position an error vector composed of a uniformly random angle, and a length chosen from the normal distribution with a variance ranging from 5-20% of the communication range.

The results are surprising and counter-intuitive. Using two metrics, results show that the difference in accuracy between the *lcv* in a perfect environment and one with position estimation error are statistically insignificant. Motivated by this observation we enumerate a complete base set of node configurations that may be seen by *lcv*. Our analysis reveals that *lcv* is immune to two of these configurations. Further simulations show the frequency of false-positives and false-negatives imposed by a third, ambiguous, configuration is low. The frequency of the ambiguous configuration is 10% in the worst case, for all networks tested. We conclude that the geometric properties underlying *lcv* are responsible for its resilience to error.

Our paper develops as follows. A brief discussion of related works in Section II precedes an overview of the *lcv* algorithm in Section III. Where the organization of our paper differs from most is in its presentation of simulation results in Section IV before the analysis, prompted by the results, in Section V. We conclude with remarks in Section VI

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

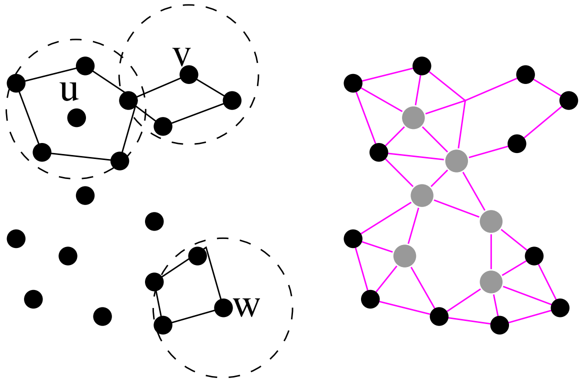


Fig. 1. (Left) Nodes u, v, w compute their local convex hull; only v and w declare *lcv* status. (Right) The black nodes sit on their local convex hulls and closely approximate the network edge.

II. Related Work

For a discussion of *lcv* in the context of the existing body of research, we direct our reader to [1]. We focus instead on edge detection algorithms in the presence of error. To our knowledge this is the first study that deals explicitly with edge node identification in the presence of position estimation error. We classify algorithms as statistical, topological, or geometric, according to the taxonomy first presented in [13].

Statistical solutions assume that network structure exhibits unique characteristics at the network edge [14], [15]. Their success relies on global communication and high network densities. It is unclear whether statistical methods might suffer in the presence of estimation error. Topological solutions [6], [16], [17] exploit the connectivity of the network to identify nodes along the network edge. We expect that their use of connectivity would render them immune to estimation errors. However, topological solutions are generally expensive and centralised in nature. Geometric solutions [1], [18] such as *lcv* use node positions to reveal structural information. This class of solution is simple and localised but also most vulnerable to position errors.

III. Overview of Local Convex View

In a previous study we investigated the local convex view (*lcv*) as a method to identify a subset of nodes that describe the network edge in a wireless network [1]. In this study we extend our previous work to investigate *lcv* in the presence of position estimation error. While not the main contribution of this work, we summarize the *lcv* here for completeness.

Our method relies on the hypothesis that within the local view of each node there exists some structural information. Our algorithm makes only the assumptions that generated or assigned node IDs within each neighbourhood are unique, and that distance measurements are available. In this study we look for real-world applicability by assuming erroneous position estimation. We consider a deployment of a large wireless network where, initially, nodes lack any knowledge of their positions.

The local convex view is inspired by $\alpha-$hulls [9], a geometric construct that captures the "shape" of a set of points. We approximate the $\alpha-$hull by computing convex hulls locally. The idea is illustrated in Figure 1 where we compute the local convex view for each node. A node's *lcv* consists of the set of nodes that comprise the convex hull of the neighbourhood in view. (Thus there are as many local convex views in a network as there are network nodes.) Observe in Figure 1 that by taking the set of "outside" nodes from the set of local convex views, the network edge begins to emerge.

The complete process is reproduced in Algorithm 1. A detailed discussion of its steps, its correctness, and a simple probabilistic model to contend with missing information may be found in [1]. In this study we seek insight into the viability of *lcv* in a practical setting by assuming that distance measurements in Step 1 are erroneous. In [1] Step 2 describes a localization process where neighbour positions are triangulated relative to other nodes in the local neighbourhood. (This process creates a local coordinate system sufficient for the purpose of computing the local convex view.) Triangulation, in the presence of inaccurate distance measurements, may fail to resolve node locations. For this reason *lcv* allows for any localisation scheme to be substituted in Step 2.

