Local Barrier Coverage with Wireless Sensor Networks

Ai Chen
Department of Computer Science and Engineering
The Ohio State University
Columbus, OH 43210
chen.1109@osu.edu

Santosh Kumar
Department of Computer Science
The University of Memphis
Memphis, TN 38152
skumar4@memphis.edu

Ten H. Lai
Department of Computer Science and Engineering
The Ohio State University
Columbus, OH 43210
lai@cse.ohio-state.edu

ABSTRACT

Global barrier coverage that requires much fewer sensors than full coverage, is known to be an appropriate model of coverage for movement detection applications such as intrusion detection. However, it has been proved that given a sensor deployment, sensors can *not* locally determine whether the deployment provides global barrier coverage, making it impossible to develop localized algorithms, thus limiting its use in practice. In this paper, we introduce the concept of *local barrier coverage* to address this limitation. Motivated by the observation that movements are likely to follow a shorter path in crossing a belt region, local barrier coverage guarantees the detection of all movements whose trajectory is confined to a slice of the belt region of deployment. We prove that it is possible for individual sensors to locally determine the existence of local barrier coverage, even when the region of deployment is arbitrarily curved. Although local barrier coverage does not deterministically guarantee global barrier coverage, we show that for thin belt regions, local barrier coverage almost always provides global barrier coverage. To demonstrate that local barrier coverage can be used to design localized algorithms, we develop a novel sleep-wakeup algorithms for maximizing the network lifetime, called *Localized Barrier Coverage Protocol (LBCP)*. We prove that LBCP guarantees local barrier coverage and show that LBCP provides close to optimal enhancement in the network lifetime, while providing global barrier coverage most of the time. They outperforms an existing algorithm called Randomized Independent Sleeping (RIS) by up to 6 times.

1. INTRODUCTION

Several important applications of wireless sensors involve movement detection, such as when deploying sensors along international borders to detect illegal intrusion, around a chemical factory to detect the spread of lethal chemicals, on both sides of a gas pipeline to detect potential sabotage, etc. *Barrier coverage*, which guarantees that every movement crossing a barrier of sensors will be detected, is known to be an appropriate model of coverage for such applications [9].

Barrier coverage has several advantages over *full coverage*, a model requiring every point in the deployment region to be covered. First, barrier coverage requires much fewer sensors than full coverage [9]. Second, the *sleep-wakeup* problem, that determines a sleeping schedule for sensors to maximize the network lifetime, is polynomial-time solvable for barrier coverage even when sensor lifetimes are not equal [11]. For the full coverage model, on the other hand, the sleep-wakeup problem is NP-Hard even if sensor lifetimes are identical [15].

A major limitation of the barrier coverage model, however, is that unlike full coverage, individual sensors can *not* locally determine whether a network *does not* provide barrier coverage [9], making it impossible to develop localized algorithms. Consequently, almost all algorithms developed so far for barrier coverage, including the optimal sleep-wakeup algorithm, are centralized [11]. (The only exception is the Randomized Independent Sleeping (RIS) scheme [9], which does not require any message exchange.) Given the large scale and unattended nature of wireless sensor networks, localized algorithms are essential for scalability. A localized algorithm is also more adaptive to changes in the network, which is expected to be quite frequent in wireless sensor network due to unattended outdoor deployments. Therefore, in order to realize the benefits of the barrier coverage model in movement detection applications, there is a strong need to develop a new model that enables the development of localized algorithms, while essentially retaining the benefits of barrier coverage.

We also observe that the notion of barrier coverage [9], which we will refer to as *global barrier coverage*, requires *every* crossing path to be covered, no matter how long it is. Thus, the sensor deployment in a 50m × 500km border (belt region) as shown in Figure 1, is regarded as *not* providing global barrier coverage due to the existence of an uncovered crossing path (which is more than 499km long). In real life, intruders are highly unlikely to follow such paths; it is more likely that a short path across the belt region is taken.

Motivated by these observations, we introduce in this paper the concept of *L-local barrier coverage*. It will be formally defined in Section 4, but informally, L-local

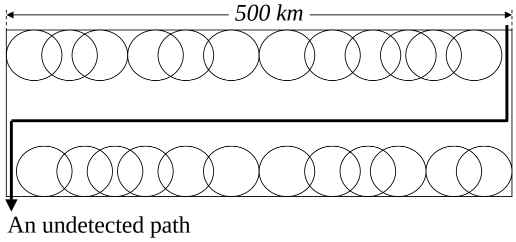

Figure 1: A belt is not global 1-barrier covered because of the existence of a long uncovered crossing path.

barrier coverage guarantees the detection of all crossing paths whose trajectory is confined to a slice (of length L) of the belt region of deployment. In other words, if the bounding box that contains the entire trajectory of a crossing path, has a length at most L, then this crossing path is guaranteed to be detected by at least one (or k) sensor(s). For example, the crossing path in Figure 2 is guaranteed to be detected since its bounding box is of length less than L, if the sensor network deployed over this belt region provides L-local barrier coverage. The concept of L-local barrier coverage not only enables the development of localized algorithms, it also generalizes the (global) barrier coverage model; when L is equal to the length of the entire deployment region, L-local barrier coverage is equivalent to global barrier coverage.

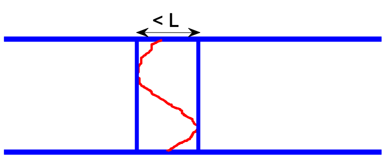

Figure 2: L-local barrier coverage ensures that crossing paths within a box of length L be detected.

A key question regarding L-local barrier coverage is how to determine whether a sensor network provides L-local barrier coverage. This question is nontrivial since there are infinitely many bounding boxes (each of length L). In this paper, we prove a theorem that allows a convenient discretization so that instead of checking each of the infinite bounding boxes to establish that a sensor network provides L-local barrier coverage, one only needs to check if the neighborhood of each sensor is barrier covered.

Although local barrier coverage does not deterministically guarantee global barrier coverage (when L is less than the length of the deployment region), we show (by simulation) that for thin belt regions, local barrier coverage almost always provides global barrier coverage. This means that for thin belts, checking locally for the existence of local barrier coverage is sufficient to ensure global barrier coverage in practice. Intuitively, this holds because as the width of the deployment region approaches zero, local barrier coverage and global barrier coverage become equivalent.

To demonstrate that local barrier coverage can be used to design localized algorithms, we develop a sleep-wakeup algorithm for extending the network lifetime, called *Localized Barrier Coverage Protocol (LBCP).* We prove that LBCP guarantees local barrier coverage and show that LBCP provides close to optimal enhancement in network lifetime, while providing global barrier coverage most of the time. It outperforms an existing algorithm called Randomized Independent Sleeping (RIS) by up to 6 times.

Organization: Section 2 describes the network model, and Section 3 mentions some related work. Section 4 constitutes the theoretical foundation of L-local barrier coverage. A critical design issue is addressed in Section 5. Section 6 describes our localized sleep-wakeup protocol and proves its correctness. Simulation results appear in Section 7. Section 8 concludes the paper. The proof of a theorem is presented in the Appendix.

2. THE NETWORK MODEL

The network model adopted in this paper is similar to that in [9]. We review here some definitions (with necessary modifications to suit the purpose of this paper).

A sensor network, N, is a collection of sensors with their locations known. We use u to denote both a sensor node as well as the point of its location. We assume that a sensor network is deployed over a belt region. An example belt region is illustrated in Figure 3. To formally define a belt region, let $d(x, y)$ denote the Euclidean distance between two points x and y; and for a point x and a curve l, let $d(x, l)$ be the distance between x and l, i.e., $d(x, l) = \min\{d(x, y) : y \in l\}$. Two curves l_1 and l_2 are said to be *parallel with separation* w if $d(x, l_2) = d(y, l_1) = w$ for all $x \in l_1$ and $y \in l_2$.

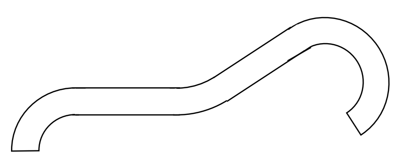

Figure 3: A general belt with two parallel boundaries

Definition 2.1. **[Belt of Width** W**]** *If l_1 and l_2 are two parallel curves with separation W, the region between l_1 and l_2 is referred to as a* belt (region) *of width W. The two curves l_1 and l_2 are the belt's* parallel boundaries*.*

For ease of presentation, we envision a belt region as roughly going from left to right. With such a convention, the belt's two parallel boundaries may be referred to as the top and the bottom boundary; and the other two boundaries, the left and the right.

Intrusion movement is assumed to occur one parallel boundary to the other. Thus, a path is said to be a *crossing path* if it crosses from one parallel boundary to the other. A crossing path is *orthogonal* if its length is equal to w, the belt's width. Orthogonal crossing paths are straight lines and, therefore, often referred to as orthogonal crossing lines. For rectangular belts, orthogonal crossing lines are parallel to the belt's left and right sides.

A point p is covered (monitored) by a node u if their Euclidian distance is less than or equal to the sensing range, denoted by r. The sensing region of a node u is the set of all points covered by u. A crossing path is *k-covered* if it intersects the sensing regions of at least k distinct sensors. Finally, a sensor network N provides *k-barrier coverage* over a deployment belt region $\mathcal{D}$ if all crossing paths through region $\mathcal{D}$ are k-covered by sensors in N.

DEFINITION 2.2. **[Coverage Graph, $\mathcal{G}(N)$]** *[9] A coverage graph of a sensor network N is constructed as follows. Let $\mathcal{G}(N) = (V, E)$. The set V consists of a vertex corresponding to each sensor. In addition, it has two virtual nodes, s and t to correspond to the left and right boundaries. An edge exists between two nodes if their sensing regions overlap in the deployment region $\mathcal{D}$. An edge exists between u and s (or t) if the sensing region of u overlaps with the left boundary (or right boundary) of the region.*

A node can be active or sleeping. When constructing a coverage graph, only active nodes are used. Using coverage graphs, the following theorem enables us to determine whether a belt region is k-barrier covered.

THEOREM 2.1. *[9] A network N provides k-barrier coverage if and only if there exist k node-disjoint paths between the two virtual nodes s and t in $\mathcal{G}(N)$.*

Remark: Although we use a disk model here for the sensing region, our results hold for all other models for which a coverage graph can be constructed. We address this in Section 4.3.

Remark: We also note here that sensors do not continuously sample the environment and every time a sensor begins to sample the environment there is some startup latency. So, a sensor may not be able to detect an intruder if the intruder just touches the sensor's sensing region. However, if we assume the intruder's maximum movement speed is known, then for a given sampling frequency and a given startup latency, a conservative, smaller-than-actual sensing range can be calculated and used such that if an intruder ever touches this conservative sensing region, then he will stay in the actual, larger sensing region for sufficient time and the sensor will detect the intruder with very high probability.

