

# A Cognitive model of spatial path-planning

David Reitter · Christian Lebiere

Published online: 22 September 2010

© Springer Science+Business Media, LLC 2010

**Abstract** Planning a path to a destination, given a number of options and obstacles, is a common task. We suggest a two-component cognitive model that combines retrieval of knowledge about the environment with search guided by visual perception. In the first component, subsymbolic information, acquired during navigation, aids in the retrieval of declarative information representing possible paths to take. In the second component, visual information directs the search, which in turn creates knowledge for the first component. The model is implemented using the ACT-R cognitive architecture and makes realistic assumptions about memory access and shifts in visual attention. We present simulation results for memory-based high-level navigation in grid and tree structures, and visual navigation in mazes, varying relevant cognitive (retrieval noise and visual finsts) and environmental (maze and path size) parameters. The visual component is evaluated with data from a multi-robot control experiment, where subjects planned paths for robots to explore a building. We describe a method to compare trajectories without referring to aligned points in the itinerary. The evaluation shows that the model provides a good fit, but also that planning strategies may vary with task loads.

**Keywords** Cognitive Modeling, Navigation, Path-Planning, Vision, Search, Cognitive Imagery, ACT-R, Multi-Robot Control

## 1 Introduction

Planning how to get to an intended location has long been an essential challenge for many species. From the simple task of reaching fruits in a tree or gaining access

---

D. Reitter, C. Lebiere

Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213, USA

E-mail: reitter@cmu.edu

C. Lebiere

E-mail: cl@cmu.edu

to the window in an office filled with desks, to the complex problem of driving to a destination in an urban jungle, we need to employ cognitively efficient, yet practically effective strategies to reach our goals. Navigation, like general planning, is about making choices on the way to reaching a goal from an initial state. Many levels of cognitive mechanisms are relevant to this process, including perceptual scanning, knowledge-based search and memory-based decision-making.

The cognitive model of human path-planning proposed here addresses the question: how does a process model perform when given cognitively motivated bounds on memory retention, decision-making based on experience and visual attention? We propose a cognitively constrained model that addresses two core aspects of human path-planning. The first describes how paths to a goal location are bootstrapped with the help of the perceptual system. The second describes how previously found paths are stored and preferentially retrieved to plan other routes.

Among the guiding hypotheses in developing the visual component of the model is the idea that most local planning problems can be solved visually, with minimal involvement of higher cognition (memory and rule-based algorithms). The model proposes that humans use egocentric principles even when a partial map is viewed from a top-down vantage point in order to get to a given goal. A local search algorithm explores the affordances that are local to the current focus of attention, with easy-to-calculate heuristics optimizing globally to ensure the model makes progress towards a given goal.

Situations that provide a visual impression of the path (such as a map) enable grounding turning points along the path in visible locations (landmarks). More complex problems require the recognition of previously visited locations, and yet more complex planning problems require non-geometrical knowledge about reachable goals for many existing locations. Even then, subsymbolic information is used to estimate the effort involved in taking a specific route, and to retrieve a good intermediate step given start and end points. As in any rational model, such information is acquired in a way that optimizes the overall performance.

Once spatial knowledge is established, processes of memory-based decision-making determine navigation. We will adopt a deductive approach here, taking a basic means-end algorithm to plan a path and formulating it within the ACT-R theory of cognition (Anderson, 2007), which provides a framework of established limitations to storage and retrieval of knowledge. The subsymbolic mechanisms implemented in this framework are what distinguishes ACT-R models from symbolic algorithms of path-planning that reflect robotic competence, but not human performance. Those mechanisms can serve as a powerful heuristic to prune the search space, as can be seen in chess, where humans select only a small set of all possible moves for evaluation.

We will evaluate the model first through parameter exploration, showing that near-optimal performance can be reached where cognitive parameters are at levels validated against human performance for other cognitive models in the ACT-R framework. For instance, visual memory will be bounded, and declarative memory is affected by decay and noise at levels considered realistic. We then show how visual and memory-based strategies scale with the size of the problem, generating predictions. Finally, we investigate how the visual component can produce paths similar to those planned by human subjects in a robot-control experiment.

In the following sections, we will discuss other cognitive approaches to path-planning. We describe first the memory-based part of the model including a series of simulations, investigating the scaling properties of the model. We then introduce the perceptual planning component, which is used in a simulation of maze navigation and evaluated against experimental data from human operators in a multi-robot control simulation.

## 2 Recent Work

Models of spatial navigation have addressed path-planning as well as representation mechanisms. For instance knowledge of locations may be arranged in arrays of two or more dimensions (e.g., Kosslyn, 1980; Glasgow and Papadias, 1998), or multiple layers of spatial representations (Kosslyn, 1994). A representation supporting path-planning will not only need to store distances between locations, but also the affordances of connections between them: some routes may be unavailable or suboptimal. Regarding path-planning algorithms, Fum and del Missier (2000) present data showing how humans develop spatial plans in a 2D environment that contains various numbers of obstacles. Subjects' performance was measured by the time to find the path and the number of unnecessary steps taken compared to the optimal path to the goal (errors). The number of turns in the path was found to be a crucial predictor of planning time. In the visual component presented in this paper, visually guided local planning results in long, straight lines with few turns.

The integration of cognitive architectures with visual and spatial processing represents a third field of work. (Chandrasekaran, 2006; Dye, 2007) attempt to blend cognitive and perceptual factors in navigation and planning applications in their work on implementing diagrammatic reasoning in cognitive architectures. Lathrop and Laird (2007) extend the Soar architecture with a visual system that is able to render imaginary drawings in tasks that involve reasoning about the relationships of geometric objects. They demonstrate that these tasks can be carried out on a non-visual, largely symbolic level but that visual imagery, however, speeds up the process (in line with data). Within the context of the ACT-R architecture, much work has been done on the problem of spatial planning and representation, including issues of adaptivity in planning (Fu, 2003), encoding of spatio-temporal stimuli (Johnson et al., 2002), visualization capacity of spatial paths (Lyon et al., 2008), path-planning in virtual environments (Best and Lebiere, 2006), spatial perspective-taking for planning (Hiatt et al., 2004) and architectural modules for neurally plausible navigation in 3D spaces (Schunn and Harrison, 2001). In general, such approaches carry out route planning and other visual operations with visual representations held in memory (visual imagery). Here, we address the case of route planning by combining externally available information filtered through the perceptual system with abstract partial path knowledge held in long-term memory.

## 3 Model Overview

The model's objective is to plan a path to arrive at a goal point, given a current location and a partially observable environment, which constrains individual steps.

Any cognitively plausible model of path-planning must give rise to core behavior observed in humans. *Creativity*: Humans can find new paths even with limited visual information reflecting only the immediate surroundings. *Adaptivity*: Humans become accurate navigators once a mental model of the environment has been acquired. Usually, it is postulated that humans build a *topological map*. *Decay*: over time, the mental model degrades, causing more explorative behavior and navigation errors. Our model addresses these points using a combination of storage and recall from memory on the one hand, and visual search on the other. There are further human and non-human capabilities that we abstract away from: e.g., integrating first-person (3D) and top-down (2D map) perspectives or taking into account head position to correlate observed visual features with a mental map.