Algorithm 1 Boundary Node Identification Algorithm at any node u.

1: Share the distance measurements to single hop neighbours.
2: Set u as the origin of, and construct, local coordinate system.
3: Compute the local convex view (lcv).
4: **return** $u \in lcv$

IV. Simulation Results

We evaluate the performance of *lcv* when error is introduced in simulated networks of varying density and distribution. Network nodes are distributed in a 200x200 unit space, each node with a fixed range of 8 units. We vary node density by changing the network size. Note that by changing size instead of communication range we can vary the density without affecting the maximum network diameter. Network sizes are 1500, 2500, and 3500 nodes. (In the uniform networks this results in average neighbourhood sizes of 7, 12, and 17 nodes.) We tabulate and experiment over the largest connected component of each network. Experiments repeat over 25 runs of each network generated using non-overlapping random streams.

Node locations are chosen from a normal or skewed (Pareto) distribution in addition to the uniform distribution traditionally used to generate wireless network topologies. Uniformly distributed networks may be sufficient to provide insight yet are poor representations of many real deployments. Normal coordinates are generated with an average of 100 (the network center) and a standard deviation of 40. Skewed coordinates are chosen from the Pareto distribution with scale parameter 1.0

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

and shape parameter 100.5. Graphical representations of these networks may be found in [1].

Error is added to the system by blurring the position of nodes from their actual locations. This blurring occurs from the perspective of each node so that two nodes may see a common neighbour in two different positions. Before computing the local convex view, each coordinate is shifted. We shift coordinates by adding a vector consisting of an angle chosen from the uniform distribution, and a length chosen from a parametrized normal distribution. We use the edge proximity and regional proportionaly metrics from [1] to evaluate the efficacy of the *lcv* method in the presence of error.

A. Edge Proximity

In this section we measure the proximity of *lcv*-nodes to the edge of the network. *Edge proximity* is defined as the probability that the location X of an *lcv*-node lies in region x. This gives the likelihood that nodes sitting on their local convex view also sit close to the network boundary.

Edge proximity plots appear in Figure 2. Subfigures are organised such that network size and density changes across each row, while the underlying network distribution varies down each column. For each network we plot the edge proximity for varying error values, parametrized by increasing the variance from 0 to 20% of the communication range. Each plot represents the cumulative distribution over partitions of the network. Networks are partitioned in a manner appropriate to the underlying distribution, according to the following criteria.

- Uniform networks are partitioned into quadrilateral 'rings'. Each ring is of width equivalent to $0.25R$, where R is the communication range.
- Normal networks are partitioned into rings that are 0.25 standard deviations in width. The statement "80% of reporting nodes sit outside 2σ" may be interpreted as 80% of reporting nodes sit amongst the outermost 5% of network nodes.
- Skewed networks are partitioned into diagonals that span from the $y-$ to the $x-$ axis, 10 units apart. So, for example, nodes that report in the 180 region have x and y coordinates with a sum greater than 180.

We first summarize from [1] the performance of *lcv*, in general. In uniformly random networks the *lcv* method is greatly affected by density. Among normal networks depicted in Figures 2d- 2f, approximately 80% of *lcv* nodes report among the furthest 5% of network nodes. Similarly, convergence to 1 in skewed networks shown in Figures 2g- 2i occurs quickly. As network sizes increase in normal and skewed networks the curves shift to the left, indicating the point at which the cumulative distributions converge to 1 occurs further from the network center. One important note: in normal and skewed networks the network edge physically occurs further from the origin as the network grows large. Therefore the conclusion that increases in network size are responsible for an increase in accuracy, and hence a shift of the curves to the left, should be avoided.

With respect to the effect of error on the performance of *lcv* we find the observations to be somewhat counter-intuitive. Within each subfigure, each curve represents a different degree of error. Curves within each subfigure show identical trends with differences in accuracy that are largely statistically insignificant. (Confidence intervals have been omitted for clarity.) This would indicate that, in all tested networks, the accuracy of *lcv* is largely unaffected by error. We reserve a discussion of the causes for Section V.

We proceed in the next section with an evaluation using a second metric to confirm our observations.

