Using Local Geometry for Tunable Topology Control in Sensor Networks

Sameera Poduri[1], *Member, IEEE,* Sundeep Pattem[2], *Member, IEEE,* Bhaskar Krishnamachari[1,2], *Member, IEEE,* and Gaurav S. Sukhatme[1,2], *Senior Member, IEEE*

***Abstract*— Neighbor-Every-Theta (*NET*) graphs are such that each node has at least one neighbor in every theta angle sector of its communication range. We show that for $\theta < \pi$, *NET* graphs are guaranteed to have an edge-connectivity of at least $\lfloor \frac{2\pi}{\theta} \rfloor$, even with an irregular communication range. Our main contribution is to show how this family of graphs can achieve tunable topology control based on a single parameter θ. Since the required condition is purely local and geometric, it allows for distributed topology control. For a static network scenario, a power control algorithm based on the *NET* condition is developed for obtaining k-connected topologies and shown to be significantly efficient compared to existing schemes. In controlled deployment of a mobile network, control over positions of nodes can be leveraged for constructing *NET* graphs with desired levels of network connectivity and sensing coverage. To establish this, we develop a potential fields based distributed controller and present simulation results for a large network of robots. Lastly, we extend *NET* graphs to 3D and provide an efficient algorithm to check for the *NET* condition at each node. This algorithm can be used for implementing generic topology control algorithms in 3D.**

***Index Terms*— topology control, k-connectivity, distributed deployment of mobile nodes, 3D networks**

I. Introduction

Topology control for ad-hoc wireless and sensor networks has traditionally only dealt with *uncontrolled* deployments, where there is no explicit control on positions of nodes [1]–[3]. The primary mechanisms proposed are power control and sleep scheduling. These methods involve removing edges from an existing, well connected communication graph in order to save power while ensuring that the resultant sub-graph preserves connectivity. Controlled deployments, feasible when positions of individual nodes can be altered, present a different and interesting scenario for topology control. Since connectivity properties directly depend on the positions of nodes, position control can be leveraged for effective topology control. There is increasing evidence that such deployments will be possible in, as well as be required by, a number of future sensor network applications. Consider the following examples - sensors implanted in civil structures at select, unobtrusive locations *e.g.* SHM [4], sensors in water bodies either anchored to stay in place or mounted on boats *e.g.* NAMOS [5], sensors that are not inexpensive enough to afford high redundancy *e.g.* MASE [6], small and medium scale networks maintained by a robot or human *e.g.* energy harvesting using robots [7], and mobile sensor networks *e.g.* NIMS [8]. Traditional approaches are ill-suited in such scenarios since they are not designed to exploit control of the motion and placement of the nodes.

[1]Department of Computer Science, [2]Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089. Partial support for this research was provided by the NSF through the STC (CCR-0120778), ITR (CNS-0325875), CAREER (CNS-0347621 and IIS-0133947), and NeTS-NOSS programs (CNS-0627028 and CNS-0520305). Sukhatme acknowledges support from the Okawa Foundation, and Poduri was supported by a merit fellowship from the USC Women in Science and Engineering (WiSE) Program.

In this paper, we develop a general approach that supports traditional methods like power control and also allows new designs for controlled deployments. Our main contributions are:

- a k-connectivity result, based on local conditions and independent of communication model
- topology control algorithms based on the k-connectivity result for a) power control with static nodes, and b) distributed deployment of mobile nodes
- 3D extensions of 2D results and an efficient algorithm for topology control in 3D.

We define *Neighbor-Every-Theta* (*NET*) graphs in which each node (except those at the boundary) have at least one neighbor in every θ sector of its communication range. Our key theoretical result is that for $\theta < \pi$, a *NET* graph has an edge-connectivity of at least $\lfloor \frac{2\pi}{\theta} \rfloor$ irrespective of the communication model. This implies that the condition only depends on the angles between the neighbors of each node and holds for arbitrary edge-lengths. This feature is particularly relevant for sensor networks using low-power radios that have irregular communication range. For the special case with an idealized disk communication model, we derive conditions for maximizing the sensing coverage area (defined as the total area sensed by at least one node) while satisfying the k-connectivity constraint, and for obtaining proximity graphs such as the Relative Neighborhood Graph. It should be emphasized that the *NET* condition is local since each node only requires relative position information of its (communicating) neighbors.

NET graphs are naturally suited for distributed power control. We implement a typical power control protocol using satisfaction of *NET* condition at each non-boundary node as the termination criterion and show that a power-efficient network with edge connectivity $\lfloor \frac{2\pi}{\theta} \rfloor$ can be achieved even with realistic, irregular links. This is in contrast to the sector based topology control algorithms in literature including CBTC [9], [10], that rely on an idealized disk communication model to guarantee connectivity properties. In CBTC based power control [11], for a network to be k-connected, each node must either have a neighbor in every $\theta = \frac{2\pi}{3k}$ sector or operate at full power. Our results imply that a sector angle of $\theta = \lfloor \frac{2\pi}{k} \rfloor$ is sufficient for k-connectivity. A three times larger value of sector angle implies a much lower power requirement. We present simulation results to establish this in Section VI. Even though we do not consider sleep-scheduling mechanisms in depth, it is possible to think of designs where densely deployed nodes make sleep/wake decisions in a distributed manner by using local, pair-wise negotiations based on combinations of *NET* condition satisfaction and other criteria.

Position control is ideal for obtaining topologies that are *NET* graphs. Given the properties guaranteed by the *NET* condition, an external agent deploying a static sensor network can make decisions on the best positions for deployment of new nodes based on the geometry of the existing network. Self-deploying mobile nodes can use this condition to decide their motion strategy. We present a distributed algorithm for self-deployment of mobile nodes to concretely demonstrate the use of the local and geometric conditions to implement coverage and connectivity tradeoffs. The algorithm involves achieving *NET* condition satisfaction through purely pair-wise negotiations between neighbors. The performance of this algorithm is studied through an implementation on the Player/Stage simulation platform [12]. Results illustrate the tunable tradeoff between connectivity and coverage and show that the distributed algorithm can achieve approximations of some coverage-maximizing tiling structures that also satisfy *NET* conditions.

It has been argued [13] that topology control in three dimensions (3D) is much more complex. There are several operations that are common to a wide range of topology control algorithms that have increased and sometimes prohibitively high complexity in 3D. Examples include obtaining angular information of and ordering neighbors and checking for intersections of overlapping regions [14]–[17]. As a result, there is very little work on topology control algorithms in 3D. We argue that controlled deployment and *NET* graphs are well suited for sensor networks in 3D and present extensions to our 2D results. We propose an efficient algorithm for identifying the largest empty cone of a node's communication range. This algorithm can be used in conjunction with several deployment and topology control mechanisms for implementing their 3D extensions.

The paper is organized as follows. The next section defines the problem and section III presents *NET* graphs and analysis of their connectivity properties which are extended to 3D in section IV. Sections V and VI present applications of *NET* graphs for distributed deployment of mobile robots and distributed power control respectively. Finally we put our work in context with a discussion of related work in section VII and conclude in section VIII.

II. Problem Formulation

Local conditions that guarantee global network properties are the key to designing distributed topology control algorithms. In particular, we study local conditions that guarantee global k-edge-connectivity of the network. Once such conditions are found, they can be integrated with controls available in order to design topology control algorithms. Further, it is important to understand if the results can be extended to 3D networks in an efficient manner. A detailed description of these three problems follows.

Problem 1: k-connectivity certificates

Given a network, find local geometric conditions between node positions that can guarantee global k-edge-connectivity.

A graph is said to be *k-edge-connected* if at least k edges must be removed to disconnect it. By Menger's theorem, this is equivalent to saying that there exist at least k edge-disjoint paths between any two nodes in the graph. The graph that we consider is the communication network where two nodes are said to have an edge between them if they can communicate. In this case, high edge-connectivity is desirable because it implies high fault-tolerance and path diversity.

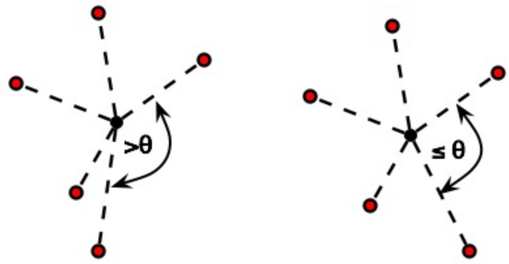


Fig. 1. Illustration of the *NET* condition. Red circles are symmetric neighbors. In a) node does not satisfy *NET* condition. It has one sector greater than θ with no neighbor. In b) node satisfies *NET* condition, the largest sector with no neighbors is smaller than θ.

By 'local' we mean that each node only has information about positions of its communicating neighbors relative to its own position. The global coordinates of nodes are not available. The conditions we seek must be based only on this local information so that each node can independently decide whether it satisfies the condition. Note that, for the k-connectivity certificates, we do not make any assumption on the communication model. In particular, we do not assume that the communication model is an idealized disk where two nodes can communicate if and only if they are within a fixed distance.

The next step is to apply these conditions in the design of efficient topology control algorithms. We consider two ways of modifying the topology of a network - (1) by controlling the positions of nodes and (2) by controlling the communication power of nodes.

Problem 2: Topology Control

How can the local geometric conditions for k-connectivity be integrated with the controls available, like communication power and node positions, for efficient topology control?

We have the following sub-problems.

a) Distributed deployment of mobile nodes: *Given N mobile nodes, design a distributed control law such that they move to maximize sensing coverage while maintaining global k-connectivity.*

b) Power control in a static network: *Given a static network of N nodes, how should they choose their communication power such that the network is k-connected and the energy is minimized?*

Lastly, we study the extension to networks in three dimensions.

Problem 3: Extensions to networks in three dimensions

Does the geometric analysis for 2D networks extend to 3D networks?

It is well known that several geometric results in 2D either do not extend to 3D or are significantly more complex in 3D [13]. We want to extend our results to 3D and study their correctness and efficiency.

III. Neighbor-Every-Theta (*NET*) Graphs

In this section, we will define *Neighbor-Every-Theta* graphs and show that they give us k-connectivity certificates.

Definition 3.1: The *Neighbor-Every-Theta* (*NET*) condition (Fig. 1) for a node embedded in the 2D plane is defined as requiring at least one symmetric neighbor in every θ sector of its communication range.
Nodes A and B are symmetric neighbors if A can communicate to B and B can communicate to A.

For finite networks, nodes on the network boundary cannot satisfy such a condition. The *boundary* of a network can be defined as a cycle of nodes such that every other node lies *inside* the cycle [18]. A node that does not belong to the boundary is called an *interior* node.

Definition 3.2: A *NET* graph is one in which every interior node satisfies the NET condition for a given θ.

