Analytical Characterization of Computationally Efficient Localization Techniques

S. Alireza Motevallian, Guoqiang Mao, Brian D.O. Anderson

Abstract—Trilateration-based localization techniques have been widely used in sensor networks due to their computational efficiency and distributedness. However in sparse networks or in the boundary area of networks, trilateration-based techniques often fail to localize all localizable nodes. Bilateration-based techniques emerge as a generalization of trilateration techniques to a broader class of networks. Compared with trilateration-based techniques, the main benefit of bilateration-based schemes is that they can localize a higher percentage of nodes while still maintaining the low computational complexity and distributedness properties. One potential drawback of bilateration-based schemes is that the number of estimated possible positions (hence the memory required to store these positions) may grow exponentially with the number of nodes in the network. Despite the empirical observations reported in the literature that such exponential growth is a rare event, there is a lack of rigorous analysis quantifying the complexity of bilateration-based schemes. In this paper, we tackle the challenge by first characterizing a broad subclass of the set of *critical sub-networks* within which the number of possible estimated positions grows exponentially with the size of these sub-networks. Then using mathematical techniques from percolation theory, we prove that, in random geometric networks, with very high probability the size of these critical sub-networks, which constitute the worst case for bilateration-based localization, is bounded. Therefore the complexity of bilateration-based localization technique does not grow exponentially with the size of the entire network. The significance of this result is to analytically demonstrate that bilateration-based techniques not only localize a higher fraction of nodes than their trilateration counterpart, but also they can be implemented in a very efficient (low computational cost) manner.

Index Terms—Distributed Localization, Bilateration, Trilateration

I. INTRODUCTION

Range-based localization techniques are being increasingly used in wireless sensor networks, vehicular networks and cellular networks etc. to provide important localization information. In a typical setting, such techniques estimate the location of each node by using the known positions of a few nodes (called *anchors*) and some inter-sensor distance measurements [11]. The network being considered can usually be modeled by its *grounded graph* $G = (V, E)$ where for each node in the network there is a corresponding vertex v in V and

S. A. Motevallian and Brian D.O. Anderson are with the Australian National University and National ICT Australia. Emails: {alireza.motevallian,brian.anderson}@anu.edu.au

G. Mao is with the School of Electrical and Information Engineering, the University of Sydney and National ICT Australia. Email: guoqiang.mao@sydney.edu.au

$(u, v) \in E$ if the distance between the nodes corresponding to u, v is known. The distance between the nodes corresponding to vertices u, v is shown by $d(u, v)$.

The computational complexity of a range-based localization algorithm is, in general, exponential in the number of sensor nodes [4]. However for networks with certain topology, computationally efficient localization algorithm, where the complexity increases only polynomially with the number of nodes, can be designed. More specifically, it is well known that if the grounded graph of a network is *trilaterative*, i.e. has a *trilateration ordering* where the first three vertices induce a triangle and every other vertex is connected to at least *three* nodes preceding it in the ordering, a *sequential localization* algorithm [1] known as *trilateration* can be designed that localizes nodes sequentially using the previously estimated positions of nodes ahead in the ordering and distance measurements to these nodes, and with a computational complexity increasing only polynomially with the number of nodes. The computational efficiency and distributedness of trilateration-based localization algorithms have made them the most popularly used range-based localization algorithms [5], [6], [8]–[10]. Further, it is known that if a network has three anchors and the grounded graph of the network is trilaterative, the network is *localizable*. A network is said to be localizable if there is a *unique* set of node positions that are consistent with the known positions of anchors and distance measurements. Note that the ground graph being trilaterative is a sufficient but *not necessary condition* for the network to be localizable. Consequently, trilateration-based localization algorithms often fail to localize all localizable nodes, particularly in sparse networks or in the boundary area of networks where there are comparatively few number of distance measurements.

Bilateration-based techniques emerge as a generalization of trilateration techniques that are applicable to a broader class of networks. In bilateration-based techniques, a reduced number of distance measurements is required. Particularly, bilateration-based techniques can be implemented in the class of *bilaterative* networks, where a bilaterative network is a network whose grounded graph has a *bilaterative ordering*: a vertex ordering in which the first three vertices induce a triangle and every other vertex is connected to at least *two* nodes preceding it in the ordering. Compared with trilateration-based techniques, the main benefit of bilateration-based schemes is that they can localize a higher percentage of nodes while still maintaining the low computational complexity and distributedness properties. This benefit is obtained at the expense of an increasing number of possible estimated positions generated in the course of implementing the bilateration-based algorithm and hence a larger amount of memory required to store these possible estimated positions. More specifically, consider a special case where a network with n nodes with a bilaterative ordering v_1,