3. RELATED WORK

Although full (or blanket) coverage, which requires that every point in the deployment region be covered by sensors, has been extensively studied in the literature (e.g., in [3, 4, 6, 7, 10, 15, 16, 17, 18, 19, 20, 21]), research on barrier coverage is still in its infancy.

The concept of *barrier coverage* (which we call *global barrier coverage* in this paper) is introduced in [9]. A centralized algorithm to determine whether a network provides global barrier coverage is provided there. The problem of deriving a reliable estimate for ensuring global barrier coverage in a random deployment is solved in [2, 13].

An optimal sleep-wakeup algorithm for achieving global barrier coverage is proposed in [11]. This is a centralized algorithm. Our LBCP algorithm, on the other hand, is a localized algorithm that provides near-optimal performance, while ensuring global barrier coverage most of the time.

The model of full coverage has been extensively studied. A localized algorithm for determining whether a network does not provide full coverage is presented in [7]. Several heuristic algorithms for sleep-wakeup exist that attempt to maximize the network lifetime while maintaining full coverage [3, 6, 10, 15, 21]. As the sleep-wakeup problem is NP-Hard, no polynomial-time optimal algorithm (centralized or local) exist for this model.

Since local determination of global barrier coverage is not possible [9], our localized algorithm of course cannot deterministically guarantee global barrier coverage. However, it can ensure local barrier coverage for appropriately selected values of L, and thereby ensure that all crossing paths that are confined to a box of length at most L will surely be detected. In the unlikely event that a crossing path stretches to more than a length of L across the belt's length, it may still be detected, but is not guaranteed. In summary, since most crossing paths are likely to follow the shortest or close-to-shortest paths, our algorithm is practically sufficient for ensuring barrier coverage, while extending the network lifetime to close to optimal via local computation.

4. L-LOCAL BARRIER COVERAGE

The concept of L-local barrier coverage is introduced in Section 1. In this section, we formalize this new concept and address a key question: *Given a sensor deployment over a belt region, how does one determine if the deployment provides L-local barrier coverage?*

L-local barrier coverage and its properties are easier to describe and understand in a rectangular belt than in a general belt. So we begin with rectangular belts and then generalize the results for general belts. We first employ the sensing disk model, and then remark on the modifications necessary when other sensing models are used.

4.1 Rectangular Belts

We begin with some definitions. Consider a rectangular region with sensors deployed over it. Recall the definitions of *parallel boundaries* and *orthogonal crossing lines* made in Section 2. Figure 4 illustrates the following definitions.

DEFINITION 4.1. **[Zone]** *A zone, Z, is a slice of the belt region. Two of its edges coincide with the belt's two parallel boundaries (referred to as its parallel boundaries), and the other two edges are orthogonal crossing lines (referred to as its orthogonal boundaries).*

In rectangular belts, the length of a zone Z, denoted by L_Z, is the distance between the zone's two orthogonal boundaries. Note that a zone's two orthogonal boundaries happen to be parallel here, but as will be seen later, they are not necessarily parallel in a general belt.

DEFINITION 4.2. **[L-zone]** *For a positive number L, an L-zone is a zone of length L.*

DEFINITION 4.3. **[$2d$-zone(u)]** *For a positive value d, the $2d$-zone of a sensor node u, denoted by $2d$-zone(u), is an L-zone with $L = 2d$, in which the orthogonal crossing line passing through u divides the L-zone into two sections of equal length (each of length d).*

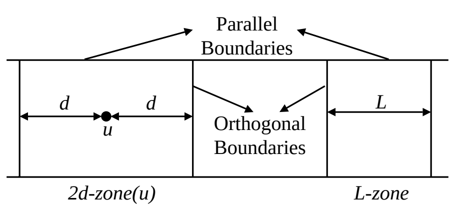

Figure 4: L-zone and $2d$-zone(u)

Recall the definition of k-barrier coverage in Sec. 2.

DEFINITION 4.4. **[L-Local k-Barrier Coverage]** *For a positive number L and a positive integer k, a belt region is said to be L-local k-barrier covered if every L-zone in the region is k-barrier covered.*

Note that if a network provides L-local k-barrier coverage, then it provides M-local k-barrier coverage as well, for all $0 \leq M \leq L$. When $k = 1$, L-local k-barrier coverage is simply referred to as L-local barrier coverage.

We now address the above mentioned question of how to determine if a belt region is L-local k-barrier covered. We begin with a couple of lemmas. The first lemma indicates a condition under which any wide enough L-zone must contain at least one active sensor. Recall that r indicates the sensing range of each sensor node.

LEMMA 4.1. *Consider a rectangular belt with at least one active node. If $d > r$ and $2d$-zone(u) for every active node u is k-barrier covered, then every L-zone with $L \geq 2r$, must contain at least one active node.*

Proof: Assume there is no node in an L-zone with $L \geq 2r$. Consider the node a closest to this L-zone. Without loss of generality, assume a is to the left of the L-zone. Then, the nodes to the right of a must be to the right of the L-zone. Therefore, the distances between any nodes to the right of a and any nodes to the left of a (including a and those on the same orthogonal crossing line as a) are larger than L. Since $L \geq 2r$, there is no overlap between the coverage area of the nodes to the right of a and that of the nodes to the left of a (including a and those on the same orthogonal crossing line as a). Then the nodes on the left side of a (including a and those on the same orthogonal crossing line) should provide k-barrier coverage for $2d$-zone(a), which is impossible because $d > r$. Therefore, there must be at least one active node in every L-zone with $L \geq 2r$. □

Even if two zones with overlap are individually k-barrier covered, their union as a single zone is not necessarily k-barrier covered. We prove in the following lemma a condition under which the union of two zones is k-barrier covered. The condition is that one of the two zones is k-barrier covered in a special way, and the other zone is relatively narrow.

LEMMA 4.2. *Let A and B be two zones with intersection, with A of length $L_A \geq r$ and B of length $L_B \leq r$ as shown in Figure 5. Suppose A is k-barrier covered, but no node in $A - B$ covers A's orthogonal boundary l_3 that is contained in B. Then, $A \cup B$ is k-barrier covered.*

Proof: Because A is k-barrier covered, there must be at least k nodes covering A's orthogonal boundary l_3. Since $L_A \geq r$, these nodes must not be to the left of A. Furthermore, because no node in $A - B$ covers l_3, these nodes must be in the zone B or to the left of B. Since $L_B \leq r$, these nodes' sensing disks must also cover B's orthogonal boundary l_4 (this boundary is also an orthogonal boundary of $A \cup B$). Then, the coverage graph for A is a subgraph of the coverage graph for $A \cup B$ (by Definition 2.2). Since A is k-barrier covered, $A \cup B$ is also k-barrier covered (by Theorem 2.1). □

The following theorem indicates when, and for what value of L, we can conclude that a rectangular belt is L-local k-barrier covered.

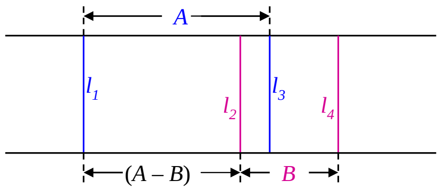

Figure 5: Visualizing the proofs of Lemma 4.2

THEOREM 4.1. *Consider a rectangular belt with at least one active sensor node. If* $2d$*-zone*(u) *for every active node* u *is* k*-barrier covered for some* $d > r$*, then the entire belt is* L*-local* k*-barrier covered, with* $L = \max\{2d - 2r, d + r\}$*.*

Proof: Consider two possible cases: $d \geq 3r$ or $r < d < 3r$.

Case 1: $d \geq 3r$. In this case, $\max\{2d - 2r, d + r\} = 2d - 2r$. Let $L_1 = 2d - 2r$. We need to show that every L_1-zone is k-barrier covered. Given any L_1-zone as illustrated in Figure 6, there must be at least one active node in its center $2r$-zone according to Lemma 4.1. Let's say node b is in the $2r$-zone. Then, L_1-zone $\subseteq$ $2d$-zone(b) and the k-barrier coverage of $2d$-zone(b) implies the k-barrier coverage of the L_1-zone.

Case 2: $r < d < 3r$. In this case, $\max\{2d - 2r, d + r\} = d + r$. We will show every L_2-zone is k-barrier covered, where $L_2 = d + r > 2r$. Given any L_2-zone as shown in Figure 6, there must be an active node in the L_2*-zone* according to Lemma 4.1. If there is a node n in the center $(d - r)$-zone, then L_2-zone $\subseteq$ $2d$-zone(n) and therefore the k-barrier coverage of $2d$-zone(n) implies the k-barrier coverage of the L_2-zone.

If there is no node in the center $(d - r)$-zone, then there are nodes in the left or in the right r-zone. Without loss of generality, assume there are nodes in the left r-zone, and let m be the one closest to the center $(d-r)$-zone. By the assumption, $2d$-zone(m) is k-barrier covered. For ease of presentation, let A be $2d$-zone(m) and B be the right r-zone. Since there is no node in the center $(d - r)$-zone and m is the node closest to the center $(d - r)$-zone and $d > r$, no node in $A - B$ ever covers A's orthogonal boundary in B. The length of A is $2d > r$ and the length of B is r. According to Lemma 4.2, $A \cup B$ is k-barrier covered. Since L_2-zone $\subseteq A \cup B$, the L_2-zone is also k-barrier covered. This completes the proof. □

We require $d > r$ in Theorem 4.1. If $d \leq r$, then the k-barrier coverage of every $2d$*-zone*(u) does not imply the k-barrier coverage of every L-zone. We prove this in the next Theorem.

