

No Map, No Problem: A Local Sensing Approach for Navigation in Human-Made Spaces Using Signs

Claire Liang¹, Ross A. Knepper¹, Florian T. Pokorny²

Abstract—Robot navigation in human spaces today largely relies on the construction of precise geometric maps and a global motion plan. In this work, we navigate with only local sensing by using available signage — as designed for humans — in human-made environments such as airports. We propose a formalization of “signage” and define 4 levels of signage that we call complete, fully-specified, consistent and valid. The signage formalization can be used on many space skeletonizations, but we specifically provide an approach for navigation on the medial axis. We prove that we can achieve global completeness guarantees without requiring a global map to plan. We validate with two sets of experiments: (1) with real-world airports and their real signs and (2) real New York City neighborhoods. In (1) we show we can use real-world airport signage to improve on a simple random-walk approach, and we explore augmenting signage to further explore signs’ impact on trajectory length. In (2), we navigate in varied sized subsets of New York City to show that, since we only use local sensing, our approach scales linearly with trajectory length rather than freespace area.

I. INTRODUCTION

Human-made environments give just enough hints for people to reliably navigate them without ever glancing at a map. Large numbers of visitors navigate places such as airports, malls, and stadiums every day. Oftentimes, the only information these people have to navigate are signs with high level directions, scattered throughout the environment. Robots may serve a variety of roles in these spaces: as guides, as teammates, as assistants, etc. For robots to naturally interact with humans it is helpful to understand the spatial cues that humans use. Signs are placed intentionally, and from them, humans can piece together a plan to their destination as they go, with minimal instruction. Building planners have purposely provided directions to pedestrians by means of signage, but may not have provided detailed metric maps to perform traditional path planning. Building planners and human navigators have an implicit contract on how to interpret these signs as a complete set of directions. Signage is well understood to be a rich source of semantic information for humans [1]; robots should be able to tap into this same source to understand the context of their space.

In this work, we hypothesize that, when sufficient signage is available in environments such as airports, a robot does not need to perform global mapping and can plan solely using available signage and local sensing. We test this hypothesis by creating a formalization of these signs and proposing an



Fig. 1: Examples of real signage in an airport algorithm for navigation that only requires local sensing and does not involve any global motion planning.

Robots that have actually been deployed in real world spaces are often unable to use natural signs [2]; they instead typically perform traditional path planning on a pre-generated, user-provided or robot-generated, precise metric map [3].

The amount of environment-specific engineering necessary to deploy robots — in particular, low-cost sensing robots — in dynamic human environments adds barriers to widespread adoption. Signage already provides information that deployed robots typically entirely ignore. Robots could use the directions intended for human users to reduce the amount of customization for each individual space. Airports, malls, and stadiums are transitory spaces, constantly evolving to account for emergencies and necessary adjustments. A building manager should be able to put up a temporary detour sign and assume that robots in the building will use it correctly.

In this paper we first propose one possible formalization of signage. We then define four properties of signage placement to quantify the quality of a given signage. We provide an algorithm that automatically checks the signage level of an environment and also present an algorithm that “interprets” signs on the fly to navigate a robot to its goal based only on local sensing and without global map construction. We then experimentally evaluate the impact of signage level on trajectory length.

II. RELATED WORK

Humans can use both metric and topological spatial reasoning and trade between them seamlessly. When trying to walk near the edge of a cliff, a human will likely focus on metric precision, while, when providing directions to a

¹Claire Liang and Ross A. Knepper did this research with The Department of Computer Science, Cornell University, Ithaca, NY, USA cliang@cs.cornell.edu, rak@cs.cornell.edu

²Florian T. Pokorny is with the Division of Robotics, Perception and Learning, EECS, KTH Royal Institute of Technology, Stockholm, Sweden

pizza delivery-person over the phone, a person can provide sparse, topological instructions. Work in neurology shows that different parts of brain fire when planning and navigating topological aspects of the environment versus metric aspects [4, 5]. Being able to strike a balance between different kinds of planning for navigation allows humans to be resilient and adapt to different environments. Modern robot systems, on the other hand, are engineered to rely heavily on metric maps and precise sensing [6]. Semantic instructions and contextual clues need to be converted into those metrically precise terms, often resulting in a complex conversion of representations that may lose information or generate extraneous noise and misleading representations [7, 8]. Similarly, in cases where the conversion is possible, such as for self-driving vehicles that use signs, this process often relies on the structured rules of the road to constrain the action space [9].

Robots have been deployed in airports and grocery stores [3] to explore interaction with humans [10, 11]. In one such study, Joosse and Evers [11] note that planning inefficiencies are detrimental for smooth interaction: “re-planning was perceived as taking quite long, participants’ general impression was that the robot was less suited for guiding passengers in a congested area, especially if they were under time pressure”. Our method plans rapidly with only local information, reducing long delays from re-planning.

The area of semantic mapping addresses many interesting parts of navigating human-made environments. Methods from this area jointly estimate hybrid metric, topological, and semantic representations of an environment and use this representation to navigate. However, these techniques often require a hand annotated topological graph representation of landmarks within a scene [12, 13], and thus do not scale well for use across a large range of environments. They are also less applicable to environments that have rapidly changing configurations or less distinct landmarks [14, 15].

