Target Counting under Minimal Sensing: Complexity and Approximations*

Sorabh Gandhi, Rajesh Kumar, and Subhash Suri

Department of Computer Science,
University of California,
Santa Barbara, CA-93106

Abstract. We consider the problem of counting a set of discrete point targets using a network of sensors under a minimalistic model. Each sensor outputs a single integer, the *number* of distinct targets in its range, but targets are otherwise indistinguishable to sensors: no angles, distances, coordinates, or other target-identifying measurements are available. This minimalistic model serves to explore the fundamental performance limits of low-cost sensors for such surveillance tasks as estimating the number of people, vehicles or ships in a field of interest to first degree of approximation, to be followed by more expensive sensing and localization if needed. This simple abstract setting allows us to explore the intrinsic complexity of a fundamental problem, and derive rigorous worst-case performance bounds. We show that even in the 1-dimensional setting (for instance, sensors counting vehicles on a road), the problem is non-trivial: target count can be estimated within relative accuracy of factor $\sqrt{2}$ and this is the best possible in the worst-case. We then address additional questions related to constructing *feasible* target placements, and noisy counters. In two dimensions, the problem is considerably more complicated: a constant-factor approximation is impossible. Our algorithms and analysis can easily handle some of the non-idealities of real sensors, such as asymmetric ranges and non-exact target counts.

1 Introduction

Inexpensive smart sensors coupled with ad hoc wireless networking provide a compelling and cost-effective technology for what is variously called ubiquitous computing or situational awareness. Specifically, there has been a growing interest in the *networked* power of many cheap and low-fidelity but unattended and geographically-distributed sensors. Because of their low cost, both in hardware that can be several orders of magnitude cheaper than their "mainframe" counterparts, and the untethered, self-organizing architecture that makes them attractive for deployment at large geographic scale without costly human management, pervasive sensor networks hold great potential for "environmental monitoring." The hardware costs and availability, however, are only part of the solution. In order to

* This research was supported in part by the National Science Foundation under grants CNS-0626954 and CCR-0514738.

S. Fekete (Ed.): ALGOSENSORS 2008, LNCS 5389, pp. 30–42, 2008.
© Springer-Verlag Berlin Heidelberg 2008

realize the full potential of these networked smart sensors, significant challenges in algorithms, software, and signal processing must be addressed, many of which arise from the "minimalistic" nature of this sensing and computing platform.

In this paper, we examine some of these key issues in the context of *counting and localizing* targets in a physical space under minimal sensing assumptions. We focus on target counting, as opposed to the more-widely studied target tracking problem, for two reasons: (1) counting is an important problem in its own right; in many environmental monitoring and unattended surveillance applications, for which sensor networks are an ideal platform, accurately estimating a population (e.g. animals in natural habitats, intruders in sensitive areas) is a fundamental end goal; and (2) a good estimate on the target count is often a pre-requisite for robust tracking; for instance, many popular tracking heuristics such as those based on particle filters need a good educated guess on the number of unknown targets to avoid getting stuck.

We frame our research within a minimalistic sensing model to align it with the primary motivation behind the appeal of sensor networks: *low cost and small form factor*. As a result, the binary sensing model has received a great deal of attention for target tracking and other monitoring applications, both in theory and practice (for instance, see [1,2,3,4,5,6]). While the binary sensing model has been shown to achieve excellent performance for tracking a single target [5], for multiple targets it is useful only in settings where the targets are pairwise widely-separated, as was formalized in [6]. As a result, provable-quality tracking and counting of targets requires a richer class of sensors.

In this paper, we work with an abstract model of a *counting sensor*: each sensor outputs an integer value, representing the number of distinct targets in its sensing range. Each target is modeled as a point. The sensor produces no other information about the targets, such as their locations, angles, distances, or any other distinguishing identifiers. While a convenient abstraction for our theoretical investigation of the fundamental limits of target counting and localization, such a sensor is also a fairly good first-order approximation of low-cost radar sensors that can detect the presence of multiple targets but cannot localize them individually. Other sensors including infra-red sensors or acoustic sensors also exhibit this characteristic. In low-cost camera systems as well, achieving reliable calibration or coordinating multiple snapshots for depth and location is both difficult and error-prone [7,8,9]. Furthermore, the measurements are often so noisy that systems actually improve performance by using only the simplest and most robust information content; for instance, Oh et al. [10] report that the variability in the signal strength of their PIR (passive infrared) motion sensors was so great that they actually *improved* the performance of their tracking system by using them as *binary* sensors.

