Sensor Placement for Triangulation Based Localization

Onur Tekdas, Student Member, IEEE, and Volkan Isler, Member, IEEE

Abstract
Robots operating in a workspace can localize themselves by querying nodes of a sensor-network deployed in the same workspace. This paper addresses the problem of computing the minimum number and placement of sensors so that the localization uncertainty at every point in the workspace is less than a given threshold. We focus on triangulation based state estimation where measurements from two sensors must be combined for an estimate.

This problem is NP-hard in its most general form. For the general version, we present a solution framework based on integer linear programming and demonstrate its application in a fire-tower placement task. Next, we study the special case of bearing-only localization and present an approximation algorithm with a constant factor performance guarantee.

Note to Practitioners
Sensor networks can provide robust and scalable solutions to the localization problem which arises in numerous automation tasks. A common method for localization is triangulation in which measurements from two sensors are combined to obtain the location of a target.

In this work, we study the problem of finding the minimum number, and placement of sensors in such a way that the uncertainty in localization is bounded at every point in the workspace when triangulation is used for estimating the location of a target. We present an efficient geometric algorithm for bearing-only localization which can be used for the deployment of camera-networks. We also present a generic framework for arbitrary uncertainty metrics, and demonstrate its utility in an application where watchtowers are deployed to detect forest fires.

Index Terms
Sensor network deployment, localization, approximation algorithms.

SHORT PAPER
Corresponding Author is Onur Tekdas.
Email: tekdas@cs.umn.edu
Tel: +1-612-625-4002
Fax: +1-612-625-0572
Mailing Address:
University of Minnesota
Department of Computer Science and Engineering
4-192 EE/CS Building
200 Union Street SE
Minneapolis, MN 55455, USA

Volkan Isler
Email: isler@cs.umn.edu
Tel: +1-612-625-1067
Fax: +1-612-625-0572
Mailing Address:
University of Minnesota
Department of Computer Science and Engineering
4-192 EE/CS Building
200 Union Street SE
Minneapolis, MN 55455, USA

Both authors are with the Department of Computer Science and Engineering, University of Minnesota. Emails: {tekdas, isler}@cs.umn.edu. Corresponding author is Onur Tekdas. Mailing Address: 4-192 EE/CS Building, 200 Union Street SE, Minneapolis, MN 55455, USA. Tel: 612-625-4002 Fax: 612-625-0572.
Earlier versions of the results in this paper appeared in [1], [2]. In this full version, we improve the approximation ratio of the placement algorithm presented in [1] (Section IV). This work is supported in part by NSF grants 0907658, 0917676 and 0936710.
Sensor Placement for Triangulation Based Localization

Abstract—Robots operating in a workspace can localize themselves by querying nodes of a sensor-network deployed in the same workspace. This paper addresses the problem of computing the minimum number and placement of sensors so that the localization uncertainty at every point in the workspace is less than a given threshold. We focus on triangulation based state estimation where measurements from two sensors must be combined for an estimate.

This problem is NP-hard in its most general form. For the general version, we present a solution framework based on integer linear programming and demonstrate its application in a fire-tower placement task. Next, we study the special case of bearing-only localization and present an approximation algorithm with a constant factor performance guarantee.

I. INTRODUCTION

The process of determining the location of an object from two sensor measurements is commonly referred to as triangulation. For example, a forest fire is localized by triangulating bearing measurements from two fire-towers whose relative locations are known. Similarly, images taken from two calibrated cameras can be used to localize an object.

As sensors are becoming inexpensive, deploying many sensors in a workspace to provide localization services is becoming feasible (see e.g. [3], [4]). We believe that this technology provides a valuable alternative to on-board localization for mobile robots for the following reasons. First, on-board localization requires a sensor and a processor on each robot. In scenarios where many inexpensive, specialized robots (e.g. Roombas) operate in the same workspace, we find the best pair of sensors for that location. Second, localization with external sensors can be more robust than on-board localization especially when there are not many distinct features in the environment. Finally, localization with external sensors may be the only alternative in scenarios such as localizing adversarial entities. We note that the additional burden of calibrating external sensors can be relieved by using fully automated techniques [5]-[7].

For example, occlusions caused by the environment may prevent a sensor from participating in the triangulation process.

In this paper, motivated by these factors, we study the following problem: given an environment model, an estimation model and an uncertainty threshold, what is the minimum number and placement of sensors to guarantee that the uncertainty in estimating the location of a target in the workspace is not greater than the given threshold regardless of the target’s location?