$v_2,...,v_n$ and each node v_i, $i > 3$, has distance measurements to *exactly two* previous nodes in the ordering, i.e. v_{i-1} and v_{i-2}. Let v_1, v_2 and v_3 be the anchor nodes. It is easy to establish that given two distance measurements to v_2 and v_3 only, v_4 has two possible positions. Subsequently, given the position of v_3 and *one* estimated position of v_4, v_5 has two possible positions. Since v_4 has two possible positions, v_5 has four possible positions. Continuing the above process iteratively, it can be established that v_i, $i > 3$, has at least 2^{i-3} possible positions. Therefore the number of possible positions in bilateration-based localization algorithms may *possibly* increase exponentially with the number of nodes. In reality however, many possible positions can be eliminated when extra information is available, e.g. more distance measurements than two are available, when the sequential localization process meets an anchor, the information that two nodes are *not* neighbors of each other also allows the removal of some possible positions [4]. In fact, in the practical implementation of bilateration-based localization algorithms, it was reported *empirically* that the maximum number of possible positions that may occur during the localization process is less than 256 with 99.5% chance [3].

An accurate analysis on the maximum number of possible positions that may occur in the course of bilateration-based localization process is important because it decides the computational complexity of bilateration-based localization algorithms which in turn determines whether the benefit of bilateration-based localization algorithms (localizing more nodes compared with trilateration-based techniques) outweighs its drawback (increased computational complexity).

In this paper, we tackle the above challenge by presenting an analytical study on the maximum number of possible positions that may occur in the course of bilateration-based localization process in a class of networks modeled by random geometric graphs. The main contributions of the paper are:

- First, we define the class of *critical topologies*, in which the maximum number of possible positions, that may occur in the course of bilateration-based localization process, grows exponentially with the number of nodes in the graph. Then we characterize a broad subclass of critical graphs which commonly occur in random geometric networks.
- Second, by resorting to techniques from *percolation* theory [2], we show that *asymptotically almost surely* the size of sub-networks whose grounded graphs have critical topologies, is bounded and does not increase with the number of nodes in the entire network.
- By combining the above two results, we show that in large networks, the complexity of bilateration-based localization techniques does not grow exponentially with the size of the entire network.

The significance of the above results are that they analytically demonstrate that bilateration-based techniques not only localize a higher fraction of nodes than their trilateration counterpart, but also they can be implemented in a very efficient (low computational cost) manner. Therefore they provide strong support for the use of bilateration-based localization techniques.

The organization of the paper is as follows: Section II introduces the network model. Section III contains the formal statement of the problem, the solution analysis and simulation results. Concluding remarks appear in Section IV.

II. NETWORK MODEL

We consider networks with nodes Poissonly distributed on a sufficiently large 2D area with density λ. Any two nodes can directly communicate with each other, hence can measure their distance, if their true Euclidean distance is smaller than a given threshold R, known as the *transmission range*. That is, the so-called unit disk model is adopted. Because the main focus of this paper is on studying the *maximum* number of possible estimated positions in the bilateration-based localization process that occur *in the entire network*, when the network area is sufficiently large, the impact of boundary effect on the above maximum value becomes comparatively small and hence is ignored in the paper.

We tessellate the entire network area into small grids of size $\frac{R}{\sqrt{5}} \times \frac{R}{\sqrt{5}}$. Note that the size of the grids has been chosen such that *any two nodes* located in *adjacent* grids are directly connected. Two grids are said to be adjacent if they share at least one point. Following common terminology used in discrete percolation theory [2], we also call these grids *sites*. A site is said to be *occupied* if there is at least one node in it; otherwise the site is said to be *empty*. Let p be the probability that a site is occupied, obviously

$$p = 1 - e^{-\lambda R^2/5}$$

If a site is occupied, we place a vertex in the center of the site; if a site is empty, no vertex is placed inside the site. For convenience, from now on, we normalize all distances by $\frac{R}{\sqrt{5}}$ such that these grids become unit grids with size 1×1. Denote by (x_i, y_i) and (x_j, y_j) the coordinate of two vertices v_i and v_j. We place an edge to connect v_i and v_j if and only if $|x_i - x_j| \leq 1$ and $|y_i - y_j| \leq 1$. If there exists an edge between v_i and v_j, its distance is considered to be known.

The above procedure allows us to construct a discrete percolation graph out of the earlier random network. Fig. 1 illustrates some fully connected grids and random grids for examples. In the rest of this paper, we shall analyze the maximum number of possible estimated positions of the random networks by analyzing the maximum number of possible estimated positions in its associated discrete-percolation graph. Note that due to the relation between the grid size and the transmission range, if there are more than one node in a grid, conclusions can be readily drawn that: a) these nodes are directly connected to each other (hence their distances are all known); b) these nodes are directly connected to nodes in adjacent grids, if exist. Therefore the maximum number of possible estimated positions of the random networks is less than that in its associated discrete-percolation graph. We denote the discrete-percolation graph by $G(p)$.