Our work goes a step beyond classical semantic mapping by honing in specifically on signage and trying to avoid global map representations entirely. Map-free robotics work ranges from reactive control [16] (which does not provide global guarantees about plan completeness) to visual-cue only navigation [17] (which focuses on identifying the visual cues to identify goals or waypoints). SLAM methods also range widely across the spectrum of metric to semantic methods [18] but cannot make global completeness guarantees for map-free planning. Our work aims to eschew the global map, but enable a formalization that can, when sufficient signage is available, guarantee desirable completeness properties.

III. METHODS

A. Assumptions

We assume an environment $\mathcal{E} \subset \mathbb{R}^2$ modelled as a polygon with internal polygonal obstacles. A robot’s configuration is modelled as a point and defined by its position $(x, y) \in \mathcal{E}$. The robot is not provided a map of \mathcal{E} but is assumed to have a sensing radius r that allows it to sense the set $\mathcal{E} \cap \mathbb{B}_r(x, y)$, where $\mathbb{B}_r(x, y) = \{(x', y') : (x - x')^2 + (y - y')^2 \leq r^2\}$. In

this work, we assume noise-free perception and odometry for simplicity. We assume that $r > 2d$, where d is the diameter of the largest ball that can be inscribed at any point in \mathcal{E} . This version of the problem assumes static obstacles¹.

Informally, this means that the sensor radius must be larger than twice the width of the widest open-space. However, the robot only needs to be able to detect presence of obstacles and signs, with no further visual details. We believe that in practice, this sensing radius requirement aligns with the capabilities of modern cameras and is valid for most of our considered environments.

B. Choice of Skeletonization: The Medial Axis

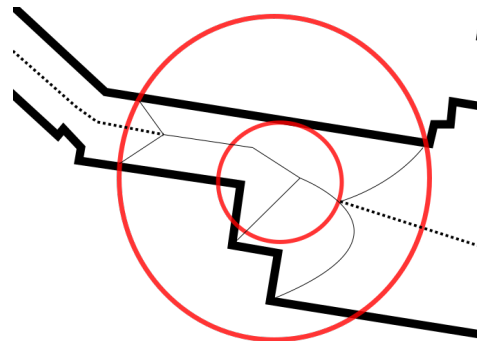


Fig. 2: An example of a local medial axis constructed within r . Bolded curves are generated from within the sensing circle, dotted lines represent segments from global medial axis.

Our approach relies on a graph-representable skeletonization of $\mathcal{E} \cap \mathbb{B}_r(x, y)$. While multiple such skeletonizations may be considered in general, we propose an approach based on the medial axis representation. We show in Lemma III.1 that we can provably reconstruct a local neighborhood of the global medial axis within $r/2$ of the robot’s current position using only local sensing. This insight is adapted from existing work on digital geometry analysis and processing [19].

Recall that the medial axis of a polygon \mathcal{E} is defined as follows (see [20]): let $CORE(\mathcal{E})$ be the set of maximally inscribed disks in \mathcal{E} . The Medial Axis $\Gamma(\mathcal{E})$ is then the set of the centers of all disks in $CORE(\mathcal{E})$ [20]. An example construction of a medial axis is included in Fig. 2 [21]. The medial axis is a subset of the Voronoi diagram generated by the boundary of \mathcal{E} [22].

The medial axis $G = \Gamma(\mathcal{E})$ forms a graph. We denote the vertices of G by V and edges by E . Each edge connects two vertices. Vertices fall into three main categories: *forks* if they have degree greater than 2, *continuation vertices* if they have degree 2, and *end points* if they have degree 1. Degree zero occurs only for connected components of \mathcal{E} that are disks.

The following result allows the robot to reconstruct a part of the *global medial axis* $\Gamma(\mathcal{E})$ within $r/2$ of the robot’s position given *only local sensing up to radius r* .

Lemma III.1. *Let $\mathcal{E} \subset \mathbb{R}^2$ be a polygon and $(x, y) \in \mathcal{E}$. Assume that $r > 2d$, where d is the diameter of the largest*

¹Dynamic agents, such as humans, in the scene could be dealt with in an adapted version of Section III-D

disk that can be inscribed in \mathcal{E} . Then

$$\Gamma(\mathbb{B}_r(x, y) \cap \mathcal{E}) \cap \mathbb{B}_{\frac{r}{2}}(x, y) = \Gamma(\mathcal{E}) \cap \mathbb{B}_{\frac{r}{2}}(x, y).$$

Proof. Let $(x', y') \in \Gamma(\mathbb{B}_r(x, y) \cap \mathcal{E}) \cap \mathbb{B}_{\frac{r}{2}}(x, y)$ and consider the maximal inscribing disk $\mathbb{B}_s(x', y')$ at (x', y') in $\mathbb{B}_r(x, y) \cap \mathcal{E}$. This disk is also maximal in \mathcal{E} itself because any disk in \mathcal{E} containing it has diameter at most d and must in fact be contained in $\mathbb{B}_r(x, y)$ by the triangle inequality. Conversely, let $(x', y') \in \Gamma(\mathcal{E}) \cap \mathbb{B}_{\frac{r}{2}}(x, y)$, and consider the inscribing disk $\mathbb{B}_s(x', y')$ at (x', y') and maximal in \mathcal{E} . Its center lies in $\mathbb{B}_r(x, y) \cap \mathcal{E}$ and is also maximal in $\mathbb{B}_r(x, y) \cap \mathcal{E}$ since it is maximal in \mathcal{E} by assumption. \square

From this point on we will refer to the $\frac{r}{2}$ ball in which the local medial axis aligns with the global medial axis as the “planning ball”.