THEOREM 4.2. *If* $d \leq r$*, then for any given value of* $L > 0$*, there exists a sensor deployment such that even if* $2d$*-zone*(u) *for every node* u *is* k*-barrier covered, the belt region is not* L*-local* k*-barrier covered.*

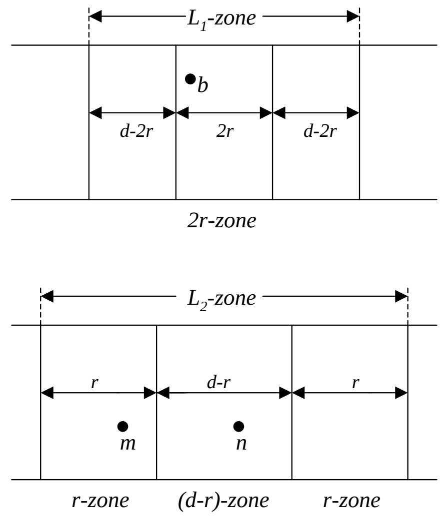

Figure 6: Visualizing the proofs of Theorem 4.1

Proof: Consider a rectangular belt of length $2(L + r)$ or more. Place k sensors in the belt along a same orthogonal crossing line, but otherwise no other nodes in the belt. Now, $2d$-zone(u) for every node u is k-barrier covered, since $d < r$; but there apparently exists an L-zone which is not even 1-barrier covered. □

The L established in Theorem 4.1 is the largest such value possible. In other words, if $L > \max\{2(d-r), d+r\}$, then even if $2d$-zone(u) for every active node u is k-barrier covered, the sensor network does not necessarily provide L-local k-barrier coverage to the entire belt. We prove this in the next theorem.

THEOREM 4.3. *If* $L > \max\{2d-2r, d+r\}$*, then even if* $d > r$*, there exists a sensor deployment such that* $2d$*-zone*(u) *for every active node is* k*-barrier covered but the region is not* L*-local* k*-barrier covered.*

Proof: If $d \geq 3r$, then $2d-2r \geq d+r$ and $L > 2d-2r$. First consider $k = 1$. The example in Figure 7(A) shows that $2d$-zone(u) for all nodes u is 1-barrier covered but there exists an L-zone that is not barrier covered. For $k > 1$, replicate each node in Figure 7(A) k times. Now, the $2d$-zone of each active node is k-barrier covered but there exists an L-zone that is not even 1-barrier covered.

If $r < d < 3r$, then $2d - 2r < d + r$ and $L > d + r$. The argument for this case is similar to the previous case except that we now use the deployment shown in Figure 7(B). □

4.2 General Belts

Now, consider a belt of any shape as defined in Definition 2.1. For mathematical reasons, assume that curvature[1] exists everywhere on the belt's parallel boundaries, and that the largest curvature value over the en-

[1]Informally, for a curve in the plane, if curvature exists at

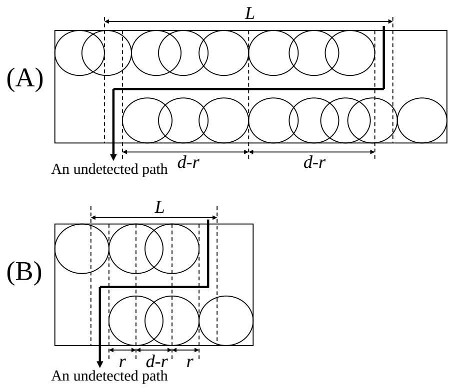

Figure 7: Visualizing the proofs of Theorem 4.3.

tire belt boundaries is $1/R$ for some $R > 0$. Assume $R \gg r$, which informally means that the belt is not steeply curved as compared to the sensing range r.

We need to define L-zone for general belts, where, as in the case of rectangular belt, L is supposed to indicate the zone's length. Now there is a question — a zone's two parallel boundaries in a general belt are mostly of different lengths; which length should be regarded as *the* length of the zone? We resolve this issue by measuring the zone's middle line, as defined below.

Definition 4.5. [**Middle Line**] *The middle line of a belt is the curve parallel to, and at the middle between, the belt's two parallel boundaries. (So, the width between the middle line and either parallel boundary is one half of the belt's width.)*

Definition 4.6. [L**-zone**] *For a positive value L, an L-zone on a belt is a zone with the length of the zone's middle line being L.*

Definition 4.7. [$2d$**-zone**(u)] *For a positive value d, the $2d$-zone of a sensor node u, denoted by $2d$-zone(u), is an L-zone with $L = 2d$, in which the orthogonal crossing line passing through u divides the L-zone's middle line into two sections of equal length (each of length d).*

These concepts are illustrated in Figure 8 using a circular belt. Note that Definitions 4.6 and 4.7 are consistent with Definitions 4.2 and 4.3, respectively. The length of a zone's middle line is equal to the distance between this zone's two orthogonal boundaries in a rectangular belt.

With L-zone and $2d$-zone defined as above, let "L-local k-barrier covered" be defined as before (Definition 4.4).

a point, then there is a circle that best fits with the curve at that point. The reciprocal of that circle's radius is the curvature value at that point. Thus, the curvature value of any point on a circle with radius R is $1/R$, and the curvature value of any point on a straight line is 0.

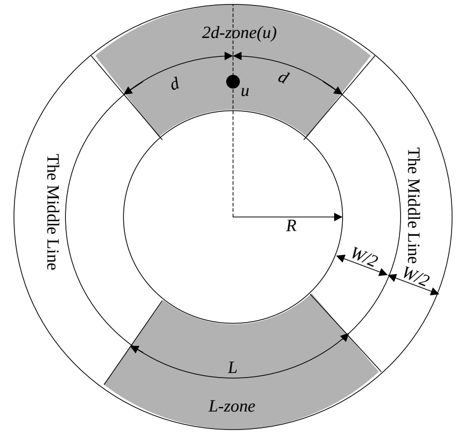

Figure 8: Middle line, L-zone, $2d$-zone

The following theorem generalizes Theorem 4.1, and can be proved along a similar line of reasoning, even though the details are considerably more involved. In the theorem, W refers to the belt's width, and as before, r is the sensing range. We write $\arcsin(r/2R)$ to mean $\arcsin(r/(2R))$ — the parentheses are omitted for simplicity. The quantity $2(R + W/2)\arcsin(r/2R)$ and the formula for L can be more easily explained using a circular belt. In a circular belt, the inner circle's radius is R. If a zone (i.e., its middle line) is of length $2(R + W/2)\arcsin(r/2R)$, then the zone's inner boundary's chord is of length r. Similarly, a zone of length $2(R + W/2)\arcsin(2r/2R)$ has its inner boundary's chord being of length $2r$. If a zone is of length $(R + W/2)r/(R + W)$, then the zone's outer boundary is of length r. These formulas are explained in more detail in the theorem's proof, which we defer to the Appendix in order not to interrupt the flow of presenting our main ideas.

Theorem 4.4. *Consider a belt region with at least one active node deployed in it. Let $1/R$ be the largest curvature value on the belt's two parallel boundaries. If $2d$-zone(u) for every active node u in this belt is k-barrier covered for some $d > 2(R + W/2)\arcsin(r/2R)$, then the entire belt is L-local k-barrier covered, where L equals*

$$\max\left\{2d - 2\left(R + \frac{W}{2}\right)\arcsin\left(\frac{2r}{2R}\right), d + \left(\frac{(R + W/2)r}{R + W}\right)\right\}$$

Note that Theorem 4.4 is indeed a generalization of Theorem 4.1. As R approaches infinity, the belt becomes rectangular and the L in Theorem 4.4 approaches the L in Theorem 4.1.

4.3 Other Sensing Models

In Sections 4.1 and 4.2, we assume that the sensing region is a disk. However, all of our results can be easily extended to other sensing models. For example, suppose that the sensing ranges are different in different directions, but there exist a maximum and a minimum value for the sensing ranges. Let $r_{\max}$ be the maximum value of the sensing ranges and $r_{\min}$ be the minimum. The following theorems correspond to Theorems 4.1 and 4.4, and can be proved in a similar fashion.

THEOREM 4.5. *Consider a rectangular belt with at least one active node deployed in it. If the* $2d$*-zone*(u) *for every active node* u *in this belt is* k*-barrier covered for some* $d > r_{\max}$*, then the entire belt is* L*-local* k*-barrier covered, with* $L = \max\{2d - 2r_{\max}, d + r_{\min}\}$.

THEOREM 4.6. *Consider a belt region with at least one active node deployed in it. Let* $1/R$ *be the largest curvature value on the two boundaries of the belt. If* $2d$*-zone*(u) *for every active node* u *in this belt is* k*-barrier covered for some* $d > 2(R+W/2)\arcsin(r_{\max}/2R)$*, then the entire belt is* L*-local* k*-barrier covered, where* L *equals*

$$\max\left\{2d - 2\left(R + \frac{W}{2}\right)\arcsin\left(\frac{2r_{\max}}{2R}\right), d + \frac{(R+W/2)r_{\min}}{R+W}\right\}$$

Note that we always get $L \geq d$ in Theorems 4.5 and 4.6 regardless of which sensing model is used as long as $r_{\min} \geq 0$. Therefore, we always can achieve any desired value of L if we make the value of d large enough.

5. IDENTIFYING A 2D-ZONE

Theorems 4.1, 4.4, and 4.6 indicate that in order to determine whether a network provides local barrier coverage, it is sufficient to check whether for some appropriate value d, the $2d$-zone of each node is barrier covered. If a sensor is able to identify its $2d$-zone, it can construct a coverage graph and then determine whether its $2d$-zone is k-barrier covered (by using Theorem 2.1).

The main issue, therefore, is to equip sensors with a mechanism to locally determine their $2d$-zones. This job is trivial if the belt is rectangular or circular. For a general belt, especially when it is extremely long such as one along an international border, it may sometimes be unrealistic to assume that every sensor has knowledge of the belt's curvatures in its neighborhood. In such cases, it is nontrivial to recognize a node's $2d$-zone. We develop a heuristic for this task. Sensor nodes are assumed to have knowledge of the belt's width (W) but not its shape.

Consider a node u which needs to identify its $2d$-zone. It is difficult, if not impossible, to recognize $2d$-zone(u) without knowing the belt's boundaries. Fortunately, as will be seen in Section 6, our purpose of recognizing a $2d$-zone is to ensure that it is k-barrier covered, thus it suffices for us to (1) identify a region that encloses the $2d$-zone and (2)ensure that the identified region is k-barrier covered. This will imply that the original $2d$-zone in question is k-barrier covered.

Our heuristic for local identification/estimation of $2d$-zones is outlined in Figure 9 as Procedure IDENTIFY, which will be explained shortly. When invoked by a node u, the procedure normally returns an estimation of $2d$-zone(u) (which encloses the actual $2d$-zone(u)). However, in some situations, the procedure may detect that some node's $2d$-zone is not k-barrier covered, in which case it will just report this information (without returning a $2d$-zone(u))). (Procedure IDENTIFY(u) will be used in Procedure INITIAL of the next section.)