III. PROBLEM FORMULATION & ANALYSIS

In this section, we give a formal statement of the problem and present our approach to the problem.

A discrete percolation graph $G(p)$ may have several bilaterative components. Since we are interested in only these components of $G(p)$, whenever we talk about a subgraph
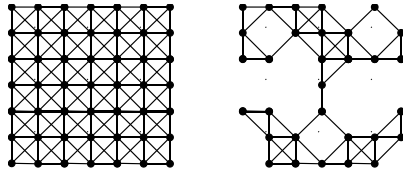
Figure 1. (a) a fully connected grid; (c) a discrete[a] percolation graph[b]

$H = (V, E)$ of $G(p)$, we implicitly assume H is a bilaterative component $H_B = (V, E, B)$ of $G(p)$ where B is one bilateration ordering for H (we drop the subscript B and simply write $H = (V, E, B)$ if the bilateration ordering is clear from the context). We denote an ordering of n vertices $v_i, i = 1..n$ by $< v_1, .., v_n >$. The *seed triangle* T of a bilateration ordering $B = < v_1, ..., v_n >$ is defined as the triangle $G(\{v_i | i = 1..3\})$. Denote by $S_B(v_i)$ (or simply $S(v_i)$ is B is clear from the context) the set of finite estimated positions of node v_i with respect to the bilateration ordering B during the localization process of any bilateration-based localization technique.

We define $s^*(H) = \max_{B \in \mathcal{B}}(\max_{v_i}(|S_B(v_i)|))$, where $\mathcal{B}$ is the set of all bilateration orderings of H (or s^* if H is clear from the context), as the maximum number of possible positions for any node in the grounded graph H taken among all possible bilateration orderings of H. Given the definition of $s^*(H)$, the main objective of this paper is then to show that $s^*(G(p)) = O(1)$ for any bilaterative component of the discrete percolation graph $G(p)$.

The main idea here is to first identify the class of graphs (subgraphs $H = (V, E)$ of $G(p)$) which can result in $s^*(H) \in \Theta(2^n)$ where $n = |V|$ is the number of vertices in H. We say that a bilaterative graph H is a *bad topology* (bad graph) if $s^*(H) \in \Theta(2^n)$. The bilaterative graph H is a *critical topology* (critical graph) if $s^*(H) = 2^{n-3}$. Assuming the bilateration ordering $B = < v_1, ..., v_n >$ for a critical topology H, by starting from v_1, v_2, v_3, each vertex $v_i, i = 4..n$ has twice the number of candidate positions as v_{i-1}.

In any bilateration-based algorithm, the node v_i is said to be finitely localized if $0 < |S(v_i)| < \infty$ holds ($S(v_i)$ is a finite non-empty set). Initially all $S(v_i)$ are set to $\emptyset$. The algorithm starts from the sensors which have at least **2** anchors in their neighborhood. In the first step, whenever a node v_i has at least two finitely localized (including anchors) neighbors, it finitely localizes itself (and creates a finite and non-empty set $S(v_i)$) by considering all the combinations of the possible positions of those neighbors. During the run of the algorithm, whenever a change happens to $S(v_i)$ (being finitely localized, uniquely localized, etc.), v_i notifies its neighbors. In each subsequent step, whenever the node v_i receives a notification from a neighbor, it checks if that position information is inconsistent with some of the elements in $S(v_i)$. If so, it prunes $S(v_i)$ by removing those inconsistent positions. The process continues until no inconsistency happens. During the construction of $S(v_i)$ only the *consistent* position pairs p_s, p_t of the finitely localized neighbors v_s, v_t are used. The two positions p_s, p_t of the nodes v_s, v_t are said to be consistent if for any node v_w whose $S(v_w)$ is used to compute both p_s, p_t (directly or indirectly), only a single $p_w \in S(v_w)$ is used (see [3] Section 3.1 for further details).

Suppose that $H_B = (V, E, B)$ is a bilaterative graph and

$H^{(i)}$ is the bilaterative subgraph of H_B induced by the first i vertices, i.e. $v_1, \ldots, v_i$ in the bilateration ordering B. Also assume that the positions of v_1, v_2 and v_3 are known to be p_1, p_2 and p_3 respectively (v_1, v_2, v_3 are anchors). For the three anchors, $S(v_i) = \{p_i\}, i = 1..3$. Denote by $Z_i(j, k) = \{(p_j, p_k) | p_j \in S(v_j), p_k \in S(v_k), (p_j, p_k) \, is \, consistent, 1 \leq j, k \leq i\}$ the set of all *consistent* pairs of candidate positions of vertices $v_j, v_k, j < k \leq i$ and its maximum by $z_i^*(H_B) = \max_{1 \leq j < k \leq i}(|Z_i(j, k)|)$. The next Lemma provides and upper bound on $z_i^*(H_B)$ (the details of the proof can be accessed in [7]):