Contributions: After formalizing the sensor placement problem in Section II-A, we present a general solution framework based on integer linear programming (Section III). Since the general problem is NP-hard, we also focus on a common special case, bearing-only localization, and present an approximation algorithm that deviates from the optimal solution only by a constant factor both in the number of cameras used and the uncertainty in localization (Section IV).

In the next section, we start with the problem formulation.

II. THE SENSOR PLACEMENT PROBLEM

A. Problem formulation

In this section, we formulate the Sensor Placement Problem for Triangulation Based Localization (SPP). In SPP, we are given a workspace \( \mathcal{W} \) which consists of all possible locations of the target. Let \( s \) be a \( k \)-tuple representing related sensor parameters which can include, for example, location and orientation of a sensor. The second input to SPP is \( \mathcal{Q} \), the domain of \( s \). In other words, \( \mathcal{Q} \) is the set of all possible placements of a single sensor. Finally, we are given a function \( U : \mathcal{Q} \times \mathcal{Q} \times \mathcal{W} \to \mathbb{R} \), where \( U(s_1, s_2, w) \) returns the uncertainty in localizing a target located at \( w \) using two sensors with parameters \( s_1 \) and \( s_2 \). In general the function \( U \) is specific to particular environment and sensing models. For example, \( U(s_1, s_2, w) \) can be infinite if the environment causes an occlusion between \( w \) and either \( s_1 \) or \( s_2 \). Similarly, if the sensors have range or field-of-view constraints, the function \( U \) can be defined to incorporate them. To simplify the notation, we left the dependency of \( U \) on environment and sensing models implicit throughout the text.

Let \( S = \{ s_1, \ldots, s_n \} \subseteq \mathcal{Q}^n \) be a placement of sensors where the \( i^{th} \) sensor has parameters \( s_i \). The quality of a placement in a workspace is determined as follows:

\[
U(S, \mathcal{W}) = \max_{w \in \mathcal{W}} \min_{s_i, s_j \in S} U(s_i, s_j, w)
\]

In other words, to establish the quality of a placement in a workspace, we take the largest uncertainty value over the entire workspace. To compute the uncertainty value for a specific location in the workspace, we find the best pair of sensors for that location.

We can now define the sensor placement problem: Given a workspace \( \mathcal{W} \), candidate sensor locations \( \mathcal{Q} \), an uncertainty function \( U \) and an uncertainty threshold \( U^* \), find a placement \( S \) with minimum cardinality such that \( U(S, \mathcal{W}) \leq U^* \).

It is easy to see that SPP is a hard problem by establishing its relation to the well-known \( k \)-center problem, which is NP-Complete. In the \( k \)-center problem, we are given a set of locations for centers and a set of targets along with a distance function \( d(i, j) \) between the centers and the targets. The objective is to minimize the maximum distance between any target and the center closest to it [9]. The converse problem, where the maximum distance from each vertex
to its center is given and the number of centers is to be minimized, is also NP-Complete [10]. The converse problem can be easily seen to be a special case of the SPP where the uncertainty function is chosen as \( U(s_i, s_j, w) = \min\{d(s_i, w), d(s_j, w)\} \). Hence, SPP is at least as hard as the converse \( k \)-center problem.

\[ y \leq \min \{d(s_i, w), d(s_j, w)\}, \text{ Hence, SPP is at least as hard as the converse } k \text{-center problem.} \]

To its center is given and the number of centers is to be minimized, is also NP-Complete [10]. The converse problem can be easily seen to be a special case of the SPP where the uncertainty function is chosen as \( U(s_i, s_j, w) = \min\{d(s_i, w), d(s_j, w)\} \). Hence, SPP is at least as hard as the converse \( k \)-center problem.

\[ y \leq \min \{d(s_i, w), d(s_j, w)\}, \text{ Hence, SPP is at least as hard as the converse } k \text{-center problem.} \]

To its center is given and the number of centers is to be minimized, is also NP-Complete [10]. The converse problem can be easily seen to be a special case of the SPP where the uncertainty function is chosen as \( U(s_i, s_j, w) = \min\{d(s_i, w), d(s_j, w)\} \). Hence, SPP is at least as hard as the converse \( k \)-center problem.

\[ y \leq \min \{d(s_i, w), d(s_j, w)\}, \text{ Hence, SPP is at least as hard as the converse } k \text{-center problem.} \]