Because our main focus is fundamental achievable limits of performance, we begin with an idealized sensing model, and then discuss the impact of these assumptions as well as generalizations to non-idealized settings. We assume that each ideal sensor has a circular sensing range of a known radius, and it

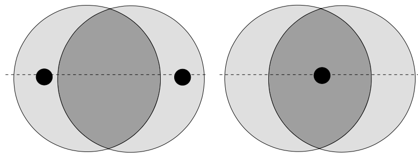

Fig. 1. The two scenarios have identical sensory information: each sensor detects 1 target, yet the total number of targets is different in the two cases

reliably counts the number of distinct targets in its range. Even with such idealization, it is easy to see that our minimal sensing model does not have enough information to accurately count targets *even in 1-dimension.* Figure 1 shows an example of two scenarios with two sensors. The sensory information of both sensors is identical in the two scenarios: both sensors detect 1 target. Thus, there is no way to distinguish between the two scenarios, and decide whether the true target count is 2 (left) or 1 (right). One can, of course, generalize this to an example where sensors cannot distinguish between n and $2n$ targets, and arrive at the impossibility result that, under our minimal sensing model, *no algorithm* can count targets with an *accuracy factor* better than $\sqrt{2}$. It turns out, however, that this is essentially the worst-possible scenario, and one can always achieve $\sqrt{2}$ approximation factor for *any configuration of targets and sensors in one dimension.*

Given our sensing model, one may feel that the best counting accuracy is achieved by non-overlapping sensing ranges—the inaccuracies arise only from multiple sensors counting the same target. Why not just deploy sensors with non-overlapping ranges and obtain the best possible results? There are at least three reasons for sensors with overlapping ranges. First, circles do not tile the two-dimensional plane, and so even in an idealized setting, one cannot achieve full coverage without overlapping circular ranges. Second, while the target count can be improved by *minimizing* the overlap among different sensing ranges, the *location accuracy*, in fact, improves with increasing the overlap [5]. Thus, there is an inherent tension between counting accuracy and the localization accuracy, which may promote sensor deployments with significantly overlapping ranges, even in one-dimensional situations, like a road environment. Finally, all of our results, in fact, hold even when the sensing ranges are not ideal disks; they just need to be connected intervals in one-dimension and any reasonable convex shape in two dimension. Thus, our theory applies to irregular, anisotropic sensing ranges of real sensors, whose overlap is both unpredictable and impossible to eliminate. Therefore, in this work we approach the problem with a worst-case viewpoint, and make no assumptions about the placement of targets or the sensors. We seek to provide worst-case guarantees for the target count for any (adversarial) choice of targets and sensor ranges.

Our approximate counting algorithm, however, is non-constructive, in that it does not necessarily produce a configuration of targets consistent with the sensing input—*it just produces upper and lower bounds on the target population.* Furthermore, it is easy to show examples where not all target counts between the lower and upper bounds are feasible, meaning that there is no possible configuration of targets that is consistent with the sensors' readings. Constructing a feasible configuration of targets is not entirely trivial, but it can be solved in polynomial time by a reduction to the shortest path problem in a graph.

Next, we consider the impact of some non-idealities on our results. In particular, we allow sensor ranges to be non-unit-disk: they can be arbitrary size segments in 1D and arbitrary convex regions in the plane, and they can be asymmetric around the sensor. The target sensing also can be "noisy," in that the number of targets detected by a sensor can lie in an uncertainty range. Specifically, we assume that if the true reading of a sensor is c, then a sensor can report any value in the range $[(1-\rho)c, (1+\rho)c]$, where ρ is the *noise* or *uncertainty* parameter, reflecting the false positives and negatives in the sensor's reading. It turns out that all our algorithms and theorems hold even in these more general and realistic models; of course, the accuracy of the target counting now depends on the parameter ρ.

