

# Navigation with Learned Spatial Affordances

Susan L. Epstein (susan.epstein@hunter.cuny.edu)

Anoop Aroor (aaroo@gc.cuny.edu)

Matthew Evanusa (matthew.evanusa@gmail.com)

Department of Computer Science, Hunter College and The Graduate Center of the City University of New York  
New York, NY 10065

Elizabeth I. Sklar (e.i.sklar@liverpool.ac.uk)

Simon Parsons (S.D.Parsons@liverpool.ac.uk)

Department of Computer Science, University of Liverpool, Liverpool, UK

## Abstract

This paper describes how a cognitive architecture builds a spatial model and navigates from it without a map. Each constructed model is a collage of spatial affordances that describes how the environment has been sensed and traversed. The system exploits the evolving model while it directs an agent to explore the environment. Effective models are learned quickly during travel. Moreover, when combined with simple heuristics, the learned spatial model supports effective navigation. In three simple environments, these learned models describe space in ways familiar to people, and often produce near-optimal travel times.

**Keywords:** spatial cognition, cognitive architecture, spatial affordances, learning, exploration

## Introduction

People somehow find their way through unfamiliar territory without a map, and with experience soon improve their ability to navigate there. This paper describes a system that simulates that skill development in an agent subject to noise and uncertainty. Our thesis is that learning to navigate is based on commonsense, qualitative reasoning, exploration, and affordances derived from perception. Our approach relies on devices well documented in people: a penchant for exploration and the representation and exploitation of perceptual experience through heuristic reasoning. Two principal results are reported here. First, a reusable, transferable, human-friendly depiction of an environment can be learned quickly. Second, such a model supports navigation in time close to that realized by an optimal path planner that contends with similar noise and uncertainty.

*Spatial cognition* learns, organizes, and applies knowledge about a spatial environment. People represent that knowledge internally as a *spatial mental model*, but the nature of that model remains an open question. Empirical evidence shows it is not an image-like metric map, even for a simple environment. Tversky's subjects displayed systematic errors incompatible with a map (Tversky, 1993). She suggested that their mental model was a gradually acquired collage of disparate knowledge types. In another study, subjects navigated no differently when their virtual environments were metrically or topologically possible or impossible (Zetsche, Galbraith, Wolter, & Schill, 2009).

Neurophysiologists have suggested that people use a sensorimotor or graph-like spatial mental model, where *spatial*

*abstractions* remove perceived but irrelevant details from spatial knowledge (Frommberger & Wolter, 2008). An *affordance* is a relation that enables one to perform an action (Gibson, 1977). Here, a *spatial affordance* is a spatial abstraction that supports navigation. This paper focuses on how a cognitive architecture combines a penchant for exploration (Speekenbrink & Konstantinidis, 2014) with the heuristic exploitation of learned spatial affordances.

Navigation, as studied here, is in a *world*, a dynamic, partially observable environment where maps are unreliable or unavailable, and landmarks may be absent, obscured, or obliterated. Examples include complex office buildings, warehouses, and search and rescue scenes. A traveler there may encounter unanticipated barriers or passageways.

*SemaFORR* is an application of the *FORR* cognitive architecture to robot navigation (Epstein, 1994). *FORR* was confirmed as cognitively plausible on human game players (Ratterman & Epstein, 1995), and has since learned successfully in a variety of application domains. *FORR* relies on multiple application-specific *rationales*, good reasons to select an action. This makes it a particularly suitable cognitive architecture for navigation, given that people also rely on multiple wayfinding strategies to select routes (Takemiya & Ishikawa, 2013; Tenbrink, Bergmann, & Konieczny, 2011). Moreover, *SemaFORR*'s rationales exploit research on the ways people perceive, envision, describe, and navigate through space (e.g., (Golledge, 1999)).