Procedure IDENTIFY(u)

1. Node u computes the values of r_1, r_2, r_3 using the values of W, d, r, and R as described. Let $r' = \max\{r_1, r_2\}$.
2. Node u then finds two nodes p and q such that $r' \leq d(p, u) \leq r_3$, $r' \leq d(q, u) \leq r_3$, and $\angle puq \geq \pi/2$. If it can not find two such nodes, then it stops and reports that at least one node's $2d$-zone is not k-barrier covered by the network.
3. It draws a line l_1 perpendicular to $\overline{pu}$ and r_1 away from u. Similarly, it draws a line l_2 perpendicular to $\overline{qu}$ and r_1 away from u.
4. Node u uses the slice S of the belt region between l_1 and l_2 as an estimate for $2d$-zone(u).

Figure 9: Procedure IDENTIFY

Now we explain the ideas behind procedure IDENTITY. Conceptually, a region enclosing $2d$-zone(u) can be found as follows. Choose a sufficiently large value r_1 such that the entire $2d$-zone(u) is enclosed by the circle C_1 of radius r_1, centered at u. Let l_1 and l_2 be two lines that are tangent to C_1 on the opposite sides of the orthogonal crossing line passing through u, and each intersect the two long boundaries of the belt. The section S of the belt region between l_1 and l_2 evidently contains $2d$-zone(u), as illustrated in Figure 10.

To carry out the above scheme, there are two essential tasks: 1) estimating the value of r_1, and 2) identifying the two lines l_1 and l_2. (We do not need to identify the top and bottom boundaries of region S, because they play no role in constructing the coverage graph.)

For the value of r_1, we want it to be as small as possible. Even though $r_1 = W + d$ is an obvious valid estimate, $r_1 = \sqrt{\left(\frac{d(R+W)}{R+W/2}\right)^2 + W^2}$ can be easily verified (using elementary geometry) to be a tighter bound, where $1/R$ is the biggest curvature value on the long boundaries of the belt. Step 1 of IDENTIFY calculates r_1 using this formula. The other two values, r_2 and r_3, will be explained soon.

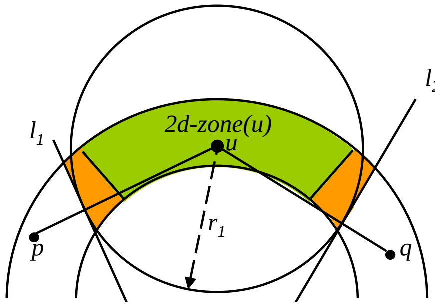

Figure 10: Identified $2d$-zone(u) is the slice of the belt region between lines l_1 and l_2, which contains the real $2d$-zone(u).

To address the second issue, which is to select l_1 and l_2 (the task of Step 3 of IDENTIFY), the main idea of our heuristic is for u to choose two far away nodes p and q that are on the opposite sides of the orthogonal crossing line passing through u and satisfy the two conditions: $d(p,u) \geq r_1$ and $d(q,u) \geq r_1$. We will shortly discuss how to identify two such nodes. Line l_1 then is a line that is perpendicular to $\overline{pu}$ and at a distance of r_1 from u. Similarly, line l_2 is perpendicular to $\overline{qu}$ and r_1 away from u. (See Figure 10.) Then, we claim that the slice S of the belt region between l_1 and l_2 contains $2d$-zone(u).

We now describe how to find the two nodes p and q — we first present a basic scheme and then a refined version. Meeting the requirement $d(p,u) \geq r_1$ and $d(q,u) \geq r_1$ is easy. But ensuring that p and q are on the opposite sides of the orthogonal crossing line passing through u is non-trivial. (Let $l(u)$ denote this crossing line.) Intuitively, if the curvature of the belt is not too large, then two far away nodes on the opposite sides of $l(u)$ should form a large angle at u. Indeed, if we assume $R \gg W$, then there exist two values r_2 and r_2' such that for any two nodes p and q in the belt region with $r_2 \leq d(p,u) \leq r_2'$ and $r_2 \leq d(q,u) \leq r_2'$, it holds that $\angle puq \geq \pi/2$ if and only if p and q are on the opposite sides of $l(u)$. Using some elementary geometry, it can be shown that if $R \geq 3W$, then we can set $r_2 = \sqrt{2}\left(W + \frac{2W^2}{R}\right)$ and $r_2' = \sqrt{2}\left(R - \frac{2W^2}{R}\right)$. Now, u can select two nodes for p and q such that $r' \leq d(p,u) \leq r_2'$, $r' \leq d(q,u) \leq r_2'$, and $\angle puq \geq \pi/2$, where $r' = \max\{r_1, r_2\}$.

We now discuss an optimization to the process of searching for p and q (this constitutes Step 2 of IDENTIFY). Since R may be considerably larger than W, so may r_2' much larger than r'. In that case, letting u search all nodes in the range between r' and r_2' will be inefficient. To cut down the search domain, we will use a smaller value r_3 in place of r_2'. Lemma 5.1 is for this purpose. Note that we attempt to keep r_3 as small as possible.

LEMMA 5.1. *Let r' be as described above and $r_3 = \sqrt{(2r+r')^2 + 2W^2}$. Consider two circles C' and C_3 centered at u with radii r', r_3, respectively. Let S_1 and S_2 be the two slices of the belt region that are between C' and C_3. If there is no node in S_1 or S_2, then at least one node's $2d$-zone is not k-barrier covered in the network.*

Proof: According to Lemma A.2 in the appendix, if the $2d$-zone(u) of every active node u is k-barrier covered, then each L-zone with $L \geq 2(R + W/2)\arcsin(2r/2R)$ must contain at least one active node. Using some elementary geometry, it can be shown that if $r_3 = \sqrt{(2r+r')^2 + 2W^2}$, the maximum size of the zone in S_1 (or S_2) is larger than or equal to $2(R+W/2)\arcsin(2r/2R)$. Therefore, If there is no node in S_1 or S_2, then at least one node's $2d$-zone is not k-barrier covered in the network. □

Now, if all nodes' $2d$-zones are k-barrier covered, then p and q exist in S_1 and S_2, respectively (by Lemma 5.1). Otherwise, at least one node's $2d$-zone is not k-barrier covered in the network. It can be checked that if $R \gg r$, $R \gg d$, and $R \gg W$, then $r_2' > r_3$, and therefore a node u can use r_3 in place of r_2' in its search for p and q.

We summarize the above discussion as a theorem for subsequent references.

THEOREM 5.1. *Suppose that at the beginning when all nodes are active, every node's actual $2d$-zone is k-barrier covered. Then for every node u, Identify(u) returns an estimated $2d$-zone(u) that encloses the actual $2d$-zone(u).*

6. LOCALIZED SLEEP-WAKEUP PROTOCOLS

In this section, we use local barrier coverage concept to design a localized sleep-wakeup algorithm, called *Localized Barrier Coverage Protocol (LBCP)*, for barrier coverage to maximize the network lifetime. In Section 7, we will show that the protocol usually have close-to-optimal performances and provide global barrier coverage most of the time for thin belt regions.

6.1 Assumptions

We first state some assumptions. We assume that each node has a unique ID as is common in newer platforms such as TelosB [14]. We also assume that the network has been localized so that each node knows its own location. In the event of localization inaccuracies, the identified $2d$-zone of a node u may not contain the real $2d$-zone(u). However, the error of the location, denoted by ϵ, only slightly affects the performance of LBCP. For example, in a rectangular belt we only need to increase the value of d to $d' = d + \epsilon$ to insure that the identified $2d'$-zone of a node u contains the real $2d$-zone(u). Further, we assume that two nodes u and v are able to communicate with each other if u is in v's (identified) $2d$-zone or v is in u's (identified) $2d$-zone. With the communication range increasing to $1000ft = 304.8m$ (see Mica2 data sheet [1]), this should be possible in thin belts. We also assume that every node is able to estimate its remaining lifetime (if it stays active) by observing its battery drainage. Battery drainage rate can be observed in recent mote platforms [12]. Finally, we assume that the MAC protocol does not introduce too much latency; all LBCP packets are sent or received almost immediately.

6.2 Protocol

We now describe the LBCP protocol. The protocol consists of a main program and three procedures called INITIAL, ACTIVE, and WAKEUP. At any time, each node is in one of three states: *active, sleeping*, or *waking-up*. As shown in Figure 11, initially, every node is in the active state and executes Procedure INITIAL to perform some preliminary work such as initializing variables. Every node executes Procedure INITIAL only once. Then, every active node periodically invokes Procedure ACTIVE to check if it should stays active or go to sleep. During Procedure ACTIVE, a node may decide to go to sleep due to node redundancy. It is important that a node's falling asleep should not jeopardize the belt's L-local k-barrier covered property. A sleeping node periodically wakes up and invokes Procedure WAKEUP to check if it should become active or go back to sleep.

Protocol LBCP

1. Initially, every node is active and calls Procedure INITIAL once and only once.
2. After initialization, every active node periodically invokes Procedure ACTIVE to decide whether to stay active or go to sleep. In the latter case, the node becomes sleeping.
3. Every sleeping node periodically wakes up, enters the waking-up state, and invokes Procedure WAKEUP.

Figure 11: Protocol LBCP

Procedure INITIAL is shown in figure 12. Its main functions are for each node u 1) to initialize some variables, 2) to compute its $2d$-zone, 3) to recognize the set of nodes $N_u = \{v : v \in 2d\text{-zone}(u) \text{ or } u \in 2d\text{-zone}(v)\}$, and 4) to collect information about those nodes in N_u. Thus, in the first step every node u initializes a set $A(u) = \phi$, which will be used in Procedure ACTIVE. In the second step, the node invokes Procedure IDENTIFY (as described in Figure 9) to compute $2d$-zone(u). As remarked in the previous section, the *computed* $2d$-zone(u) encloses but not necessarily equals the *actual* $2d$-zone(u). In this procedure and throughout the rest of this section, the term "$2d$-zones" refers to *computed* $2d$-zones unless otherwise stated. If IDENTIFY fails to return a $2d$-zone, that means the current sensor deployment is insufficient to provide L-local barrier coverage. Node u sends out a message to indicate this fact. In this case, the value of L needs to be reduced or more sensors need to be deployed. If on the other hand IDENTIFY succeeds in returning a $2d$-zone, node u sends out an *Inform_I* packet to all other nodes in $2d$-zone(u), informing them of u's ID, position, and lifetime. The second half of Step 3 is more subtle. As computed $2d$-zones may differ from actual $2d$-zones, for any two nodes x, y, the condition $x \in 2d$-zone(y) does not necessarily imply $y \in 2d$-zone(x). (Note that $x \in N_y$ implies $y \in N_x$.) Hence, after all nodes have broadcast an *Inform_I* packet (and the packets have reached their destinations), every node u will only have information about a subset of N_u, namely $N'_u = \{v : u \in 2d\text{-zone}(v)\}$. In order for u to collect information about nodes in $N_u - N'_u$ ($= \{v : u \notin 2d\text{-zone}(v) \text{ and } v \in 2d\text{-zone}(u)\}$), we let every node v in $N_u - N'_u$ reply with an *Inform_B* packet upon receiving u's *Inform_I* packet — this constitutes the second part of Step 3. Step 4 is self-explanatory.