C. Signs

Sign Representation: For a given goal point g , each sign s in the environment consists of two pieces of data: the position of the sign (x, y) and the symbolic information σ on the sign. σ is the mapping of the real-life sign’s content to edge direction assignments on $\Gamma(\mathcal{E})$. The σ mapping can vary depending on perception and sensing capabilities; choosing the right mapping is an independently interesting research problem that will be addressed in future work. In our implementation, we represent σ as a list of mappings from vertices on $\Gamma(\mathcal{E})$ to their attached edges’ direction assignments, one mapping for each endpoint of each nearest edge² to the sign’s position. In other words, each mapping in σ provides direction assignments for all edges that have endpoints at either vertex (v_1, v_2) of the sign’s nearest edges.

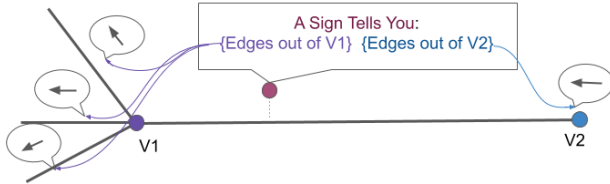


Fig. 3: A sign gives directions to the outgoing edges of the vertices for all edges nearest to the sign

This means that all edges attached to v_1 will have some direction assignment d which is either: ‘outgoing’, ‘ingoing’, or ‘NONE’ (and the same for v_2).

We assume the robot is able to determine which mappings belong to which endpoint of a sign’s nearest medial axis edge and which mappings belong with which edge (if there is more than one equidistant edge).

Sign Interpretation: Each time a sign is detected, if the robot is at a point on $\Gamma(\mathcal{E})$ nearest to the sign, it “interprets” the sign by acquiring the set-valued interpretation function

²We assume perfect distance measurements. Multiple edges can be equidistant to a sign.

$f : V \rightarrow \mathcal{P}(E \times \{\text{‘ingoing’}, \text{‘outgoing’}, \text{‘NONE’}\})$ for each of its current edge’s endpoints.³ Let a robot have a set of stored interpretation functions I . When the robot arrives at any fork or endpoint v' , it “evaluates” any of interpretation functions associated with v' in I and stores the resulting edge directions onto the $\Gamma(\mathcal{E})$ graph. As a result, each evaluation step results in a *medial axis mixed graph*: a graph where some edges are directed and some are not. For an additional optional optimization, at any continuation vertex there are two attached edges: e_1 and e_2 , the robot arriving from an edge e_1 can continue on using the same direction onto e_2 .

We describe the process of interpreting a single sign in [Algorithm 1](#). We assume that, during initialization, the robot has moved to a point on the medial axis that lies on an edge (this is always possible because there’s always a collision free path to the medial axis). So consider the case where the robot starts on an edge of the medial axis and there exists a sign within its planning ball. The robot first moves as close as it can to the sign without leaving its current edge ([Lemma III.2.i](#)). At this point, the robot is able to determine if its current edge is a global closest edge to the sign ([Lemma III.2.iii](#)). If this is the case, we “interpret” the sign by acquiring the interpretation functions described above.

Algorithm 1: Interpreting local signage.

Input: Local sensing sphere (B_r), Local Medial Axis Mixed Graph edge (e), sign (s)
Result: Acquires interpretation functions for sign s

```

if  $s \in B_{r/2}$  then
    move(nearestPt( $e, s$ ))
    if isNearestEdge( $e, s$ ) then
        |  $G = \text{update}(G, s, \sigma)$ ;
    end
end

```

Lemma III.2. Let $\mathcal{E} \subset \mathbb{R}^2$ be a polygon and $r > 2d$, where d is the diameter of the largest disk that can be inscribed in \mathcal{E} . Let $s \in \mathbb{B}_{\frac{r}{2}}(x, y)$ and $(x, y) \in \Gamma(\mathcal{E})$ and let e be an edge of $\Gamma(\mathcal{E})$ containing (x, y) . The answers to the following questions can be computed explicitly and solely from the local medial axis part $\Gamma(\mathcal{E}) \cap \mathbb{B}_{\frac{r}{2}}(x, y)$:

- i) Is (x, y) a nearest neighbor to s on e ?
- ii) If the nearest neighbor to s on e is not (x, y) , can we determine the direction along e towards the nearest neighbor?
- iii) Is (x, y) a nearest neighbor to s globally on $\Gamma(\mathcal{E})$?