B. Regional Proportionality

We find that, using the edge proximity metric, *lcv* seems largely unaffected by errors in position estimation. In this section we seek further insight by evaluating regional proportionality. *Regional proportionality* is the proportion of nodes that declare *lcv* status within each partition as described in Section IV-A, versus those nodes that claim not to be *lcv*. We call this the *lcv*- vs. regular-node ratio.

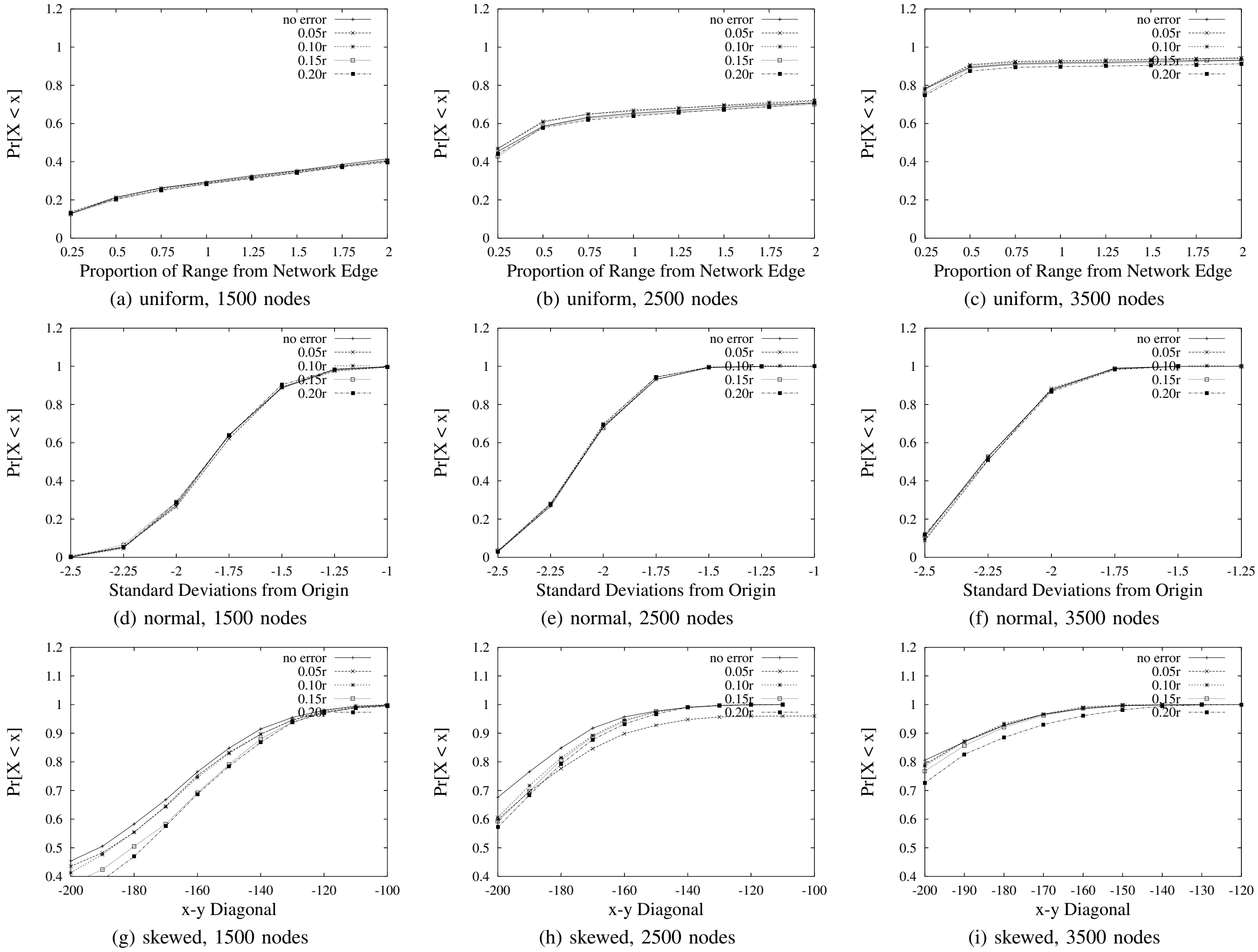
We plot regional proportionality in Figure 3. Subfigures are organised such that network size and density changes across each row, while the underlying network distribution varies down each column. For each network we plot the edge proximity for varying error values, parametrized by increasing the variance from 0 to 20% of the communication range. Note the leftward shift of curves as size increases among normal and skewed networks. Recall from previous that this is an artifact of the network's edge shifting further from the origin as the networks grow.