*SemaFORR* makes navigation decisions for a simple autonomous robot. The robot has no map; instead, it has only a *local view* of its immediate surroundings, a form of low-level sensorimotor perception. This view is provided by a *wall register*, a set of limited-range sensors that calculate

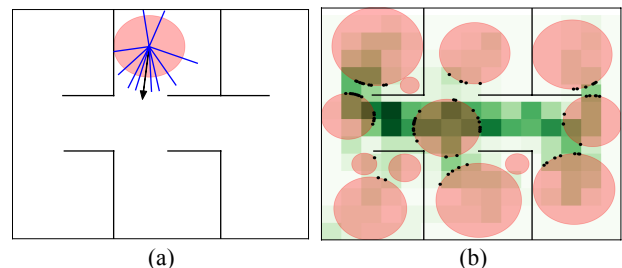


Figure 1: How *SemaFORR* perceives space and what it learns. (a) From its heading (arrow), the robot has only a local view. The inferred region is shown as a circle. (b) A learned spatial model for world A. Dots are regions' exits.

the robot’s distance to the nearest wall in 10 directions, as in Figure 1(a).

Through heuristic analysis of both the robot’s perception and its travel in a world, SemaFORR quickly learns a *spatial model* as a set of spatial affordances. Figure 1(b) superimposes an example of a learned spatial model on the true map for a simple office space (*world A*). The circles and squares represent regions and conveyors, respectively. (Both are explained in the next section.) The model clearly captures world A’s rooms and hallway. Although this learned collage is only approximately correct, we show here that it supports effective navigation.

SemaFORR is ultimately intended to make decisions for the robots of *HRTeam* (Human-Robot Team) (Sklar et al., 2011). An HRTeam robot is autonomous, inexpensive, and has simple perceptual devices. It is challenged both by actuator noise (imperfectly executed intended actions) and uncertainty (e.g., in its perceived location or from friction).

Because a SemaFORR robot is also intended to collaborate with a human team member, properties of our approach become particularly important. SemaFORR’s decision structure allows a robot to explain the reasons for its actions. Because those reasons are readily understandable by people, human-robot collaboration can be more natural for the person. In addition, a cognitively plausible mental model can be shared with the person at a level of abstraction that is both meaningful and parsimonious.

The next section of this paper describes the navigation task, FORR, and SemaFORR. Subsequent sections include the experimental design and results for several navigators constructed within SemaFORR. The paper concludes with related research and a discussion that includes current work.

## Navigation, FORR, and SemaFORR

For SemaFORR, a *task* requires the robot to *visit* (come within  $\epsilon$  of) a target. A *location* in a two-dimensional world is a real-valued pair  $(x,y)$  that denotes a point in a coordinate plane. At any instant, the robot’s *position* is its location and its *heading* (allocentric forward direction). Given the target’s location and its own location (from overhead cameras), the robot can compute the Euclidean distance between them.

Classical robot navigation either assumes a map or has the robot navigate to construct one. In such a map, the A\* algorithm can find an optimal (i.e., shortest) path between any two locations (Hart, Nilsson, & Raphael, 1968). For A\*, a continuous map is *discretized*, that is, a coordinate grid is superimposed on the environment and each cell is treated as a node in a graph. A\* then finds a *plan*, the shortest path from the robot’s start cell to the target’s cell. A\* is ill-suited to unknown territory, however, because it requires a complete and correct map. Many variations on A\* address dynamic or uncertain environments, but when a robot’s plan fails, the robot still must repair it or replan. This paper explores what a robot can achieve without planning.

SemaFORR’s robot senses only at a *decision point*, its location when it selects its next action. In a FORR-based system, action selection is the product of a *decision cycle*. The

input to a decision cycle is the current state, a set of possible actions, and world knowledge. In SemaFORR, the current state includes the wall register, the robot’s list of targets to visit, its current position, and a history of its decision points on the way to its current target. For cognitive efficiency, rather than generate possible actions in a continuous space, SemaFORR has a discrete repertoire of qualitative actions: 5 forward linear moves of various lengths (henceforward, simply *moves*), 10 clockwise or counterclockwise rotations of various degrees (*turns*), and a *pause* (do nothing).

SemaFORR is implemented with a simulator that replicates the errors observed in our laboratory on a Surveyor SRV-1 Blackfin, a small robot platform with a webcam and 802.11g wireless. Its larger moves and turns incur larger actuator discrepancies. SemaFORR’s world knowledge is its spatial model, represented as descriptives, described next.