To determine the best route between a start and an end point, the model first attempts to retrieve a path from declarative memory. If none is found, visual search produces the next intermediate step. Visited locations and pathways traveled are stored in declarative memory, and reinforced upon repeated presentation. They can be retrieved by the memory-based component should the need arise: either, when no visual information is available, or, when the visual system reaches its limitations. In particular, when the need to identify past locations requires the use of memorized knowledge rather than visual finds, the integrated model will abandon the visual strategy. When presented with an unknown maze, search will be visual initially, progressing to memory-based navigation later. However, visual search can execute some of the path-planning that would otherwise require mental maps (or graphs); simulations with an artificial maze world demonstrate this.

The visual system has access to the part of the visual scene that it attends to at the time (cf., the visual representations in Glasgow and Papadias's (1998) model). Perceptual planning is used whenever the memory-based planning strategy cannot produce an itinerary for a local solution, for instance when the locations involved have not been visited before, or the local circumstances have changed (e.g., new obstacles are present), or previous experiences are not available. The visual component represents possible paths as largely straight lines from the point of attention to reachable (possibly intermediate) goals. Given a first-person perspective, paths will equal lines of sight; given a two-dimensional (2D) representation of the maze (as in our experiments), possible paths are detected as straight lines that are uninterrupted by walls. In the latter (2D) case, knowledge of previous decisions is strongly grounded in the visual world. Memory that would otherwise be declarative on the abstract level is externalized in the visual scene. The role of visual perception is to identify traversable shapes and select promising ones: the adopted heuristic is to choose the route that ends up as close to the goal as possible. Thus, perception is to convey a sense of distance from the end of straight lines to the goal.

The visual and memory-based planning strategies have complementary strengths and weaknesses that govern their suitability to various circumstances. Learning to navigate around a city, for instance, will be a memory task, while most navigation in a park unknown to the subject would be better suited to the visual approach. But the strategies also display fundamental similarities, as would be somewhat expected because they both leverage cortical areas that, while specialized to their functions (vision and memory), also share basic biological mechanisms and organizational principles. One similarity is an emphasis on a means-end approach of efficiently (i.e., in a single

cognitive step on the order of a few hundred milliseconds) obtaining a partial, approximate solution from their respective architectural modules, then subgoaling the rest of the solution. Another similarity lies in the principles by which those solutions are obtained, namely a mix of similarity-based generalization (e.g., between the end point of the partial solution and the target) and heuristic quality measures (lack of barriers in the visual model, activation strengths of memory traces in the memory model).

With our model we aim to put forward a stochastic but rational view of the path-planning problem, extending symbolic and computational algorithms (Wray et al., 2007). The model demonstrates how path-planning can be achieved despite the independently motivated and validated constraints of human performance provided by a cognitive architecture. Thus, it is compatible with theories of higher-level spatial planning, exploration and localization (Kelley, 2006). The specific cognitive architecture chosen here (Anderson, 2007) and the encoding of spatial locations in declarative memory by our model explain how humans ground those locations in observable landmarks (O’Keefe and Nadel, 1978) through cue-based memory retrieval. Concretely, the abstract information in topological maps is grounded in visual concepts (landmarks) by the perceptual portions of the architecture. Landmarks may serve as cues to retrieve location-based information during planning. The abstract representation of affordances alone would be insufficient to explain classic mental imagery data (Shepard and Metzler, 1971).

## 4 High-level Planning

This model component is based on learning and retrieving locations and paths between them; this way the component stores a topological map in declarative memory that makes the affordances of locations available to the control module. The representation of the topological map is distributed among traditional chunks linked to each other in hierarchical structures, and does not make any additional assumptions about specialized representations or processes.

The representation in this part of the model is symbolic and memory-based. Given the immediate surroundings and a goal, options for a next step can be determined. The abstract representation allows us to store locations, their local options (i.e., outgoing paths) and how useful these options are with respect to a given goal. The abstract situation is encoded as declarative knowledge, indicating the efficiency of the route to traverse from point *A* to point *B*. Such situations may be recalled with the help of cues: this could be a goal location *C*, but also a preceding traversal. This makes goal-directed planning possible and predicts that sequences of decisions rather than just individual steps are learned and combined during path-planning. The traversal of a maze according to such abstract representations amounts to traversing a graph structure.

### 4.1 Memory-based Strategy

The memory-based component primarily leverages the subsymbolic mechanisms of declarative memory in the ACT-R architecture. While subsymbolic mechanisms also

play a large role in procedural memory, there were a number of reasons for focusing initially on declarative subsymbolic processes:

- Declarative memory provides a more direct integration path with symbolic and perceptual information since ACT-R stores those sources of information initially (and perhaps largely) using this type of memory.
- Decision-making typically follows a path by which it starts with declarative processes that are then ultimately (if possible) compiled into procedural structures. As such, a declarative account is an enabling account to a subsequent procedural one.
- Subsymbolic procedural processes largely consist of a utility calculus determining the selection of production rules. While this mechanism bears a strong resemblance to reinforcement learning techniques that have been applied extensively to navigation and planning tasks (Millán and Torras, 1992), rules are too coarse to represent all but the most practiced experience. Thus, most fine-grained knowledge about path quality will be represented declaratively.
- Declarative subsymbolic mechanisms reflect more complex and discriminating statistical and semantic factors than procedural subsymbolic mechanisms that give them greater power in complex domains (cf., an ACT-R model of the game of Backgammon: Sanner et al., 2000).

Before describing the subsymbolic mechanisms, we will briefly sketch out the symbolic level of the model that leverages those mechanisms, more specifically the declarative representation of the problem and the production rules that manipulate them. In this model, declarative memory items, *chunks*, are of two basic types: location chunks that define states of the system, and path chunks that define transitions from one state to another. In addition, there is one basic goal chunk type representing the planning process that holds three pieces of information: the current state, the desired (goal) state and process information such as the current step of the process and ultimately an intermediate state determined by that level of planning. These goals are constructed during path-planning and enter declarative memory when completed. There, they constitute a record for purposes of backtracking as well as learning across planning episodes that would allow previous partial solutions to be reused. As such, one can view specific paths between states as a special case of past planning solutions. Similarly, the productions that act on those representations are equally straightforward:

1. The key production retrieves a path from memory. This is the key step that leverages declarative subsymbolic processes.
2. A subsequent production checks if the path starts at the current location. If so, it advances along the path and schedules a task to move from the endpoint of the chosen path to the final destination.
3. If the path does not start at the current location, another production schedules getting from the current location to the path's starting point as a subgoal, and changes the current goal to get from that point to the final destination for later resumption.
4. If a subgoal (such as the ones set by the previous productions) has been completed, then another production attempts to retrieve the next highest goal and resume it.
5. If no such higher goal exists, the process terminates in success.

6. If no path can be found from the current location that has not already been taken (to avoid looping), the process terminates in failure (or rather, resorts to the visual planning component).