Lemma 1. *Suppose that $H_B = (V, E, B)$ is a bilaterative graph with $|V| = n$. Then $|S(v_i)| \leq 2^{i-3}$ and $z_i^*(H_B) \leq 2^{i-3}$ always hold for any $4 \leq i \leq n$.*

Corollary 2. *Assume that $H_B = (V, E, B)$ is a bilaterative graph with $|V| = n$. Then $s^*(H_B) \leq 2^{n-3}$ always holds.*

Corollary 2 implies that among all bilaterative graphs on n vertices (shown by $\mathcal{G}$), critical topologies are the ones with maximum possible s^* and the maximum itself is 2^{n-3} ($\max_{|V(H)|=n, H \in \mathcal{G}}(s^*(H)) = 2^{n-3}$).

Therefore, as the first step towards an analytic study of the performance of bilateration-based techniques, we first characterize a broad subset of the class of critical topologies. We then derive an upper bound on the size of such subgraphs of $G(p)$. Due to tight relation between this bound and $s^*(G(p))$, it immediately follows that $s^*(G(p))$ is also bounded and provides a rough estimate of $s^*(G(p))$. This will be done by computing an upper bound on the cumulative probability distribution of the length of some maximal critical subgraph of the random grid $G(p)$ and show that it is bounded by a constant value. We should mention that it is not only the maximal critical topologies that can cause $s^*(G(p)) \in \Omega(2^n)$. There could be some *bad* topologies which are formed by concatenating several critical topologies and globally rigid subgraphs producing larger $s^*(G(p))$ than the one that could be estimated by only maximal critical topologies (maximal TC_t subgraphs). It is still an open problem (and subject of our further studies) to investigate these cases to see if there are bad topologies producing a bigger value for $s^*(G(p))$ (its true value) than the maximal TC_t subgraph. Therefore, the obtained upper bound is rather an approximation of $s^*(G(p))$ and should be considered as the first step towards analytically studying the performance of bilateration-based localization techniques.

A. Characterization of a Subclass of Critical Topologies

We first introduce a class of bilaterative graphs called *twisting chains* which will be shown to generate a commonly occurring subclass of critical topologies.

Definition 3. A graph $H = (V, E)$ is called a twisting chain of order $t \in \mathcal{Z}^+$ (denoted by TC_t) if $|V| \geq 3$ and the edge set includes only the following edges:
  - $(v_1, v_2) \in E$.
  - $\forall v_i, i = 3..|V| : v_i$ is adjacent to v_{i-1} and exactly one of $\{v_j | j = (i - 2 - t)..i - 2\}$.

The class of TC_0 graphs are simply called *chains* and they are the graphs obtained from a single edge $((v_1, v_2))$
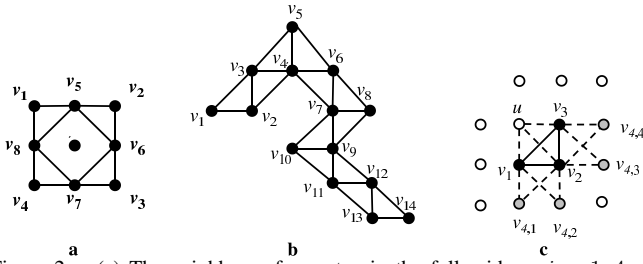
Figure 2. (a) The neighbors of a vertex in the full grid. $v_i, i = 1..4$ are the corner neighbors and $v_i = 4..8$ are the siblings. (b) An example of a TC_1 graph. Notice the difference with the chains: v_7 is adjacent to v_4, v_6 instead of v_5, v_6; (c) neighbors of the triangle v_1, v_2, v_3. Only four of them (gray-colored circles) can be part of a TC_1 together with the triangle.

by subsequently adding the new vertex v_{i+1} to the vertices v_{i-1}, v_i. We also define TC_∞ (or simply TC) as a twisting chain where v_i is connected to v_{i-1} and any single other vertex preceding it in the bilateration ordering.

Remark 4. If the graph G is a TC_t, then it is necessarily a TC_u for any $u \geq t$.