## Descriptives capture affordances

In FORR, a *descriptive* is a data item whose value is computed on demand, with functions that determine how and when to update it. The current values of all descriptives are computed as input at the beginning of each decision cycle. Spatial affordances are represented as descriptives that evolve as the robot travels to new targets.

When the robot reaches a target, SemaFORR reviews its *true path*, the sequence of decision points that brought it there and the wall register at each of them. SemaFORR then revises the true path to reduce expended cognitive and physical effort. (This is similar to people’s use of return paths in (Hamburger, Dienelt, Strickrodt, & Röser, 2013), but with decision points rather than landmarks or viewpoints.) SemaFORR uses the wall register at each decision point to identify a better (i.e., more direct) choice. In the resultant *corrected path*, edges represent better moves that were possible actions, as in Figure 2(a).

SemaFORR’s learned spatial model summarizes its perceptual and travel experience with three kinds of descriptives: conveyors, regions, and exits. A *conveyor grid* covers the world with cells about 1.5 times the size of the robot’s footprint. It tallies how often all corrected paths intersect each cell. A *conveyor* is a cell with a high count; it represents an area through which many successful paths have traveled, as in Figure 2(b). A *region* approximates a confined, connected, open space (e.g., a room). As in Figure

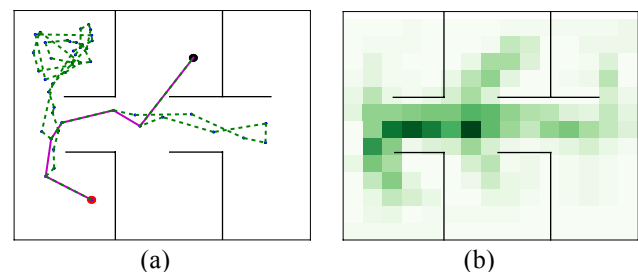


Figure 2: In world A, (a) a true path (dashed) to a target in the upper center, and its corrected (solid line) version based on local perception. (b) Conveyors after travel to 40 targets. Corrected paths passed through darker cells more often.

1(a), a region is a circle centered on the robot’s location, with radius equal to the shortest wall-register distance. (Regions are reminiscent of areas in online mapping (Thrun et al., 1998), but do not require that the robot map all walls first.) Larger regions subsume smaller ones, and regions do not overlap. An *exit* from a region, shown here as a dot on its edge, is formed wherever a true path intersects its perimeter. To make a decision, SemaFORR applies commonsense and its descriptives through its Advisors.

### Advisors capture high-level reasoning

In FORR, an *Advisor* is a boundedly rational (i.e., resource-limited) procedure that represents a rationale. Given the current state, world knowledge, and a possible action, an Advisor produces its opinion on the action as a *comment*. The *strength* of a comment reflects the degree to which the Advisor’s rationale supports or opposes the action. SemaFORR has 22 Advisors in all; Table 1 lists their rationales.

Most rationales produce one Advisor for moves and another for turns. A turn Advisor looks ahead to how it could use the same rationale after a turn. For example, GREEDY’s comments on moves have strengths proportional to how close the moves are expected to bring the robot to the target. GREEDY’s comments on turns have strengths based on how close the robot could come to the target if it were to turn that way and then move in the resultant direction. A turn decision is not a plan - it recommends a turn and anticipates a subsequent move, but does not commit to one.

SemaFORR has three rationales that exploit its learned spatial affordances. CONVEY supports moves to high-count conveyors, with preference for those further from the robot. (When high-count conveyors are near one another, CONVEY promotes travel through them rather than to them.) Exits support loose connectivity among the regions, as follows. If the robot is in region *R*, a *leaf region* is one with exits only to *R*. (With perfect knowledge, a leaf region would be a dead-end.) If the target is in region *T*, UNLIKELY opposes actions into a leaf region other than *T*, and EXIT supports actions toward exits from *R* that do not go to a leaf other than *T*, in the spirit of (Björnsson & Halldórsson, 2006).

Table 1. SemaFORR’s Advisor rationales. \* = rationale uses spatial affordances. † = rationale applies only to turns.