Within each subfigure we can compare the curves against the network-wide proportion of *lcv*-nodes, represented by the horizontal line. The network average permits a clearer interpretation of the results. For example, in Figure 3a the network-wide proportion of *lcv*-nodes is 0.17. From the same figure we see that the proportion of *lcv*-nodes in the outer-most $0.25R$ region is 0.64 when there is zero error. We can conclude that, with no error, nodes in the outer-most ring are almost 4 times as likely to identify with the edge of the network.

The regional proportionality metric seems to reinforce the observation that error, as tested, has little-to-no effect on the performance of *lcv*. Similar to the edge proximity metric in Section IV-A, plots within each subfigure show identical trends with differences in accuracy that are largely statistically insignificant. However, there are subtle noteworthy observations. We refer our reader first to plots derived from uniformly generated networks in Figures 3a- 3c. We can see that error has a more pronounced effect on *lcv* accuracy as the network density increases, but only in the outer-most region of the network.

We emphasize that the differences in accuracy, statistically speaking, are insignificant - with one exception. A dense uniformly generated and bounded network will eventually capture the shape enforced by the bounds. In our experiments this shape is a quadrilateral. It is directly responsible for the

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.



(a) uniform, 1500 nodes (b) uniform, 2500 nodes (c) uniform, 3500 nodes

(d) normal, 1500 nodes (e) normal, 2500 nodes (f) normal, 3500 nodes

(g) skewed, 1500 nodes (h) skewed, 2500 nodes (i) skewed, 3500 nodes

Fig. 2. Edge proximity distributions reveal the proximity of *lcv*-nodes to the network edge. The value r represents the communication range.

loss in accuracy in the outer-most ring of the network, as density increases. We develop this idea next in Section V.

V. Discussion of Results

The previous section revealed little-to-no degradation in the performance of *lcv* in the presence of errors. This idea is counter-intuitive, and so we use this section to enumerate and discuss the scenarios faced by the local convex view method.