We then consider the target counting problem in two-dimensions and prove that, in the worst-case, no fixed approximation is achievable. An easy $\sqrt{m}$ approximation is possible if the maximum degree of overlap among sensor ranges is m. (This is in contrast to the 1-dimension, where the approximation factor does not depend on the degree of sensing overlap.) All of these results extend to the "noisy" sensor model. All the theorems in this pre-proceedings version are without proofs, the proofs will be included in the conference proceedings.

2 The Counting Sensor Model

We begin with an idealized model of sensing. Each target is modeled as a point, and each sensor is assumed to have a unit-disk sensing range, with perfect sensing: each sensor is able to count precisely the number of targets present in its range. Neither of these assumptions are critical to our algorithms and analysis, as we later discuss, but provide a convenient framework to understand the fundamental limits of target counting. Because the communication requirements of our collaborative counting are so minimal (each sensor only needs to communicate its reading), we abstract away all networking issues in our discussion. In particular, we assume that all the processing occurs at a base station, or a tracker node, that knows the precise geometry of the sensors' locations and ranges. We make no assumptions about the geographic distribution of sensors or targets: *our results are worst-case.*

Throughout, we assume that the targets have fixed locations, and sensors' readings represent a snapshot of the target locations. This view is valuable even in tracking applications when no a priori information is available about the motion of the targets and where the targets can be deliberately evasive, creating an adversarial situation. In such settings, a tracking algorithm is forced to interpolate the motion across snapshots, and therefore must solve the target counting and localization problem considered here.

We begin our discussion by considering the problem in a one-dimensional setting. We imagine targets as points arranged on a line, and a collection of sensors, each with a unit-interval sensing range. It turns out that the exact counting of targets is non-trivial even in this simple setting, and leads to some interesting results. The 1-dimensional setting is also a useful framework in many practical situations, such as counting targets along a road or counting objects in a crowd using far away cameras.

3 Counting and Localization Targets in One Dimension with Ideal Sensors

We begin by repeating our earlier example to argue that precise counting is not possible even in one dimension, and even with idealized counting sensors.

Theorem 1. *If sensors have overlapping ranges, then precise counting of targets is impossible even with idealized counting sensors. Thus, for arbitrary arrangements of sensors and targets, no algorithm can determine the target count precisely.*

Fortunately, it turns out that this is the worst possible scenario, and the $\sqrt{2}$ approximation of the target count is possible for any (adversarial) placement of targets and sensors in 1-dimension.

3.1 Target Count Approximation

Let $S = \{s_1, s_2, \ldots, s_n\}$ denote the set of sensors, and let $C = \{c_1, c_2, \ldots, c_n\}$ denote their sensing counts; that is, c_i is the number of targets detected by s_i in its range. We denote the set of sensing ranges by R, and the union of all these ranges by U. Recall that each sensing range is an interval on the line containing the sensors and the targets. We assume that U is a contiguous range, if not, we run our algorithm on the disconnected contiguous subsets of U separately and add the counts to get the approximate count.

Our algorithm for approximating the number of targets, which we call the SCAN algorithm, is as follows. We compute a *non-redundant* subset $R' \subset R$ of the sensing ranges, where non-redundancy means that union of the ranges in R' equals U, and no range $r \in R'$ is covered by the union of the remaining ranges in the set. In other words, no range can be deleted from R' without losing some coverage of the domain.

Let us denote the set of sensors associated with R' by $S' = \{s'_1, s'_2, \ldots, s'_{n'}\}$ and their readings by set $C' = \{c'_1, c'_2, \ldots, c'_{n'}\}$. Our algorithm outputs $C_A = \frac{S_{C'}}{\sqrt{2}}$ as the target count, where $S_{C'}$ is the sum of readings of the set C', namely, $S_{C'} = \sum_{1 \leq i \leq n'} c'_i$. The algorithm for finding the set R' is given in Algorithm 1.

It is easy to verify that this algorithm can be implemented in worst-case time $O(n \log n)$. We now prove the main result of this section that C_A is a factor $\sqrt{2}$ approximation of the true count, which we denote as C_{OPT}.