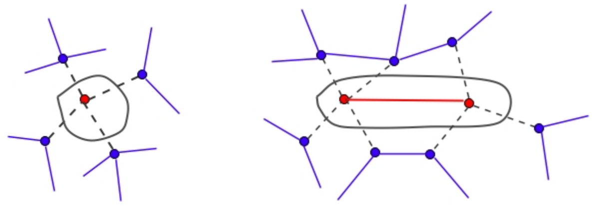
We now analyze the connectivity and coverage properties of *NET* graphs. For large networks where the number of boundary nodes is small compared to the network size, we show that *NET* graphs have an edge connectivity of at least $\lfloor \frac{2\pi}{\theta} \rfloor$, independent of the communication model. With the stronger assumption of a idealized disk communication model, *NET* graphs, for specific values of θ, contain proximity graphs such as the Relative Neighborhood Graph (RNG). An upper bound on the sensing coverage is shown to be obtained from a symmetric arrangement of nodes and can be computed as a function of θ. *NET* graphs form a family of graphs based on the single parameter θ - as θ becomes smaller, the graphs become denser with an increasing level of connectivity.

A. Connectivity Analysis of NET Graphs

The edge-connectivity of any graph is at most its minimum node degree and can in general be arbitrarily low irrespective of the node degree [19]. For some special graphs, higher node degree implies higher edge-connectivity. One example of such a graph is the random geometric graph, where an average node degree of $O(\log N)$ (where N is the network size) guarantees an edge-connectivity of 1 with high probability [20]. It turns out that for em NET graphs also there is relation between minimum node degree and edge-connectivity. In *NET* graphs every interior node has a degree of at least $\lfloor \frac{2\pi}{\theta} \rfloor$. We will show that the edge-connectivity is also at least $\lfloor \frac{2\pi}{\theta} \rfloor$ (except in pathological cases where boundary nodes can be disconnected by removing fewer edges). This means that the easiest way to disconnect a *NET* graph is to disconnect a single interior node by removing $\lfloor \frac{2\pi}{\theta} \rfloor$ edges. We will now formally establish this property by first considering the simple case of graphs that are very large so that edges close to the boundary are not significant. An extreme case of such graphs are the "boundary-less" graphs defined below. Later, we will analyze the impact of boundary nodes on connectivity.

Definition 3.3: A *NET* graph is **boundary-less** if every node satisfies the *NET* condition. For $\theta < \pi$, such a graph must span the entire 2D plane.

Definition 3.4: Given a graph $G = \{V, E\}$, a **cut set** is a set of edges $E' \subseteq E$ such that $G' = \{V, E - E'\}$ has more than one component. The edge-connectivity a graph, λ, is the size of its smallest cut set, C. Note that if G is connected, C must divide it into exactly two components.



(a) $\parallel V_1 \parallel = 1$ (b) $\parallel V_1 \parallel = 2$

Fig. 2. The red nodes and edges represent G_1, dotted edges form the cut set C.

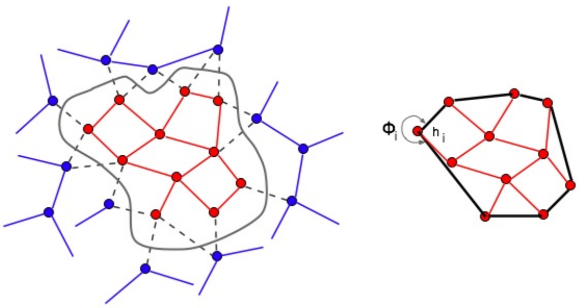


Fig. 3. $\parallel V_1 \parallel \geq 3$ The red nodes and edges represent G_1, the dotted edges form the cut set C. The corresponding convex hull H is shown with solid black lines.

Definition 3.5: For a graph embedded in the Euclidean plane a **cut** is a curve that partitions the graph into two or more components.

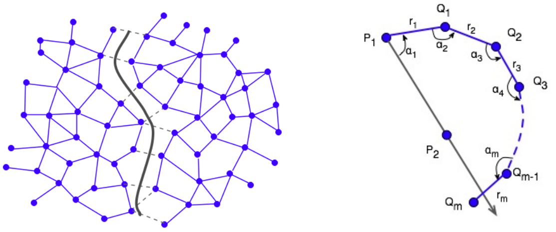
Theorem 3.6: For $\theta < \pi$, a boundary-less *NET* graph has an edge-connectivity $\lambda \geq \lfloor \frac{2\pi}{\theta} \rfloor$
Proof: Consider a NET graph $G = \{V, E\}$. Let $C \subseteq E$ be the smallest cut set of G so that $\lambda = \parallel C \parallel$. Let the two components of $G' = \{V, E - E'\}$ be $G_1 = \{V_1, E_1\}$ and $G_2 = \{V_2, E_2\}$. WLOG, $\parallel V_1 \parallel \leq \parallel V_2 \parallel$.

If $\parallel V_1 \parallel = 1$ then $\parallel C \parallel \geq MinDegree \geq \lfloor \frac{2\pi}{\theta} \rfloor$ (Fig 2(a)). If $\parallel V_1 \parallel = 2$ then $\parallel C \parallel \geq 2\left(\lfloor \frac{2\pi}{\theta} \rfloor - 1\right) \geq \lfloor \frac{2\pi}{\theta} \rfloor$ since $\theta < \pi$ (Fig. 2(b)).

Suppose $\parallel V_1 \parallel > 2$. Construct the convex hull, H_1 of V_1. Then $H_1 \subseteq V_1$. Define angle ϕ_i at $h_i \in H_1$ as shown in Fig 3. There will exist at least 3 vertices[1] h_i such that $\phi_i > \pi$ *i.e.* , $\angle h_i < \pi$. WLOG, assume that ϕ_1, ϕ_2 and $\phi_3 > \pi$. Since $G = \{V, E\}$ is a *NET* graph, for $i = 1, 2, 3$, ϕ_i must contain $\geq \lfloor \frac{\pi}{\theta} \rfloor$ edges $\in C$. Therefore, $\parallel C \parallel \geq 3\lfloor \frac{\pi}{\theta} \rfloor \geq \lfloor \frac{2\pi}{\theta} \rfloor$. □

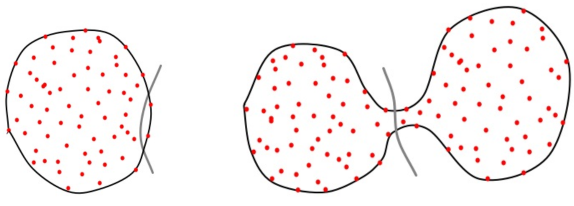
The above result holds for the general case where the graph has boundary nodes provided the cut is completely in the interior of the network, *i.e.* if all the nodes in V_1 satisfy the *NET* condition. If on the other hand, the cut mostly consists of boundary edges then it is not interesting because it is unlikely to impact network performance. Now consider cuts that intersect the network boundary twice - when entering and leaving the graph (fig. 4(a)). WLOG, assume that the cut is minimal in length *i.e.* it is the shortest curve corresponding to its cut set. The case when the cut intersects itself is covered by the above theorem - the number of edges cut inside the loop alone must be at least $\lfloor \frac{2\pi}{\theta} \rfloor$. Assume that the cut does not intersect itself and by way of contradiction,

[1] Since H is a convex polygon, $\angle h_i \leq \pi, \forall i$. Suppose all except 2 internal angles are $= \pi$, say $0 < \angle h_1, \angle h_2 < \pi$ and $\angle h_i = \pi$ for $i = 3, 4, ...k$, where k is number of vertices of H, then $\sum_1^k \angle h_i = ((k-2)\pi + \angle h_1 + \angle h_2) > (k-2)\pi$. Contradiction since $\sum_1^k \angle h_i = (k-2)\pi$.



(a) (b)

Fig. 4. (a) A finite graph where all non-boundary nodes satisfy the *NET condition*. The think line $\mathcal{L}$ is a cut through the graph (b) $\mathcal{L}$ cuts polygons



(a) (b)

Fig. 5. Degenerate cases of cuts through *NET* graphs that result in poor connectivity. (a) A cut close to the boundary (b) pinched boundary

that it disconnects the network by cutting less than $\lfloor\frac{2\pi}{\theta}\rfloor$ edges. We will now analyze the nature of such a cut and prove that away from the network boundary, the distance along the cut between two consecutive edges must be less than $R_c.sec(\theta/2)$, where R_c is an upper bound on the edge-length. This implies that away from the boundary the cut cannot be longer than $\lfloor\frac{2\pi}{\theta}\rfloor \cdot R_c.sec(\theta/2)$. If θ is not close to π this expression is bounded. For example, for $\theta = 0.9\pi$ it is $\approx 13 \cdot R_c$ and for $\theta = \frac{2\pi}{3}$ it is $6 \cdot R_c$. For large networks, a "short" cut like this can only exist close to the boundary (fig. 5(a)) or if the boundary is "pinched" [21](fig. 5(b)).

Lemma 3.7: For $\theta < \pi$, the length of a minimal cut between two consecutive edges in a boundary-less *NET* graph is less than $R_c \cdot sec(\theta/2)$.

Proof: Let $\mathcal{L}$ be the cut and l its length between two consecutive edges it cuts, say (u_1, v_1) and (u_2, v_2) (figure 4(b)). Then $l \leq \max\{d(u_1, u_2), d(u_1, v_2), d(v_1, u_2), d(v_1, v_2)\}$, where $d()$ is the euclidean distance function. Let the node pair that leads to maximum distance be P_1, P_2. Now P_1 must have at least one neighbor in each of the two θ sectors adjoining $\overrightarrow{P_1P_2}$. Let these be Q_1 at an angle $0 \leq \alpha_1 \leq \theta$ and Q_1' at an angle $0 \leq \alpha_1' \leq \theta$ with $\overrightarrow{P_1P_2}$. $0 \leq \alpha_1 + \alpha_1' \leq \theta < \pi$. WLOG, $\alpha_1 < \pi/2$. Similarly Q_1 must have a neighbor Q_2 at an angle $\alpha_2 \leq \theta$, and so on till some $\overline{Q_{m-1}Q_m}$ intersects $\overrightarrow{P_1P_2}$. Note that the point of intersection cannot be in between P_1 and P_2 because otherwise $\mathcal{L}$ would cut $\overline{Q_{m-1}Q_m}$ contradicting the assumption that (u_1, v_1) and (u_2, v_2) are consecutive edges.