To its center is given and the number of centers is to be minimized, is also NP-Complete [10]. The converse problem can be easily seen to be a special case of the SPP where the uncertainty function is chosen as \( U(s_i, s_j, w) = \min\{d(s_i, w), d(s_j, w)\} \). Hence, SPP is at least as hard as the converse \( k \)-center problem.

\[ y \leq \min \{d(s_i, w), d(s_j, w)\}, \text{ Hence, SPP is at least as hard as the converse } k \text{-center problem.} \]
The resulting placement has 32 variables. Since the number of branch-and-bound operations which took $2.680 \times 10^7$ CPU hours, we use only one of the duplicate pairs: $x_{ij}$ and $x_{ji}$. This reduces the number of $x_{ij}$ variables to 113, 136.

The optimal placement can not afford to place cameras too far from the workspace.

Let $R = \sqrt{U^*}$. The proposed placement algorithm consists of two phases. In the first phase, we choose a set of centers $C$ which will be used to determine the location of the cameras. In the second phase, we place cameras on circles whose centers coincide with the chosen centers and whose radii are at most $2R$. We will show that this placement achieves bounded deviation from the optimal one in terms of both the number of sensors and the performance guarantee.

The centers are chosen by the following algorithm:

**Algorithm selectCenters(workspace $W$):**

- $C = \emptyset, W \leftarrow W$
- while $W \neq \emptyset$
  - $c \leftarrow$ an arbitrary point in $W$
  - $C \leftarrow C \cup \{c\}$
  - $W \leftarrow W \setminus \{w : d(c, w) < 2R, w \in W\}$

The following lemma shows that the number of centers is small with respect to $|OPT|$.

**Lemma 1:** Let $C$ be the set of centers chosen by selectCenters and OPT be an optimal placement. $|OPT| \geq |C|$.

**Proof:** For each center $c \in C$, let us define $D(c, R)$ to be a disk centered at $c$ with radius $R$. Since the distance between the centers is at least $2R$, the disks $D(c, R)$ are pairwise disjoint. We claim that each disk $D(c, R)$ contains at least one camera in OPT, which proves the lemma.

Suppose the claim is true and let $c$ be a center such that OPT has no cameras in $D(c, R)$. But then, for any $s_i, s_j \in OPT$, the uncertainty in observing $c$ will be:

$$U(s_i, s_j, c) = \frac{d(s_i, c)d(s_j, c)}{|\sin \angle s_i cs_j|} \geq d(s_i, c)d(s_j, c) > R^2 = U^*$$

However, this means that OPT exceeds the uncertainty threshold on $c$. A contradiction! □

In the second phase, we use the set of centers to determine the placement of cameras.

**Algorithm placeSensors(centers $C$):**

- for each $c \in C$
  - $W_c \leftarrow \{w : d(c, w) < 2R, w \in W\}$
  - Let $T$ be any equilateral triangle whose circumcircle is $D(c, 2R')$ where $R' = \frac{\sqrt{3}}{2}R$
  - Place three sensors $s_1, s_2, s_3$ on the vertices of $T$ (See Figure 4).

In Figure 3 we illustrate the two phases of the algorithm. We only show the first two selected disk centers and the placement of sensors inside their disks.

Clearly, algorithm placeSensors places at most $3|OPT|$ cameras. (Lemma 1). All we need to show is that for any point $w$ in the workspace, we can find two sensors $s_i$ and $s_j$ such that $U(s_i, s_j, w) < 5.5U^*$. The next lemma shows the existence of such camera pairs.

**Lemma 2:** Let $S = \{s_1, s_2, s_3\}$ be the set of three cameras placed by placeSensors inside $D(c, 2R)$. For any point $w \in W_c$, there exists an assignment of two cameras $s_i$ and $s_j$, such that $U(s_i, s_j, w) < 5.5U^*$ where $1 \leq i, j \leq 3$.

**Proof:**
of the triangle such that the assignment of $P$ such that $D_{1P} = D(c, 2R') \setminus P^1$ into 6 regions by bisectors of triangle $T$. In this figure we consider the area $P_{23}^4$ shown in Figure 4. For any $w \in P_{23}^4$, the assignment of $s_2$ and $s_3$ satisfies $U(s_2, s_3, w) \leq 5.5U^*$.

We partition $P^2_3$ into 6 equal parts using the bisectors of triangle $T$ (See Figure 4). Suppose $w$ lies in region: $P_{23}^4$. Let $m_{13}$ be the midpoint of $[s_1, s_3]$, and $w'$ and $m_{13}$ be the intersection points of the boundary of $D(c, 2R')$ with $s_2$ and $s_2m_{13}$, respectively.