Proof. i) and ii) Any edge e is either linear or a quadratic Bezier curve by construction of the medial axis, so the direction of travel towards the nearest neighbor can be computed explicitly by analyzing gradients of the squared distance function along e . In particular, testing whether the distance function to s locally around (x, y) exhibits a local

³Where $\mathcal{P}(X)$ for a set X denotes the powerset of X

minimum is sufficient to verify i) since the edge segment is either linear or quadratic.

iii) Observe that any nearest neighbor on $\Gamma(\mathcal{E})$ to s has distance at most $\frac{d}{2}$ to s by definition of d . If $\|s - (x, y)\| \geq \frac{d}{2}$, (x, y) is therefore not a nearest neighbor, while if $\|s - (x, y)\| \leq \frac{d}{2}$, we have $\frac{d}{2} \leq \frac{r}{4}$ and $\mathbb{B}_{\frac{d}{2}}(s) \subseteq \mathbb{B}_{\frac{r}{2}}(x, y)$, so the nearest neighbor still occurs within $G' = \Gamma(\mathcal{E}) \cap \mathbb{B}_{\frac{r}{2}}(x, y)$. Since G' is a finite structure with simple linear or quadratic edges, we can compute the nearest neighbors explicitly to check if (x, y) is among them. \square

Levels of Signage:

Real life signs are imperfect. There are many airports in the world where there are few signs, or signs that disagree with each other. We need a way to identify if a given set of signs in a space provides a plan to get a robot to its goal using our approach. We define three different levels of signage (*completeness, consistency, and full-specification*) as a way to formally categorize a set of signs in a space. These three levels are important for efficiency but are not necessary for the robot to reach the goal. We also define *validity*. If signage is invalid, the robot will not be able to reach its goal but will be able to identify the signs' implied directions on the graph as invalid. Further, the robot can identify the specific subgraph with invalid direction assignments.

For each of the following definitions, let $G \subset \mathbb{R}^n$ be the mixed graph that is output from the medial axis sign interpretation function f with vertex set V and edge set E . We assume as input the set of signs S that correspond to the input goal g . We will continuously refer to the medial axis structure in this section as G since the actual sign levels of requirement definitions are dependent only on mixed graph edge assignments rather than geometric constraints of the medial axis.

Definition 1 (Consistent Graph Signage). Given a goal point $g \in \mathcal{E}$ and a set of sign positions S , for an arbitrary pair of signs $s, s' \in S$ and their interpretation functions f_s and $f_{s'}$, let $\psi = \text{image}(f_s)$ and $\psi' = \text{image}(f_{s'})$. S is consistent if $\forall (e, d) \in \psi$ and $\forall (e', d') \in \psi'$, if $e = e'$ then $d = d'$ or $d = \text{'None'}$ or $d' = \text{'None'}$. That is, no pair of signs is ever interpreted to two opposing directions for the same edge in E . See an example in (a) in Fig. 4.

The three following definitions assume consistent signage.

Definition 2 (Fully-specified Graph Signage). Assume a goal point g , and a set of signs S and the mixed graph $G = (V, E)$ generated using the interpretation functions for all $s \in S$. S is fully-specified if every edge in E is directed. See an example in (c) in Fig. 4.

Definition 3 (Valid Graph Signage). Assume a goal point g , and a set of signs S and the mixed graph $G = (V, E)$ generated using the interpretation functions for all $s \in S$. Let F be the set of fork vertices s.t. the distance from any $v \in F$ to g is greater than $\frac{r}{2}$ and for which v 's associated edges are directed. S is valid if $\forall v \in F$ there exists at least one outgoing edge and at least one incoming edge. See an

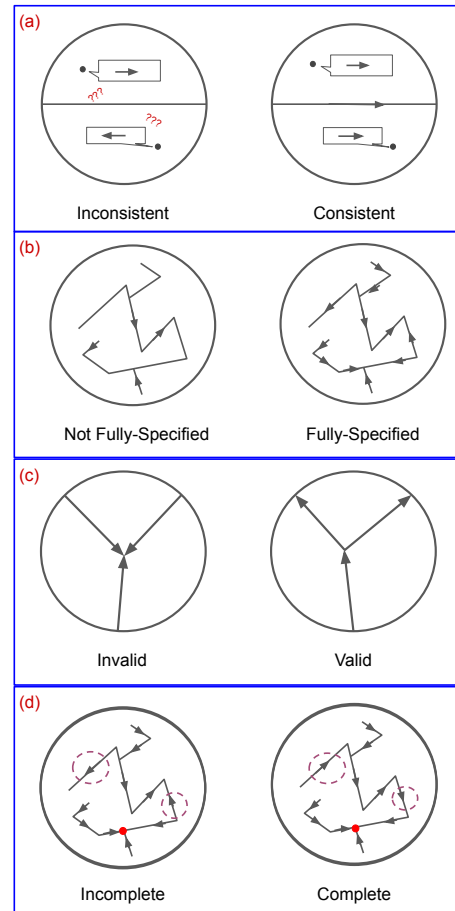


Fig. 4: Examples of (a) inconsistent direction assignment compared to an consistent direction assignment (b) a graph without fully-specified direction assignments compared to an graph with fully-specified direction assignments (c) invalid direction assignment compared to an valid direction assignment. The goal is not within sensing radius in both cases (d) a graph with incomplete direction assignment compared to a graph with complete direction assignment. Differences are circled and the goal is illustrated with a red point example in (b) in Fig. 4.

Definition 4 (Complete Graph Signage). Assume a goal point g , and a set of signs $S \subseteq \mathcal{E}$ and the mixed graph $G = (V, E)$ generated using the interpretation functions for all $s \in S$. S is complete if from any point $p \in G$, there exists a path to g that only takes directed edges in G . See an example in (d) in Fig. 4.

If a set of signs does not meet any of the aforementioned levels the set can potentially be augmented or modified with artificial signage.