The *length* of a TC_t graph (for some t) $H = (V, E)$ is define as the number of vertices in H. The following lemma proves that every TC_t graph is a critical topology:

Lemma 5. *Assume that $H_B = (V, E, B)$ a bilaterative graph with $H_B \in TC_t$ for some $t < n - 2$ and $B = <v_1, .., v_n>$. Then $s^*(H_B) = 2^{n-3}$ holds, i.e. H_B is a critical topology.*

Proof: Since H_B is a TC_t graph, the vertex v_4 is connected to 2 vertices $v_i, v_j, i, j \in 1..3$. Inductively, *given each pair of (generic) positions (p_i, p_j) for the vertices $v_i, v_j, 3 \leq i \leq n, j \in (i - t - 2)..(i - 1)$, there are two possible positions for node v_{i+1}*. Form Lemma 1 and by noting that in any TC_0 graph G with n vertices, $z_i^*(G) = 2^{i-3}$, we conclude that the maximum number of (p_i, p_j) pairs is 2^{i-3}. Therefore, $S(v_{i+1}) = 2 \times 2^{i-3} = 2^{i-2}$ holds which implies $s^*(H_B) = 2^{n-3}$ and hence H_B is a critical topology. ■

In the following Lemma we prove that in a critical topology no vertex can be adjacent to more than two vertices preceding it in the ordering.

Lemma 6. *The bilaterative graph $H_B = (V, E, B)$ has the maximum number of candidate positions $s^*(H_B) = 2^{n-3}$ (is a critical topology) only if for each vertex $v_i \in V$, $|N(v_i) \cap \{v_j | j < i\}| = 2$.*

Proof: We prove it by contradiction. Assume that H_B is a critical topology and up to the step $i - 1$, $|S(v_{i-1})| = 2^{i-4}$ holds. Also assume that v_i is connected to three vertices $v_{i-1}, v_j, v_k, j < k < i - 1$ (and possibly others again, but this is not relevant to the calculations). The vertex v_i is connected to $H^{(i-1)}$ by a trilateration operation. Therefore, for each combination of the positions of v_{i-1}, v_j, v_k, there is exactly one position for v_i. Define $\hat{z} = max_{s,t \in \{j,k,i-1\}}(|Z_{i-1}(s,t)|)$ as the maximum number of consistent pairs between v_j, v_k, v_{i-1}. According to Lemma 1, $\hat{z} \leq 2^{i-4}$. Therefore, $|S(v_i)| \leq 2^{i-4}$ holds. Now even if for all $i + 1 \leq j \leq n$ steps, a bilateration operation results in doubling the size of each $S(v_j)$, $s^*(H_B) \leq 2^{i-4} \times 2^{n-i} = 2^{n-4} < 2^{n-3}$ contradicting the fact that H_B is a critical topology. ■

Lemma 6 intuitively reinforces the fact that TC_t graphs form a fairly broad subclass of critical topologies and hence

are worth to be studied (detailed explanation of this fact can be found in [7]). Therefore, in the remainder we concentrate on this subclass of critical topologies. One commonly occurring class of these bad topologies are TC_1 graphs. Figure 2b show an example of such graphs.

Definition 7. A TC_t subgraph $C = (V_C, E_C)$ of the graph $G = (V, E)$ with a specific value of t is termed *maximal* if there is no vertex $u \in V$ such that $G(V_C \cup \{u\})$ is also a TC_t graph.

B. Maximal TC_1 Subgraphs: Cumulative Distribution Function

In the discrete percolation model $G(p)$ almost all occurrences of maximal TC_t subgraphs of $G(p)$ are actually TC_1(refer to [7] for details of the proof). Therefore, it suffices to concentrate on maximal TC_1 subgraphs of $G(p)$ and derive a rough bound on the cumulative distribution of their lengths. To be able to better study TC_1 subgraphs, we investigate how such topologies can grow inductively from a triangle (which is the basis of any bilaterative graph).

As is depicted in Figure 2c, any triangle in the random grid consists of two orthogonal sides and one chord. Among the vertices in the neighborhood of the triangle $T = G(\{v_1, v_2, v_3\})$ there are only four (out of 12) neighbors which are candidate for the fourth node in the bilateration ordering (the shaded nodes in Figure 2c). Notice that the vertex u cannot be the fourth vertex as it is connected to 3 vertices in the ordering and therefore in any realization of the graph $|S(u)| = |S(v_3)|$ will hold. Therefore, the TC_1 graphs can only grow from one of their orthogonal sides and not from the chord. Also notice that to extend the TC_1 graph from T, on each side of the triangle (e.g. v_1, v_2) at most one of the two neighboring vertices can exist (either $v_{4,1}$ or $v_{4,2}$ but not both of them), i.e. if they exist simultaneously one of them will be adjacent to 3 vertices preceding it in the ordering violating the definition of TC_1 graphs (the growing process will stop).

The next vertex in the ordering (for example $v_{4,1}$) also defines a triangle with neighboring vertices in the triangle (v_1, v_2). As can be seen in Figure 2c, one of the orthogonal sides of this new triangle is in common with T. Therefore, there is only one side in this triangle from which the TC_1 graph can grow. By a proper reordering, the same extension is possible from the other side of T which implies that the number of sides a TC_1 graph can grow from remains fixed (2 as in the initial triangle). We call these extendible sides as the *ends* of the TC_1 graph.