## 4.2 Storage of Locations in Declarative Memory

To focus on determining the ability of subsymbolic mechanisms to find the right path, we avoided in this basic model the use of backtracking mechanisms that could assure successful planning through exhaustive consideration of all possible routes. However, such a process could be easily integrated to leverage the record of previous (sub)goals in declarative memory, and would be triggered at the last step of the process described above. The key step, therefore, is the retrieval from memory of the next path to be considered, which attempts to maximize the activation of the chunk retrieved. Memory activation is a sum of three terms (not counting a stochastic noise term to be discussed later), each of which aims to capture a separate statistical factor in the path selection process. It is defined by the following equation:

$$A_i = B_i + \sum_j W_j * S_{ji} + \sum_k MP * Sim(v_k, d_p)$$

The first term, *base-level activation*, captures frequency and recency, following the power laws of practice and forgetting, respectively, and can be interpreted as the log-odds that a given path will be the right one. In contrast to this context-free term, the second term, *spreading association* captures the log-likelihood that this path is the right one given the current context elements. Technically, each context element spreads activation to related chunks: in this model, the current and destination locations serve as cues in the retrieval of partial paths between those locations. In practice, the additive spreading activation mechanism embodies the naïve Bayes assumption of independence between context elements, so there is no guarantee that a path related to (not necessarily connected to) both locations would be a good one to get from one to the other. Hence there is a need for a more explicit semantic factor reflected in the third term, *partial matching*, which imposes a penalty on chunks proportional to their dissimilarity with the pattern specified in the retrieval request. In practice, since the location chunks to be retrieved are represented with their starting and destination locations, we aim for the starting location to be as similar as possible to the current location, and for the end location of the retrieved path to be as similar as possible to the final destination.

These factors added together effectively balance the requirements that paths selected in the retrieval process reflect the constraints favoring common paths (a powerful heuristic reflected in all human actions), those often taken from and to the specific locations, and those that make progress toward the final goal.

Those factors should ultimately be learned from experience using architectural learning mechanisms. In the absence of a dataset that would allow us to acquire such parameters dynamically, we followed the Rational Analysis process<sup>1</sup> (Anderson, 1991; Simon, 1991) and set them to reflect idealized values, at which the rational

---

<sup>1</sup> Rational Analysis assumes that cognitive mechanisms tend to be optimal given the environment and their computational limitations. It proposes that the task holds clues to understanding the mechanisms themselves, as the cognitive apparatus reflects constraints imposed by the task.

learner will arrive through long-term learning. Accordingly, the base-level activation of the chunk for path  $i$  is set to the log of the odds that  $i$  would be on the shortest path between two locations  $j$  and  $k$ , averaged over all possible starting and ending location  $j$  and  $k$ , respectively:

$$B_i = \log \frac{1}{n^2} \sum_{j,k}^n odds(i, j \rightarrow k)$$

Similarly, the strength of association from location  $j$  to path  $i$  is set to the log odds that  $i$  is part of the shortest path from  $j$  to or from another location  $k$  (paths are not necessarily symmetric), averaged over all other locations  $k$ :

$$S_{ji} = \log \frac{1}{n} \sum_{k=1}^n odds(i, j \rightarrow k)$$

Finally, the similarities between locations  $j$  and  $k$  used in matching the path starting and destination locations are set to reflect the semantics of the domain, specifically a negative exponential function of the distance of the shortest path between those locations:

$$Sim(j, k) = e^{-d(j \rightarrow k)} - 1$$

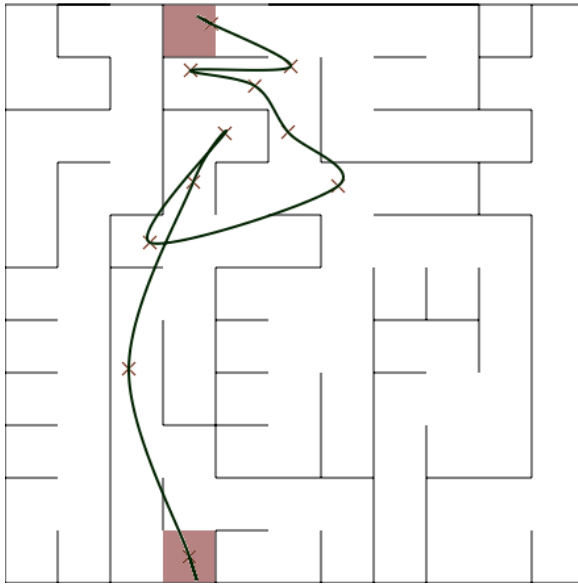
Unlike the first two factors, which reflect the statistics of previous problem-solving experience, the third factor most likely reflect the interaction with the domain itself such as resulting from the visual level. The use of these activation factors is similar to that of Lyon et al.'s (2008) model of 3D path-planning.

## 5 Perceptual Path-Finding

The perceptual portion of navigation refers to the externally available visual representation only and does not require declarative memory. The task being modeled initially is to find a route from given start and end locations on a two-dimensional representation of the topology; the model reflects visual attention and a strategy used to determine whether there is a way to get to the target. We present initial simulations on small mazes, but then evaluate the strategy using a realistic task that involves subjects finding itineraries for robots around a building.

The component itself uses a number of concepts commonly used in the ACT-R literature, including *visual finsts*, a constraint limiting the recognition of visited locations (Pylyshyn, 1989). Neither declarative memory nor the acquisition of procedures are involved; we do not specify a set of ACT-R-style production rules as they would not yield testable predictions beyond the algorithms discussed. The concept of underspecifying parts of the model that cannot be motivated empirically, and that do not contribute to the explanation of empirical variance, is realized in an underlying re-implementation of ACT-R called *ACT-UP* (Reitter and Lebiere, 2010), which allows more efficient prototyping and evaluation of models.

The visual navigation strategy is applicable whenever the goal is easy to reach. Such a goal could also be a subgoal present during planning with a topological map. Thus, the visual strategy ties in to the cognitive strategy to find local solutions and



**Fig. 1** The visual strategy can find relatively straight paths easily, with limited capability to backtrack. Start point at bottom. Crosses mark decision points. Lines smoothed for visualization.

guide exploratory behavior. It should be noted that visual navigation does not require the declarative, explicit storage of branching points. Any backtracking is visually guided and constrained by visual finsts. As a consequence, the visual strategy inherits the limitations of visual memory: primarily, there is only a small number of finsts available. Without a memory-based component storing branching points explicitly, the visual component can only identify 5 or so locations as previously visited (if we accept the default assumption of about 5 finsts). The model may, given a sufficiently complex navigation task, even get caught in a loop, visiting the same locations again and again. This is especially true when the visual component is used alone without the benefit of its memory-based counterpart.

When navigating perceptually, the model can plan incrementally and commit to the first moves early. For comparison, Fum and del Missier's (2000) model also proposes that subjects choose locally optimal paths (a hill-climbing strategy). Our visual strategy is similar in that the cognitive level also prefers to backtrack locally in order to avoid long-term memory needs.

### 5.1 Visual Strategy

The visual navigation strategy always chooses the best straight line segment originating at the current point of visual attention. This could be interpreted as a *line of sight* in the corresponding first-person perspective environment. Fig. 1 illustrates the algorithm on a simple maze. (We will use such mazes to explain the algorithm and, later, to evaluate aspects of the model.) The visual strategy comprises the following steps.