Generating Artificial Signage:

We provide an naive algorithm to generate artificial signage that augments an original set of signs to reach a desired level of signage in Algorithm 2. Let our original set of signs be S' . We start by creating a fully directed graph G' . We assign directions to the $\Gamma(\mathcal{E})$ graph by performing any search algorithm for a path to the robot's goal from the vertices of $\Gamma(\mathcal{E})$. We assume that Algorithm 1 S' 's generated direction assignments are consistent with G' 's directions. Then, for each vertex, we place a sign at the centroid of each of

the vertex’s attached edges. These signs contain only the respective vertex’s direction information. Since each directed edge maps to at least one sign in this process, following [Algorithm 1](#) on the augmented set of signs will output the same mixed graph as was input to [Algorithm 2](#).

Algorithm 2: Generating Signage

Input: Medial Axis Mixed Graph ($G : (V, E)$), Original signs S'

Output: set of signs S

$S = \{\}$

for $v \in V$ **do**

for $e \in v.edges$ **do**

$s = (p_i = e.centroid,$

$\sigma = [\sigma_v = v.edges.directions,$
 $\sigma_{other v} = directionless]);$

$S.add(s);$

end

end

return S

D. Planning

Any policy that follows the directions returned by [Algorithm 1](#) can use our approach to navigate to its goal. As a concrete example, we present the following policy:

Let the robot ρ store: (1) a list of seen signs, (2) interpretation functions (3) and direction assignments for edges.

For each timestep, ρ senses, then moves to the nearest point on the medial axis if it is not already on it. It then updates its directions, interprets signs. If ρ is positioned at a vertex, it converts stored sign information corresponding to its current vertex into directions and picks a new edge to traverse. The robot prioritizes edges that are labeled with an outgoing direction. If there are no outgoing edges, the robot chooses randomly from the remaining undirected edges. The robot follows its chosen edge until it reaches the next point where a sign can be evaluated on a vertex. This process is illustrated in [Fig. 5](#). Since ρ ’s movement at timestep t is restricted to actions that end within its current sensing circle, ρ ’s sensing circle at t will have nonempty overlap with its sensing circle at time $t-1$. This property, in conjunction with odometry, keeps edge directions and geometry consistent across timesteps.

IV. EVALUATION

We evaluate the following hypotheses each with its own set of experiments.⁴

- 1) Using our formalization for real-world signage improves navigation trajectory length. Additionally, more signage (with correct directions) can improve performance.
- 2) Our local-sensing only approach means computation time and resource usage scale linearly with trajectory length rather than size of the space.

⁴The airport maps, real life signage annotations, and code will all be made available on the github repo.

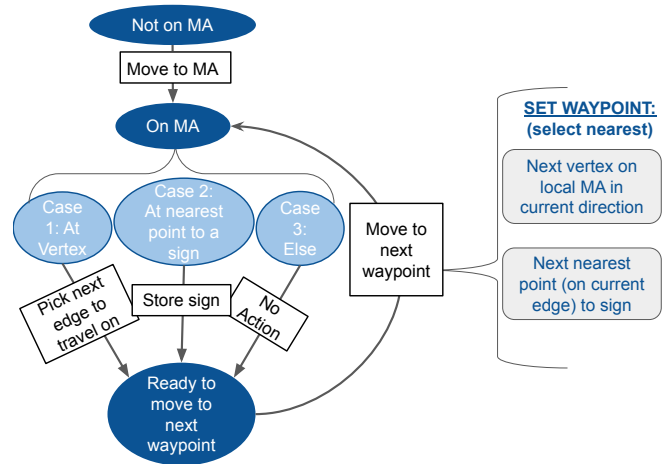


Fig. 5: Flow chart of control policy to navigate to the medial axis and follow along prescribed directions while resolving signs along the way.

For (1) we use real airport maps and real signs and further augment with artificial signage to evaluate the impact of signage levels on trajectories taken. For (2) we examine navigation time on real world city maps with artificially generated signage to show that using only local sensing allows navigation time to scale with trajectory length rather than the size of the space.

1) *Real Airport Maps:* We compiled 20 airport maps available via Google Maps and annotated them with the actual signs from Google Streetview (with the exceptions of ITH and ORD, which were hand-gathered and hand-labeled). The labeling process was done by volunteers who found signs using Google Streetview and noted the directions they interpreted from the signs. These steps are further illustrated in [Fig. 8](#). Real human-made environments like these almost never have what we define as “complete, fully-specified, and consistent signage.” With our sign formalization, we wanted to investigate the impact of density of (correct) signs in a space to the trajectory the robot would take. Therefore, we additionally evaluated our technique on the original dataset as well as an augmented dataset for each signage level.