There are two events that stop a TC_1 graph from being further extended: (a) first (as mentioned before) is when the next vertex to join the graph is connected to it by 3 edges; (b) none of the two possible neighbors of any of the two ends exist. According to Definition 7, this subgraph of $G(p)$ is a maximal TC_1 sub-graph as it is not possible to extend it further.

Based on the above observations we compute an upper bound on the cumulative distribution function (CDF) of the length of maximal TC_1 subgraphs in $G(p)$. Let us denote the maximal TC_1sub-graph of G including an arbitrary site O, by $C(O)$. We are interested in computing an upper bound on

$Pr_p(|C(O)| \geq l)$ for different values of l. In this notation p is the parameter determining the CDF. First we need to compute the probability of the existence of a triangle including O, i.e. $Pr_p(|C(O)| \geq 3)$.

1) CDF of a triangle on O: Assume that $N(O)$ is the set of vertices in the neighborhood of O excluding O itself. We define $A_i, i = 0, ..., 8$ as the event that the number of neighbors of O is exactly i ($A_i \equiv |N(O)| = i$). For ease of notations let us denote the event of having at least a triangle containing O, by $B \equiv |C(O)| \geq 3$. It is obvious that $Pr_p(B) = \sum_{i=0}^{8} P_p(B|A_i) \times P_p(A_i)$. We have $P_p(B|A_0) = P_p(B|A_1) = 0$ as for having the desired triangle, O must have at least two neighbors.

Figure 2a shows the neighborhood of node O. If $|N(O)| \geq 5$, then at least a pair of those vertices are adjacent which together with the vertex O form a triangle implying that $Pr_p(B|A_i, i = 5..8) = 1$. The probability of having exactly i vertices in $N(O)$ is obtained from a binomial distribution with success probability p (since the occurrence of each vertex is independent of the others and is with probability p). Therefore, $Pr_p(A_i) = \binom{8}{i} p^i (1-p)^{8-i}$.

The probability $Pr_p(B|A_i, i = 2..4)$ is not as straightforward to compute as there are some configurations with $2 \leq |N(O)| \leq 4$ which cannot form a triangle with vertex O. In the sequel we consider each case separately:

- Exactly 2 neighbors. Generally there are $\binom{8}{2} = 28$ possible choices for picking two neighbors of O. However, as Figure 2a shows, only 12 choices of the vertex pairs out of $\binom{8}{2}$ lead to an edge forming a triangle together with O. Therefore, we conclude that $P(B|A_2) = \frac{12}{28}$.

- Exactly 3 neighbors. The total number of choices to select 3 vertices out of the 8 possible neighbors of O is $\binom{8}{3} = 56$. To form a triangle together with O, it suffices for the choice of three vertices to have at least two of them adjacent. It is simpler to count the number of choices where all three vertices are non-adjacent: (a) $\binom{4}{3} = 4$ possible choices for when all the three vertices are on the corners of the neighborhood; (b) 4 possible configurations when two of the vertices are at the corners and 1 is a sibling. All other choices result in configurations where at least two of the vertices are adjacent. This means that, out of the 56 choices, $56 - 8 = 48$ configurations can induce a triangle including O and hence $Pr_p(B|A_3) = \frac{48}{56}$.

- Exactly 4 neighbors. In this case there are $\binom{8}{4} = 70$ distinct choices of the vertices. It is easy to see that the only possible configuration which does not induce any edge is when the 4 vertices are the 4 corner neighbors of O. This implies $Pr_p(B|A_4) = \frac{69}{70}$.

Putting all these together, we conclude that the probability of having a triangle ($Pr_p(|C(O)| \geq 3)$) on the vertex O is:

$$Pr_p(|C(O)| \geq 3) = 12p^2(1-p)^6 + 48p^3(1-p)^5$$
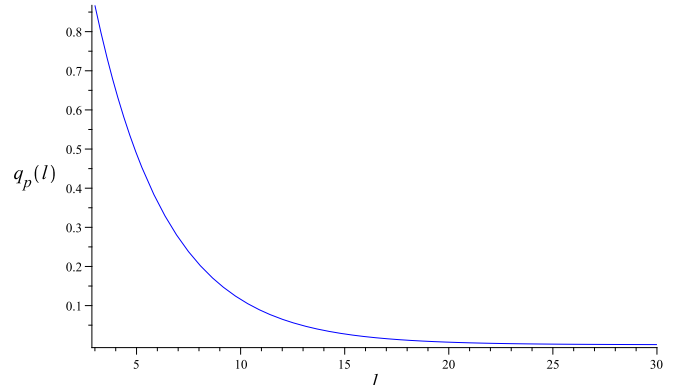$$+ 69p^4(1-p)^4 + \sum_{i=5}^{8} \binom{8}{i} p^i (1-p)^i$$