We have,

$$\begin{aligned}
l &= \|P_1P_2\| \\
&\leq r_1 \cdot cos(\alpha_1) + r_2 \cdot cos(\alpha_1 + \alpha_2 - \pi) + \cdots \\
&\quad +r_m \cdot cos(\alpha_1 + \alpha_2 + \cdots + \alpha_m - (m-1)\pi) \\
&\quad (\text{where } \pi/2 > \alpha_1 > \alpha_1 + \alpha_2 - \pi > \cdots \\
&\quad > \alpha_1 + \alpha_2 + \cdots + \alpha_m - (m-1)\pi) > -\pi/2) \\
&\leq R_c(cos(\alpha_1) + cos(\alpha_1 + \alpha_2 - \pi) + \cdots \\
&\quad +cos(\alpha_1 + \alpha_2 + \cdots + \alpha_m - (m-1)\pi)) \\
&\quad (\text{where } R_c \text{ is an upper bound on edge length}) \\
&\leq R_c((cos(\alpha_1) + cos(\alpha_1 + \theta - \pi) + \cdots \\
&\quad +cos(\alpha_1 - (m-1)(\pi - \theta))) \\
&\quad (\text{since } cos \text{ is a concave function in } [-\pi/2, \pi/2]) \\
&= R_c \frac{sin\left(\frac{m\cdot(\theta-\pi)}{2}\right) \cdot cos\left(\alpha_1 + \frac{(m-1)\cdot(\theta-\pi)}{2}\right)}{sin\left(\frac{\theta-\pi}{2}\right)} \\
&\leq R_c \frac{1}{cos(\theta/2)}
\end{aligned}$$

□

It must be emphasized that the connectivity result only needs the largest edge in the network to be bounded and holds even for a non-ideal and irregular communication models. In the following subsections, we present further interesting properties of *NET* graphs with a stronger assumption of an idealized disk communication model.

B. Proximity Graphs

Proximity graphs such as the Relative Neighborhood Graph (RNG), Gabriel Graph (GG) and the Delaunay Graph (DelG) have several properties such as connectivity, sparseness, efficient network routes, *etc.* that make them desirable network topologies [14], [22]–[24]. We will now analyze conditions under which *NET* graphs will contain these proximity graphs. Here we assume an idealized disk communication model but allow each node to have a different communication range. Further, the number of boundary nodes is assumed to be small compared to the network size. The analysis presented is non-trivial since the edge-lengths in *NET* graphs are restricted by the communication range of nodes, while in proximity graphs they depend only on the relative positions of nodes and very long edges are possible.

The set of edges E for various proximity graphs is defined as follows [19]. In what follows, we use the name and location of a node interchangeably.

Disk graph ($DG(V, E, R)$): The directed graph containing all outgoing edges of a node, $u \in V$, not longer than $R(u)$.

$$E = \{(u, v)|u, v \in V \text{ and } d(u, v) \leq R(u)\}$$

Given positive $r \in \mathbb{R}$, let $C(p, r)$ be the circle consisting of points whose distance from point p is strictly less than r. Define the lune, denoted $L(p, q)$, to be the intersection of two circles, both of radius $d(p, q)$, centered at these points, that is, $L(p, q) = C(p, d(p, q)) \cap C(q, d(p, q))$.

Relative Neighborhood Graph (RNG(V)): The undirected graph containing an edge (u, v) if there is no point $w \in V$ that is simultaneously closer to both u and v. Equivalently, (p, q) is an edge if $L(p, q) \cap V = \emptyset$.

$$E = \{(u, v)|u, v \in V \text{ and } \exists \text{ no } w \in V \ni$$
$$d(u, w) < d(u, v) \text{ and } d(v, w) < d(u, v)\}$$

Gabriel Graph (GG): The undirected graph containing an edge (u, v) if the disk whose diameter is edge (u, v) does not contain any other points of V, that is, if $C(\frac{u+v}{2}, \frac{d(u,v)}{2}) \cap V = \emptyset$.

$$E = \{(u, v)|u, v \in V \text{ and } C(\frac{u+v}{2}, \frac{d(u, v)}{2}) \cap V = \emptyset\}$$

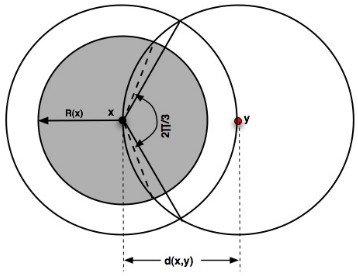


Fig. 6. RNG sector condition. a) The circle of radius $d(x,y) > R_c$ subtends an angle of $\frac{2\pi}{3}$ at x. The lune contains an area larger than a $\frac{2\pi}{3}$ sector. b) The limiting case, when $d(x,y) = R(x)$.

Delaunay Graph (DelG): The undirected graph containing an edge (u,v) if the Voronoi regions of u and v have non-empty intersection. From the properties of Voronoi diagrams it follows the edges of triangle (u,v,w) are in $DelG(V)$ if their circumcircle does not contain any other points of V.

These graphs are hierarchically related:
$RNG(V) \subseteq GG(V) \subseteq DelG(V)$.

In the proofs that follow, we assume that each node, u, in the network has a communication range $R(u)$ so that the communication graph of the network is a disk graph. Our next step is to find conditions under which the RNG of the network is contained in the communication graph. This is equivalent to finding conditions under which no edge (u,v) of an RNG is longer than either $R(u)$ or $R(v)$. The following theorem presents this condition. Later, we prove similar results for the GG and DelG.

Theorem 3.8: If each node $x \in V$ has at least one neighbor in every $\frac{2\pi}{3}$ sector of $C(x, R(x))$, the communication graph is a supergraph of $RNG(V)$. Moreover, $\frac{2\pi}{3}$ is the largest angle that satisfies this property.

Proof: Consider any node $x \in V$. Suppose x has at least one neighbor in every $\frac{2\pi}{3}$ sector of $C(x, R(x))$. We first show that for any node y outside $C(x, R(x))$, the edge $(x,y) \notin RNG(V)$. The lune $L(x,y)$ will contain a sector of at least $\frac{2\pi}{3}$ (Fig.6). By premise, $\exists$ a node in $L(x,y)$. This implies that $RNG(V)$ does not have any edges incident on x that are longer than $R(x)$.

Next we show that for any node $z \in V$ inside $C(x, R(x))$ such that $d(x,z) > R(z)$ *i.e.* $(x,z) \in DG(V,E,R)$ but $(z,x) \notin DG(V,E,R)$, the edge $(x,z) \notin RNG(V)$. Since z also has a neighbor in every $\frac{2\pi}{3}$ sector of $C(z, R(z))$ and $d(x,z) > R(z)$, from the above argument $(x,z) \notin RNG(V)$.

Therefore $RNG(V) \subseteq DG(V,E,R(x))$.

Now suppose a node $x' \in V'$ has two neighbors with a sector angle of $\frac{2\pi}{3} + \delta$ $(\delta > 0)$ between them (Fig.7). We can place a node y' outside $C(x', R_c)$ such that the edge $(x', y') \in RNG(V')$. Therefore, $\frac{2\pi}{3}$ is the largest angle for which this condition holds.$\square$

We have shown that for $\theta \leq \frac{2\pi}{3}$, *NET* graph contains the RNG assuming an idealized disk communication model. We can prove a similar property for GG and DelG, except that in this case the value of θ is a function of the distance to neighbors.

Lemma 3.9: If each node $x \in V$ has at least one neighbor in every $\theta = 2\arccos(\frac{r}{R(x)})$ sector of $C(x,r)$ $(r \leq R(x))$, the communication graph is a supergraph of $GG(V)$. *Proof:* See Appendix.

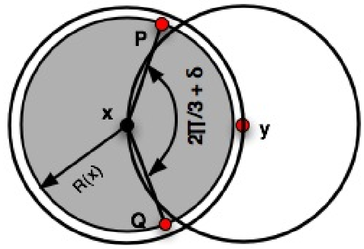


Fig. 7. $\theta = \frac{2\pi}{3}$ bound for RNG - $L(x,y)$ is empty

Corollary 3.10: If a node x has at least one neighbor in every $\theta = 2\arccos(\frac{r}{R(x)})$ sector of $C(x,r)$ $(r \leq R(x))$, the communication graph is a supergraph of $DelG(V)$. *Proof:* See Appendix.

In real networks, it is not possible for the boundary nodes to satisfy the conditions required by the theorems. We show using simulated deployments (in section V) that the assertions can be validated in spite of these exceptions.

C. *Coverage Analysis of* NET *Graphs*

Having listed the conditions that guarantee global connectivity properties, we now turn to the problem of maximizing coverage. We assume that all nodes have a idealized disk sensing model with a sensing radius of R_s. Maximizing sensing coverage is equivalent to packing problem with a constraint placed on the communication neighbors. This is a very hard problem given an irregular communication model and even harder to solve locally. Therefore, we make a simplifying assumption that the communication model is also an idealized disk with a radius R_c for all nodes. Suppose that in order to satisfy the *NET* sector conditions, a node must have $k = \lfloor \frac{2\pi}{\theta} \rfloor$ neighbors. From the node's local perspective, all neighbors must be located on the perimeter of the communication range to maximize coverage. Intuitively, the nodes must also be placed symmetrically on the perimeter. We prove this result for the special case when $R_c = R_s$.

Lemma 3.11: For $R_s = R_c$, the area coverage is maximized when the $k \geq 3$ nodes are placed at the edges of k disjoint $\frac{2\pi}{k}$ sectors of $C(x, R_c)$.

Proof: Consider a node x and k nodes $y_1, y_2, ..., y_k$ placed on the perimeter of $C(x, R_c)$. Let these nodes be placed in anticlockwise order at angles $\beta_1 = 0, \beta_2, ..., \beta_k$ respectively. Define

$$\begin{aligned} \theta_i &= \beta_{i+1} - \beta_i, 1 \leq i \leq k-1 \\ \theta_k &= 2\pi - \beta_k \end{aligned} \qquad (1)$$

We need to find θ_i such that the total area covered by these $k+1$ nodes is maximized. The open disks of all nodes lie within $C(x, 2R_c)$ and are tangent to this disk at exactly one point (T_i) each. The disks of adjacent nodes i and $i+1$ intersect at point I_i (and at X). The total coverage lies between πR_c^2 and $4\pi R_c^2$ and is maximized when the area $\sum_{i=1}^{k} T_i I_i T_{i+1}$ is minimum (Fig.8).

$$\begin{aligned} &T_i I_i T_{i+1} \\ &= T_i X T_{i+1} - I_i Y_i T_i - I_i Y_{i+1} T_{i+1} - I_i Y_i X Y_{i+1} \\ &= (2R_c)^2 \cdot \frac{\theta_i}{2} - 2R_c^2 \cdot \frac{\theta_i}{2} - \frac{1}{2}(2R_c \sin\frac{\theta_i}{2})(2R_c \cos\frac{\theta_i}{2}) \\ &= R_c^2(\theta_i - \sin\theta_i) \end{aligned}$$

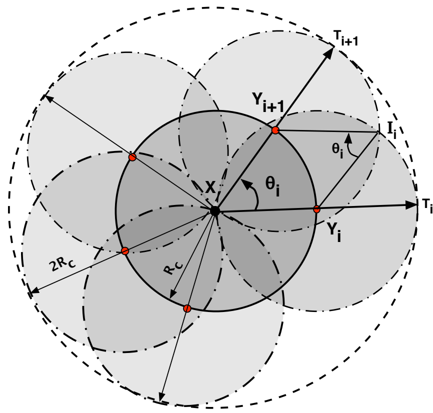


Fig. 8. Coverage with k nodes placed on the communication perimeter of node x. The shaded area is the total coverage.