Algorithm 1. SCAN

Sort the segments in R in increasing order of left endpoints.
$R' = \emptyset$, $r =$ first segment in the sorted set R
$R' = R' \cup \{r\}$, $R = R \setminus \{r\}$
while $R \neq \emptyset$ **do**
  $T =$ set of segments in R that intersect with r.
  Let r' be the segment in T with the rightmost endpoint.
  $R = R \setminus T$, $R' = R' \cup \{r'\}, r = r'$.
end while
Output the total count of targets associated with ranges in R divided by $\sqrt{2}$.

Theorem 2

$$\frac{C_A}{\sqrt{2}} \leq C_{OPT} \leq \sqrt{2}C_A$$

In effect, the algorithm SCAN above outputs a range $[a, b]$ with a guarantee that the true target count lies between a and b, and $b \leq 2a$. By predicting the geometric mean of these two bounds as an approximation, the algorithm can guarantee that its prediction is within a factor of $\sqrt{2}$ of the true count.

Unfortunately, the counting scheme presented so far is non-constructive—it tells us bounds on the number of targets, but offers no actual placements of targets satisfying the readings of all the sensors. In the following section, we address this fundamental problem of producing target placements consistent with the sensors' readings.

3.2 Target Placement

Consider the example in Figure 2. For this example, the algorithm SCAN outputs the target range $[2, 4]$, which clearly is consistent with the sensors' readings. However, a moment's reflection shows that there is no realizable (feasible) *target placement* that is consistent with the target count of either 2 or 3. Indeed, the only feasible target placement satisfying the sensors' readings needs 4 targets, as shown. Even for feasible target counts, the algorithm does not provide an actual placement of targets. We address these shortcomings in the following.

Consider a set $S = \{s_1, s_2, \ldots, s_n\}$ of n sensors along the X-axis, and let $C = \{c_1, c_2, \ldots, c_n\}$ denote the readings associated with these sensors. Let $P =$

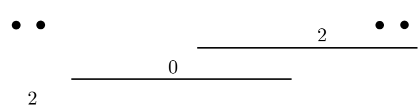

Fig. 2. An example of 3 sensors on a line, where the first and the third sensor has target count 2, while the middle sensor has count 0. The SCAN algorithm outputs a target range of $[2, 4]$. Only the target count of 4 is realizable as a physical configuration of targets consistent with the sensors' readings.

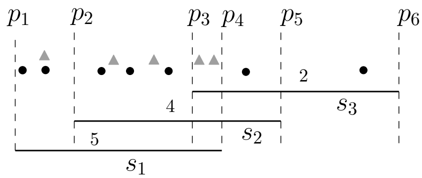

Fig. 3. An example with 7 targets and 3 sensors. The true target positions are shown as solid circles. The sensor readings are shown by the numbers placed above each sensor's range. The output placement computed by our algorithm is shown using lightly shaded triangles.

$\{p_1, p_2, \ldots, p_{2n}\}$ denote the set of $2n$ points defining the start and the end points of the sensor ranges, sorted in order of increasing x-coordinates; that is, the x-coordinate of p_i is less than the x-coordinate of p_{i+1}.

We introduce a set of variables $Z = \{z_1, z_2, \ldots, z_{2n}\}$ where z_i represents the *total* number of targets lying to the left of point p_i. By definition, therefore, we have the following constraint:

$$z_1 \leq z_2 \leq \ldots \leq z_{2n}, \tag{1}$$

because the number of targets to the left of p_{i+1} is at least as large as the number of targets to the left of p_i.

Next, if p_j and p_k are the starting and end points associated with the range of sensor s_i, then $z_k - z_j$ denotes the number of targets in s_i's range. This introduces another constraint:

$$z_k - z_j \;=\; c_i \tag{2}$$

We have one such constraint for each sensor. Any assignment of z_i's satisfying these constraints, together with $z_1 = 0$, corresponds to a feasible placement of targets for our problem. In particular, a feasible solution can be obtained by placing $z_i - z_{i-1}$ targets spaced equally between points p_{i-1} and p_i, for $2 \leq i \leq 2n$. The set of constraints described above can be solved as an integer linear program. Unfortunately, in general, integer linear programming is NP-Hard. Fortunately, the special structure of our problem admits a rather efficient (polynomial time) solution, by a transformation to a shortest path problem. In particular, all the constraints in our problem have the form of a *difference constraint.* We explain the reduction to the shortest path problem, using an example.