For clarity, we use the same notation as in Figure 6, i.e. $\alpha = \angle s_2w_3s_1$, $\beta = \angle s_2w's_3$, $\gamma = \angle w_3ws_1$ and $\delta = \angle s_1w_3m_{13}$. For any $w \in P_{23}^4$, the following inequalities hold:

(i) $\beta = \frac{\pi}{4}$, (ii) $0 \leq \gamma \leq \delta$, (iii) $\frac{\pi}{4} \leq \delta \leq \frac{\pi}{2}$, (iv) $\alpha = \beta + \gamma$. Finally, using (i)–(iv), we can bound $\alpha = \angle s_2w_3s_1$: $rac{\pi}{4} \leq \angle s_2w_3s_1 \leq \frac{\pi}{2}$. The distances $d(s_2, w)$ and $d(s_3, w)$ are upper bounded by: $d(s_2, w) \leq d(s_2, m_{13}) = 4R'$, $d(s_3, w) \leq d(s_3, m_{13}) = 2R'$.

Hence, $U(s_2, s_3, w) \leq \frac{4R'^2}{\sin \frac{\pi}{3}} = 3.6659U^* < 5.5U^*$.

Case $(w \in P^3)$:

Again, we partition $P^3$ into 6 equal regions using bisectors of triangle $T$ (See Figure 4). For any point $w$ inside $P_{13}^3$, we assign cameras $s_1$ and $s_2$. Let us say that $w$ is the intersection point of the boundary of $D(c, 2R')$ with $c_w$. Similarly, suppose that $w''$ and $s_3''$ are intersection points of the boundary of $D(c, 2R')$ with $c_w$ and $c_{s_2}$, respectively. To obtain a bound on the uncertainty, we first establish a lower bound on $\sin(\angle s_1w_3s_3) = \sin(\text{Angle}(r, \theta))$, followed by an upper bound on the product $d(s_1, w)d(s_3, w) = \text{Mult}(r, \theta)$. Finally, we show that both bounds are reached at the same point: $w = s_1''$.

For the remaining part of the proof, we will use polar coordinates as shown in Figure 7. We define uncertainty function $U_{\text{uncert}}(r, \theta)$ in polar coordinates as follows:

$$U_{\text{uncert}}(r, \theta) = \frac{\text{Mult}(r, \theta)}{\sin(\text{Angle}(r, \theta))}$$

$$\text{Mult}(r, \theta) = \sqrt{(r^2 - 4R'^2 \sin \theta + 4R'^2)(r^2 - 4R'^2 \cos \theta + 4R'^2)}$$

$$\text{Angle}(r, \theta) = \frac{\pi}{4} + \arctan\left(\frac{2R'^2}{r \cos \theta}\right) + \arctan\left(\frac{2R'^2}{r \sin \theta}\right)$$

where for all $w \in P_{13}^3$, $-\pi/6 \leq \theta \leq \pi/6$ and $2R' < r \leq 2R$.

First, we show that for any $w \in P_{13}^3$, $\text{Mult}(r, \theta) \leq \text{Mult}(2R, \pi/6)$, then we show $\sin(\text{Angle}(r, \theta)) \geq \sin(\text{Angle}(2R, \pi/6))$ where $(2R, \pi/6)$ corresponds to $s_3''$.

By Euclid’s exterior angle theorem (in any triangle the angle opposite the greater side is greater), we have $d(s_1, w) \leq d(s_1, w'')$ and $d(s_3, w) \leq d(s_3, w'')$. Therefore, $\text{Mult}(r, \theta) \leq \text{Mult}(2R, \pi/6)$.

By the extreme value theorem: $\text{Mult}(2R, \pi/6) \leq \text{Mult}(2R, \pi/6)$, $\text{Mult}(2R, \pi/6) \leq \text{Mult}(2R, \pi/6)$. Hence, for any $w \in P_{13}^3$, $\text{Mult}(r, \theta) \leq \text{Mult}(2R, \pi/6)$ holds.