1. Direct visual attention to the start point.

2. Identify straight lines (uninterrupted by obstacles) in all directions from the current point of visual attention, up to a length *LineLen* (a free parameter). For some domains, lines have a non-negligible width (*LineWidth* parameter).
3. Choose the line whose end point is expected to help us to reach the goal according to a *distance* heuristic; try to avoid points that have been recently visited.
4. Move visual attention along the chosen line towards the end point. While doing so: If moving across a recently visited location (backtracking), identify *openings* to the left and right, that is, uninterrupted straight lines beginning at the current point of attention and extending orthogonally to the current track. If openings are found, abandon the movement to the end point and continue with 2.
5. Terminate if goal reached<sup>2</sup>.

*Distance heuristic:* This heuristic is task-specific: it is sufficient in the case of mazes to consider the direct geometric distance to the ultimate goal. When applying the model to the case of a building map, the heuristic also examines the average density of obstacles seen along the direct line between potential decision point and goal; a dense connection indicates the presence of walls and other obstacles and would be avoided. If the building map shows unexplored territory, the density of such areas is a domain parameter (*UnexploredDensity*).

*Recently visited locations:* While traveling to the next decision point, the model notes its location; the most recent locations are accessible in form of *visual finsts* (Pylyshyn, 1989). This mechanism allows the model to distinguish previously visited locations from novel portions of the maze. The number of visual finsts is limited, so that the model can only recognize a few recently visited locations, while other locations are considered novel. We expect this model to be compatible with more recent, graded views of visual salience (Byrne, 2006), even though the visual component alone does not require any subsymbolic representation beyond the raw image of a maze.

*Location:* In the maze domain, we consider a single cell as a location. In the building-map domain, we use cells of 0.5m width (the corridors in the buildings in our dataset are 2.95m wide).

*Openings:* The model detects an opening to the left or right of the track if the distance to the next obstruction is more than one standard deviation greater than the moving average.

*Timing:* Following Salvucci (2001), the time to shift attention and encode the new location is determined as a function of location frequency  $f_i$  and eccentricity  $e_i$  (distance in units of visual angle) and constants  $K$  and  $k$ :  $T_{enc} = K * [-\log f_i] * e^{ke_i}$ . In the context of the maze, where most locations are novel, we assume  $f_i = 0.5$ . The model does not take additional motor activity into account.

---

<sup>2</sup> For purposes of evaluation, we also terminate if number of steps exceeds  $w^{1.5}$ , where  $w$  is the width of the maze in squares to determine the maximum number of steps.

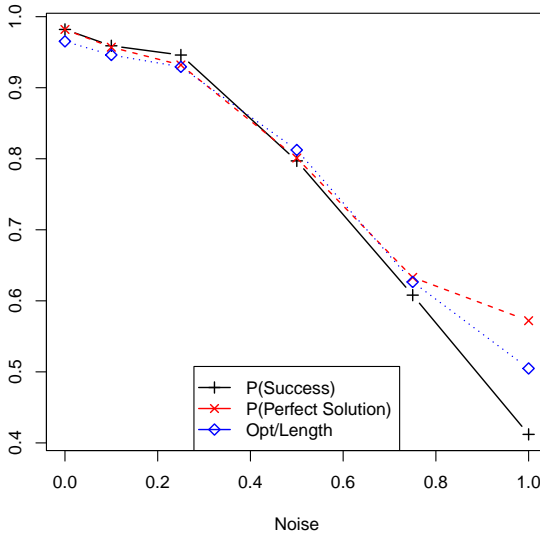
## 6 Evaluation

The model's two essential components, memory-based and perceptual, have been motivated and constrained differently. The memory component is based on the independently validated ACT-R theory, which describes abilities and limitations of human memory in great detail. An evaluation of the memory component should show whether it can reproduce desirable behavior within the constraints of the architecture, that is, with noise and learning parameters that have been found to be adequate in other studies. The perceptual component, however, is less constrained by the cognitive architecture, as the standard implementation of the visual module does not include many of the factors at issue here. As a consequence, we will present a series of evaluations. The first focuses on the performance of the memory component under realistic cognitive parameters and with a number of prototypical organizational layouts of topological maps. Next, we demonstrate the adequacy of the perceptual component using an exemplary maze world, before turning to an empirical study to evaluate the correspondence between model performance and human data. The evaluation avoids the pitfalls of an extrinsic, single-measure end result describing the performance of a complex system. It reflects the convergence of top-down (theory-driven, deductive) and bottom-up (data-driven, inductive) modeling approaches.

### 6.1 Memory Parameters and the Topological Map

Cognitive models are often evaluated by comparing their performance against human data at a specific complexity point for a particular task. While this approach has the advantage of precision and specificity, it is also susceptible to parameter manipulations, as every model includes parameters whose values can be set to achieve specific performance levels and thus "fit" the data. Viewing cognitive models as computational artifacts, computational theory does not evaluate the performance of an algorithm by concentrating on performance at a specific complexity point. Such performance would be affected by many factors unrelated to the fundamental nature of the algorithm. Instead, computational approaches study how performance scales with problem complexity. This approach has been applied previously to the evaluation of cognitive models (e.g., Lebiere and Wallach, 2001; Gluck and Pew, 2005) and will be the focus of our evaluation of the memory-based model component.

We tested the memory-based high-level planning component in two idealized but naturally scalable environments: *trees* and *grids*. A tree structure is meant to approximate the hierarchical organization of structures such as buildings, with heavily travelled central connectors (elevators, main hallways) and increasingly localized destinations (wings, rooms). A grid structure is meant to approximate the regular pattern of city streets. We ran our model component over structures of both types over a range of complexity, including trees of depths 3 and 4 (including root level) and branching factors of 2 and 3 (approximately the most levels of organization in hierarchical networks such as large buildings or road networks). Grids were sized 2x2 to 5x5 (which might seem small but provides many combinatorial possibilities and could be applied in a self-scaling manner). The only parameter that we manipulated in the model was the amount of activation noise. This was designed both to reflect the stochastic nature of human cognition (which may seem like a purely limiting factor but is actually a



**Fig. 2** Model performance degrades with increased noise levels w.r.t. three measures: success rate, rate of perfect solution and optimal-to-obtained-path length ratio.

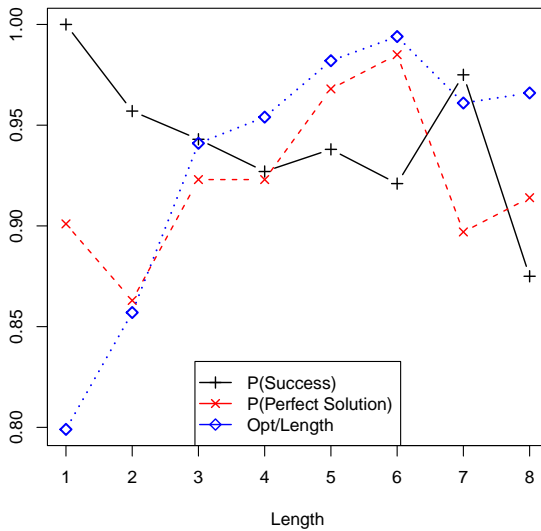
powerful feature to avoid both predictability and local minima in search, e.g. West et al., 2005) and temper the assumption that the activation calculus parameters were set to idealized values that learning mechanisms would not perfectly reach.