Procedure INITIAL

1. Node u calculates the value of d according to the desired value of L (using Theorem 4.4) and initializes sets $A(u) = \phi$ and $N_u = \phi$.
2. Node u invokes Procedure IDENTIFY(u) to identify its $2d$-zone. If IDENTIFY reports that some node's $2d$-zone is not k-barrier covered, u informs the base station of this fact. Otherwise, u broadcasts over $2d$-zone(u) an *Inform_I* packet containing u's ID, position and lifetime.
3. When node v receives another node u's *Inform_I* packet, it records u's ID, position and lifetime, and sets $N_v = N_v \cup \{u\}$. Furthermore, if $u \notin 2d$-zone(v), then v replies with an *Inform_B* packet, containing v's ID, position and lifetime.
4. When node u receives another node v's *Inform_B* packet, it records the contained information and sets $N_u = N_u \cup \{v\}$.

Figure 12: Procedure INITIAL

After executing Procedure INITIAL, each node periodically invokes Procedure ACTIVE as described in Figure 13 to decide whether to go to sleep. Roughly, a node u can go to sleep if for all active nodes v such that $u \in 2d$-zone(v), $2d$-zone(v) will be k-barrier covered without u. However, two nodes each eligible for going to sleep may sometimes cause damage in barrier coverage if they *both* go to sleep. So, each node u maintains a set $A(u)$ to address this subtle problem. Initially, $\forall u, A(u) = \phi$ (Step 1 of INITIAL). As time elapses, $A(u)$ will indicate the set of nodes to which u has recently granted a permission for them to go to sleep but they have not made their decisions. In Step 1, node u checks if it and all nodes in $A(u)$ go to sleep, will there be any harm to barrier coverage? If the answer is no, u then consults the nodes in N_u about its going to sleep (by broadcasting a *Query_A* packet). In Step 2, after receiving u's inquiry, node v will grant a permission (*Not_Required_A*) for u to go to sleep if, and only if, u and all nodes in $A(u)$'s going to sleep will not jeopardize $2d$-zone(v)'s k-barrier coverage. In this case, u is added to $A(v)$. Step 3 indicates that a node u can go to sleep only if it has received a positive response (permission) from all active nodes in N_u. Whatever u decides — to go to sleep or stay active — u informs all active nodes in N_u of its decision, so that they know of its status. If u decides to go to sleep, it then changes to sleeping state until T time later or until the first active node in N_u is expected to die, whichever occurs earlier. Waking up after T time is to protect against unanticipated sensor failures, or if the estimation of remaining lifetime is inaccurate. In Step 4, v removes u from $A(v)$ since u has already made its decision. If u thinks of going to sleep in the future, it will need to get a new permission (*Not_Required_A*) from v. Note that if node u stays active, then every T time units it will invoke procedure ACTIVE again. For efficiency, u may first check whether there have been new active nodes added in $2d$-zone(u) in the past time T (or since its last broadcast of *Query_A*). If so, then u invokes ACTIVE; otherwise, u waits for another period of T. The value of T is pre-specified.

When a sleeping node wakes up, it executes the procedure WAKEUP shown in Figure 14, to decide whether to become active or go back to sleep. In Step 1, u removes its record of any other node since u maybe have missed other nodes' status-update messages (if any). Then, u queries other nodes in N_u to see if they need it for their $2d$-zones (by broadcasting a *Query_W* packet). In Step 2, node $v \in N_u$ replies to u indicating whether or not it requires u to become active, depending on whether or not the condition "$2d$-zone(v) is not k-barrier covered and u is located in $2d$-zone(v)" is satisfied. During u's sleeping, some nodes in N_u may have changed their status. So, whether or not v requires u to be active, v replies with a (*Required_W* or *Not_Required_W*) packet containing its ID, position and lifetime. This will enable u to keep an updated record

Procedure ACTIVE /* to be invoked by active nodes u */

1. An active node u checks if $2d$-zone(u) will be k-barrier covered without u itself and the nodes in $A(u)$; if so, u sends a *Query_A* packet to the nodes in N_u.
2. Whenever an active node v receives a node u's *Query_A* packet, if $2d$-zone(v) will be k-barrier covered without $A(v) \cup \{u\}$, then v adds u to $A(v)$ and replies with a *Not_Required_A* message. Otherwise, v replies with a *Required_A* message.
3. After issuing a *Query_A*, if u receives a *Not_Required_A* packet from every active node that is in N_u, then it decides to go to sleep. In that case, u sends a *Decision_Sleep* packet to the nodes in N_u and goes to sleep until T time later or until the first active node in N_u is expected to die, whichever occurs earlier. Otherwise, u stays active and sends to the nodes in N_u a *Decision_Continue* packet containing u's ID, position and lifetime.
4. Whenever an active node v receives a node u's *Decision_Sleep* packet, v removes u from its set of active nodes and removes u from $A(v)$; but if v receives u's *Decision_Continue* packet, v only removes u from $A(v)$ and renews u's information.

Figure 13: Procedure ACTIVE

of active nodes in N_u. Step 3 indicates that u can go back to sleep only if it receives *Not_Required_W* packets but no *Required_W* packet. Step 4 ensures that a sleeping node periodically wakes up (and invokes Procedure WAKEUP). Note that u does not need to inform other nodes of its going back to sleep, for in other nodes' records u has already been non-active. In Step 5, u informs all active nodes in N_u if it decides to become active.

Remark: In some steps in Procedures ACTIVE and WAKEUP, a node u needs to send a packet to the (active) nodes in N_u. To implement this efficiently, we may let all nodes u calculate in the Initial phase a value $\gamma_u = \max\{d(u,v) : v \in N_u\}$, and then in order to broadcast a packet over N_u, u just broadcasts the packet (possibly in multihop) in the range of γ_u. There may be nodes $v \notin N_u$ lying in range γ_u of u. When such nodes v receive the packet, they just discard the packet. Note that $v \in N_u$ iff $u \in N_v$, so v is capable of testing whether $u \notin N_v$.

6.3 Analysis

Procedure WAKEUP /* to be invoked by waking-up nodes u */

1. A waking-up node u resets its record of active nodes to null, and sends a *Query_W* packet to the nodes in N_u.
2. When an active node v receives a *Query_W* packet from a node u, if u is in $2d$-zone(v) and the latter is currently not k-barrier covered, then v replies with a *Required_W* packet containing its ID, position and lifetime. Otherwise, v replies with a *Not_Required_W* packet containing its ID, position and lifetime.
3. If u receives any *Required_W* packet, or if u does not receive any reply — *Required_W* or *Not_Required_W* — to its *Query_W* packet (meaning there is no active node in N_u), then u becomes active and sets $A(u) = \phi$. Otherwise, u goes back to sleep. Upon receiving a packet of either type, u records the ID, position and lifetime contained in the packet.
4. If u decides to go back to sleep, u sleeps until T time units later or until the first active node in N_u is expected to die, whichever occurs earlier.
5. If u decides to become active, u sends to the nodes in N_u a *Decision_Active* packet containing u's ID, position and lifetime.
6. When an active node v receives u's *Decision_Active* packet, v adds u to its list of active nodes and records u's ID, position and lifetime.

Figure 14: Procedure WAKEUP

In LBCP, a node u communicates only with nodes in N_u ($= \{v : v \in 2d\text{-zone}(u) \text{ or } u \in 2d\text{-zone}(v)\}$). When the length of the belt increases, while keeping the density constant, the computing and communication cost of a node remains invariant for a given value of d. In this sense, LBCP is a *localized* algorithm.

The LBCP protocol's goal is to let as many nodes as possible go to sleep while maintaining the property that every active node's actual $2d$-zone is k-barrier covered and, therefore, by Theorem 4.4, the entire belt region is L-local k-barrier covered. The protocol attempts to maximize the network lifetime. The performance of the LBCP protocol varies as d or T is varied. We will investigate their effects in Section 7.2. In this subsection, we show that the LBCP protocol does maintain the belt's L-local k-barrier coverage property during the network's lifetime. We assume that there is no packet loss in communication (as ensured by the underlying network or transport layer) and no unanticipated sensor failures.

In LBCP, every active node u keeps record of other active nodes in N_u. We show that u's record is accurate.

LEMMA 6.1. *Active nodes u never mistakenly record a nonactive node as an active one.*

Proof: During the lifetime of the network, consider three cases: (1) u is in the initial phase (invoking Procedure INITIAL), (2) u just wakes up and becomes active, and (3) u has been active for a while. In the initial phase, all nodes are active, so, it is impossible that u records a nonactive node as active. (We implicitly assume that all nodes' lifetimes are longer than the length of the initial phase, which is in the order of seconds.) We now consider cases 2 and 3.

Case 2: u just wakes up and becomes active. Upon waking up, u invokes Procedure WAKEUP, in which it abolishes its old record and sends a *Query_W* to collect updated information. Clearly, only active node can receive this packet and reply with a *Required_W* or *Not_Required_W* packet. So, after executing Procedure WAKEUP and becomes active, u does not record any nonactive node's information.

Case 3: u has been active. By Cases 1 and 2, u's record was correct at the beginning of the current active period. Since then, if an active node v_1 in N_u has gone to sleep, v_1 must have executed ACTIVE, in which u must have received v_1's *Decision_Sleep* (since $u \in N_{v_1}$) and thereby removed v_1 from u's set of active nodes. If an active node $v_2 \in N_u$ died during u's current active period, u would know of v_2's death (since u knew v_2's lifetime) and remove v_2 from its active nodes list. If a node $v_3 \in N_u$ woke up and became active during u's current active period, u would receive a *Decision_Active* from v_3 and correctly recorded v_3's information. □

In order to apply Theorem 4.4 to guarantee L-local k-barrier coverage, it is essential that there is at least one active node. The following lemma ensures this.

LEMMA 6.2. *If currently there is an active node in the belt, then there will be an active node anytime before a first node dies.*

Proof: Suppose the lemma is not true, i.e. at some point of time t before any node dies, there is no active node in the belt. Then, consider the very last active node u that went to sleep. Before going to sleep, u must have invoked the ACTIVE procedure. By Lemma 6.1, u knew that no other node in $2d$-zone(u) was active. Thus, in Step 1 of ACTIVE, u would find that if it went to sleep $2d$-zone(u) would not be k-barrier covered; and Therefore, u would stay active. This is a contradiction. □

We are now ready to prove that Protocol LBCP guarantees the belt's L-local k-barrier coverage. The following two lemmas are crucial for this purpose. Lemma 6.3 indicates that the belt is L-local k-barrier covered before any node dies; and Lemma 6.4 indicates that if a node's death ever jeopardizes the belt's L-local k-barrier coverage, it is only transient and the coverage will be resumed quickly. For simplicity, throughout the rest of this section, unlike in the above, "$2d$-zone" without an adjective will refer to an actual $2d$-zone.