Consider the example shown in Figure 3, with 7 targets and 3 sensors. The true target positions are shown as solid circles. The sensor readings are shown by the numbers placed above each sensor's range. The first set of constraints that enforce the conditions $z_i \leq z_{i+1}$, for $1 \leq i \leq 2n$, can be written as the following set of difference constraints:

$$z_1 - z_2 \leq 0, \quad z_2 - z_3 \leq 0, \quad z_3 - z_4 \leq 0$$
$$z_4 - z_5 \leq 0, \quad z_5 - z_6 \leq 0$$

Each of the equality constraint encoding the count of each sensor (Eq. 2) can be written as a pair of difference constraints:

$$z_4 - z_1 \le 5, \quad z_1 - z_4 \le -5$$
$$z_5 - z_2 \le 4, \quad z_2 - z_5 \le -4$$
$$z_6 - z_3 \le 2, \quad z_3 - z_6 \le -2$$

These inequalities can be transformed into the formulation of a shortest path problem in a graph as shown in Figure 4. In this graph, there exists a node for each variable z_i, and an edge for each difference constraint. In particular, the difference constraint $z_i - z_j \le \ell$ maps to an edge directed from node w_j to node w_i, with weight ℓ. In addition, we add an artificial node s, and introduce 0-weight edges from s to all other nodes in the graph. We now observe that this graph has well-defined shortest paths from s to all other nodes *if and only if* there is no negative-weight cycle in the graph. More precisely, if there is a negative cycle in the graph, then the set of inequalities are inconsistent, and there is no feasible solution. Otherwise, the shortest path distances to the nodes z_i correspond to a feasible solution.

Solving the shortest path problem on the graph gives the following shortest path distances from s: $z_1 = -5$, $z_2 = -4$, $z_3 = -2$, $z_4 = z_5 = z_5 = 0$. We can enforce $z_1 = 0$ by adding 5 to all these variables, without violating any constraints. We then get: $z_1 = 0$, $z_2 = 1$, $z_3 = 3$, $z_4 = z_5 = z_6 = 5$. The placement of targets corresponding to these variable settings is shown in Figure 3 by lightly shaded triangles.

We can solve the shortest paths problem in the graph using the Bellman-Ford algorithm; this algorithm either determines that the graph contains a negative-weight cycles, or computes valid shortest path distances in worst-case time $O(|V||E|)$, where $|V|$ and $|E|$, respectively, are the number of vertices and edges in the graph [11] . In our setting, both the number of vertices and edges is $O(n)$, so the algorithm has time complexity $O(n^2)$. We can now state the main result of this section.

Theorem 3. *Given a set of n counting sensors on a line and their target counts, we can find in $O(n^2)$ time a placement of targets consistent with all the sensors' counts, or determine that the sensors' readings are inconsistent.*

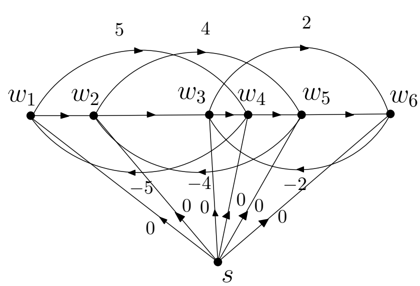

Fig. 4. The graph corresponding to the example of Figure 3

The Bellman-Ford shortest path algorithm, in fact, has an interesting property: the algorithm minimizes the maximum difference between the difference variables. In other words, the algorithm not only finds a feasible assignment of variables, but actually finds an assignment satisfying

$$\min \quad \max_{1\leq i\leq 2n, 1\leq j\leq 2n} |z_i - z_j|$$

In our setting, the maximum difference is between the variables z_{2n} and z_1 (Equation 1). But $z_{2n} - z_1$ equals the total number of targets in the solution, and so our algorithm finds a feasible solution with the least possible number of targets consistent with the sensors' counts.