We show the performance of the model using three dependent measures:

- $P(\text{Success})$ : the proportion of successful runs (i.e., reaching the goal), which are far from assured given the constraint of never using the same path twice and the lack of backtracking
- $P(\text{Perfect Solution})$ : the probability of finding the shortest path assuming success
- $\text{Opt}/\text{Length}$ : the ratio of shortest path to actual path length (again assuming success). A value of 1 would indicate that the model has chosen the shortest possible path.

Fig. 2 displays the performance of the model as a function of activation noise in terms of these three measures, averaged over 8 different environment structures (four trees, four grids as above) and sizes and 10 repetitions each. While, as expected, all three performance measures decrease sharply (in remarkably correlated fashion) as a function of activation noise, a typical noise value of 0.25 used in many models of memory-based decision-making (e.g., Gonzalez and Lebiere, 2005) provide over 90% performance on all measures.

Fig. 3 displays the performance of the model as it scales to paths of increasing lengths. An interesting pattern arises for paths of length 2 to 6, displaying an inverse relationship between decreasing probability of success but increasing probability of finding a path of minimal length assuming success. This pattern is however not present for paths of length 7 and 8, which consist only (for artifactual reasons) of grid structures, thus it largely originates from the likelihood of getting lost in tree structures given the lack of backtracking. In general, the probability of success seems to



**Fig. 3** The model predicts lower success rates for longer solutions. When a solution is found, it is usually less than 20% longer than the optimal solution. Noise level at 0.25.

scale well (over this admittedly limited range) as a function of path length, especially considering the wide sampling of high noise values reflecting in this average.

## 6.2 Visual Planning: Scalability

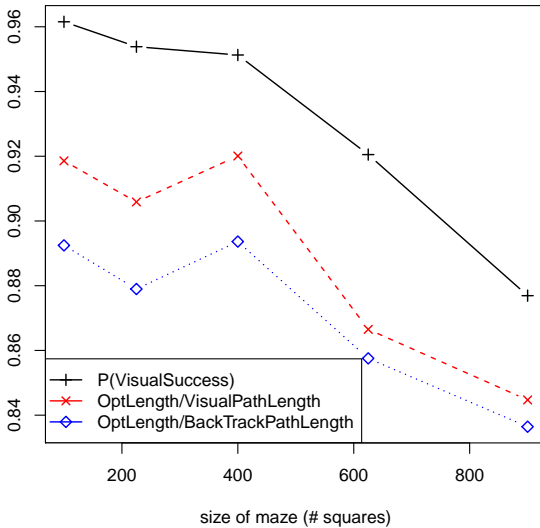
In line with our original motivation to model navigation tasks that are comparable to real-life navigation over short distances, we investigated path-planning using *mazes*. Our mazes are of size 10 by 10 squares or larger. They are relatively easy to solve: they do not generally contain many long dead-end routes, requiring the visual component to backtrack only occasionally. Other, more complex mazes, would define the task primarily as a memory problem: there, remembering previous branching points is the most important sub-task of solving algorithms (as long as other external solutions, such as marking branches visually, are not allowed).

Mazes were generated using a dynamic programming algorithm commonly known as *Eller's algorithm*. Start and end points were always located on opposite sides of the maze.

We ran the visual component using randomly generated mazes, varying the size of the maze (100–900 squares) and also the number of visual finsts (2–40). As for the memory-based model, we show the proportion of successful runs within the allotted maximum number of steps ( $P(\text{VisualSuccess})$ ), as well as the proportion of the minimal path length (from start to goal,  $\text{OptLength}$ ) and the length of the path suggested by the visual component ( $\text{VisualPathLength}$ ). Additionally, we show a comparative baseline, where we note the results obtained from a trivial backtracking model, expressed in proportion to  $\text{OptLength}$ . The baseline backtracking model moves square



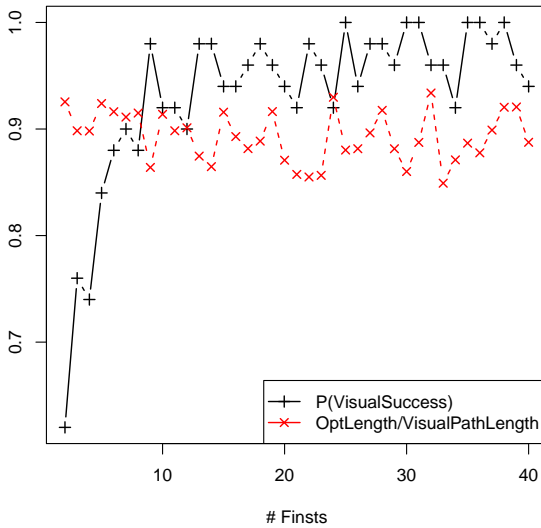
**Fig. 4** Predicted solution times increase linearly with the size of the maze.



**Fig. 5** The performance of the visual component drops in mazes bigger than about 20 by 20 squares. Data aggregated over number of finsts. Curves displayed are success rates, ratio of minimal vs. obtained path length, and, for comparison, ratio of optimal vs. backtracking-based path length. The backtracking baseline delivers consistently longer paths than the visual strategy.

by square and backtracks to previously visited choice-points, but prefers squares that are visually closer to the goal. It will always succeed.

The visual model component predicts a mostly linear increase in time-to-solution with size (side length) (Fig. 4). This is expected as the model generally optimizes decisions locally rather than searching the whole maze. The model predicts good performance with respect to the length of the solutions (Fig. 5); for mazes of any size, the solutions resemble the optimal solution better than a simple backtracking algorithm that explores the maze square by square. Performance drops in mazes larger than about 400 squares. This drop in performance also holds with finsts held at 10. Even though the number of finsts is critical for the model to succeed, 5–10 finsts are sufficient to solve the mazes (Fig. 6), and performance does not improve beyond that.



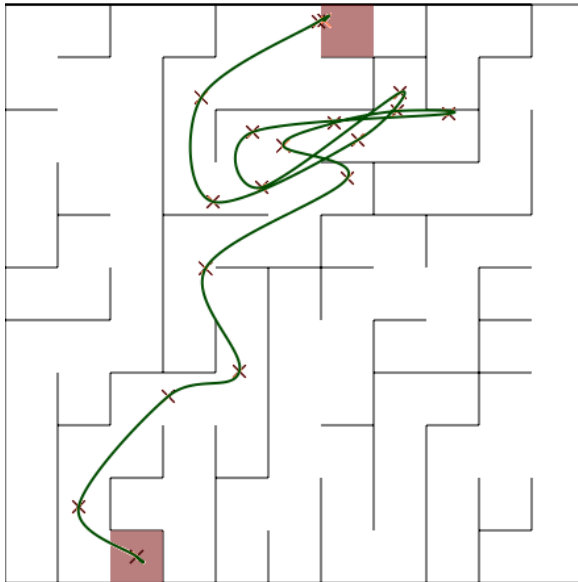
**Fig. 6** Visual component performance appears to stabilize after 5–10 visual finsts. Data aggregated over maze sizes.