Lemma 6.3. *If presently there is at least one active node and every active node's $2d$-zone is k-barrier covered, then at any moment before a first node dies, every active node's $2d$ zone will be k-barrier covered (despite that nodes may change their states from active to sleeping and vice versa).*

Proof: By Lemma 6.2, there will be always an active node between now and the moment of the first node's death. For contradiction, assume the lemma is false. Then there must be an earliest moment t at which some active node u's $2d$-zone becomes non-k-barrier covered. There are two reasons for this to happen, which we will show actually are impossible.

Case 1: u has been active and $2d$-zone(u) has been k-barrier covered, but $2d$-zone(u) suddenly becomes non-k-barrier covered at time t. This may occur only if some active node v in $2d$-zone(u) goes to sleep and thereby leaves $2d$-zone(u) non-k-barrier covered. By Theorem 5.1, this node v is also in computed $2d$-zone(u). Before v goes to sleep, it must have invoked ACTIVE(v), sent a *Query_A* to u, and received back a *Not_Required_A* from u. However, u could send back a *Not_Required_A* only if the computed $2d$-zone(u) (and hence the actual $2d$-zone(u)) would be k-barrier covered without v. Thus, v's going to sleep cannot cause $2d$-zone(u) to become non-k-barrier covered.

Case 2: u was sleeping and $2d$-zone(u) was not k-barrier covered before time t, but u wakes up and becomes active at time t. We prove the impossibility of this scenario by showing that no any sleeping node may become active before any node dies. Assume the contrary and let u' be the first sleeping node to become active. Note that u' was once active, later went to sleep, and now it wakes up and becomes active again. Before u' went to sleep, u' invoked Procedure ACTIVE. By an argument similar to that for Case 1, the computed (and hence actual) $2d$-zones of all active nodes in $N_{u'}$ were still k-barrier covered when u' went to sleep. While u' was sleeping, for any active node v in $N_{u'}$, some nodes in v's computed $2d$-zone might go to sleep, but v's computed $2d$-zone remained k-barrier covered. When u' wakes up and invokes Procedure WAKEUP to check if it needs to become active, u' will only receive *Not_Required_W* packets, and therefore will go back to sleep, which is a contradiction. □

A set of nodes A is *feasible* if $\forall x \in A$, $2d$-zone(x) is k-barrier covered, provided that all nodes in A are active.

Lemma 6.4. *After a node dies, if the remaining sensor network is feasible, then all active nodes' $2d$-zones will be k-barrier covered in finite time.*

Proof: Suppose node v dies (out of power). Consider any active node u whose actual $2d$-zone (and hence computed $2d$-zone) becomes non-k-barrier-covered because of v's death. All sleeping nodes in the computed $2d$-zone(u) that know v's lifetime (and thus its death) will immediately wake up and invoke Procedure WAKEUP. And in at most T time units, the other sleeping nodes in the computed $2d$-zone(u) will have also waked up and invoked Procedure WAKEUP. u will ask waking-up sensors to become active as long as the computed $2d$-zone(u) is not k-barrier covered. Since $2d$-zone(u) is k-barrier covered if all remaining nodes in $2d$-zone(u) are active (by the assumption of feasibility), $2d$-zone(u) will be k-barrier covered in T time units at most.

If a new active node w's $2d$-zone is non-k-barrier covered, evidently it will become k-barrier covered in at most T time units since all sleeping nodes in $2d$-zone(w) will have waked up and invoked Procedure WAKEUP in T time units. Since the number of sensors in the belt is finite, LBCP protocol will make all active sensors' $2d$-zones become k-barrier covered in finite time. □

In Section 7.2, we will investigate the time it may take for all active nodes' $2d$-zones to recover from a node's death. As will be seen, after a node dies, all active nodes' $2d$-zones either continue to be k-barrier covered or resume so immediately. This is because sufficient nodes have been scheduled to wake up at the end of a node's lifetime (by Step 3 of Procedure ACTIVE and Step 4 of WAKEUP.)

The following theorem that establishes the LBCP protocol's correctness, follows directly from Lemmas 6.3 and 6.4 and Theorems 4.1 and 4.4.

Theorem 6.1. *The belt under LBCP is guaranteed to be L-local k-barrier covered as long as the sensor network remains feasible, except for possible recovery times after nodes die.*

Message Complexity: The number of messages transmitted by a node in LBCP protocol is insignificant. For any node u, let the number of nodes in N_u be at most D. Every node executes Procedure INITIAL once. Assume there are no *Require_N* transmitted; otherwise, every node stops working. There are at most $(1 + D)$ packets (1 *Inform_I* packet and D *Inform_B* packets) transmitted when a node executes Procedure INITIAL.

Let T be $f(\leq 1)$ times the lifetime of a node. Then, a node invokes Procedure ACTIVE at most $1/f$ times. There are at most $(2+D)$ packets (1 *Query_A* packet, D *Not_Required_A* or *Required_A* packets, and 1 *Decision_Sleep* or *Decision_Continue* packet) transmitted when a node invokes the ACTIVE procedure. So, a node sends at most $(2+D)/f$ packets in executing the ACTIVE procedure in its entire lifetime.

We assume there are at most m disjoint sets of nodes in a node's $2d$-zone such that the nodes in each set provide local barrier coverage for this $2d$-zone. Then, a node sleeps at most mF $(= mT/f)$ time units in its entire lifetime, where F is the life time of a node. There are two possible reasons making a sleeping node wake up. One reason is that it has slept for T time units; another reason is that some active node in its $2d$-node is going to die. a node wakes up at most mF/T $(= m/f)$ times for the first reason and at most D times for the second reason. There are at most $(2+D)$ packets (1 *Query_W* packet, D *Required_W*or *Not_Required_W* packets, and 1 *Decision_Active* packet) transmitted when a node invokes the Procedure WAKEUP . So, a node sends at most $(2+D)(m/f+D)$ packets in executing the Procedure WAKEUP in its entire lifetime.

Therefore, the total number of messages sent by a node in its entire lifetime is at most $(1+D)+(2+D)((1+m)/f+D)$. If $D=100$, $f=0.1$ and m=10, each node will transmit a maximum of $21,521$ packets. Given that transmitting a 60-byte packet consumes 0.01μAh on a Telos mote [14], transmissions of LBCP messages consume about 0.22 mAh, which is insignificant compared to more than $2,000$ mAh of energy reserve in a pair of AA batteries. Note that this analysis gives an upper bound and the real energy consumption may be much smaller than this upper bound.

7. PERFORMANCE EVALUATION

We have implemented LBCP protocol in MATLAB. We have three main results: 1) local barrier coverage almost always implies global barrier coverage when belts are thin, 2) the LBCP protocol provides close to optimal network lifetime while providing global barrier coverage most of the time, and 3) changing the belt from a rectangle to a general belt does not adversely affect the aforementioned performance. We define the network lifetime as the total time when the network is local barrier covered or the total time when the network is global barrier covered.

We use a belt region of dimension $2,000m \times 100m$, unless stated otherwise. Sensors are deployed randomly with uniform distribution. The default sensing range (r) is $30m$, and $k=1$. For the LBCP protocol, lifetime of each node is 10 weeks, $d=100m$, and $T=0$. For every simulation case, 5 random scenarios have been simulated unless stated otherwise. We assume no packet loss, which can be ensured with a suitable reliable data transfer layer.

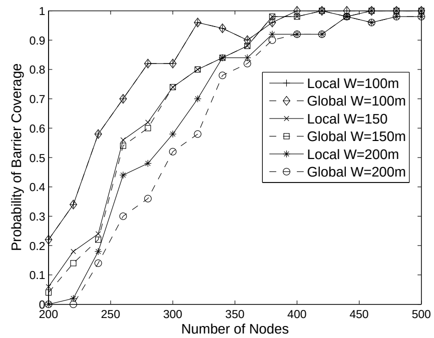

Figure 15: How often is the network local barrier covered vs. global barrier covered when $d = 31m$ and $W = 100m$, $150m$, or $200m$?

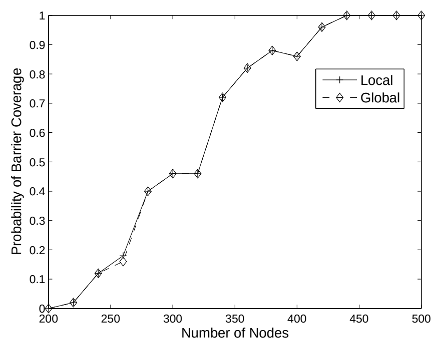

Figure 16: How often is the network local barrier covered vs. global barrier covered when $d = 100m$ and $W = 200m$?

7.1 Local Barrier Coverage vs. Global Barrier Coverage

We vary the density of nodes in random deployments to study the density at which the network begins to provide local barrier coverage and compare it with that for global barrier coverage. For every simulation case, 100 random scenarios have been simulated. To determine if the network provides local barrier coverage, we use Theorem 4.1, which ensures that if $d > r$, then barrier coverage of $2d$-zones of all nodes is sufficient to ensure L-local barrier coverage with $L = \max\{2(d-r), d+r\}$. Hence, we only need to check that the $2d$-zones of all nodes are barrier covered, rather than checking each of the L-zones, of which there are infinitely many.

The results of simulation appear in Figure 15. As can be seen from this figure, when the width (W) is $100m$, the network always provides global barrier coverage whenever it provides local barrier coverage, even if we use a value of d that is close to r. As the width of the region is increased, local barrier coverage does not always ensure global barrier coverage for small d. But, if a larger value of d (e.g., $100m$) is used (which implies a larger value of L in L-local barrier coverage), then local barrier coverage implies global barrier coverage even when the width is large as shown in Figure 16. In summary, for thin belts, local barrier coverage is sufficient for ensuring global barrier coverage, in practice.

7.2 Lifetime Maximization With LBCP

We investigate four main issues here.

1) What level of lifetime improvement is achieved using LBCP and how often does it provide global barrier coverage?