Figure 3. Analytical bound on CDF of maximal TC_1 subgraphs of length at least l for different values of l when $p = p^* = 0.5315$

2) CDF of the length of maximal TC_1 subgraphs: So far we have computed the probability of having a triangle including an arbitrary vertex O. Now we use that result to derive an upper bound on $Pr_p(|C(O)| \geq l)$ in $G(p)$ for $l > 3$. We do this by recursively deriving an upper bound on $Pr_p(|C(O)| \geq i)$ having $Pr_p(|C(O)| \geq i+1)$.

Denote the event that there is a TC_1 subgraph of length *at least* i which includes vertex O by A_i ($A_i \equiv |C(O)| \geq i$) and the graph itself by G_i. As we know any TC_1 graph of length $l \geq 3$ can be further extended from at most two ends. According to Figure 2c there are at most 4 possible nodes that can form the extension with the current TC_1 subgraph ($v_{4,i}, i = 1..4$). It is not hard to see that $Pr_p(A_{i+1}) = Pr_p(A_{i+1}|A_i) \times Pr_p(A_i)$ holds.

We denote the upper bound on $Pr_p(A_i)$ by $q_p(i)$ and compute it recursively. First notice that for $i = 3$ Subsection III-B1 gives the exact value for $Pr_p(A_3)$, the probability that there exists a triangle including vertex O. We set $q_p(3) = Pr_p(A_3)$. Since $Pr_p(A_{i+1}) = Pr_p(A_i) \times Pr_p(A_{i+1}|A_i)$ holds, we can write $q_p(i+1) = q_p(i) \times q_p(i+1, i)$ where $q_p(i+1, i)$ is an upper bound on $Pr_p(A_{i+1}|A_i)$. The upper bound $q_p(i+1, i)$ can be easily obtained by noting that there are at most 4 vertices that can potentially be the $i + 1$-th vertex in the extended TC_1 graph. Since in Figure 2c either $v_{4,1}$ or $v_{4,2}$ (also one of $v_{4,3}, v_{4,4}$) can exist but not both of them, the probability that G_i is not extendible from any side is $(1 - 2p(1-p))^2$. Based on this we define $q_p(i+1, i)$ as the probability that G_i is extendible from at least one side:

$$q_p(i+1, i) = Q_p = 1 - (1 - 2p(1-p))^2$$

Note that this upper bound is independent of i. Hence we have $q_p(i+1) = q_p(i) \times Q_p$ as the upper bound on $Pr_p(A_{i+1})$. By recursion we are able compute the upper bound on $Pr_p(A_l)$:

$$Pr_p(|C(O)| \geq l) = Pr_p(A_l) \leq q_p(l)$$
$$q_p(l) = q_p(3) \times Q_p^{l-3} \tag{1}$$

Figure 3 plots $q_p(l)$ for a specific value of $p = p^*$ (p^* is the value maximizing $q_p(l)$ which can be derived from Equation 1). For example the probability that the maximal TC_1 subgraph in $G(p)$ including an arbitrary vertex O to be longer than $9, 11, 13, 15, 19$ and 27 are respectively $0.2, 0.1, 0.05, 0.03, 0.01, 0.001$. This leads to the following important theorem which summarizes the contribution of the paper:
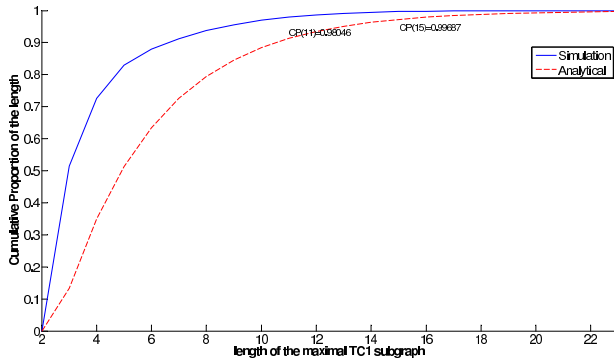
2135

Figure 4. Analytical bound on CDF and the average Cumulative Proportion of the length of the maximum TC_1 subgraphs in $G(p)$ with $p = 0.5$ for 100 networks.