For each of these maps we compare against a simple baseline that does not use signage. This baseline performs a random walk along the medial axis graph. We then create two sets of signage with the original signage plus augmented signage. One set of augmented signage is “complete”, and we take a random subset of this signage such that the directions are “consistent” but not “complete.” We compare the performance of the random walk baseline to a robot using our algorithm on real signs, the consistent signs, and the complete signs. We also compare our performance to the optimal trajectory length given a complete global map; this serves as an upper bound for performance. The point of doing this comparison is to show the benefit that more correct signs provides for trajectory length and evaluate our algorithm with respect to the upper bound in performance. We initialize our robot with a random starting location and starting direction for each trial. The results for all twenty real-life airport maps are shown in Table 1. The airports are

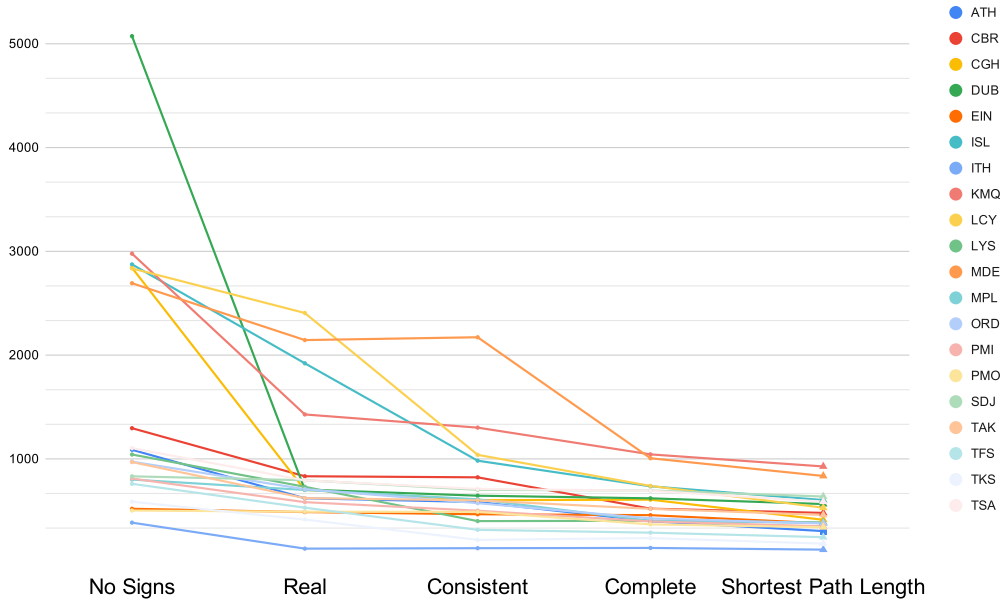


Fig. 6: 20 airports’ average trajectory length across 50 runs for each of 4 signage levels: (1) no signs (equivalent to a random walk), (2) real signage, (3) augmented consistent signage, and (4) augmented complete signage. For comparison, the graph includes triangle points on the rightmost side: the upper bound for robot performance (shortest path length along the medial axis graph).

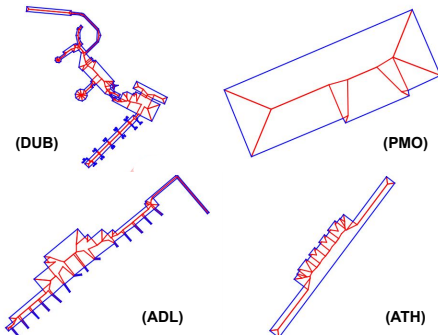


Fig. 7: Dublin (DUB), Palermo (PMO), Adelaide (ADL), and Athens (ATH) Airports from our real airport dataset pictured with their medial axis.

listed in order of increasing average branch factor of the medial axis graph.

In Fig. 9 we show how each step of artificial sign augmentation adds to the mixed graph $\Gamma(\mathcal{E})$ for Komatsu airport (seen in Table I’s KMQ row). In Table I, one can see that the performance between “real” and “consistent” signage is not that different. This lack of change is because the consistent signage generated for this instance does not have any directions for the small corridor at the top of the airport (the magnified portion in subfigure c) and the robot is still performing a pure random walk on that subgraph.

2) *City maps*: We also demonstrate how our technique scales (since we use only local sensing rather than needing to process the entire global map) using real city maps with artificially generated signage. Each map receives signage that is valid, complete, fully-specified, and consistent. We initialize our robot with a random starting location and starting direction for each trial.

The city maps we use for these trials are OpenStreetMap data gathered from New York City. City blocks generally lie on a fixed grid; this provides a consistent measure of scale.

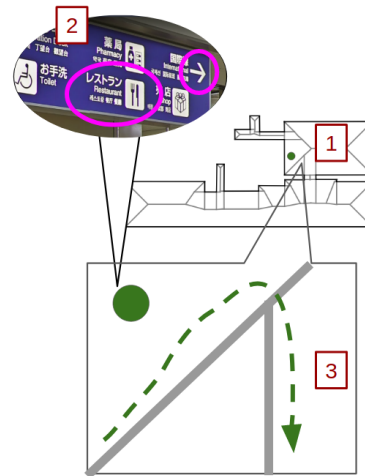


Fig. 8: This figure illustrates an example from the labeling process done to collect real life signs from Komatsu airport (KMQ) the steps are as follows: (1) locate the sign on the map and note its location (2) interpret the content of the sign’s directions (3) convert the sign’s content to edge assignments on the medial axis (according to the sign representation described in Section III-C).

A sample from each size are in Fig. 10 and results are in the accompanying table.

Observations about signage: The results from experiment 1 (Fig. 6) show that the interpretation of consistent, sufficient, and complete signage as a proxy for a map and global trajectory plan generates robot trajectory lengths close to the shortest path in each space. Further, we show that sufficiency, completeness, and consistency each improve the robot’s ability to use signage to navigate to its goal. However, even incomplete, inconsistent, and not fully-specified signage is better than no signage at all.