Name	Rationale
<i>Tier 1 in the order Advisors are considered</i>	
VICTORY	Go to a target within range
AVOIDWALLS	Do not go within $\epsilon$ of a wall
NOTOPPOSITE†	Do not oscillate in place
<i>Tier 3 heuristic Advisors vote to choose an action</i>	
*CONVEY	Go to frequent, distant conveyors
*EXIT	Leave a region via an exit
*UNLIKELY	Do not enter a target-free leaf region
BIGSTEP	Make a large move or turn toward one
CLOSEIN	When nearby the target, go closer
ELBOWROOM	Go where there is room to move
EXPLORER	Go to unfamiliar locations
GOAROUND†	Turn to avoid obstacles directly before you
GREEDY	Go closer to the target

Two SemaFORR rationales advocate exploration as a way to reduce uncertainty, a requisite human behavior in noisy, dynamic domains (Speekenbrink & Konstantinidis, 2014). EXPLORER encourages movement to novel (or rarely visited) locations, those that minimize the total Euclidean distance to previous decision points in the current task only. NOTOPPOSITE prevents oscillation in place by vetoing turns that change the previous rotation direction after a pause.

GREEDY and the remaining rationales rely only on commonsense and local perception. When there is no intervening wall, VICTORY supports a move directly to the target, or a turn that aligns the robot’s heading with the target. When the robot is near the target, CLOSEIN supports actions with comment strengths based both on distance to the target and the heading correction necessary to reach it. AVOIDWALLS opposes actions that would bring the robot too close to a wall, and thereby risk collision due to noisy actuators. BIGSTEP supports the largest possible action in each direction, with comment strengths proportional to that action’s size. Because a broader expanse offers more alternatives and allows larger movements, ELBOWROOM prefers actions that maintain a reasonable distance from any wall. Finally, when the robot is facing a wall, GOAROUND veers away from it, and prefers larger turns more strongly when a wall is closer. Disagreements among Advisors are anticipated, and resolved during a decision cycle.

### Hierarchical decision making

To reach a target, SemaFORR repeatedly selects one action at a time with the decision cycle shown in Figure 3. SemaFORR alternately chooses a move (or pause) on one decision cycle and a turn on the next decision cycle. Pauses allow extended turns in one direction. If SemaFORR chooses an action other than pause, it sends a command that drives the robot’s motors for some period of time. This actuation either turns the robot or propels it forward, subject to the simulated actuator error described above.

FORR partitions Advisors into three tiers, which its decision cycle treats hierarchically. Advisors assumed to be correct are placed in *tier 1*. All other Advisors, in *tier 3*, are heuristics. (Tier 2 is not used here; it includes planners, and

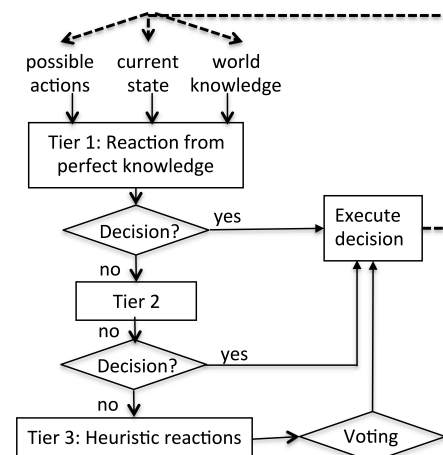


Figure 3: The FORR decision cycle.

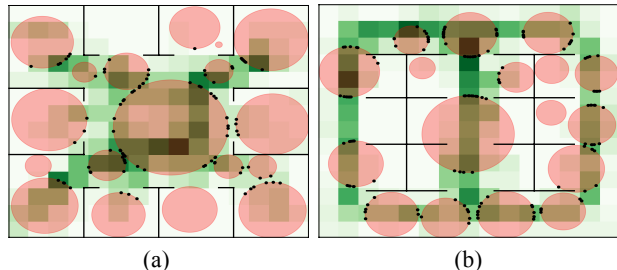


Figure 4: After navigation to 40 targets, a learned spatial model superimposed on the true map for (a) world B and (b) world C. For a model of world A, see Figure 1(b).

is the focus of current work.) In FORR, a comment from a tier-1 Advisor either forces an action or vetoes one. If any tier-1 Advisor selects an action, it becomes the decision. Otherwise, tier-1 Advisors may forbid some actions, and FORR invokes tier 3 on the actions that remain.