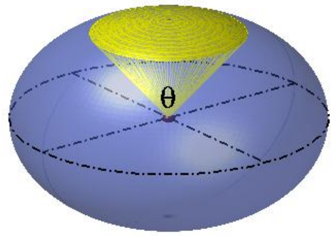


Fig. 9. *NET* graph: each cone of angle θ must have at least one neighbor.

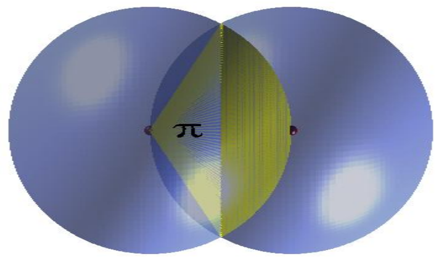


Fig. 10. RNG cone angle condition

$$\sum_{i=1}^{k} T_i I_i T_{i+1} = R_c^2\left(\sum_{i=1}^{k}\theta - \sum_{i=1}^{k}\sin\theta_i\right) = R_c^2\left(2\pi - \sum_{i=1}^{k}\sin\theta_i\right) \quad (2)$$

The problem now reduces to finding $\max \sum_{i=1}^{k} \sin\theta_i$ subject to $\sum_{i=1}^{k}\theta_i = 2\pi$ and $0 \leq \theta_i \leq 2\pi$. Since sin is non-negative and concave in $[0,\pi]$ and non-positive in $(\pi, 2\pi]$, it follows that the solution is $\theta_i = \frac{2\pi}{k}, 1 \leq i \leq k$.□

For $k = 3, 4, 6$, it is possible to place nodes such that each node has its neighbors placed symmetrically on its communication range. The resulting communication graph will be a tiling of the space: hexagonal, square and triangle for $k = 3, 4$ and 6 respectively. In these cases, since each node maximizes coverage locally, the total coverage of the network will also be the maximized. For values of k other than $3, 4$ and 6 such an arrangement is not possible since the corresponding tiling structures do not exist. Therefore in these cases, the local condition for optimizing coverage will not necessarily optimize global coverage. Due to their symmetry, tiling graphs possess some other desirable properties. It can be verified that the hexagonal tiling ($\theta = \frac{2\pi}{3}$) is an RNG and in this case, GG ≡ RNG. This holds for the square tiling ($\theta = \frac{\pi}{2}$) as well. The triangle tiling ($\theta = \frac{\pi}{3}$) is a DelG.

Based on the above theorem, we can compute an upper bound on the maximum achievable coverage of a *NET* graph as a function of θ. Consider the arrangement in Fig. 8 with all the neighbors placed symmetrically around x. If each overlap area within the sensing range of x is divided by the number of nodes that cover it, then the sum of these weighted areas will give the maximum possible per-node coverage. In this computation we ignore the boundary nodes which cannot have a symmetric placement of neighbors. Therefore, this is an asymptotic bound. This upper bound on coverage is plotted in Fig. 13 along with the lower bound for edge-connectivity derived in theorem 3.11.

In the next section we extend *NET* graphs to 3D and study their connectivity and coverage properties.

IV. *NET* GRAPHS IN THREE DIMENSIONS

Network configuration in 3D is significantly more complex than in 2D [13]. There are several operations that are common to a wide range of sensor network configuration algorithms in 2D that have increased and sometimes prohibitively high complexity in 3D. We argue that controlled deployment and *NET* graphs are well suited for sensor networks in 3D and present extensions to our 2D results. We propose an efficient algorithm for identifying the largest empty cone of a node's communication range. This algorithm can be used in conjunction with a number of generic deployment and topology construction mechanisms for implementing their 3D extensions.

A node satisfies the *NET3D* condition if it has at least one symmetric neighbor in every cone of solid angle θ. A *NET3D* graph is one in which every node except those on the boundary satisfy the *NET3D* condition. If the network is very large compared to the size of the boundary, the following is our conjecture for the edge-connectivity.

Conjecture 4.1: For $\theta < 2\pi$, a boundary-less *NET3D* graph has an edge-connectivity $\lambda \geq \lfloor \frac{2\pi}{\theta} \rfloor$ □

Note that, like in 2D, the connectivity result is independent of the communication model. For specific values of θ, *NET3D* graphs will contain proximity graphs such as RNG, GG, and DelG.

Lemma 4.2: If each node $x \in V$ has at least one neighbor in every $\theta = \pi$ cone of $S(X, R(x))$, the communication graph is a supergraph of $RNG(V)$.

Lemma 4.3: If each node $x \in V$ has at least one neighbor in every $\theta = 2\pi(1 - \frac{r}{R_c})$ cone of $S(X, R(x))$, the communication graph is a supergraph of $GG(V)$ and $DelG(V)$. Moreover, $2\pi(1 - \frac{r}{R(x)})$ is the largest angle that satisfies this property.

The proofs follow from the corresponding proofs in 2D and the fact that a cone with apex angle α will contain a solid angle of $\theta = 2\pi(1 - cos(\alpha))$. For example in the proof for RNG, the

3D lune will contain a cone with apex angle $\alpha = \frac{2\pi}{3}$ and solid angle of $\theta = \pi$.

A. Integrating geometric conditions with topology control

The local conditions described above can be integrated with node placement to construct efficient topologies. A key requirement is an algorithm to check for empty cones larger than a given θ. For instance, to check for formation of RNG, $\theta = \pi$. This step is non-trivial in 3D because there exists no natural "order" of neighbors. We propose the following algorithm for finding the largest empty cone around a given node.

Algorithm 1: largestCone($G = (V, E)$**,**$v \in V$**)**
let S be the unit sphere centered at v
for each $u \in Neighbor(v)$, let $\overrightarrow{vu}$ be the direction vector from v to u
let c_u be the intersection of $\overrightarrow{vu}$ with S
let DT be spherical delaunay triangulation $c_u \forall u$
find $a_{i,j,k}$ = area of circumcircle of triangle $(u_i, u_j, u_k) \in DT$
return $max(a_{i,j,k})$

Algorithm 1: Find largest empty cone around a node

Theorem 4.4: For a given graph $G = (V, E)$ and node v, **largestCone** returns the largest empty cone around v.
Proof: The circumcircle of every (spherical) triangle in the spherical delaunay triangulation is empty [25]. Therefore the cone returned by **largestCone** is certainly empty. Suppose there exists an empty cone (whose image on the unit sphere is the circle c) that is larger than the one returned by **largestCone**. Then the center of c lies in some triangle t of the delaunay triangulation. Since c is empty, none of the vertices of t must lie inside c. This implies that the circumcircle of t will be larger than c which is a contradiction. Therefore *largestCone* correctly returns the largest empty cone.□.

The computational complexity of **largestCone** is $O(dlog(d))$ where d is the number of neighbors of a node. This is because, the complexity of spherical triangulation is $O(dlog(d))$, the number of triangles generated is $O(d)$, and sorting them will also take $O(dlog(d))$ time.

Several topology control algorithms [14]–[16], [23], [26] in 2D rely on directional information and in particular use the angle between adjacent neighbors. This algorithm can be used as a primitive for extending such algorithms to 3D.

V. DEPLOYMENT USING *NET* GRAPHS

Controlled deployments are well suited to take advantage of the global properties of *NET* graphs resulting from local geometric conditions. This applies to scenarios of deployment of static nodes by an autonomous agent and self-deployment of mobile nodes. In controlled deployment of a static sensor network, the agent can make decisions about the best locations for new nodes based on the local geometry of the existing network. Self-deploying mobile nodes can use this local condition to decide their motion strategy. The coverage can be maximized by positioning neighbors within adjacent θ sectors as far apart from each other as possible. In general, based on the coverage and connectivity requirements of an application, nodes can either be pre-configured for a certain value of θ or they can tune it dynamically.

This section presents a virtual potential fields based self-deployment algorithm for mobile nodes. We assume that all nodes have idealized disk models for communication and sensing with radii of R_c and R_s respectively. This ensures that negotiations between neighboring nodes are symmetric and simplifies the problem.

A. Distributed Deployment of Mobile Nodes

Potential field based algorithms have been widely used for the deployment of mobile networks [27]–[29]. These algorithms involve constructing local virtual forces between neighboring robots to encode their desired motion and/or placement configuration. In our algorithm, we use two kinds of forces. The first, $\mathbf{F_{repel}}$, causes the nodes to repel each other to increase their coverage and the second, $\mathbf{F_{attract}}$ constrains neighboring nodes to stay connected. These forces have inverse square law profiles - $\mathbf{F_{repel}}$ tends to infinity when the distance between the nodes decreases to zero and $\mathbf{F_{attract}}$ tends to infinity when the distance between nodes increases to R_c. They are tuned such that when two nodes apply a force of $\mathbf{F_{repel}} + \mathbf{F_{attract}}$ on each other, they settle exactly at R_c. The convergence of this controller is well known [29]. By using a combination of these mutually opposing forces, each node maximizes its coverage while maintaining the *NET* condition of having at least one neighbor in every θ sector.

The algorithm involves exchange of purely pairwise information between nodes. Each individual node then combines this information to check for *NET* condition satisfaction. The result of this check is then translated to individual decisions for each of its neighbors.

In a typical mobile deployment scenario, all nodes start in positions close to each other so that the initial network is highly well connected and the *NET* condition is trivially satisfied. Each node begins by repelling all neighbors to increase its sensing coverage. In the process, it loses communication with some neighbors that move farther than R_c. When the number of neighbors is close to the number required to satisfy the *NET* condition, the node assigns priority values to each of its neighbors based on their contribution towards satisfying its *NET* condition. This is done by computing sector angles between adjacent neighbors. The neighbors contributing to larger sector angles have a higher priority and the node applies the attractive force to hold them within its communication range. Nodes on the boundary designate all neighbors as low priority and hence allow the decision to be made based on the requirement of their neighbors. The pseudocode for this algorithm is shown below. c_1, c_2 are parameters that can be tuned for how strictly the *NET* condition is required to be satisfied.