Signage is more important in spaces like the Dublin airport (DUB) where one crucially placed sign allows the robot to

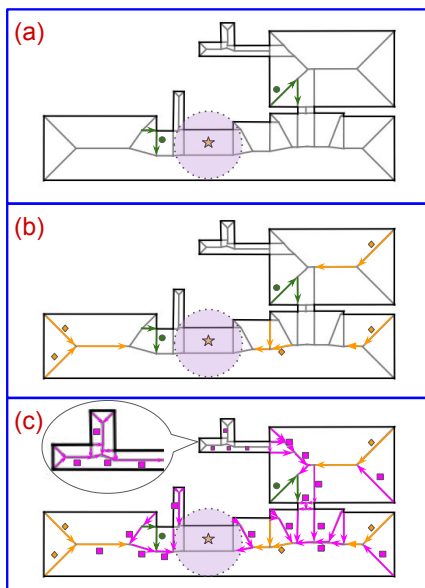


Fig. 9: Komatsu Airport (KMQ) with directions, depicted by arrows in their sign level’s respective color, assigned to their $\Gamma(\mathcal{E})$ according to (a) real signage (green circles) (b) real signage augmented to consistent signage (orange diamonds) (c) consistent signage augmented to complete signage (fuschia squares). The goal, a food court, is depicted with a yellow star, the translucent purple circle around the goal represents the locations where the robot can sense the goal. In our simulations the robot is randomly dropped in the freespace and must use signs to find its way to the goal. Signs are not required to provide directions for all adjacent edges.

escape the trap of a narrow passageway. Signage is less important in airports with a wide, open rectangular shape (e.g. Palermo Airport (PMO)), since the ratio of the size of the space to the subset of the space from which the robot can see the goal is much lower: it’s easy to see where you’re going from any point.

The use of the medial axis as the skeletonization of choice also has its weaknesses. For example, in Fig. 7, the general shapes of the Athens (ATH) and Adelaide (ADL) airports are similar, but because of the gate ramps included in the ATH, the number of edges in the medial axis graph is significantly higher. The number of signs necessary to satisfy complete signage is higher in ATH than in ADL. One could bypass this issue by pruning the medial axis.

We cannot make conclusions about how signage is actually used universally from our dataset of twenty airports. However, anecdotally, winding airports with long hallways and narrow passageways seem more reliant on signage to guide pedestrians to their destinations. This observation matches our intuition and supports the belief that the amount of signage in real world airports is proportional to the difficulty of navigating their spaces.

The results from experiment 2 (Fig. 10) show that, by using exclusively local sensing, we can remove the requirement for a global metrically precise map and still plan and navigate effectively through the space. With our technique, computational overhead scales linearly as a function of trajectory length rather than the total size of the space.

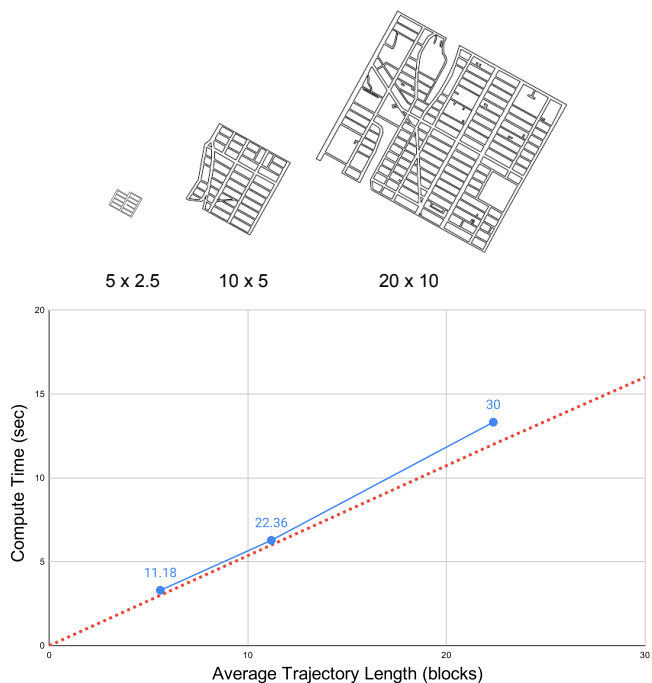


Fig. 10: Compute time of different sized snippets of New York City. Each map size has 10 different snippets of the city. The graph plots the average across trials for three different map sizes (5x2.5, 10x5, and 20x10). Each block is approximately 3 times wider than it is long.

V. DISCUSSION

A. Future Work and Limitations

Currently, our technique requires a strong assumption about the robot’s sensing radius relative to the widest corridor in the environment. To relax this assumption, we could consider any subset of the space wider than the robot’s actual sensing radius as “open space” and define a different policy for robot behavior in these subsets.

We also need to consider translating realistic sensor input to a representation that aligns with our sensing ball model. To do this, we must first sense obstacles in a scene and determine whether or not they should be omitted when computing the medial axis. Second, we must address the homography estimation problem of converting arrows on a sign seen at an angle into an executable heading angle for the robot. Third, we must address sign detection and sign text and symbol classification.

The medial axis can be used for more efficient and interesting control policies than our presented proof-of-concept policy. The medial axis is a subset of the constrained segment Voronoi graph, which can form the cell decomposition for a sequential control approach [23].