The angle $\angle s_1w_3s_3$ is always between $\angle s_1w''s_3$ and $\angle s_1w's_3$ such that $\angle s_1w''s_3 = 2\pi/3$. Therefore, $\sin(\angle s_1w_3s_3)$ is lower bounded by $\min(\sin(\angle s_1w''s_3), \sin(2\pi/3))$, i.e.
function

\[
\sin (\text{Angle}(r, \theta)) \geq \min (\sin (\text{Angle}(2R, \theta)), \sin(2\pi/3)).
\]

The function \text{Angle}(2R, \theta) has its local minima and maxima at \( \theta = -\pi/6 \) and \( \theta = \pi/6 \), respectively and it is increasing in its domain. This can be shown by investigating the domain and roots of the first derivative of \text{Angle}(2R, \theta). Therefore, \( \sin (\text{Angle}(r, \theta)) \geq \min (\sin (\text{Angle}(2R, -\pi/6)), \sin (\text{Angle}(2R, \pi/6)), \sin(\pi/3)) = \sin (\text{Angle}(2R, -\pi/6)). \)

Finally, \( U(s_2, s_3, w) \leq \frac{\text{Mult}(2R, -\pi/6)}{\sin(\text{Angle}(2R, -\pi/6))} = 5.4989U^* < 5.5U^* \).

In conclusion, \text{placeSensors} guarantees an uncertainty for all cases that is not greater than 5.5U^*, i.e. \( U(W, S) < 5.5U^* \).

Remark 1: We believe that the placement of the three sensors given in Algorithm \text{placeSensors} is optimal. To verify this, we numerically computed the error in localization as a function of the radius of the circle circumscribing the equilateral triangle formed by the three sensors. The minimum value was achieved at radius 1.25 which is very close to the radius used in the algorithm (2\( \sqrt{1/4} = 1.2599 \)). The small difference is due to discretization. To prove optimality, it remains to be shown that an optimal placement is symmetric around the center. We believe that this claim is true. However we do not have a proof of this statement.

As a comparison, we ran the ILP program for the case where the environment is a disk of radius two. Recall that the ILP takes a threshold as input, and computes a placement with the smallest number of sensors to achieve this threshold. It turns out that the precise threshold value is critical. The value given by the argument in Lemma 2, used in the analysis of the approximation algorithm is 5.4989. The number of sensors in the placement changes from three to four when the threshold changes from 5.499 to 5.498. The placement of the three sensors was identical to the one used in Lemma 2.

Is it possible to obtain a better uncertainty guarantee? In general, let us define an \( (\alpha, \beta) \)-approximation algorithm for sensor placement be an algorithm which places at most \( \beta \) times the number of cameras used in an optimal placement and guarantees a deviation of factor \( \alpha \) in uncertainty. From the results presented above, we have (5.5, 3) approximation algorithm. Clearly, there is a trade-off between \( \alpha \) and \( \beta \). Using algorithm \text{placeSensors} as a subroutine, we can obtain a class of approximation algorithms by covering each disk of radius 2R (used by \text{placeSensors}) with \( k \) disks of smaller radius. This guarantees a smaller deviation from \( U^* \). The problem now becomes a disk-covering problem: Given a disk of radius 2R, find the smallest radius \( r(k) < 1 \) required for \( k \) equal disks to completely cover the original disk. Clearly, this would guarantee a reduction of \( r(k)^2 \) in the performance guarantee of \text{placeSensors}, at the expense of increasing the number of cameras by a factor \( k \). The interested reader can find different values of \( r(k) \) in [15].

V. CONCLUSION

In this paper, we introduced a novel sensor placement problem where pairs of sensors are used to obtain estimates about a target’s location. We presented a general solution based on integer linear programming and an approximation algorithm for the special case of bearing-only localization in the absence of obstacles. We are currently working on extending the approximation guarantees to more complex environments. Preliminary results can be found in [16].

REFERENCES


Onur Tekdas is a PhD student in the Department of Computer Science and Engineering at University of Minnesota-Twin Cities. He received his MS degree (2008) in Computer Science from Rensselaer Polytechnic Institute and BS (2006) degree in Computer Engineering from Middle East Technical University, Turkey. His research interests are in robotics and sensor networks.

Volkan Isler is an Assistant Professor in the Computer Science Department, and a resident fellow at the Institute on Environment at the University of Minnesota. He obtained his MSE (2000) and PhD (2004) degrees in Computer and Information Science from the University of Pennsylvania. He obtained his BS degree (1999) in Computer Engineering from Bogazici University, Istanbul, Turkey. In 2008, he received the National Science Foundation’s Young Investigator Award (CAREER). He is currently co-chairing IEEE Society of Robotics and Automation’s Technical Committee on Networked Robots. His research interests are in robotics and sensor networks.