As a FORR-based system, SemaFORR consults its tier-1 Advisors first, in the order shown in Table 1. For example, assume that the robot in Figure 1(a) is about to select a turn. If its target is not in range, VICTORY does not comment, and AVOIDWALLS vetoes those turns to the right that would bring it too close to the wall. Then, if the last turn was to the right and the last move was a pause, NOTOPPOSITE vetoes all left turns, and SemaFORR then invokes tier 3 on the remaining turns. Because no tier-1 Advisor ever vetoes pause, there will always be at least one remaining action. If pause is selected or if only pause remains, the robot does nothing until the next decision cycle.

Unlike tier 1, all tier-3 Advisors are consulted together. In tier 3, *voting* sums the comment strengths for each action, and the action with maximum total strength is the decision. (Ties are broken at random.) For example, a long (BIGSTEP) move that gets close to the target (GREEDY) and goes where the robot has not traveled in the current task (EXPLORER) is likely to have considerable support. Those Advisors also provide a human-friendly explanation for the decision.

## Experimental design

We tested multiple navigators. *SemaFORR-A\** is our gold standard. From the map, it uses A\* to plan an optimal path as a sequence of *waypoints* from the robot’s initial location to its target. SemaFORR-A\* avoids walls and selects the action that brings the robot closest to its next waypoint in the plan. To limit actuator error, SemaFORR-A\* always choos-

es the smallest moves and turns. When a waypoint is obstructed, or when noisy actuators drive the robot too far from its next waypoint, SemaFORR-A\* replans.

SemaFORR uses all the Advisors in Table 1. To evaluate the impact of its components, we also tested ablated navigators with all the tier-1 Advisors but only subsets of the tier-3 Advisors. *SemaFORR-B*, for basic, uses only commonsense and perception: BIGSTEP, CLOSEIN, ELBOWROOM, GOAROUND, and GREEDY. *SemaFORR-E*, for explore, is SemaFORR-B plus EXPLORER. *SemaFORR-C* and *SemaFORR-R* add to SemaFORR-E only the Advisors for conveyors or regions, respectively.

Because SemaFORR is expected to improve its performance over a sequence of tasks in unfamiliar, unmarked territory, it should do *lifelong* (i.e., cumulative) learning. We tested each navigator in three worlds (A, B, and C) shown in Figures 1(b) and 4. A *setting* for a world is the robot’s starting location (here, always in the lower left) and a randomly ordered list of 40 randomly generated targets to visit. In a *trial*, the robot begins a setting in its starting location and then attempts to travel from one target to the next, in order. There is a 250-decision limit to reach any one target. If the robot *fails* (does not reach a target), it begins travel toward the next target from its current position. Performance is averaged over 5 trials in each of 5 settings, a total of 25 trials (1000 targets) per world for each navigator. Testing for the data reported here was performed in simulation.

In each world, each navigator is evaluated on its trial time, the distance it travels, and the frequency with which it reaches its target within 250 decisions. Time includes time to sense (the wall register), to decide (consult the Advisors), to act (send and execute the command), and to learn (calculate the affordances). In the following, cited results are statistically significant at  $p < 0.05$  unless otherwise noted.

## Results

Results appear in Table 2. SemaFORR-B, without exploration or spatial affordances, is surprisingly effective in world B; it finishes within 12% of the optimal travel time. Nonetheless, it fails on nearly a third of its targets in world C. When encouraged to explore (SemaFORR-E), however, the likelihood of success in worlds A and C improves considerably. Compared to SemaFORR-E, conveyors (SemaFORR-C) reduce the time and maintain the success rate in world C. Recall that SemaFORR-A\* is limited to only the smallest moves and turns. SemaFORR, however, takes larger steps

Table 2: Mean time in seconds, distance traveled, and the percentage of targets reached in 250 decision cycles. Only SemaFORR-A\* uses a map. All other navigators use some combination of commonsense, spatial affordances, and exploration. Time includes time to sense, decide, move, and learn, where applicable.