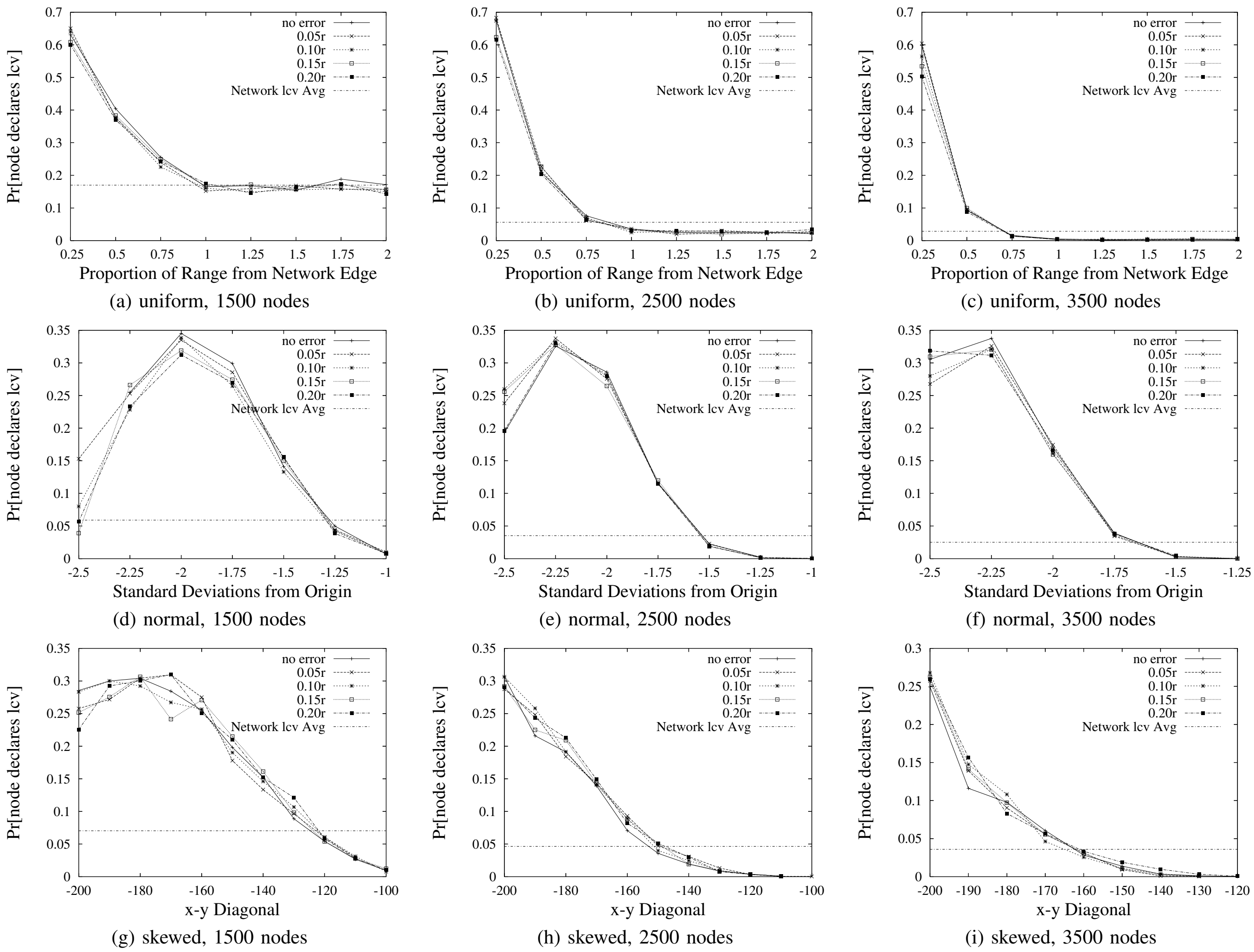
In our analysis we assume that the position of only a single node has been incorrectly estimated. This relatively benign assumption permits a clear demonstration of the effects of error on *lcv* without sacrificing accuracy or completness. Specifically, all remaining cases may be composed of the cases presented here.

The three cases under consideration by *lcv* are presented in Figure 4. For the purpose of demonstration, we consider the *lcv* operation at node u. In our example topologies, neighbours are joined to u with a solid line. The position of some neighbouring node v determines a dash-dot-dashed line that represents a threshold of interest. The neighbour in question has a position estimated by the node labeled w, with a true position that may exist anywhere inside the greyed region. Finally, the dashed poly-line corresponds to the local convex hull under consideration.

Figures 4a and 4b depict the two 'good' cases where the *lcv* computation is unaffected by error. In the first case, shown in Figure 4a, node u determines it is on the local convex view and declares itself close to the network boundary. Note that the local convex hull consists of the same nodes irrespective of the actual location of w anywhere inside the grey region. The second case shown in Figure 4b is of a similar theme. Node w is estimated to have a location that renders node u inside its local convex view. In fact, node w may sit anywhere in the grey region without affecting the local convex view. In both these cases the underlying geometry ensures the resilience of the local convex view method: the convex hull remains consistent so long as the error region of w remains entirely to one side or the other of the threshold determined by (u, v).

Figure 4c depicts the ambiguous case. Node w is estimated to have a position close enough to the threshold that its actual location may exist on either side of the threshold. In the example shown in Figure 4c, a w is estimated to have a

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

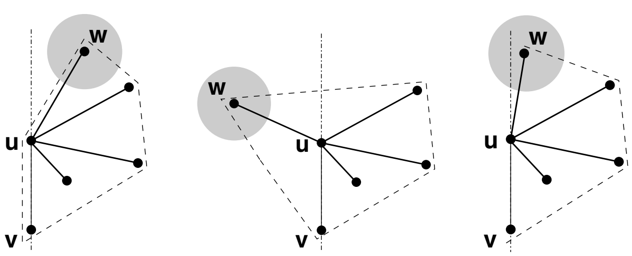


(a) uniform, 1500 nodes (b) uniform, 2500 nodes (c) uniform, 3500 nodes

(d) normal, 1500 nodes (e) normal, 2500 nodes (f) normal, 3500 nodes

(g) skewed, 1500 nodes (h) skewed, 2500 nodes (i) skewed, 3500 nodes

Fig. 3. Regional proportionality reveals the proportion of nodes in each region to declare edge-node status via *lcv*.