Algorithm 2: assignPriority(θ**,** $node$**,** $neighborList$**)**
if $node$ is on boundary
    for $q \in neighborList$
        $priority[node][q] = 1$
else
    for $q \in neighborList$
        $sectorVoid[q] = angle(q_{prev}, q) + angle(q, q_{next})$
        if $sectorVoid[q] < c_1 \cdot \theta$
            $priority[node][q] = 0$ (redundant)
        else if $c_1 \cdot \theta \leq sectorVoid[q] < c_2 \cdot \theta$
            $priority[node][q] = 1$ (low priority)
            else $priority[node][q] = 2$ (high priority)

To ensure that the forces are symmetric, nodes exchange pair-wise priority information and apply forces based on the average priority.

Algorithm 3: distributedDeployment(θ)
find $neighborList$
$degree = sum(neighborList)$
if $(degree > (\frac{2\pi}{\theta} + 1))$
  repel all neighbors
else if $(degree > \frac{2\pi}{\theta}$ or $boundary)$
  $priority =$ assignPriority(θ, $self$, $neighborList$)
  for $q \in neighborList$,
    if $(priority[self][q] + priority[q][self]) \geq 1$
      repel + attract q
    else repel q
else repel + attract all neighbors

The condition used for classifying boundary nodes plays an important role in determining the final network structure. During deployment, the boundary of the network grows and nodes that were previously interior nodes become boundary nodes. Once a node becomes a boundary node, it is not required to satisfy the *NET* condition. If interior nodes are allowed to easily switch to becoming boundary nodes, the coverage will increase because nodes will spread out but the connectivity properties will suffer. On the other hand, if interior nodes are constrained to always satisfy *NET* condition and are not able to switch to the boundary then the network cannot spread out and coverage will be poor. We use a heuristic based on the observation that boundary nodes have large empty sectors. Initially all nodes are designated as non-boundary. If a node has an empty sector greater than α_b for time τ, then it declares itself as a boundary node. Similarly, if a boundary node has no empty sector greater than α_b for time τ, then it becomes an interior node.

B. Simulation Results

The algorithms were implemented in the Player/Stage software platform which simulates the behavior of real sensors and actuators with high fidelity [12]. It does not provide support for realistic communication models and hence we use a simple, idealized disk communication model. In our simulations, R_s was chosen equal to R_c. This is not a requirement for the algorithm and the coverage results are intended to be illustrative. We note that the connectivity results are independent of R_s [28]. If $R_s < \frac{R_c}{2}$, then the per node coverage is very high and in fact close to πR_s^2 because nodes can satisfy edge constraints without any overlap of sensing areas. If on the other extreme, R_s is significantly larger than $2 \cdot R_c$ then the per node coverage is low compared to πR_s^2 because of large overlaps. The interesting behavior happens when R_s and R_c are comparable.

The parameters c_1 and c_2 in **assignPriority** can be tuned depending on how strictly *NET* satisfaction is desired. For the results presented in this section, a conservative choice was sought and the empirically determined values are shown in Table. I. The sector parameter α_b is used by individual nodes for boundary detection. Choosing a large value for α_b will result in many boundary nodes considering themselves interior nodes trying to ensure *NET* satisfaction and impedes the spreading. On the other hand, choosing a small value leads to interior nodes switching to boundary nodes and decreases network connectivity. The appropriate setting of boundary sector α_b depends on the θ value and the empirically determined settings are shown in Table. I.

θ	$\frac{2\pi}{3}$	$\frac{\pi}{2}$	$\frac{2\pi}{5}$	$\frac{\pi}{3}$	$\frac{2\pi}{7}$	$\frac{\pi}{4}$
c_1	1.15	1.15	1.25	1.0	1.1	1.1
c_2	1.45	1.5	1.6	1.3	1.3	1.3
α_b	$\frac{21\pi}{20}$	$\frac{11\pi}{12}$	$\frac{5\pi}{6}$	$\frac{5\pi}{6}$	$\frac{5\pi}{6}$	$\frac{5\pi}{6}$

TABLE I
PARAMETER SETTINGS

Results are shown for deployments of 100 robots with each experiment repeated 10 times. Fig.11 show the final network configuration from sample runs of the distributed algorithm for varying θ values. For $\theta = \frac{2\pi}{3}$, hexagonal tiling is difficult to achieve in a distributed manner and combined with the conservative choice of c_1 and c_2, the algorithm pushes the deployment to a square tiling (Fig.11a). Large blocks of square tiling are achieved for $\theta = \frac{\pi}{2}$ (Fig. 11b). For the non-tiling angle $\theta = \frac{2\pi}{5}$ (at least 5 neighbors required), in most cases it is actually beneficial to settle for 6 neighbors and tile triangularly. Fig. 11c shows that the algorithm correctly allows for the highly stable tiling in large portions of the final configuration. This is also obtained for the tiling angle $\theta = \frac{\pi}{3}$, as shown in figure Fig. 11d. Samples for $\theta = \frac{2\pi}{7}$ and $\theta = \frac{\pi}{4}$ are also shown. No tiling is possible for these angles. Fig.12 shows that the deployment algorithm adapts well to obstacles. On encountering an obstacle, the nodes continue to spread while avoiding it and surround it with little impact on the *NET* graph. Obstacles have the effect of increasing the number of boundary nodes in the network and as a result, the network structure is not as uniform as before.

Fig. 13a compares the coverage obtained from the deployment algorithms to an asymptotic upper bound on coverage obtained from Lemma 3.11 by considering node overlaps as described in Section.V. The algorithm's coverage is close to the bound for smaller values of θ and the difference grows with θ. The difference can be attributed to a conservative choice of parameters c_1 and c_2. This is reflected in the fact that the average node degree resulting from the deployment algorithm is always greater than $\lfloor \frac{2\pi}{\theta} \rfloor$ (Fig. 13c) even though the boundary nodes have smaller node degrees. Also note that the coverage upper bound is tight only in case of tiling angles. For non-tiling angles, it is not clear what the optimal deployment is.

Fig. 13b shows the edge connectivity values. For the k-connectivity calculation, boundary nodes and one hop neighbors of boundary nodes are not considered. For the chosen c_1 and c_2 values a connectivity of $k = \lfloor \frac{2\pi}{\theta} \rfloor$ is not guaranteed, but it is very often achieved and connectivity $\lfloor \frac{2\pi}{\theta} \rfloor - 1$ is almost always assured. Using lower values for c_1 and c_2 can provide a guaranteed level of connectivity at the cost of some coverage. Fig. 13d shows the number of sectors of interior nodes that violate the *NET* condition. A 10 % leeway was allowed for deviation from θ, i.e., a violation occurs if sector angle $> 1.1 \cdot \theta$. There is a drop in violations for $\theta = \frac{2\pi}{5}$ since it defaults to the 6-neighbor triangular tiling while there is a sudden increase for $\theta = \frac{2\pi}{7}$ and $\theta = \frac{\pi}{4}$ since there are no tiling angles in sight making it difficult for the pair-wise negotiations to attain symmetrical placement of neighbors. The asymmetry and small value of θ result in increased violations. Note that this does not adversely affect the k-connectivity.

Fig. 14 shows a comparison of the communication graph obtained from deployment with $\theta = \frac{\pi}{3}$ with its corresponding

RNG. In Fig. 14c, the dark lines represent the edges in the RNG that are absent in the communication graph. The communication graph differs from the RNG in exactly 3 edges at the boundary of the network. Even in the presence of boundary nodes in real networks, these results validate the assertion in Theorem 3.8 that the communication graph will contain the RNG for $\theta \leq \frac{2\pi}{3}$. Also, the communication graph contains only a few non-boundary edges that are not in the RNG and will therefore inherit the sparseness properties of the RNG.

VI. Power Control with *NET* graphs

In this section we illustrate how *NET* graphs can be used for topology control of a static network by varying the communication power. The result is a power control mechanism that can guarantee k-edge-connectivity without any restriction on the communication model. In contrast, most existing power control algorithms, including the well-known CBTC [11], [26] algorithm, require an idealized binary disk communication model.

In the power control problem, we are given a network deployed with a sufficieltly large density so that connectivity (k-connectivity) is guaranteed when nodes operate at full power. The objective is to find an assignment of transmission powers to nodes so that the reduced graph retains connectivity (k-connectivity) while expending minimum energy. In some applications it is also desirable to maintain other properties such as planarity, sparseness, spanner, *etc.* *NET* graphs are naturally suited for distributed power control because

- they are based on local geometric conditions that guarantee global k-connectivity,
- communication range for the low power radios used in sensor network applications is highly irregular [30], [31]. The connectivity properties of *NET* graphs are independent of the communication model.

Consider a sensor network such that when nodes operate at full power, they satisfy the *NET* condition for a given θ. Using a typical power control protocol and satisfaction of *NET* condition at each internal node as the termination criterion, a power-efficient k-connected network can be achieved.

We have implemented a completely distributed power control algorithm as follows. Starting with a minimum value, each node incrementally increases its transmission power till the *NET* condition is satisfied. We assume that the orientation of neighbors can be computed either from angle-of-arrival of messages or localization information. Each node computes its empty sectors and increments power if 1) *NET* condition is not satisfied or 2) if it has an asymmetric incoming link from a node that does not satisfy *NET* condition. Since boundary nodes (detected using an algorithm such as [18]) are not required to satisfy *NET* condition, they only increase power in response to incoming asymmetric links. This is a simple implementation with scope for further power optimization which we plan to pursue in future. Our objective here is to demonstrate that for a given θ the resulting topologies preserve $\lfloor \frac{2\pi}{\theta} \rfloor$ edge-connectivity. In simulations, we used realistic statistical models for wireless links developed in [31] for which the authors have made code available online [32].

Fig. **??** shows typical topologies resulting with 500 nodes distributed uniformly at random on a 80m x 80m square. Nodes that cannot satisfy the *NET* condition at full power are identified as boundary nodes (shown as red squares). The initial transmission power was set to $-10dB$ and incremented in steps of $1dB$. Links that had a packet reception rate of at least 90% were considered active. Note that the figures only show bidirectional links. The simulations were terminated when all non-boundary nodes satisfied *NET* condition.

Fig. **??** (a) and (b) show the comparison the topologies resulting from *NET* based power control and CBTC(α) [11] for 2-edge-connectivity. According to CBTC(α) a node must either have a neighbor in every $\theta = \frac{2\pi}{3k}$ or operate at full power to guarantee k-edge-connectivity. We show that having a neighbor in every $\theta = \frac{2\pi}{k}$ is sufficient to guarantee k-edge-connectivity. As a result, *NET* condition results in a significantly sparser, and hence efficient, topologies compared to CBTC(α).