Navigator	World A			World B			World C		
	Time	Distance	Success	Time	Distance	Success	Time	Distance	Success
SemaFORR-A*	1035.89	400.06	100.00%	884.58	335.14	100.00%	1119.93	437.87	100.00%
SemaFORR-B	1947.64	927.82	90.30%	991.97	501.14	98.40%	2999.65	880.08	71.40%
SemaFORR-E	1375.11	942.68	98.90%	1022.78	627.87	98.90%	1476.82	917.87	99.50%
SemaFORR-C	1243.31	885.04	99.60%	937.56	646.51	99.30%	1350.26	896.61	99.80%
SemaFORR-R	1330.93	920.29	100.00%	979.24	612.62	98.90%	1492.19	915.14	99.30%
SemaFORR	1279.04	915.89	99.50%	1010.99	686.24	99.30%	1458.90	967.27	99.40%

that may incur larger actuator errors, and thus travels about twice as far to reach the same targets.

Although how hard it is to travel in these worlds is clearly correlated with learning time, both SemaFORR-A\* and SemaFORR spend most of their time in travel. SemaFORR devotes 17%–18% of its time to decisions; SemaFORR-A\* devotes about 19%, including planning; Moreover, SemaFORR’s learning is relatively fast; it spends 0.78% of its time learning in world A, 0.77% in B, and 0.89% in C.

From local perception, SemaFORR learns the global spatial models shown in Figures 1(b) and 4. Overall, after 40 targets these models varied little across settings and trials. Because wall register values depend upon the robot’s heading, they are necessarily approximations (e.g., the region in the center of world C crosses undetected walls and an upper room in world B is not captured). Emphases (e.g., the upper left conveyors in world A) are artifacts of the setting that generated the model. Nonetheless, regions capture the rooms in world A, and conveyors learn its hallway. In world B, SemaFORR learned a diagonal conveyor “highway” along with regions that captured every room it entered. In world C, conveyors learned the center aisle and the periphery, with regions chained together by their exits.

### Related Work

An early application of FORR to navigation (Epstein, 1998) was restricted to a grid world, where the robot occupied an entire cell. Its sensors had no range limit, its actuators were perfect, and it moved only orthogonally. Because its learning was not based on what is now known about human spatial perception, that system did best in grids with randomly generated obstructions or centralized open space. Built spaces like those here proved considerably more difficult.

SemaFORR draws from both empirical and theoretical research on spatial mental models and navigation. It embodies this knowledge in how it perceives its environment, in what it learns, and in the multiple ways it integrates that information with high-level reasoning. SemaFORR is similar in spirit to the Spatial Semantic Hierarchy (Kuipers, 2000). SemaFORR also considers moves and turns separately, and senses at the lowest level (the wall register) to build more complex representations (corrected paths and regions, which in turn support conveyors). Rather than culminate in places and paths with a single control rule, however, SemaFORR’s multiple rationales use spatial affordances: empty spaces and ways to move through and among them.

SemaFORR’s sensorimotor experience uses a simple view (its wall register), similar to human reference frames (Meilinger, 2008). The construction of a corrected path from a true one and the wall registers recorded along it are a form of incremental model development that relies on human memories of visited locations (Battles & Fu, 2014). Two of SemaFORR’s spatial affordances are well documented in people. Regions are often noted as fundamental to wayfinding (e.g., (Hölscher, Tenbrink, & Wiener, 2011; Reineking, Kohlhagen, & Zetsche, 2008)), and conveyors are similar to activation spread for navigation (Meilinger, 2008). Im-

proved performance with spatial affordances, and with more of them, confirms much empirical work (e.g., (Battles & Fu, 2014; Tenbrink et al., 2011)).

Like SMX (Zetsche, Gerkenmeysner, Schmid, & Schill, 2012), SemaFORR considers the actions it can perform and has only a partial view of its environment. SMX’s affordances are its landmarks; SemaFORR’s are its descriptives. Moreover, SemaFORR learns its affordances and how to reason with them. Advisors that enter or exit from regions, without a plan for what to do next, emulate empirical observations on people (Battles & Fu, 2014). Finally, the GREEDY Advisors represent subjects who navigated primarily by direction when they chose routes as they navigated (Hölscher et al., 2011).