To determine the improvement in lifetime, we compare the performance of LBCP with the optimal (centralized) algorithm of [8] and with Randomized Independent Sleeping (RIS) of [9], which is a localized algorithm. We vary the number of nodes from 500 to $2,000$. The simulation results are shown in Figure 17. We make three key observations from this figure. First, although LBCP only strives to provide local barrier coverage, it always provides global barrier coverage as well in our simulations. Second, it outperforms the RIS algorithm by up to 6 times (e.g., providing a lifetime of 246.7 weeks as opposed to 40.3 weeks for RIS when the number of nodes is $2,000$). Third, it provides very close to the optimal network lifetime.

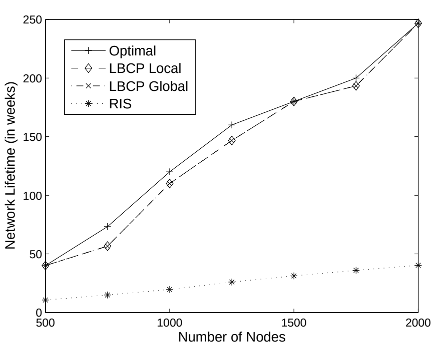

Figure 17: "LBCP Local" denotes that the network is local barrier covered with LBCP and "LBCP Global" denotes that the network is global barrier covered with LBCP. Optimal algorithm and RIS algorithm are both for global barrier coverage.

2) How does the performance of LBCP vary as d is varied?

In Figure 17, we use $d = 100$. For smaller values of d, LBCP does not always provide global barrier coverage; it only ensures local barrier coverage as can be seen in Figure 18. Although local barrier coverage may be sufficient in practice since most movements are expected to follow shortest or close to shortest paths, increasing the value of d ensures global barrier coverage, as well.

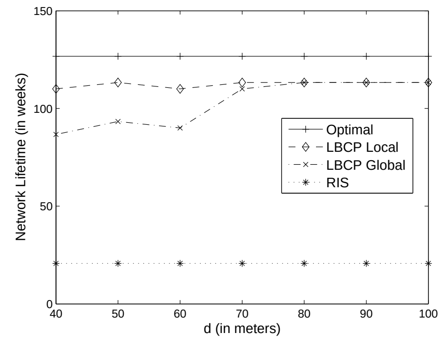

Figure 18: Network lifetime achieved with LBCP as the value of d is varied when 1000 nodes is randomly deployed in the network.

3) How does the performance of LBCP vary as T (the time period for checking the existence of local barrier coverage) is varied?

As can be seen in Figure 19, the performance of LBCP reduces with an increase in T. If T is equal to a node's lifetime, an active node continues to be active until dead, which may reduce the network lifetime. On the other hand, if $T = 0$, an active node checks immediately after a new node becomes active in its $2d$-zone if it can go back to sleep. Using a value close to 0 for T maximizes the network lifetime but involves significant overhead since an active node has to spend significant energy in periodic checking. Notice, however, that when $T = 0$, a sleeping node a wakes up only when the first active node in $2d$-zone(a) is expected to die. We suggest using $[0, 0.1]$ of a node's lifetime for T since even when $T = 0.1$ of a node's lifetime, an active node checks only 9 times in its entire lifetime, while the network life time can still reach 69% of the optimal solution.

4) When will all active nodes' $2d$-zones become k-barrier covered after a node's death makes some active nodes' $2d$-zones temporarily not k-barrier covered?

To make our evaluation more practical, the initial lifetimes of the sensor nodes are uniformly randomly distributed in the set $\{0.1, 0.2, 0.3, \ldots, 10.0\}$ (unit: week). And $T = 1.03$ weeks. Therefore, it is unlikely that a node u dies exactly when a sleeping node has slept for T time units and wakes up. In the simulation, we omit the time for executing the procedures in LBCP. The simulation results show that in all 1899 cases, all active nodes' $2d$-zones become k-barrier covered immediately after a node u's death makes some active nodes' $2d$-zones temporarily not k-barrier covered. The reason is that after u dies, some sleeping nodes who know u's lifetime (and thus its death) will immediately wake up, invoke Procedure WAKEUP, and become active, which results in that not only these original active nodes' $2d$-zones but all the new active nodes' $2d$-zones become k-barrier covered.

7.3 The Performance for General Belts

In this section, we investigate the performance of LBCP combined with the heuristic method developed in Section 5 for determining the barrier coverage in a $2d$-zone for general belts. We consider a semicircular belt whose middle line is $\pi * 1,050m$ long. All other parameters (r, d, W, node lifetime) are the same as described in the beginning of Section 7. Since [9] did not indicate how to set the value of p in the RIS algorithm for a non-rectangular belt, we only compare the performance of LBCP with the optimal algorithm. We vary the number of nodes from 500 to $1,000$. The simulation results are shown in Figure 20. We make two key observations: 1) LBCP provides close to the optimal network lifetime, and 2) LBCP always provides global barrier coverage although it only strives to provide local barrier coverage, indicating that our heuristic (of Section 5) works well in practice.

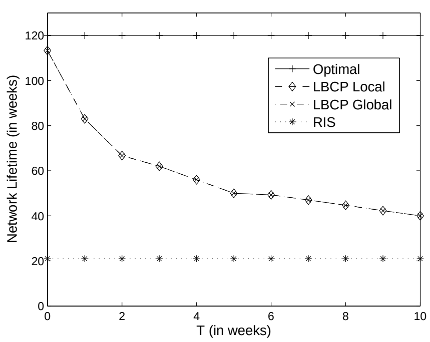

Figure 19: Network lifetime achieved with LBCP as the value of T is varied when 1000 nodes is randomly deployed in the network.

8. CONCLUSION AND FUTURE WORK

We proposed a new notion of coverage called *local barrier coverage* that is more appropriate for applications than the existing global barrier coverage. We then provided a localized algorithm for sensors to determine whether the sensor network provides local barrier coverage. In simulations, we observed that for thin belt regions, the network provided global barrier coverage whenever it provided local barrier coverage. We leveraged the concept of local barrier coverage to develop the first localized sleep-wakeup protocol for movement detection applications that provided close to optimal enhancement in the network lifetime. We proved that our

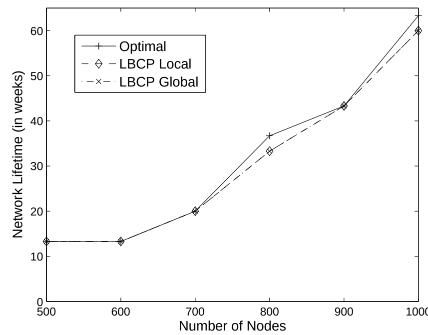

Figure 20: Network lifetime achieved with LBCP as the number of nodes is varied in a general belt.

protocol guarantees local barrier coverage. We showed that in addition to ensuring global coverage most of the time, local barrier coverage also ensured connectivity under some mild assumptions. By enabling the development of localized algorithms for barrier coverage, our work may have opened up many interesting research problems. For instance, localized algorithms for other tasks such as barrier-coverage network repair may now be explored.

The concept of L-local barrier coverage can also be used to measure the quality of barrier coverage provided by a sensor network. We have recently reported some interesting results [5].

As for future work, connectivity will be an interesting problem. If all active nodes' $2d$-zones are k-barrier covered, the network is connected under the assumption that a node is able to communicate (directly or indirectly) with all nodes in its $2d$-zone. This claim can be easily proved. Although the assumption that a node is able to communicate with all nodes in its $2d$-zone is reasonable, we will study, in our future work, under what conditions local barrier coverage implies connectivity without this assumption.

9. REFERENCES

[1] *Mica2 Datasheet.* http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf.

[2] P. Balister, B. Bollobás, A. Sarkar, and S. Kumar. Reliable density estimates for coverage and connectivity in thin strips of finite length. In *Thirteenth Annual International Conference on Mobile Computing and Networking (ACM MobiCom)*, Montreal, Canada, 2007.

[3] M. Cardei, M. Thai, and W. Wu. Energy-Efficient Target Coverage in Wireless Sensor Networks. In *IEEE INFOCOM*, 2005.

[4] B. Cărbunar, A. Grama, J. Vitek, and O. Cărbunar. Redundancy and Coverage

Detection in Sensor Networks. *ACM Transactions on Sensor Networks (TOSN)*, 2(1):94–128, 2006.

[5] A. Chen, T. H. Lai, and D. Xuan. Measuring and Guaranteeing Quality of Barrier-Coverage in Wireless Sensor Networks. In *Proc. of ACM MobiHoc'08*, 2008.

[6] T. He and et al. Energy-Efficient Surveillance System Using Wireless Sensor Networks. In *International Conference on Mobile Systems, Applications, and Services (ACM Mobisys)*, pages 270–283, Boston, MA, 2004.

[7] C. Huang and Y. Tseng. The Coverage Problem in a Wireless Sensor Network. In *ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, pages 115–121, San Diego, CA, 2003.

[8] S. Kumar. Foundations of Coverage in Wireless Sensor Networks. Technical report, Ph.D. Thesis, Ohio State University (OSU), Available at http://www.ohiolink.edu/etd/send-pdf.cgi/Kumar%20Santosh.pdf?acc_num=osu1154986262.

[9] S. Kumar, T. H. Lai, and A. Arora. Barrier Coverage with Wireless Sensors. In *ACM MobiCom'05*, 2005.

[10] S. Kumar, T. H. Lai, and J. Balogh. On k-Coverage in a Mostly Sleeping Sensor Network. In *Tenth Annual International Conference on Mobile Computing and Networking (ACM MobiCom)*, pages 144–158, Philadelphia, PA, 2004.

[11] S. Kumar, T. H. Lai, M. E. Posner, and P. Sinha. Optimal Sleep Wakeup Algorithms for Barriers of Wireless Sensors. In *Proc. of BROADNETS'07*, 2007.

[12] B. Kuris and T. Dishongh. SHIMMER Mote:Hardware Guide. *Online at http://www.eecs.harvard.edu/~konrad/projects/shimmer/references/SHIMMER_HWGuide_REV1P3.pdf*, 2006.

[13] B. Liu, O. Dousse, J. Wang, and A. Saipulla. Strong Barrier Coverage of Wireless Sensor Networks. In *Proc. of MobiHoc'08*, 2008.

[14] J. Polastre, R. Szewczyk, and D. Culler. Telos: Enabling Ultra-Low Power Wireless Research. In *Proc. of IPSN'05*, 2005.

[15] S. Slijepcevic and M. Potkonjak. Power Efficient Organization of Wireless Sensor Networks. In *ICC*, Helsinki, Finland.

[16] W. Wang, V. Srinivasan, and K. C. Chua. Trade-offs Between Mobility and Density for Coverage in Wireless Sensor Networks. In *Proc. of MobiCom'07*, 2007.