Separately, interpreting signs is valuable for applications beyond robotics. For example, our approach could be used on a smartphone as a navigational aid for humans.

B. Conclusion

In this work we present an approach for robots to navigate human spaces by using locally-sensed signs. For suitable signage and sensing radii, our approach provides global plan

completeness guarantees with only local sensing, bringing robot navigation closer to its human counterpart.⁵ Robots should not need extra infrastructure to operate in human spaces. In this work we take a step in this direction; the end goal of this line of research is to deploy robots in unmodified airports, malls, and stadiums among real human pedestrians.

ACKNOWLEDGMENT

The first two authors of this work are supported by the National Science Foundation under Grant No. 1646417. We are grateful for this support.

REFERENCES

- [1] J. Meis and Y. Kashima, “Signage as a tool for behavioral change: Direct and indirect routes to understanding the meaning of a sign,” *PLoS one*, vol. 12, no. 8, 2017.
- [2] A. Valada, J. Vertens, A. Dhall, and W. Burgard, “Adapnet: Adaptive semantic segmentation in adverse environmental conditions,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 4644–4651.
- [3] R. Triebel, K. Arras, R. Alami, L. Beyer, S. Breuers, R. Chatila, M. Chetouani, D. Cremers, V. Evers, M. Fiore *et al.*, “Spencer: A socially aware service robot for passenger guidance and help in busy airports,” in *Field and service robotics*. Springer, 2016, pp. 607–622.
- [4] A.-H. Javadi, B. Emo, L. R. Howard, F. E. Zisch, Y. Yu, R. Knight, J. P. Silva, and H. J. Spiers, “Hippocampal and prefrontal processing of network topology to simulate the future,” *Nature communications*, vol. 8, p. 14652, 2017.
- [5] E. Patai and H. Spiers, “Human navigation: occipital place area detects potential paths in a scene,” *Current Biology*, vol. 27, no. 12, pp. R599–R600, 2017.
- [6] K. Charalampous, I. Kostavelis, and A. Gasteratos, “Recent trends in social aware robot navigation: A survey,” *Robotics and Autonomous Systems*, vol. 93, pp. 85–104, 2017.
- [7] T. Taketomi, H. Uchiyama, and S. Ikeda, “Visual slam algorithms: a survey from 2010 to 2016,” *IPSJ Transactions on Computer Vision and Applications*, vol. 9, no. 1, p. 16, 2017.
- [8] S. Oßwald, M. Bennewitz, W. Burgard, and C. Stachniss, “Speeding-up robot exploration by exploiting background information,” *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 716–723, 2016.
- [9] A. De La Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armingol, “Road traffic sign detection and classification,” *IEEE transactions on industrial electronics*, vol. 44, no. 6, pp. 848–859, 1997.
- [10] T. Kanda, M. Shiomi, Z. Miyashita, H. Ishiguro, and N. Hagita, “A communication robot in a shopping mall,” *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 897–913, 2010.
- [11] M. Joosse and V. Evers, “A guide robot at the airport: First impressions,” in *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 2017, pp. 149–150.
- [12] M. R. Walter, S. Hemachandra, B. Homberg, S. Tellex, and S. Teller, “Learning semantic maps from natural language descriptions,” in *Robotics: Science and Systems*, 2013.
- [13] K. Charalampous, I. Kostavelis, and A. Gasteratos, “Robot navigation in large-scale social maps: An action recognition approach,” *Expert Systems with Applications*, vol. 66, pp. 261–273, 2016.
- [14] H. Choset and K. Nagatani, “Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization,” *IEEE Transactions on Robotics and Automation*, vol. 17, no. 2, pp. 125–137, 2001.
- [15] Y. Wei, E. Brunskill, T. Kollar, and N. Roy, “Where to go: Interpreting natural directions using global inference,” in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 3761–3767.
- [16] O. Arslan and D. E. Koditschek, “Sensor-based reactive navigation in unknown convex sphere worlds,” *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 196–223, 2019.
- [17] F. Bonin-Font, A. Ortiz, and G. Oliver, “Visual navigation for mobile robots: A survey,” *Journal of intelligent and robotic systems*, vol. 53, no. 3, p. 263, 2008.
- [18] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.
- [19] J. Giesen, B. Miklos, M. Pauly, and C. Wormser, “The scale axis transform,” in *Proceedings of the twenty-fifth annual symposium on Computational geometry*. ACM, 2009, pp. 106–115.
- [20] H. I. Choi, S. W. Choi, and H. P. Moon, “Mathematical theory of medial axis transform,” *pacific journal of mathematics*, vol. 181, no. 1, pp. 57–88, 1997.
- [21] H. Blum, “A transformation for extracting new descriptors of form,” *Models for the perception of speech and visual form*, pp. 362–380, 1967.
- [22] T. K. Dey and W. Zhao, “Approximate medial axis as a voronoi subcomplex,” in *Proceedings of the seventh ACM symposium on Solid modeling and applications*. ACM, 2002, pp. 356–366.
- [23] S. R. Lindemann and S. M. LaValle, “Simple and efficient algorithms for computing smooth, collision-free feedback laws over given cell decompositions,” *The International Journal of Robotics Research*, vol. 28, no. 5, pp. 600–621, 2009.

⁵Code, data and additional information is available at cs.cornell.edu/~cliang