### Discussion

Worlds A, B, and C simulate office space, a rotunda, and a warehouse, “built” spaces familiar to people and constructed by them. For all our navigators, world B is the easiest and C the most difficult. Commonsense reasoning without spatial affordances suffices in world B. SemaFORR’s time there is close to optimal, and the model it provides is human-friendly. In world C, however, exploration was essential to reach the targets. Conveyors further improved time in both A ( $p < 0.08$ ) and C ( $p < 0.05$ ).

With experience, SemaFORR reaches its targets faster. By design, however, some targets are more difficult to reach. To demonstrate online learning, Figure 5 normalizes task time by how difficult it is to reach each target, underestimated (because it excludes turns) as the distance to the target in an A\* plan. SemaFORR’s regression trend lines for the ratio of task time to the A\* distance ( $y$ -axis) decline across 40 targets ( $x$ -axis). In other words, on results controlled for task difficulty, SemaFORR improves its performance over 40 tasks in worlds A and C. In contrast, the trend lines for SemaFORR-E show no improvement.

We also experimented with learning during travel to a target, rather than learning once the robot arrived there. Use of the corrected path, however, significantly improves conveyors, and the corrected path is only available once the target is reached. (Data omitted.) This interplay between remembered perception and reasoning over it is, we believe, both novel and important.

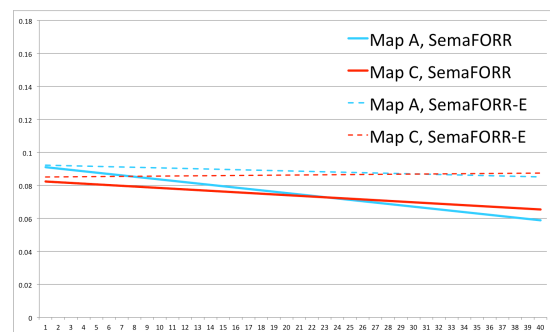


Figure 5: Regression trend lines show that SemaFORR’s normalized time ( $y$ -axis) improves across 40 tasks ( $x$ -axis), but SemaFORR-E’s (dashed) trend lines do not.

Current work includes planning and multiple robots. Simple reactive planners will be rationales for tier-2 Advisors that exploit the learned spatial model with situated cognition (Tenbrink et al., 2011). SemaFORR-A\* now controls autonomous Blackfins on our laboratory floor, each with its own copy of the software but a shared knowledge of their environment. A team of these robots addresses a setting simultaneously, with each assigned some of the 40 points. To adapt SemaFORR for multiple robots, we have designed several additional Advisors to avoid robot-robot collisions and discourage crowding.

When people navigate in unfamiliar territory, they process local percepts to construct representations that support their goal. SemaFORR produces rapid skill development, and translates perceptual signals into symbolic representations that become a long-term collage of semantic information. This spatial model supports effective navigation and can readily be conveyed to people.

### Acknowledgments

This work was supported by the National Science Foundation under #IIS-1117000 and #IIS-1116843, and by a CUNY Collaborative Incentive Research Grant, #1642. We thank A. Tuna Ozgelen for the three world maps.