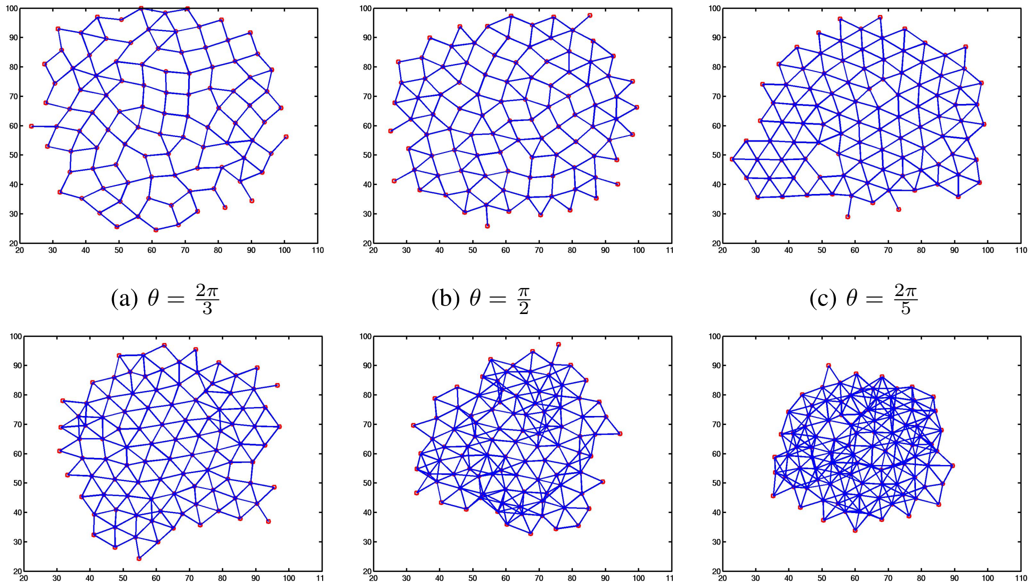
Fig. 16 shows the edge-connectivity and average power over 100 iterations. Edge-connectivity was computed as the minimum number of paths consisting of symmetric links between any two internal nodes. In all experiments, the edge-connectivity was always greater than $\lfloor \frac{2\pi}{\theta} \rfloor$. This validates our key theoretical result, Theorem 3.4. The average power used increases quickly as θ decreases (fig. 16(b)). This implies that for efficient topology control it is important to choose the largest value of θ possible depending on the application requirements. Because *NET* graph requires an angle ($\theta = 2\pi/k$) that is three times bigger than the angle required by CBTC(α) ($\theta = 2\pi/3k$), the power saved will very large. For example, for 2-edge-connectivity, CBTC(α) requires an average power of $4dB$ while *NET* graph requires $-6dB$.

These initial results are promising because we are able to achieve power-efficient topologies that guarantee k-connectivity through distributed power control even with realistic, irregular link models.

VII. Related Work

Sector based graphs have existed for a long time. The Yao graph [33] (also called θ-graph) introduced by A. C. Yao in 1977 is constructed by dividing the area around each node into equal sectors of θ each and adding an edge to the closest node in each sector if there exists one. The symmetric Yao graph [10] is a subgraph of the Yao graph containing only the edges chosen by both of the two end nodes. Every *NET* graph is a symmetric Yao graph but the converse is not true. This is because in a Yao graph, the angle between two adjacent neighbors of a node can be greater than θ depending on how the sectors were initially defined but in a *NET* graph this angle cannot be greater than θ. As a result, the properties we prove for *NET* graphs do not hold for Yao graphs. $\theta \leq \frac{\pi}{3}$ guarantees that the Yao graph contains the RNG whereas a much larger angle of $\theta \leq \frac{2\pi}{3}$ is sufficient to guarantee that the *NET* graph contains the RNG. Similarly, for $\theta = \pi$, *NET* graphs are connected [34] but the Yao graph is not necessarily connected.

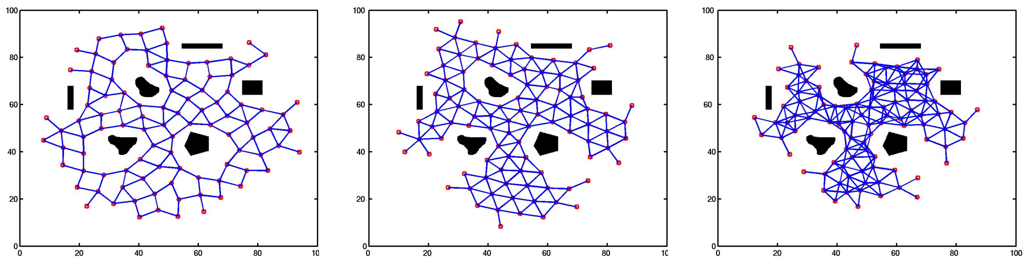
Sector based conditions were introduced for topology control of wireless ad-hoc networks by Li, Wang, Bahl and Wattenhofer [9] and Li, Wan, and Wang [10]. In the Cone Based Topology Control mechanism (CBTC) [9], each node either has a neighbor in every θ sector or operates at full power. Under the assumption of an idealized disk communication model, it is shown that the graph is connected if $\theta \leq \frac{2\pi}{3}$. Further, if the full power graph is k-connected then for $\theta \leq \frac{2\pi}{3k}$, the reduced graph retains k-connectivity [11]. In comparison, *NET* graphs achieve k-connectivity with a much larger angle (and hence significantly lower power) of $\theta = \frac{2\pi}{k}$ and do not require a the idealized disk model for communication. The difference



(a) $\theta = \frac{2\pi}{3}$ (b) $\theta = \frac{\pi}{2}$ (c) $\theta = \frac{2\pi}{5}$

(d) $\theta = \frac{\pi}{3}$ (e) $\theta = \frac{2\pi}{7}$ (f) $\theta = \frac{\pi}{4}$

Fig. 11. Sample deployments with 100 nodes and $R_c = R_s = 8.0$.

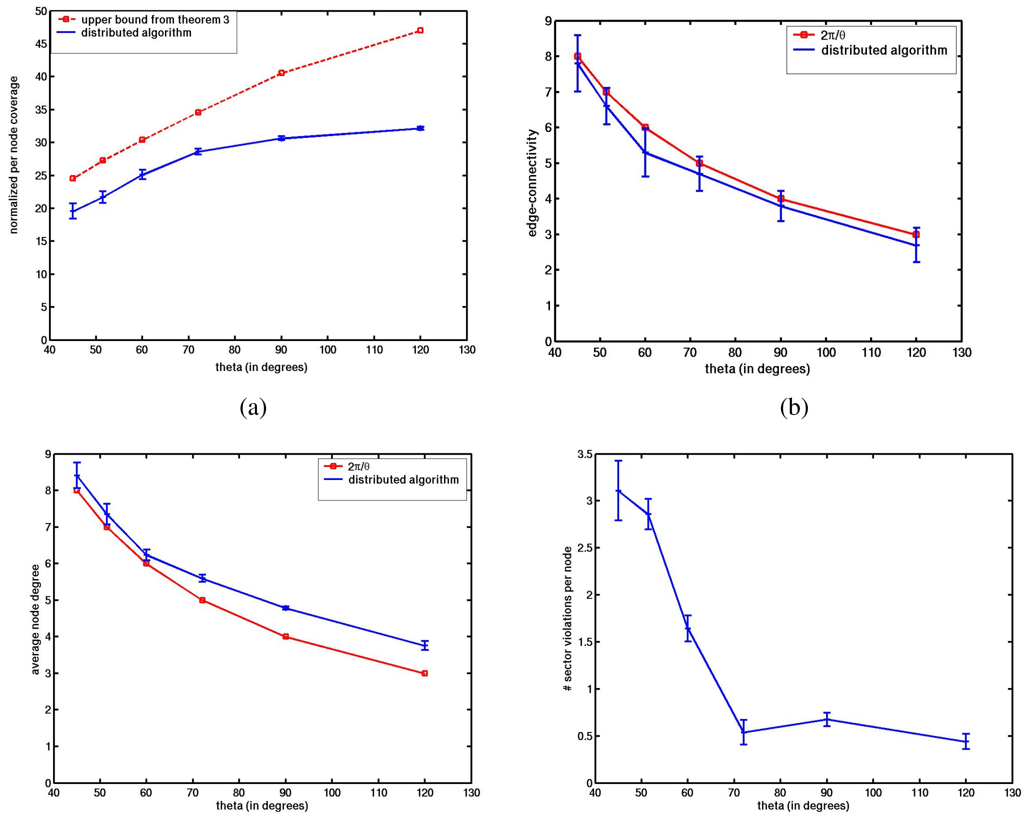


(a) $\theta = \frac{2\pi}{3}$ (b) $\theta = \frac{2\pi}{5}$ (c) $\theta = \frac{\pi}{3}$

Fig. 12. Sample deployments in the presence of obstacles with 100 nodes and $R_c = R_s = 8.0$.

between the two techniques is established in Section VI. Li *et al* proposed using geometric graphs such as RNG, GG, Delaunay and Yao graphs for topology control and to address the non-planarity and high node degree of Yao graph extended it to symmetric Yao graph, YaoGG, YaoYao graph, reverse Yao graph, and SθGG [10]. In [35] a modified Yao structure is proposed for k-vertex connectivity, where each node must have at least $k+1$ neighbors in each sector of angle less than $\frac{\pi}{3}$ around it. This implies that each node must have at least $6 \cdot (k+1)$ neighbors which is 6 times the corresponding number required for k-edge-connectivity using *NET* graphs. For scatternet formations, Stojmenovic proposed protocols that apply geometric graphs such as Yao graphs, RNG, and GG on the scatternet graph to limit node degree and ensure planarity while retaining connectivity [36]. Recently, Xue and Kumar [37] have defined θ-coverage condition which is equivalent to the *NET* condition and analyzed the critical radius for asymptotic θ-coverage for a randomly deployed 2D network. Further, they prove using geometric arguments that for $\theta \leq \pi$, θ-coverage implies 1-connectivity. The same result was simultaneously established by D'Souza *et al* [34] for a *weak monotonicity* communication model which is more general than the idealized disk communication model. Our paper extends this result to address k-connectivity as a function of θ, for any arbitrary communication model. There is a rich body of literature on topology control with a wide variety of techniques in addition to sector based techniques (see [14]).