Theorem 8. *Assume that $G(p)$ is a random grid and the length of its maximum TC_1 subgraph is denoted by $l(G(p))$. Then $l(G(p)) \in O(1)$ with respect to the size of the network.*

The maximum TC_1 subgraph of $G(p)$ is a good approximation of its longest critical subgraph (as most of critical subgraphs of $G(p)$ are TC_t and with high probability the maximal TC_t subgraph is TC_1). Since its length is $\in O(1)$, we conclude that $s^*(G(p)) \in O(1)$ also holds (with high probability) which means that *we highly expect the number of estimated positions for each node to be constant and independent of the size of the network.*

Since $Pr_p(|C(O)| \geq l)$ is an upper bound on $Pr_p(|C(O)| = l)$, we can also use $q_p(l)$ to compute an upper bound on the average length of the maximal TC_1 subgraph in $G(p)$ containing an arbitrary vertex O according to the formula $E(|C(O)|) \leq \hat{\mu}_p(G(p)) = \sum_{i=0}^{\infty} l \times q_p(l)$. With a simple calculation we can see that this sum is finite and its maximum over all p is $\hat{\mu}_{p^*} = 18.65$.

C. Simulation Results

In this section we report on extensive simulations to validate the above result. We generate 100 discrete percolation graphs (obtained from a 15×15 full grid) and find the length of the maximal TC_1 subgraphs in each. Then we derive the cumulative proportion (CP) of maximal TC_1 subgraphs which can be compared with the theoretical result in Section III-B. Figure 4 plots the average CP of the maximal TC_1 subgraph of the generated networks for $p = 0.5$.

As shown in the figure, the probability of having a maximal TC_1 subgraph of length greater than 13 is around %2. As can be seen the analytical result obtained in Subsection III-B2 (Figure 3) is an upper bound on the probability of long maximal TC_1 subgraphs.

IV. CONCLUSION

In this paper we studied the performance of the class of computationally efficient localization techniques known as bilateration-based localization techniques. Despite their advantage in localizing a high portion of the nodes, compared with their trilateration-based counterparts, in these techniques the number of estimated possible positions (and hence the memory required to store those positions) may **grow exponentially with the number of nodes** in the network. However, empirical observations reported in the literature that such exponential

growth is a rare even though there was a lack of rigorous analysis quantifying the complexity of bilateration-based schemes. We tackled this challenge by characterizing a broad subset of the class of *critical sub-networks* within which the number of possible estimated positions grows exponentially with the size of these sub-networks. We proved that in random geometric networks, with very high probability, the size of these critical sub-networks is bounded. The significance of this result is to analytically demonstrate the computational efficiency of bilateration-based techniques in comparison with the trilateration-based counterparts while being able to localize a higher fraction of nodes.

This work forms the first steps to use techniques from percolation theory to analytically characterize the performance of localization algorithms. Although we have characterized the critical topologies which cause the maximum number of estimated positions among all graphs with the same size, there are still some sub-networks of low probability whose maximum number of estimated positions can be exponential with respect to their size. One possible future direction of this work would be to further analytically study all bad topologies which could result in a memory blow-up.

REFERENCES

[1] J. Fang, M. Cao, A. Morse, and B. Anderson. Sequential localization of sensor networks. *SIAM J. Control & Optimization*, 48(1):321, 2010.

[2] M. Franceschetti and R. Meester. *Random networks for communication: from statistical physics to information systems*. Cambridge Univ Pr, 2007.

[3] D. Goldenberg, P. Bihler, M. Cao, J. Fang, B. Anderson, A. Morse, and Y. Yang. Localization in sparse networks using sweeps. In *Proc. 12th ann. int. conf. Mobile computing and networking*, pages 110–121, 2006.

[4] G. Mao, B. Fidan, and B. Anderson. Wireless sensor network localization techniques. *Computer Networks*, 51(10):2529–2553, 2007.

[5] L. Meertens and S. Fitzpatrick. The distributed construction of a global coordinate system in a network of static computational nodes from internode distances. *Kestrel Institute TR KES. U*, 4, 2004.

[6] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. In *Proc. 2nd int. conf. Embedded networked sensor systems*, pages 50–61, 2004.

[7] S. A. Motevallian. Extended proof of the paper: Analytical characterization of computationally efficient localization techniques. at http://users.cecs.anu.edu.au/~motevallian/papers/wcnc13/exprf.pdf.

[8] D. Niculescu and B. Nath. Dv based positioning in ad hoc networks. *Telecommunication Systems*, 22(1):267–280, 2003.

[9] A. Savvides, C. Han, and M. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proc. 7th ann. int. conf. Mobile computing and networking*, pages 166–179, 2001.

[10] A. Savvides, H. Park, and M. Srivastava. The bits and flops of the n-hop multilateration primitive for node localization problems. In *Proc. 1st ACM int. workshop Wireless sensor networks and applications*, pages 112–121, 2002.

[11] J. Wang, R. Ghosh, and S. Das. A survey on sensor localization. *Journal of Control Theory and Applications*, 8(1):2–11, 2010.