4 Extensions to Non-ideal Sensing

In this section, we make a limited attempt to address two of the most severe assumptions of the idealized sensor model, namely, the *unit disk* sensing range and *perfect target count*. In particular, we show that our algorithms can easily handle sensing ranges that are neither unit length (in 1d) or unit disk (in 2d) nor symmetric about the center. Secondly, our algorithms can gracefully handle *noisy* target counts by sensors. Specifically, if the true target count for a sensor is c, then a sensor can report any value in the range $[(1-\rho)c, (1+\rho)c]$, where ρ is the noise parameter, reflecting the false positives and negatives in the sensor's reading. We now discuss the implications of these non-idealities on our algorithms.

4.1 Target Count Approximation with Non-ideal Sensors

Let $S = \{s_1, s_2, \ldots, s_n\}$ denote the set of sensors. We denote the set of sensing ranges by R. As in the case of ideal sensors, we use the algorithm SCAN to compute the non-redundant set R'. Let $S' = \{s'_1, s'_2, \ldots, s'_{n'}\}$ denote the set of sensors associated with R' and let $C' = \{c'_1, c'_2, \ldots, c'_{n'}\}$ denote the (noisy) counts associated with these sensors. Our algorithm outputs $C_A = \frac{S_{C'}}{\sqrt{2(1-\rho^2)}}$ as the target count, where $S_{C'}$ is the sum of readings of set C', namely, $S_{C'} = \sum_{1\leq i\leq n'} c'_i$. Let C_{OPT} denote the actual count of the number of targets in the system. The following theorem analyzes the accuracy of this approximation.

Theorem 4

$$\frac{C_A}{\sqrt{\frac{2(1+\rho)}{1-\rho}}} \;\leq\; C_{OPT} \;\leq\; \sqrt{\frac{2(1+\rho)}{1-\rho}}C_A$$

It is not too difficult to see that our bounds for both ideal and non-ideal sensors are the *best possible* in the worst-case. In particular, given any value of ρ, it is possible to achieve the worst-case approximation factor (both overcount and undercount) with just two sensors. In the next section, we extend the target placement algorithm proposed for ideal sensors to estimate target placements in the presence of non-ideal sensors.

4.2 Target Placement with Non-ideal Sensing

Let the sets $C' = \{c'_1, c'_2, \ldots, c'_n\}$ and $C = \{c_1, c_2, \ldots, c_n\}$ denote the noisy and the true readings of the sensor set $S = \{s_1, s_2, \ldots, s_n\}$. Let us associate the set of variables $Z = \{z_1, z_2, \ldots, z_{2n}\}$ with the sorted set $P = \{p_1, p_2, \ldots, p_{2n}\}$ of start and end point of the sensor ranges, where, as for ideal sensors, z_i denotes the number of points to the left of point p_i. We now show that a feasible target placement can be obtained even for non-ideal sensors,

Theorem 5. *Given a set of n non-ideal sensors and their readings, we can find a placement for targets in the network which satisfies all sensor readings.*

Of course, both the target placement as well as the number of targets estimated may be different from the ones found using non-noisy counts, but the approximations are guaranteed to be within the range of accuracy given by our theorems.

In the next section, we consider the target counting problem when the sensors and the targets are scattered in a two-dimensional plane.

5 Two Dimensional Target Counting

We begin with an example to argue that, unlike in the one dimension, approximation within a constant factor is not achievable for the two-dimensional target counting problem. The construction is quite simple, and shown in Figure 5. Imagine starting with n circles, centered at the origin. (The circles represent the sensing ranges of our idealized counting sensors.) We keep one circle stationary, and translate the centers of the remaining $n-1$ circles by $\{\delta, 2\delta, \ldots, (n'-1)\delta\}$ along the positive X-axis, where δ is chosen such that $(n-1)\delta$ is less that the radii of these circles.