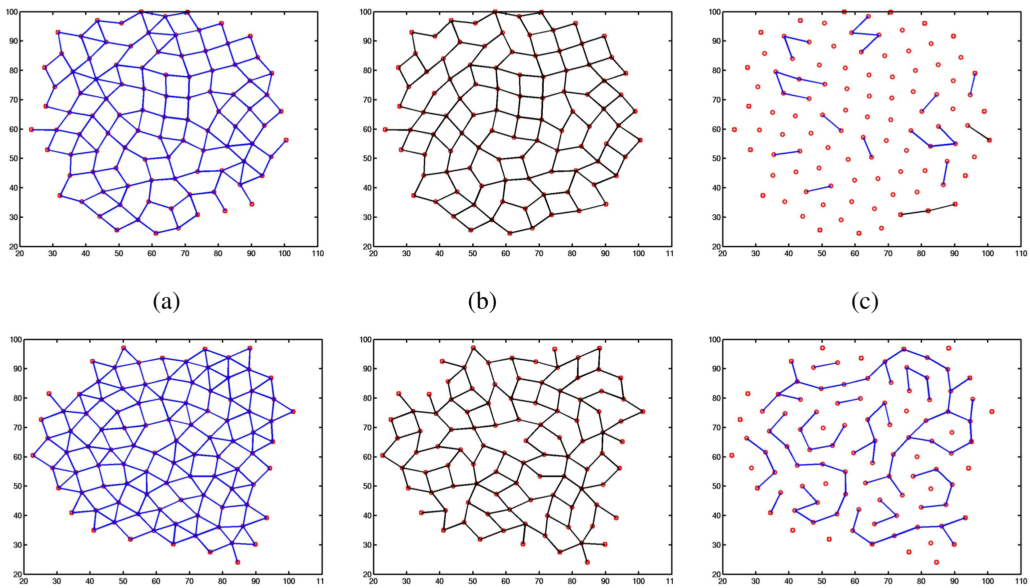
Sleep scheduling seeks to activate only a fraction of the nodes in a densely deployed network. Several researchers have studied the problem of finding the smallest set of active nodes that can simultaneously achieve complete coverage and connectivity where complete coverage is defined as every point in the network domain being within the sensing range of at least one active node. Under this definition, coverage and connectivity are not opposing goals; in fact, if the sensing range is at least twice the communication range, then complete coverage implies connectivity [15], [16]. In contrast we focus on the problem of maximizing sensing coverage given a fixed number of nodes so that higher coverage, in most cases, implies poorer connectivity. OGDC [15] and CCP [16] are sleep scheduling protocols based on geometric conditions. While OGDC seeks to minimize the number of active nodes, CCP can provide different degrees of coverage and connectivity. Interestingly, the geometric optimality conditions presented in [15], as pointed out by the authors, can be attained exactly if the positions of nodes can be controlled. Even though we do not consider sleep-



(a) (b)

(c) (d)

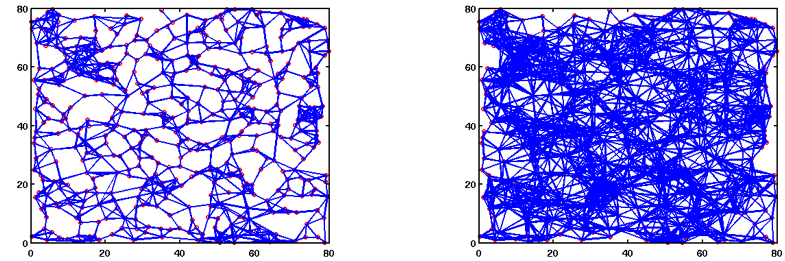
Fig. 13. Performance of the Sequential and Distributed deployment Algorithms for a deployment of 100 nodes in terms of (a) Coverage, (b) Edge Connectivity (c) Average Degree and (d) *NET* condition satisfaction



(a) (b) (c)

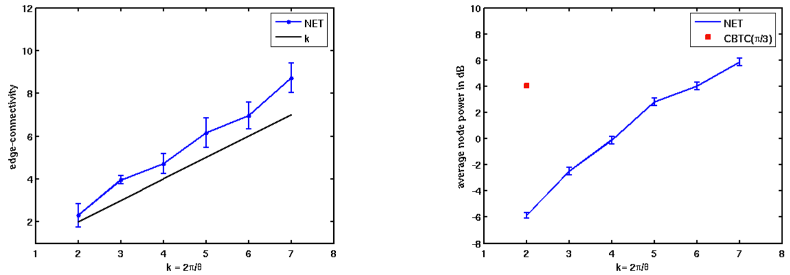
(d) (e) (f)

Fig. 14. For $\theta \leq \frac{2\pi}{3}$, the communication graph is a supergraph of RNG. (a) Communication Graph (b) RNG (c) Difference for $\theta = \frac{2\pi}{3}$. (d) Communication Graph (e) RNG (f) Difference for $\theta = \frac{2\pi}{4}$



(a) $\theta = \pi$ (b) $\theta = \frac{2\pi}{6}$

Fig. 15. Comparison of *NET* graphs and CBTC(α) for a uniform random network of 500 nodes. For 2-edge-connectivity, (a) *NET* graph requires sector angle $\theta = \pi$ and average power $-5.91dB$ while (b) CBTC requires $\theta = \frac{2\pi}{6}$ and average power of $4.10dB$.



(a) edge-connectivity (b) average power

Fig. 16. Simulation results of power control based on *NET* graphs for a uniform random network of 500 nodes averaged over 100 runs. The edge-connectivity is greater than the $\frac{2\pi}{k}$ lower bound derived in Theorem 3.4. The average communication power for achieving 2-connectivity using CBTC(α) is $4dB$ (red square) compared to $-6dB$ using *NET* graphs.

scheduling mechanisms in depth, it is possible to think of designs where densely deployed nodes make sleep/wake decisions in a distributed manner by using local, pair-wise negotiations based on combinations of *NET* condition satisfaction and other criteria.

Deployment and repair of static networks by mobile robots has been presented in [38], [39]. Typically, such algorithms have focussed on the control strategies for the robot rather than properties of the network. By providing conditions that are distributed and can be easily combined with robot controllers, our work provides a means of integrating the two objectives. Self deployment of networks of robots, where the key focus is on maximizing sensing coverage, has been well studied. Cortes *et al* [17], present a Voronoi partition based algorithm that is distributed and guaranteed to maximize coverage. Wang, *et al.*, [40] also use a Voronoi based approach to redistribute densely deployed mobile nodes to sparsely deployed areas and improve coverage. Potential field based deployment algorithms also maximize coverage [27] and have been extended to impose local constraints such as minimum degree of each node [28]. Simulations in [28] show that constraining the degree can result in the deployed network being connected with high probability. An incremental deployment algorithm that ensures line-of-sight connectivity has been presented in [41]. The above algorithms, like the *NET* based distributed deployment, require information about angle and distance to neighbors. They maximize coverage but do not guarantee network connectivity.

Topology control in 3D is significantly more complex and is relatively less studied in the literature. In [11], an algorithm has been presented to extend CBTC [9] to 3D. The computational complexity at each node is $O(d^3 log(d))$, d being the average node degree. The spherical Delaunay triangulation based algorithm that we present has complexity $O(dlog(d)$. Our algorithm has been implemented in [42] along with another technique where CBTC is applied on projections of 3D points on a 2D plane. Simulation results presented show that both techniques result in retaining connectivity with high probability. In this paper, we prove that *NET3D* guarantees k-connectivity for $\theta \leq \frac{4\pi}{k} < 2\pi$. In XTC [43] links with poor quality that can be substituted with multi-hop paths with better quality, are incrementally deleted. It does not use the disk assumption or angular information; given an initially connected network in 3D, it can retain connectivity.

VIII. Summary and Conclusions

This paper addresses distributed topology control using a construct called Neighbor-Every-Theta (*NET*) graphs. These graphs have the following two properties that allow for simple and practical algorithm design:

- tunable connectivity based on a single parameter, the sector angle θ
- connectivity guarantees do not depend on communication model

NET graphs are such that each node has at least one neighbor in every θ sector around it. We prove that for a given $\theta < \pi$, *NET* graphs have an edge connectivity of at least $\lfloor \frac{2\pi}{\theta} \rfloor$, except in pathological cases where the network can be partitioned close to its boundary. This property holds even in cases when the communication model is irregular. For the special case of an idealized disk communication model, it is shown that for specific values of θ, *NET* graphs contain proximity graphs such as the relative neighborhood graphs that are well known to be desirable

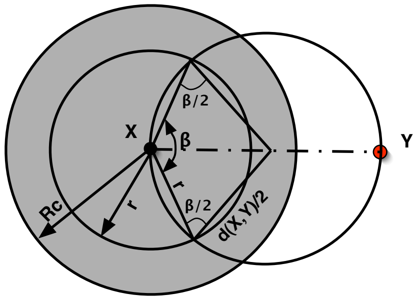


Fig. 17. Sector condition for GG

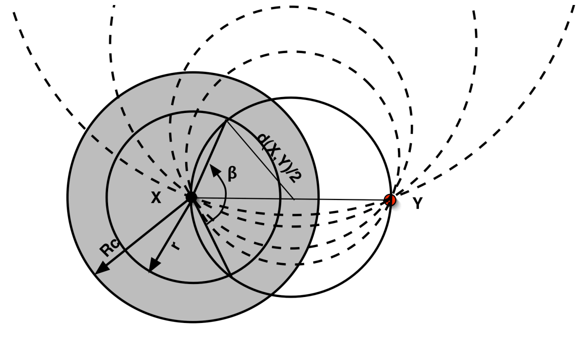


Fig. 18. Sector condition for DelG

communication topologies. We also prove the symmetric neighbor placement condition for maximizing sensing coverage under the *NET* condition.

To concretely demonstrate the use of *NET* graphs for tunable topology control, we consider two scenarios:

- For deployment of mobile nodes, we develop a distributed controller that maximizes sensing coverage while maintaining local sector conditions. This controller is based on virtual potential fields, where a combination of attractive and repulsive forces decides the motion, and pair-wise negotiations between communicating neighbors. Simulations on the Player/Stage platform provide controller performance evaluation and also serve to substantiate the analysis.
- For power control in a static network, we implement a typical protocol using satisfaction of *NET* condition at each internal node as the termination criterion and show that a power-efficient network with edge connectivity $\frac{2\pi}{\theta}$ can be achieved even with realistic, irregular links.

Lastly, we consider *NET* graphs in three dimensions and study their connectivity properties. It is our conjecture that for a given solid angle $\theta < 2\pi$, *NET3D* graphs have an edge connectivity of at least $\lfloor \frac{2\pi}{\theta} \rfloor$ when partitions close to the boundary are ignored. There is very little earlier work on topology control in $3D$. Several geometric computations become complex and even intractable when extended from $2D$ to $3D$. We have developed an efficient algorithm for determining the largest empty cone around a node in $3D$ based on spherical Delaunay triangulations. The running time is $O(dlogd)$ for average node density d, which is a significant improvement over the earlier $O(d^3 logd)$ algorithm proposed in the context of CBTC [11]. The new algorithm can be used to extend several sector based topology control algorithms to $3D$. Topology control in 3D is an important area for further research.