### 6.2.1 Discussion

As is evident from the algorithm, solution times for mazes predicted by the visual navigation strategy will depend on the number of decision points, a phenomenon observed and also modeled by Fum and del Missier (2000). Unlike their model, the present model defines branching points as end points of straight lines, or points at which the visual component recognizes branching points (openings) explicitly. Still, decision points usually involve a change of direction, so Fum et al.’s turn points correspond well to decision points. Finding a new decision point takes time, but traversing to a decision point once it is chosen is fast. The temporal penalty resulting from the decision point results from visual activity (search for a new straight line) rather than from storing the location in memory.

At the same time, the model also predicts longer solution times for cases where backtracking is important. In particular, the visual component predicts long solution times (or fails to provide a solution) in cases where solution paths first lead the attentional focus away from the goal, while offering more direct paths that ultimately lead to a dead end (Fig. 7). For situations that do not need much backtracking, the visual strategy predicts efficient paths that are found quickly (Fig. 1). So, for many real-world situations that hardly compare to the pathological case of a maze with numerous dead-end paths, the visual strategy may offer a compelling explanation.

The visual strategy is, in this simulation, limited to what is available perceptually. The integration with declarative memory beyond what has been discussed could optimize local path-planning using prototypical knowledge of the surroundings. Consider again Fig. 7: the model chooses the wrong direction (East), backtracks West, and then attempts a similar direction again (North East) rather than jumping South West (leading it temporarily away from the goal). Once the large space adjacent to the goal is



**Fig. 7** The visual component fails where knowledge of the local surroundings would be helpful. See also Fig. 1.

identified as a useful location to navigate too, the model can leverage knowledge of the local surroundings to circumnavigate the Western wall. Recognizing a pattern of 3 by 3 squares would be enough to see that the wall to the West is only one square long, and that there is an opening further South. We expect this type of knowledge to be retrieved from declarative memory, once the visual pattern has been recognized. Translated to practical tasks, the model has to *know* about the affordances of a door (access to a bigger space behind it) or a chair (can be climbed over if necessary).

### 6.3 Visual planning: Empirical Evaluation

The visual portion of the path-planning model was evaluated using human data collected in a multi-robot control study (Wang et al., 2009). Robots were sent on a search and rescue mission inside a prototypical building with rooms connected by hallways. Obstacles included office furniture, victims and other robots. *USARsim* (Lewis et al., 2007) is a robot simulation platform; *Multi-robot Control System* is a multi-robot communications and control infrastructure with accompanying user interface developed for experiments in a multi-robot control (RoboCup) competition (Balakirsky et al., 2007) that was used to provide the experimental scenario that required human participants to remotely control a varying number of robots.

Fig. 8 shows a screenshot of the system. The operator selects the robot to be controlled from the colored thumbnails at the top right of the screen. Robots are tasked by assigning waypoints on a global north-up map (bottom right). They can also be controlled through more fine-grained teleoperation (top left, unused in the present dataset). Initially, the building is unknown, and only as visual information (through a simulated laser range finder) is acquired does the structure of the building become

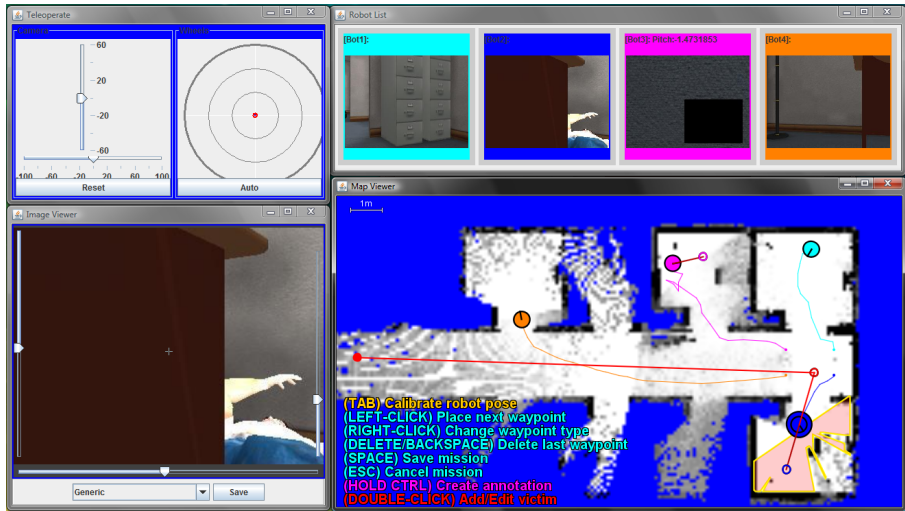


Fig. 8 The Multi-robot Control System graphical user interface.

visible. The previously explored parts of the building are then shown to the participant, making this task an example of a case where navigation would proceed visually rather than by recalling paths and locations from memory.

### 6.3.1 Dataset

45 participants from the University of Pittsburgh community took part in Wang et al.'s (2009) experiment for compensation. Subjects controlled 4, 8, and 12 robots in succession. An *exploration* condition in the experiment had subjects focus solely on the discovery of the building layout; these data (from 15 subjects) are used in the evaluation of the model. One subject's data were discarded, as the participant failed to understand the task. The exploration condition was chosen in the model evaluation because navigation and path-planning were the primary objective of these participants making their task comparable to the goal-directed planning of the model. Two other conditions had different groups of subjects either just search for victims (with robots moving autonomously) or carry out the full search and rescue task.

Some findings of Wang et al.'s (2009) study were that exploration performance saw large increases in area explored between 4 and 8 robots and a flattening from 8 to 12 robots. Workload was high when evaluated with the NASA Task Load Index (Hart and Staveland, 1998); it increased proportionally with the number of robots. The comparison of the exploration, search and full task condition showed that *navigation* and *path-planning* were the primary contributors to the difficulties in performing the full task.

Itineraries in the dataset are defined, for our purposes, as a start and end point (participants chose their end points) and a set of waypoints defined by the participant, which were then sent to a robot to be executed autonomously. Participants defined the way-points while a view of the laser-scan map was shown. As a consequence, we did not have repeated measures of items.

### 6.3.2 Evaluation Method

The model was run against subject-produced itineraries. For each itinerary, the model was given the map view as seen by the participant at the time of path-planning, and the start and end points. The model did not retain state between itineraries: as discussed, higher-level planning, including selecting robots and their target points, is not evaluated here.

We will focus solely on the planned itineraries, treating them as a continuous path. The model does not predict more task-specific variables, such as where subjects set individual waypoints. The task does not lend itself to an evaluation of timing predictions.

Itineraries were split by subjects into a development (4 subjects) and a test set (10 subjects) in order to avoid design biases and overfitting: the model was developed and parametrized using the development set; results reported here were obtained using the test set only. We exclude data from 308 trivial cases, where start and end points could be and were connected in a straight line by the participant, and 83 cases as outliers where the model failed to reach the destination (see discussion). The remaining data set comprised 699 itineraries.