### References

- Battles, A., & Fu, W.-T. (2014). *Navigating Indoor with Maps: Representations and Processes*. Paper presented at the 36th Annual Meeting of the Cognitive Science Society, Quebec City.
- Björnsson, Y., & Halldórsson, K. (2006). *Improved Heuristics for Optimal Path-finding on Game Maps*. Paper presented at the AIIDE, Marina del Rey.
- Epstein, S. L. (1994). For the Right Reasons: The FORR Architecture for Learning in a Skill Domain. *Cognitive Science*, 18(3), 479-511.
- Epstein, S. L. (1998). Pragmatic Navigation: Reactivity, Heuristics, and Search. *Artificial Intelligence*, 100(1-2), 275-322.
- Frommberger, L., & Wolter, D. (2008). Spatial Abstraction: Aspectualization, Coarsening, and Conceptual Classification 311-327. In C. Freksa, N. S. Newcombe, P. Gardenfors, & S. Wolf (Eds.), *Spatial Cognition VI* (pp. 311-327). Heidelberg: Springer.
- Gibson, J. J. (1977). The theory of affordances. In R. Shaw & J. Bransford (Eds.), *Perceiving, Acting, and Knowing: Toward an Ecological Psychology* (pp. 67-82). Mahwah: Lawrence Erlbaum.
- Golledge, R., G. (1999). Human Wayfinding and Cognitive Maps. In R. Golledge, G. (Ed.), *Wayfinding Behavior* (pp. 5-45). Baltimore: Hopkins University Press.
- Hamburger, K., Dienelt, L. E., Strickrodt, M., & Röser, F. (2013). *Spatial cognition: the return path*. Paper presented at the 35th Annual Conference of the Cognitive Science Society, Austin.
- Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100-107.
- Hölscher, C., Tenbrink, T., & Wiener, J. M. (2011). Would you follow your own route description? Cognitive strategies in urban route planning. *Cognition*, 121, 228-247.
- Kuipers, B. (2000). The Spatial Semantic Hierarchy. *Artificial Intelligence*, 119(1-2), 191-233.
- Meilinger, T. (2008). *The Network of Reference Frames Theory: A Synthesis of Graphs and Cognitive Maps*. Paper presented at Spatial Cognition VI, Freiburg.
- Ratterman, M. J., & Epstein, S. L. (1995). *Skilled like a Person: A Comparison of Human and Computer Game Playing*. Paper presented at the Seventeenth Annual Conference of the Cognitive Science Society, Pittsburgh.
- Reineking, T., Kohlhagen, C., & Zetsche, C. (2008). *Efficient Wayfinding in Hierarchically Regionalized Spatial Environments*. Paper presented at Spatial Cognition VI, Freiburg.
- Sklar, E. I., Ozgelen, A. T., Munoz, J. P., Gonzalea, J., Manashirov, M., Epstein, S. L., & Parsons, S. (2011). *Designing the HRTeam Framework: Lessons Learned from a Rough-and-Ready Human/Multi-Robot Team*. Paper presented at the Autonomous Robots and Multirobot Systems workshop, Taipei.
- Speekenbrink, M., & Konstantinidis, E. (2014). *Uncertainty and exploration in a restless bandit task*. Paper presented at the 36th Annual Conference of the Cognitive Science Society, Austin.
- Takemiya, M., & Ishikawa, T. (2013). Strategy-Based Dynamic Real-Time Route Prediction. In T. Tenbrink, J. Stell, A. Galton, & Z. Wood (Eds.), *COSIT 2013* (pp. 149-168). Heidelberg: Springer.
- Tenbrink, T., Bergmann, E., & Konieczny, L. (2011). *Wayfinding and description strategies in an unfamiliar complex building*. Paper presented at the 33rd Annual Conference of the Cognitive Science Society.
- Thrun, S., Bücken, A., Burgard, W., Fox, D., Fröhlhngaus, T., Hennig, D.,..., Schmidt, T. (1998). Map Learning and High-Speed Navigation in RHINO. In D. Kortenkamp, R. P. Bonasso, & R. Murphy (Eds.), *AI-based Mobile Robots: Case Studies of Successful Robot Systems* (pp. 21-52): MIT Press.
- Tversky, B. (1993). Cognitive Maps, Cognitive Collages, and Spatial Mental Models. In A. U. Frank & I. Campari (Eds.), *Spatial information theory: A theoretical basis for GIS* (Lecture Notes in Computer Science 716, pp. 14-24). Berlin: Springer-Verlag.
- Zetsche, C., Galbraith, C., Wolter, J., & Schill, K. (2009). Representation of Space: Image-like or Sensorimotor? *Spatial Vision*, 22(5), 409-424.
- Zetsche, C., Gerkenmeyser, T., Schmid, F., & Schill, K. (2012). *Sensorimotor Representation of Space: Application in Autonomous Systems and in a Wayfinding Assistant for Alzheimer's Disease*. Paper presented at the 11th IEEE/ACIS International Conference on Computer and Information Science.