With this arrangement of sensors, consider two different sets of target placements. In the first case (left figure), we place k targets at the origin. In the second case (right figure). we place k targets each near the top of each sensor's range. It is easy to see that in both cases, each sensor counts precisely k targets in its range, but the total number of targets present is k in the first case, and nk in the second case. Because the two cases are indistinguishable based on the

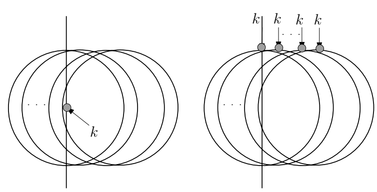

Fig. 5. Two scenarios with identical sensors information, but different total target counts: k for the left figure, and nk for the right figure

sensors' counts, no algorithm in this model can count targets with any constant factor accuracy. We summarize this result in the following theorem.

Theorem 6. *For arbitrary arrangements of sensors and targets, no algorithm can achieve a constant-factor approximation of the target counts even with idealized counting sensors in two dimensions.*

Clearly, the main source of difficulty is the overlapping sensing ranges. The construction of Figure 5 achieves the impossibility result by forcing an unbounded level of overlap among the ranges. In the following, we argue that if the degree of overlap is at most m, then one can approximate the target count within factor $\sqrt{m}$; thus, in practice, where we expect the overlap to be small, the approximation may be acceptable. By the degree of overlap, we mean the maximum number of sensing ranges that cover a point in the plane.

Theorem 7. *If the maximum degree of sensor overlap is m, then we can approximate the total number of targets in two dimensions to within a factor $\sqrt{m}$.*

If we consider the non-idealities of convex ranges and noise parameter as defined in Section 2, then for these non-ideal sensors the proof given above can be extended to obtain an approximation factor of $\sqrt{\frac{m(1+\rho)}{1-\rho}}$.

6 Related Work

The problem of detecting and tracking targets is of broad interest to many applications dealing with unattended monitoring and surveillance, with a rich literature in many disciplines, including computer vision, signal processing, ad hoc networks etc. [7,12,9,13]. The research goals in these areas, however, are different from those being pursued in sensor networks. In particular, the vision and signal processing communities are concerned with extraction of distinguishing features in detailed signals (e.g. images) and classifying the targets (e.g. tanks or cars). The mobile and ad hoc network communities have considered tracking with the goal of maintaining the state of network connectivity. In these cases, the nodes try to track other nodes using mobility models so that routing can be achieved successfully.

Counting targets is closely related to monitoring, intrusion detection and tracking targets. Counting is often the first step in most of these applications. Research in sensor networks has seen a lot of work in tracking multiple targets [14,15,16,17,18,19,20,21] and almost every piece of work assumes that the number of targets in the network is known. Our work is closely related to [5,6] in terms of deriving fundamental limits for tracking and counting targets using a minimal sensing model. These papers use a binary sensing model, which has also been considered by [1,2,3,4]. Counting sensor model is similar to the binary model in terms of minimal sensing, instead of transmitting a bit of information, counting sensors transmit an integer representing the number of targets in their range. The problem of counting targets is explored in [6] and the authors show that even in one dimension, counting targets accurately is not possible using binary sensing model unless the targets are spaced far apart from each other. Gfeller et. al. [22] add to the basic binary sensing model by considering mobile binary sensors and show that even then the problem is hard. Counting targets is also addressed in [23], the sensors considered are proximity sensors and sense the amplitude only. The target counting is then represented as peak counting problem in the aggregate sensor network, but the framework assumes that the targets are well separated. In [24], the authors look at the problem of counting the number of people in a crowd using image sensors. They subtract background from the image and then count number of visual hulls to count number of people. Their focus is on geometric hull computation and our techniques can be used on top of their algorithms to provide the bounds and counts. In [25] the authors use topological integration theory to provide expected target counts as compared to the deterministic bounds provided in this paper. The model of sensing considered is similar in the sense that the sensors give a count of the number of objects. However, the paper does not make any assumptions about sensing shapes and proves that the expected counts is the best one can hope for without geometry.