(a) Consistent (b) Consistent (c) Inconsistent

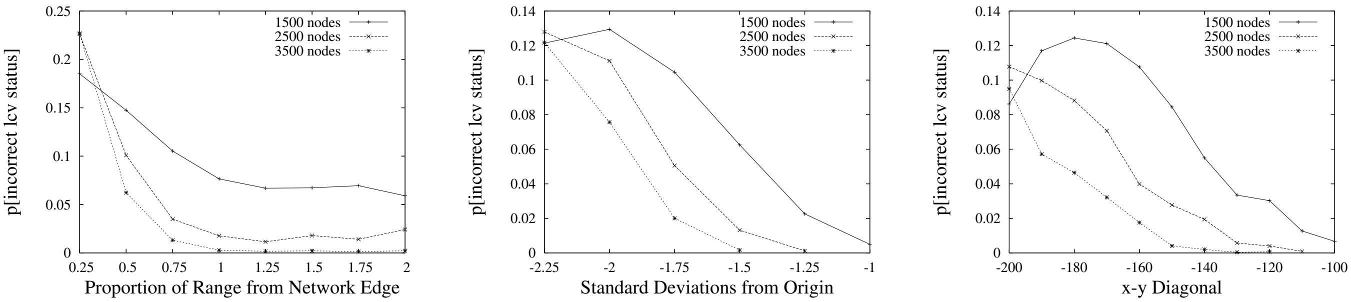
Fig. 4. In consistent cases *lcv* is unaffected by position estimation error.

position that renders u on the local convex view of u. If such a w actually sits on the the other side of the threshold then u has falsely determined that it sits on its local convex view. The reverse may occur if w is estimated to sit close to, but on the other side of, the threshold.

The observation in Section IV is that *lcv* seems relatively unaffected by error. From our analysis we have determined that any adverse effect of error to *lcv* is caused by the ambiguous scenario demonstrated in Figure 4c. Our hypothesis is that *lcv* is relatively unaffected by error because the ambiguous case occurs very infrequently. To test our hypothesis we evaluate the frequency of false positives and false negatives when error is added to *lcv*. For each type of network the results are partitioned as described in Section IV-A so that we may observe *lcv* performance in each area of the network. We plot for all networks the worst tested case in Figure 5, where the variance parameter is equal to 20% of the communication range.

Figure 5a reveals that, in uniform networks, the error in the outer-most ring of the network hovers about 20%. Interestingly, the frequency of false positives and negatives in this region climbs as density increases. The reason is that increased densities along the network edge more closely approximate the artifical lines artificially bounding the network. This causes a greater number of w nodes which are the cause of the ambiguity that leads to false positives and negatives. As we move deeper into the network where no artificial boundaries exist, the frequency of incorrect responses drops dramatically.

In the outer-most regions of normal and skewed networks, the ambiguous case appears much less frequently. For example, among the outer-most 5% of nodes in normal networks,



(a) Uniform (b) Normal (c) Skewed

Fig. 5. Frequency of *lcv* false positives and false negatives in the worst tested case, with error variance 20% of communication range.

the rate of false *lcv* responses is approximately 10%. Similarly for skewed networks. This may be explained by the lack of artificial boundaries in normal and skewed networks (ie. unlike the uniformly generated networks, these networks fail to approximate the quadrilateral region that contains them). As we move deeper into the network the rate of false positives quickly drops in both normal and skewed networks.

VI. Conclusion

In this paper we examined the ability of the local convex view (*lcv*) algorithm to identify network edge nodes in the presence of position estimation error. We engaged in extensive simulations of networks with topologies of varying size and underlying distributions. Position errors were chosen from a normal distribution with a variance up to 20% of the communication range. We assumed that an increase in estimation error would adversely reduce the accuracy of *lcv*. Observations failed to reinforce our assumption. To explain the disconnect between intuition and observation we enumerated and analysed the three base neighbour configurations that may be seen by a node. We show that the *lcv* computation is immune to errors in two of the cases. In both cases position estimation error changes the shape of the local convex view, but not the nodes that comprise it. In the third case position error leads to ambiguity, where the true position of a neighbour may lead to a false insertion or an omission of the node undergoing the *lcv* computation. Further simulation revealed the frequency of the ambiguous case to be very low, about 10% in the worst case for all networks tested. We conclude that the geometric properties underlying *lcv* are responsible for its resilience to error.