Appendix

Proof: [of Theorem 3.9] Consider a node X and a node Y outside $C(X, R_c)$. For any $r \le R_c$, let β be the angle subtended by the intersection area $C(X,r) \cap C(\frac{X+Y}{2}, \frac{d(X,Y)}{2})$ at X. From Fig.17 we have,

$$\frac{r}{2} = \frac{d(X,Y)}{2} cos(\frac{\beta}{2}) \Rightarrow \beta = 2\arccos(\frac{r}{d(X,Y)}).$$

Since $\arccos$ is strictly decreasing in $[0,1]$ and $d(X,Y) \in (R_c, \infty)$, the smallest angle subtended is

$$\theta = \inf \beta = 2\arccos(\frac{r}{R_c}). \tag{3}$$

Suppose x has at least one neighbor in every

$$\theta = 2\arccos(\frac{r}{R_c})$$

sector of $C(X,r)$. By definition, the edge $(X,Y) \notin GG(V)$ if

$$C(\frac{d(X,Y)}{2}, \frac{d(X,Y)}{2},) \cap V \neq \emptyset. \tag{4}$$

(4) is satisfied for any choice of Y if there is a node $Z \in V$ in the θ sector of $C(X,r)$. By premise such a Z exists and hence the GG will not have any edge lengths greater than R_c. This implies that $GG(V) \subseteq DG(V, E, R_c)$.□ ■

Proof: [of Theorem 3.10] Consider all circles passing through X and Y. The smallest is $C(\frac{X+Y}{2}, \frac{d(X,Y)}{2})$ and the largest are two circles of infinite radius of which the straight line through X and Y is an arc. From Fig.18, it can be seen that the intersections of these circles with $C(X,r)$ subtend an angle which is at least θ at X, where

$$\theta = \inf \beta = 2\arccos(\frac{r}{R_c}).$$

By premise, there is at least one node in every θ sector of $C(X,r)$. Therefore, every circle passing through X and Y will contain another node. So,

1) there is no $Z \in V$ such that the $\triangle XYZ \in DelG(V)$
2) trivially $C(\frac{X+Y}{2}, \frac{d(X,Y)}{2})$ also contains another node.

By 1,2 and definition of DelG, the edge $(X,Y) \notin DelG(V)$. Since the choice of Y is arbitrary, there are no edge lengths greater than R_c in the DelG. Hence $DelG \subseteq DG(V, E, R_c)$. □ ■

References

[1] P. Santi, "Topology control in wireless ad hoc and sensor networks," *ACM Computing Survey*, vol. 37, no. 2, pp. 164–194, 2005.
[2] XY Li, *Ad Hoc Networking*, chapter Topology Control in Wireless Ad Hoc Networks, IEEE Press, 2003.
[3] X. Li, I. Stojmenovic, and Y. Wang, "Partial delaunay triangulation and degree limited localized bluetooth scatternet formation," *IEEE Transactions Journal on Parallel and Distributed Systems*, vol. 15, no. 4, pp. 350–361, April 2004.
[4] K. Chintalapudi, J. Paek, N. Kothari, S. Rangwala, J. Caffrey, R. Govindan, E. Johnson, and S. Masri, "Monitoring civil structures with a wireless sensor network," *IEEE Internet Computing*, vol. 10, no. 2, pp. 26–34, Mar-Apr 2006.
[5] G. S. Sukhatme, A. Dhariwal, B. Zhang, C. Oberg, B. Stauffer, and D. A. Caron, "The design and development of a wireless robotic networked aquatic microbial observing system," *Environmental Engineering Science*, vol. 24, no. 2, pp. 205–215, 2006.
[6] R. Clayton, "Mase: Shallow subduction in central mexico," Progress Report, Sept. 2006.

[7] Mohammad H. Rahimi, Hardik Shah, Gaurav S. Sukhatme, John Heidemann, and Deborah Estrin, "Studying the feasibility of energy harvesting in a mobile sensor network," in *IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, Sep 2003, pp. 19–24.
[8] R. Pon, M. Batalin, J. Gordon, A. Kansal, D. Liu, M. Rahimi, L. Shirachi, Y. Yu, M. Hansen, W. J. Kaiser, M. Srivastava, G. Sukhatme, and D. Estrin, "Networked infomechanical systems: A mobile embedded networked sensor platform," in *IEEE/ACM Fourth International Conference on Information Processing in Sensor Networks*, April 2005, pp. 376–381.
[9] Li Li, Joseph Halpern, Victor Bahl, Yi-Min Wang, and Roger Wattenhofer, "Analysis of a cone-based distributed topology control algorithm for wireless multihop networks," in *Twentieth ACM Symposium on Principles of Distributed Computing (PODC), Newport, Rhode Island*, August 2001.
[10] Xiang-Yang Li, Peng-Jun Wan, and Yu Wang, "Power efficient and sparse spanner for wireless ad hoc networks," in *IEEE International Conference on Computer Communications and Networks (ICCCN01)*, 2001.
[11] Mohsen Bahramgiri, Mohammadtaghi Hajiaghayi, and Vahab S. Mirrokni, "Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multi-hop networks," *ACM/Kluwer Wireless Networks*, vol. 12, no. 2, pp. 179–188, 2006.
[12] B. P. Gerkey, R. T. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," in *In Proc. of the Intl. Conf. on Advanced Robotics (ICAR 2003),*, Coimbra, Portugal, June 2003, pp. 317–323.
[13] Sameera Poduri, Sundeep Pattem, Bhaskar Krishnamachari, and Gaurav S. Sukhatme, "Sensor network configuration and the curse of dimensionality," in *The Third IEEE Workshop on Embedded Networked Sensors*, 2006.
[14] P. Santi, *Topology control in wireless ad hoc and sensor networks*, Wiley, 2006.
[15] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Ad Hoc & Sensor Wireless Networks*, vol. 1, no. 1-2, pp. 89–123, Jan 2005, OGDC.
[16] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration for energy conservation in sensor networks," *ACM Transactions on Sensor Networks*, vol. 1, no. 1, pp. 36–72, Aug 2005, CCP.
[17] J. Cortes, S. Martinez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESAIM. Control, Optimisation & Calculus of Variations*, vol. 11, pp. 691–719, 2005.
[18] A. Kroller, S. P. Fekete, D. Pfisterer, and S. Fischer, "Deterministic boundary recognition and topology extraction for large sensor networks," in *SODA '06: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, New York, NY, USA, 2006, pp. 1000–1009, ACM Press.
[19] R. Diestel, *Graph Theory, Graduate Texts in Mathematics, second edition*, vol. 173, Springer Verlag, New York, NY, 2000.
[20] F. Xue and P. R. Kumar, "The number of neighbors needed for connectivity of wireless networks," *Wireless Networks*, vol. 10, no. 2, pp. 169–181, 2004.
[21] Vin de Silva, Robert Ghrist, and Abubakr Muhammad, "Blind swarms for coverage in 2-D," in *Proceedings of Robotics: Science and Systems*, Cambridge, USA, June 2005.
[22] Xiang-Yang Li, "Algorithmic, geometric and graphs issues in wireless networks.," *Wireless Communications and Mobile Computing*, vol. 3, no. 2, pp. 119–140, 2003.
[23] E. H. Jennings and C. M. Okino, "Topology control for efficient information dissemination in ad-hoc networks," in *Intl. Symposium on Performance Evaluation of Computer and Telecommunication Systems*, Jul 2002.
[24] Rajmohan Rajaraman, "Topology control and routing in ad hoc networks: a survey," *SIGACT News*, vol. 33, no. 2, pp. 60–73, 2002.
[25] Hyeon-Suk Na, Chung-Nim Lee, and Otfried Cheong, "Voronoi diagrams on the sphere," *Computational Geometry Theory Applications*, vol. 23, no. 2, pp. 183–194, 2002.
[26] R. Wattenhofer, L. Li, P. Bahl, and Y. M. Wang, "A cone-based distributed topology-control algorithm for wireless multi-hop networks," *IEEE/ACM Transactions on Networking*, Feb 2005.
[27] Andrew Howard, Maja J. Matarić, and Gaurav S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Proceedings of the International Symposium on Distributed Autonomous Robotic Systems*, 2002, pp. 299–308.
[28] Sameera Poduri and Gaurav S. Sukhatme, "Constrained coverage for mobile sensor networks," in *IEEE International Conference on Robotics and Automation*, New Orleans, LA, May 2004, pp. 165–172.
[29] M. M. Zavlanos and G. J. Pappas, "Potential fields for maintaining connectivity of mobile networks," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 812–816, 2007.
[30] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," in *The First ACM Conf. on Embedded Networked Sensor Systems (Sensys'03)*, Nov 2003.
[31] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *First IEEE Intl. Conf. on Sensor and Ad hoc Communications and Networks (SECON)*, Santa Clara, CA, Oct 2004.
[32] M. Zuniga, B. Krishnamachari, and R. Urgaonkar, "Realistic wireless link quality model and generator," available online, Dec 2005.
[33] A. C-C. Yao, "On constructing minimum spanning trees in k-dimensional spaces and related problems," *SIAM Journal on Computing*, vol. 11, pp. 721–736, 1982.
[34] R. M. D'Souza, D. Galvin, C. Moore, and D. Randall, "Global connectivity from local geometric constraints for sensor networks with various wireless footprints," in *International Conference on Information Processing in Sensor Networks (IPSN)*, 2006.
[35] X-Y Li, P-J Wan, Y Wang, and C-W Yi, "Fault tolerant deployment and topology control in wireless networks," in *MobiCom '03: Proceedings of the ACM/IEEE international conference on Mobile computing and networking*, 2003.
[36] Ivan Stojmenovic, "Dominating set based bluetooth scatternet formation with localized maintenance," in *IPDPS '02: Proceedings of the 16th International Parallel and Distributed Processing Symposium*, 2002, p. 122.
[37] Feng Xue and P. R. Kumar, "On the theta-coverage and connectivity of large random networks.," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2289–2299, 2006.
[38] Maxim Batalin and Gaurav S. Sukhatme, "The design and analysis of an efficient local algorithm for coverage and exploration based on sensor network deployment," *IEEE Transactions on Robotics*, 2006.
[39] P. I. Corke, S. E. Hrabar, R. Peterson, D. Rus, S. Saripalli, and G. S. Sukhatme, "Autonomous deployment and repair of a sensor network using an unmanned aerial vehicle," in *IEEE International Conference on Robotics and Automation*, Apr 2004, pp. 3602–3609.
[40] G. Wang, G. Cao, and T. L. Porta, "Movement-assisted sensor deployment," in *IEEE INFOCOM*, Hong Kong, March 2004.
[41] A. Howard, M. J. Matarić, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots Special Issue on Intelligent Embedded Systems*, vol. 13, no. 2, pp. 113–126, 2002.
[42] Amitabha Ghosh, Yi Wang, and Bhaskar Krishnamachari, "Efficient distributed topology control in 3-dimensional wireless networks," in *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2007.
[43] R. Wattenhofer and A. Zollinger, "Xtc: A practical topology control algorithm for ad-hoc networks," in *14th Intl. Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks*, 2004.