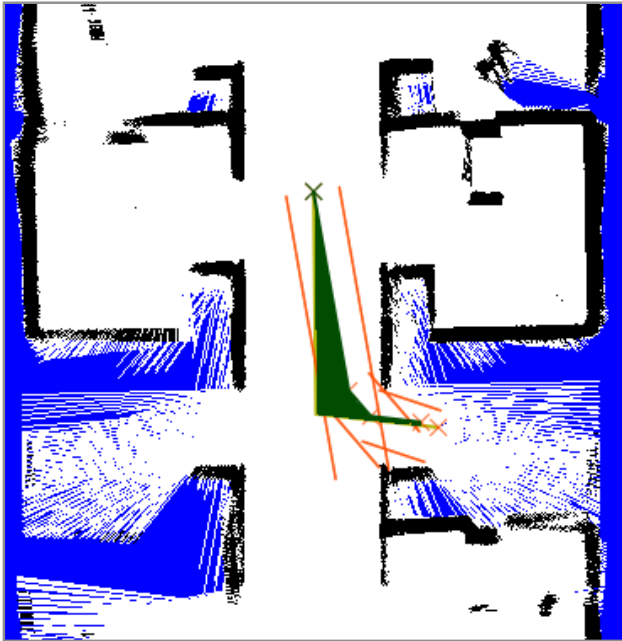
A simple method to determine model fit is to compare the length of the model's itineraries with those produced by the subject. However, when comparing such a measure between subjects, length will be confounded by the self-chosen distance of start and end points. A normalized path length measure would regress out the direct distance between start and end points; even so, the correlation between those normalized path lengths fails to penalize a model that, e.g., sometimes cuts corners where the subject does not and takes wide turns in other situations.

To define a more robust goodness-of-fit measure, we propose to measure the area defined between the two itineraries, that is, the polygon spanned by the two itineraries, whereas the start and end points of the itineraries are shared (Fig. 9). A large area implies a large deviation of the two paths; a small area means that the itineraries match well; the area is a useful measure for the more complex case of trajectories, where a temporal dimension is added (e.g., [24]). Length and area units refer to the virtual world.

*Trajectory Area Normalization* Because single decisions early in the planning process may lead to an unduly large deviation for long itineraries compared to short ones, we propose to normalize the area  $a$  by regressing out the respective effects of the lengths of the two paths, obtaining a normalized area  $a_n$ . To illustrate this, consider the case of a building with two long, parallel corridors A and B, with start and end points in connecting hallways at either ends of the corridors. If a participant chooses to send their robot down corridor A, but the model chooses B, then the penalty assessed by the area measure will depend on the length of the corridors. The normalized measure, however, will compensate for that and assign a comparable penalty for different corridor lengths.

A simple approach for this normalization would use a regression model such as:

$$a_n = \beta \frac{a}{\text{len}(\text{path}_M)\text{len}(\text{path}_S)}$$



**Fig. 9** A subject and a short model path, leading from a hallway (start point at top) into a room (on the right). Shaded area indicates (comparatively poor) model fit, straight lines parallel to path indicate width of robot as assumed by model. Markings (cross) show models decision points.

A linear model was fitted to our data, estimating the size of  $\beta$ . It failed to account fully for the effect, as residuals ( $a_n$ ) remained significantly correlated with  $a$ . We chose instead the model estimating  $\beta$  values in the equation:

$$a_n = \frac{a}{\text{len}(\text{path}_M)^{\beta_M} \text{len}(\text{path}_S)^{\beta_S}}$$

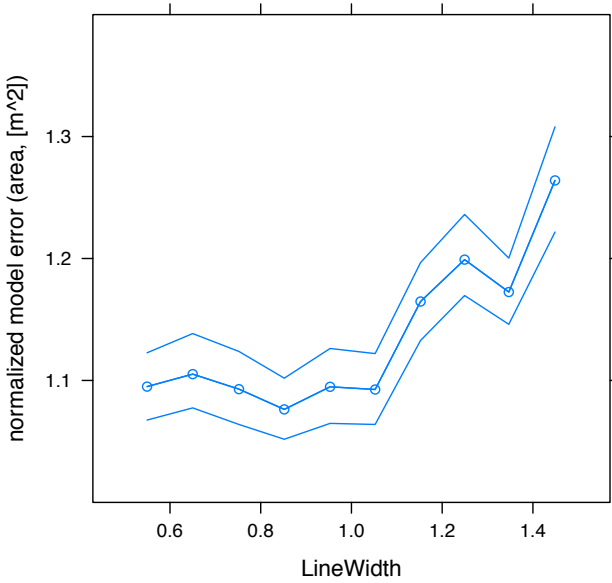
We fitted a corresponding linear fixed effects regression model ( $r^2 = 0.60$ ), estimating  $\beta_S = 1.86$  ( $p < 0.0001$ ) and  $\beta_M = -0.22$  ( $p < 0.05$ )<sup>3</sup>. The residuals  $a_n$  resulting from this regression model showed no more correlation with either path length; variance in the normalized path divergence area  $a_n$  should now be attributed to model performance and participant behavior under controlled experimental conditions.

To put the model's performance into perspective, we provide a baseline measure of a very simple, alternative model. The baseline model always plans a direct route from start to end point, regardless of walls and obstacles.

### 6.3.3 Parameter Optimization

Monte-Carlo simulations were run to explore three model parameters. We optimized against normalized areas, with subject and item-specific random variables regressed out. *LineWidth* controls the assumed width of a robot when subjects leave a margin for walls or obstacles. Larger robot widths lead to wider turns. It was set to the approximate global optimum of  $0.8m$ , suggesting an emphasis on safety in subject planning (Fig. 10). *LineLen* characterizes the longest straight line the model will examine

<sup>3</sup> Predictors were centered.



**Fig. 10** Effect of the width of a robot assumed by the model (minimum distance to walls and objects) on normalized model area. Lower areas indicate better fit the empirical data. 95% confidence intervals.

during the visual search for the next target point. High values lead to a preference for long, unobstructed paths, while low values lead the model to prefer complicated shortcuts with many turns. The optimum was at  $1m$ ; overall, the subjects' strategies were more similar to a model with shorter lines of sight, even though the effect was small (range:  $1.05m^2$  for  $LineLen=1.0$ , to  $1.15m^2$  for  $LineLen=1.15$ ,  $p < 0.0001^4$ ). The highly task-specific *UnexploredDensity* parameter was optimal at 0.6.

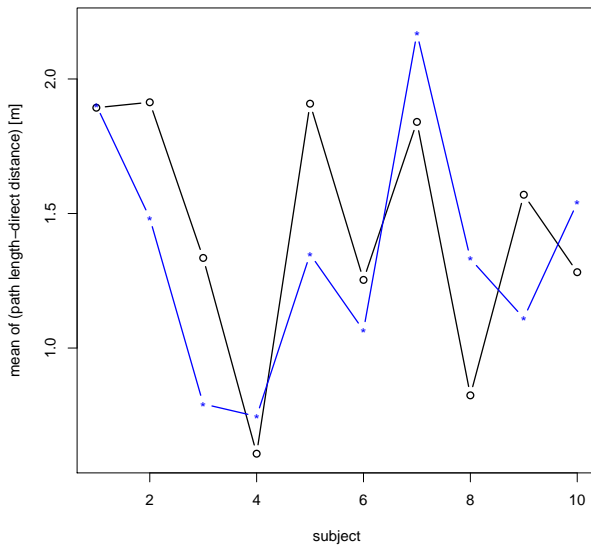
Subject-specific estimates of these parameters varied considerably; still, for reasons of simplicity, we present an evaluation of the model using optimized parameters across the dataset. The gain in normalized area through parameter optimization was, overall, relatively minor, and the model appeared to fit well across a range of these parameters.