References

[1] M. Fayed and H. T. Mouftah, "Network boundary identfication using local information," University of Ottawa, Tech. Rep. TR-2007-03, 2007. [Online]. Available: http://www.site.uottawa.ca/school/publications/techrep/2007/index.shtml

[2] M. Ben-Chen, C. Gotsman, and S. Gortler, "Routing with guaranteed delivery on virtual coordinates," in *Proceedings of the 18th Canadian Conference on Computational Geometry (CCCG'06)*, 2006.

[3] Q. Cao and T. Abdelzaher, "A scalable logical coordinates framework for routing in wireless sensor networks," *Proceedings of the 25th IEEE Real-Time Systems Symposium (RTSS)*, pp. 349–358, 2004.

[4] Q. Fang, J. Gao, L. J. Guibas, V. de Silva, and L. Zhang, "Glider: gradient landmark-based distributed routing for sensor networks." in *Proceedings of the 24th Annual IEEE Conference on Computer Communications (INFOCOM)*, Miami, FL, USA, 2005, pp. 339–350.

[5] R. Fonseca, S. Ratnasamy, J. Zhao, C. T. Ee, D. Culler, S. Shenker, and I. Stoica, "Beacon vector routing: Scalable point-to-point routing in wireless sensornets," in *Proceedings of the 2nd USENIX Symposium on Networked Systems Design and Implementation (NSDI '05)*, Boston, MA, USA, May 2005.

[6] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic routing without location information," in *Proceedings of ACM MobiCom*, San Diego, CA, September 2003.

[7] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing (MobiHoc)*, New York, NY, USA, 2003, pp. 201–212.

[8] Y. Zhao, B. Li, Q. Zhang, Y. Chen, and W. Zhu, "Efficient hop id based routing for sparse ad hoc networks," in *Proceedings of the 13TH IEEE International Conference on Network Protocols (ICNP)*, 2005.

[9] H. Edelsbrunner and E. P. Mcke, "Three-dimensional alpha shapes," *ACM Transactions on Graphics*, vol. 13, no. 1, pp. 43–72, 1994.

[10] D. Niculescu and B. Nath, "Error characteristics of ad hoc positioning systems (aps)," in *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, New York, NY, USA, 2004.

[11] A. Savvides and W. L. Garber, "An analysis of error inducing parameters in multihop sensor node localization," *IEEE Transactions on Mobile Computing*, vol. 4, no. 6, pp. 567–577, 2005, senior Member-Randolph L. Moses and Senior Member-Mani B. Srivastava.

[12] K. Whitehouse, C. Karlof, A. Woo, F. Jiang, and D. Culler, "The effects of ranging noise on multihop localization: an empirical study," in *Proceedings of the 4th international symposium on Information processing in sensor networks (IPSN)*, 2005.

[13] Y. Wang, J. Gao, and J. S. Mitchell, "Boundary recognition in sensor networks by topological methods," in *Proceedings of the 12th annual international conference on Mobile computing and Networking (MOBICOM)*, 2006, pp. 122–133.

[14] S. P. Fekete, A. Kröller, D. Pfisterer, S. Fischer, and C. Buschmann, "Neighborhood-based topology recognition in sensor networks," in *Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS)*, vol. 3121. Springer, 2004, pp. 123–136.

[15] S.P. Fekete and M. Kaufmann and A. Krller and N. Lehmann, "A new approach for boundary recognition in geometric sensor networks," in *Proceedings 17th Canadian Conference on Computational Geometry (CCCG)*, 2005, pp. 82–85.

[16] S. Funke and C. Klein, "Hole detection or: "how much geometry hides in connectivity?"," in *Proceedings of the* 22^{nd} *annual symposium on Computational Geometry (SGC)*, 2006, pp. 377–385.

[17] A. Kröller, S. P. Fekete, D. Pfisterer, and S. Fischer, "Deterministic boundary recognition and topology extraction for large sensor networks," in *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm (SODA)*, New York, NY, USA, 2006, pp. 1000–1009.

[18] Q. Fang, J. Gao, and L. Guibas, "Locating and Bypassing Routing Holes in Sensor Networks," in *Proceedings of IEEE/ACM Infocom*, Hong Kong, China, March 2004.