References

1. Arora, A., Dutta, P., Bapat, S., Kulathumani, V., et al.: A line in the sand: A wireless sensor network for target detection, classification, and tracking. Computer Networks (2004)
2. Aslam, J., Butler, Z., Constantin, F., Crespi, V., et al.: Tracking a moving object with a binary sensor network. In: SENSYS (2003)
3. Lazos, L., Poovendran, R., Ritcey, J.: Probabilistic detection of mobile targets in heterogeneous sensor networks. In: IPSN (2007)
4. Oh, S., Sastry, S.: Tracking on a graph. In: IPSN (2005)
5. Shrivastava, N., Mudumbai, R., Madhow, U., Suri, S.: Target tracking with binary proximity sensors: fundamental limits, minimal descriptions, and algorithms. In: SENSYS (2006)
6. Singh, J., Kumar, R., Madhow, U., Suri, S., et al.: Tracking multiple targets using binary proximity sensors. In: IPSN (2007)
7. Cai, Q., Aggarwal, J.K.: Tracking human motion using multiple cameras. In: ICPR (1996)
8. Nguyen, N., Venkatesh, S., West, G., Bui, H.: Multiple camera coordination in a surveillance system (2003)
9. Stauffer, C., Eric, W., Grimson, L.: Learning patterns of activity using real-time tracking. IEEE Transaction on Pattern Analysis Machine Intelligence (2000)
10. Oh, S., Chen, P., Manzo, M., Sastry, S.: Instrumenting wireless sensor networks for real-time surveillance. In: International Conference on Robotics and Automation (2006)
11. Cormen, T., Lieserson, C., Rivest, R., Stein, C.: Introduction to algorithms (2004)
12. Collins, R., Lipton, A., Fujiyoshi, H., Kanade, T.: Algorithms for cooperative multisensor surveillance. Proceedings of the IEEE 89, 1456–1477 (2001)
13. Zaidi, Z., Mark, B.: A mobility tracking model for wireless ad hoc networks. In: IEEE WCNC (2003)

14. Chen, H., Kirubarajan, T., Bar-Shalom, Y.: Multiple target tracking with multiple finite resolution. In: 5th International Conference on Information Fusion (2002)
15. Hwang, I., Roy, K., Balakrishnan, H., Tomlin, C.: A distributed multiple-target identity management algorithm in sensor networks. In: IEEE Conference on Decision and Control (2004)
16. Jung, B., Sukhatme, G.: Tracking targets using multiple robots: The effect of environment occlusion. Autonomous Robots (2002)
17. Liu, J., Liu, J., Reich, J., Cheung, P., et al.: Distributed group management for track initiation and maintenance in target localization applications. In: Zhao, F., Guibas, L.J. (eds.) IPSN 2003. LNCS, vol. 2634, pp. 113–128. Springer, Heidelberg (2003)
18. Mechitov, K., Sundresh, S.: Cooperative tracking with binary-detection sensor networks. In: SENSYS (2003)
19. Oh, S., Hwang, I., Roy, K., Sastry, S.: A fully automated distributed multiple-target tracking and identity management algorithm. In: AIAA Guidance, Navigation, and Control Conference (2005)
20. Rachlin, Y., Negi, R., Khosla, P.: Sensing capacity for discrete sensor network applications. In: IPSN (2005)
21. Shin, J., Guibas, L., Zhao, F.: A distributed algorithm for managing multi-target identities in wireless ad-hoc sensor networks. In: Zhao, F., Guibas, L.J. (eds.) IPSN 2003. LNCS, vol. 2634, pp. 223–238. Springer, Heidelberg (2003)
22. Gfeller, B., Mihalak, M., Suri, S., Vicari, E., et al.: Counting targets with mobile sensors in an unknown environment. In: Kutyłowski, M., Cichoń, J., Kubiak, P. (eds.) ALGOSENSORS 2007. LNCS, vol. 4837, pp. 32–45. Springer, Heidelberg (2008)
23. Fang, Q., Zhao, F., Guibas, L.: Counting targets: Building and managing aggregates in wireless sensor networks. PARC Technical Report (2002)
24. Yang, D., Gonzalez-Banos, H., Guibas, L.: Counting people in crowds with a real-time network of image sensors. In: International Conference on Computer Vision (2003)
25. Baryshnikov, Y., Ghirst, R.: Target enumeration in sensor networks via integration with respect to euler characteristic (2007)