[17] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration in wireless sensor networks. In *ACM Conference on Ebmedded Networked Sensor Systems (SenSys)*, pages 28–39, Los Angeles, CA, 2003.

[18] T.-T. Wu and K.-F. Ssu. Determining Active Sensor Nodes for Complete Coverage Without Location Information. *International Journal of Ad Hoc and Ubiquitous Computing*, 1(1):38–46, 2005.

[19] T. Yan, T. He, and J. A. Stankovic. Differentiated Surveillance for Sensor Networks. In *Proc. of ACM SenSys'03*, 2003.

[20] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang. Peas: A robust energy conserving protocol for long-lived sensor networks. In *International Conference on Distributed Computing Systems (ICDCS)*, 2003.

[21] H. Zhang and J. Hou. Maintaining Sensing Coverage and Connectivity in Large Sensor Networks. In *NSF International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wirelsss, and Peer-to-Peer Networks*, 2004.

APPENDIX

A. PROOF OF THEOREM 4.4

We develop a series of lemmas towards proving Theorem 4.4. Theorem 4.2 shows that d can not be too small for a rectangular belt. Indeed, we required $d > r$ in Theorem 4.1. This condition, $d > r$, was to ensure that no sensor can simultaneously cover both the orthogonal boundaries of any $2d$-zone in a rectangular belt. In order to generalize Theorem 4.1 from rectangular to general belts, we need a condition to ensure that no sensors can simultaneously cover both the orthogonal boundaries of any $2d$-zone in a general belt. The next lemma serves this purpose. In this lemma, and throughout the rest of this section, we write $\arcsin(r/2R)$ to mean $\arcsin(r/(2R))$ — the parentheses are omitted for simplicity and clarity.

LEMMA A.1. *In a general belt, if $d > 2(R + \frac{W}{2})\arcsin(\frac{r}{2R})$, then no node u can cover any orthogonal boundary of its $2d$-zone(u). Furthermore, any node on one side of the orthogonal line passing through u can not cover $2d$-zone(u)'s farther orthogonal boundary.*

Proof: First, consider a circular belt of width W, its inner circle being of radius R. That is, the curvature at any point of the belt's inner boundary is of magnitude $1/R$. Consider an L_2-zone, with a line of length r, an arc of length L_1, $\angle A$, $\angle B$ as shown in Figure 21. We know that $\angle B = 2\angle A = 2\arcsin(r/2R)$. Therefore, $L_1 = 2R\arcsin(r/2R)$, and $L_2 = 2(R + W/2)\arcsin(r/2R)$. The distance between the two orthogonal boundaries of L_2-zone is r. If a node u is at the intersection of the L_2-zone's inner circular boundary with one of its orthogonal boundaries, then u's sensing range can barely cover the other orthogonal boundary of the L_2-zone. If $L > L_2$, then the sensing range of any node on one orthogonal boundary of an L-zone can not cover the other orthogonal boundary since the distance between the two orthogonal boundaries is larger than r, and any node outside of an L-zone can not cover the farther orthogonal boundary of the L-zone. Therefore, if $d > L_2$, since $2d$-zone(u) can be divided into two d-zones at the orthogonal line passing through u, no node u can cover any orthogonal boundary of its $2d$-zone(u), and any node on one side of the orthogonal line passing through u can not cover $2d$-zone(u)'s farther orthogonal boundary.

Next, we show that the lemma holds for a circular belt of width W, whose inner circle is of radius $R' > R$. It is clear that $2(R'+\frac{W}{2})\arcsin(\frac{r}{2R'}) < 2(R+\frac{W}{2})\arcsin(\frac{r}{2R})$. So, in this belt, if $d > 2(R+\frac{W}{2})\arcsin(\frac{r}{2R})$, we also get $d > 2(R'+\frac{W}{2})\arcsin(\frac{r}{2R'})$, which implies that no node u can cover any orthogonal boundary of its $2d$-zone(u), and any node on one side of the orthogonal line passing through u can not cover $2d$-zone(u)'s farther orthogonal boundary.

Now, we consider a general belt. By assumption, the curvature at any point on the belt's parallel boundaries is of magnitude at most $1/R$. That is, at any point on the parallel boundaries, the curvature is of magnitude $1/R'$, with $R' \geq R$. Therefore, if $d > 2(R + W/2)\arcsin(r/2R)$, no node u can cover any orthogonal boundary of its $2d$-zone(u), and any node on one side of the orthogonal line passing through u can not cover $2d$-zone(u)'s farther orthogonal boundary in a general belt. □

LEMMA A.2. *In a general belt, if $d > 2(R+\frac{W}{2})\arcsin(\frac{r}{2R})$, and the $2d$-zone(u) of every active node u is k-barrier covered, then each M-zone with $M \geq 2(R+W/2)\arcsin(2r/2R)$ must contain at least one active node.*

Proof: For a circular belt, if the inner circle's radius is R and $M_2 = 2(R+W/2)\arcsin(2r/2R)$, then the length of the chord of the M_2-zone on the inner circle is $2r$, as shown in Figure 21. In this case, the sensing disks of any two nodes outside of the M_2-zone and on the opposite sides of the M_2-zone have no overlap in the M_2-zone. It is clear that if $R' > R$, $2(R' + W/2)\arcsin(2r/2R') < 2(R+W/2)\arcsin(2r/2R)$. As in the proof of Lemma A.1, in a general belt whose biggest curvature value on the two parallel boundaries is $1/R$, if $L \geq 2(R+W/2)\arcsin(r/R)$, then the sensing disks of any two nodes outside of the M_2-zone and on the opposite sides of the M_2-zone have no overlap in the M_2-zone. Clearly, the conclusion is true for any $M \geq M_2$ if it is true for M_2.

Now, assume there are no nodes in an M-zone with $M \geq M_2$. Consider the node a whose orthogonal crossing line is closest to the M-zone. Then, there is no node in the region between a's orthogonal crossing line and the M-zone (as well as the M-zone itself). Since there is no overlap between the sensing disks of any two nodes on the opposite sides of the M-zone, the nodes on the same side of the M-zone as a (including a) should provide k-barrier coverage for $2d$-zone(a) (including its farther orthogonal boundary), which is impossible according to Lemma A.1. Therefore, there must be at least one node in each M-zone. □

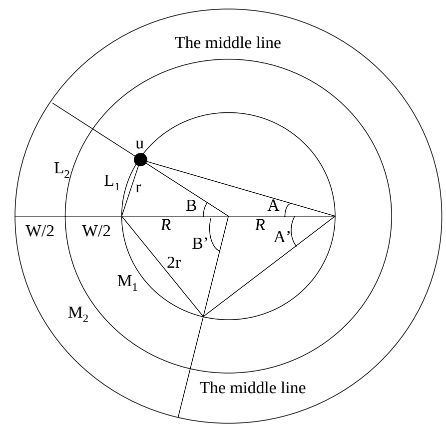

Figure 21: Visualizing the proofs of Lemmas A.1 and A.2

LEMMA A.3. *In a general belt, let A be an L_1-zone and B be an L_2-zone. Assume that the intersection of A and B is non-empty, and $L_2 \leq \frac{R+W/2}{R+W}r$. Suppose A is k-barrier covered, and no node in $A - B$ ever covers A's orthogonal boundary that is contained in B. Then, $A \cup B$ is k-barrier covered.*

Proof: Let a_1 be the A's orthogonal boundary that is in B, and a_2 be the A's orthogonal boundaries that is not in B. Let b_1 be the B's orthogonal boundary that is in A, and b_2 the B's orthogonal boundaries that is not in A. So, a_2 and b_2 are also the orthogonal boundaries of $A \cup B$.

First, we prove that the sensing disk of any node in B covers both of the orthogonal boundaries of B, b_1 and b_2. Consider see a circular blet with the inner circle's radius being R. If $L_2 = \frac{R+W/2}{R+W}r$, then the length of the longer parallel boundary (on the outer circle) of B is r and the length of the corresponding chord $\leq r$. Let p be an arbitrary node in B, and consider the passing orthogonal line b_p of p and b_1, we get a sub-zone of B. Clearly, the lengths of the parallel boundaries of this sub-zone $\leq r$, and the distance between p and $b_1 \leq r$. Therefore, the sensing disk of p covers b_1. Similarly, we can prove that the sensing disk of p covers b_2. If $R' > R$, $\frac{R'+W/2}{R'+W}r = (1 - \frac{W}{2(R'+W)})r > \left(1 - \frac{W}{2(R+W)}\right)r =$

$\frac{R+W/2}{R+W}r$. Following the idea used in the proof of Lemma A.1, we can prove that for a general belt whose biggest curvature value is $1/R$ on the parallel boundaries, if $L_2 \leq \frac{R+W/2}{R+W}r$, the lengths of the parallel boundaries of $B \leq r$, and the sensing disk of any node in B covers both of the orthogonal boundaries of B.

Because A is k-barrier covered, there must be at least k nodes covering a_1. Since these nodes are not in $A-B$, they must be in B, or outside of B but on the other side of b_2 as compared to a_1. If they are outside of B, their sensing disks must cover b_2 if they cover a_1. If they are in B, their sensing disks also must cover b_2. Therefore, the nodes making A k-barrier covered also make $A \cup B$ k-barrier covered. □

Proof of Theorem 4.4: The proof is similar to that of Theorem 4.1, but now we will use Lemmas A.2 and A.3 in place of Lemmas 4.1 and 4.2.

Assume that $2d$-zone(u) is k-barrier covered for every active node u. Let $L_1 = 2d - 2(R + W/2)\arcsin\left(\frac{2r}{2R}\right)$ and $L_2 = d + \frac{R+W/2}{R+W}r$.

Case 1: $L_1 \geq L_2$. Let $M = 2(R + W/2)\arcsin\left(\frac{2r}{2R}\right)$. By Lemma A.2, given any L_1-*zone*, there is at least one node b in its center M-zone. Then, L_1-zone $\subseteq 2d$-zone(b) and hence L_1-zone is k-barrier covered.

Case 2: $L_1 < L_2$. Given any L_2-zone, it can be divided into three parts: the center L_c-zone, and the two L_s-zones on the opposite sides of the center zone, where $L_c = \left(d - \frac{R+W/2}{R+W}r\right)$ and $L_s = \frac{R+W/2}{R+W}r$. Then, it can be proved in a similar manner as in the proof of Theorem 4.1 (case 2) using Lemma A.3 in place of Lemma 4.2 that L_2-zone is k-barrier covered.

Therefore, if the $2d$-zone of every active node is k-barrier covered, then every L-zone is k-barrier covered, where $L = \max\{L_1, L_2\}$. □