### 6.3.4 Results

The length of the itineraries produced by model and subject correlated well across subjects (Pearson  $r = 0.97$  for subject means,  $r = 0.85$  overall). This is, to a large extent, owed to the fact that models were given the same start and end points. After regressing out the direct distance between start and end points, we retain a substantial correlation of path lengths ( $r = 0.65$  for subject means,  $r = 0.46$  overall). Fig. 11 shows the varying mean itinerary length over all subjects. Subject's mean path length is  $7.43$ ; the model tends to produce similar paths of mean length  $7.53m$ . (Note that Itinerary length increased with the number of robots; specifically, participants chose longer itineraries in the 12-robot (mean  $\mu_{12} = 8.34m$ ) condition than in the 4-robot condition ( $\mu_4 = 6.40m$ ,  $p < 0.005^5$ ).

<sup>4</sup> By linear mixed-effects regression. F1/2 analysis, i.e. random terms for subjects and items were fitted.

<sup>5</sup> Wilcoxon signed rank test for  $\mu_4$  vs.  $\mu_{12}$  over paired subject means



**Fig. 11** Mean length of the itineraries for each subject in the evaluation set, excluding the effect of distance between start and end points. Subjects: \*. Model: o.

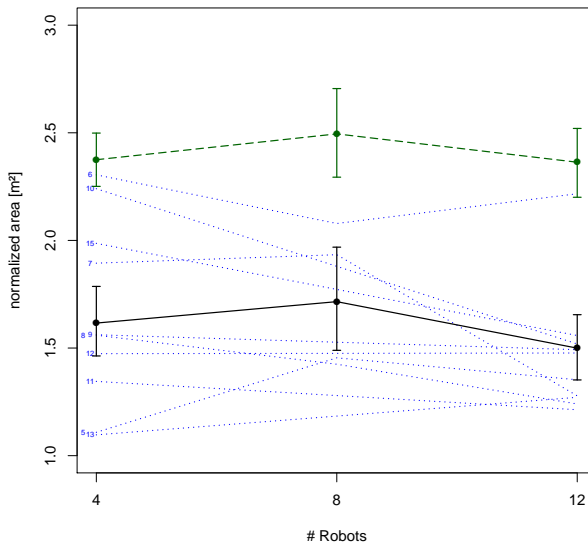
Fig. 12 shows the normalized area measures ( $\mu_{norm} = 1.58m^2$ , for un-normalized areas:  $\mu_{raw} = 2.77m^2$ ). We also show the model errors for each of the 14 subjects separately. The baseline model shows consistently larger areas and lower fit ( $\mu_{BL,norm} = 2.39$ ,  $\mu_{BL,raw} = 4.80$ ).

Comparing model error across the task load conditions, we see very small differences in normalized areas across task load conditions (4/8/12 robots: mean  $\mu_4 = 1.62m^2$ ,  $\mu_8 = 1.72m^2$ ,  $\mu_{12} = 1.50m^2$ ,  $p < 0.06^6$ ). A linear regression analysis of normalized areas predicted by condition and subject does not reveal any substantial differences in condition-specific model performance for the subjects.

### 6.3.5 Discussion

Our relatively simple, perceptual model accounts for much of the path-planning carried out by the subjects. The visual path-planning model appears to be accurate for planning situations under more and less demand (workload per se was not modeled). A qualitative analysis of cases where the model fits itineraries poorly suggests two common patterns. First, participants treat unexplored areas (solid colored areas usually adjacent to building perimeter) differently, either cutting through some of these areas, or avoiding them until a laser scan is obtained. The model expects unexplored areas to be accessible (any point's prior probability to be a wall is low), but it lacks inference about the structure of the building. Second, participants often chose to traverse corridors and doors in the center rather than cutting corners and merely optimizing distance. We interpret this as a risk minimization strategy to avoid contact with the wall, or a workload minimization strategy to limit the number of waypoints to

<sup>6</sup> Wilcoxon signed rank test for  $mu_4$  vs.  $mu_{12}$  over paired subject means.



**Fig. 12** Normalized area indicating negative model fit over conditions (4, 8 and 12 robots), i.e. smaller areas indicate better model fit. Mean across all paths (solid), and means for each subject in the evaluation set (dotted). Mean area of baseline model at top (dashed). 95% confidence intervals.

be specified. In the context of our model, this could be explained as an acquired but ultimately implicitly internalized strategy rather than part of an explicitly managed set of constraints for the task.

An analysis of a sample of those cases where the model failed to complete the path-planning showed that it was caught in small rooms or behind obstructions. This shows that the hard constraints the model applies to avoid getting too close to corners is relaxed in the subject’s planning strategy. The use of visual finst alone may allow the model to identify situations where it fails to make progress toward the goal, with no remediation available within the perceptual system. This shows a case where the memory-based component is needed.

The exploratory analysis of the model fit (Fig. 12) suggests that a few subjects may adapt their path-planning strategies as they are given a higher workload. Overall, their itineraries increase in length. This is a reasonable strategy, as it maximizes the time in which each robot will act autonomously. The differences in model fit are small. This picture is consistent with a views that subjects adapt their choice of problems (i.e., target points), but that similar visual path-planning strategies are applied regardless. While higher-level constraints seem to influence the subject’s task strategies, their low-level perceptual strategies are unaffected. Though not an identical result, the same principle is reflected in a study by Pelz and Canosa (2001), who find that look-ahead eye-movements were influenced by differences in high-level goals, but not by the “salience or conspicuity of objects in the environment.”

The visual strategy evaluated here focuses on the perceptual component of path-planning. More work is needed to actually define the reaction to task load within the higher-level path-planning model, specifically participants choosing their target locations.

## 7 Conclusion

In this paper, we have presented a dual-strategy model of human navigation, based on cognitively realistic memory and perceptual constraints from the ACT-R cognitive architecture. The model addresses two aspects of the navigation problem: how to plan a good path efficiently using a map view, given knowledge about various portions of the route, and how to plan the path given visual information about the route. The visual component may provide a good heuristic for exploration, helping in the acquisition of declarative and subsymbolic knowledge. Cognitive constraints of focused visual attention limit the visual component to a largely egocentric strategy, and the memory-based planning to incremental, local optimization.

The evaluation of the models shows that the memory-based approach, guided by subsymbolic information, is able to find efficient and near-optimal solutions within grid and tree layouts. The perceptual path-finding strategy produces reasonable paths in mazes; their performance degrades (plausibly) when larger mazes are used. In line with the literature, we infer a threshold of five to ten visual finsts, beyond which the visual component does not gain in performance. Data gathered from human operators of a multi-robot control system demonstrated validation of simulation results in a realistic scenario. Thus, the model presented was motivated both in a top-down fashion by cognitive framework, and in bottom-up fashion by empirical data.

Both routes of planning can integrate with some aspects of the environment. The visual planning route respects accessibility (both as absolute judgements of single boundaries and estimates of regions), while the memory-based route is based on known affordances, but could also use landmarks as cues to retrieve location-based information from memory. Work on data resulting from human failure to correctly plan paths (Lin and Goodrich, 2010) shows how the environment regularly influences path-planning decisions. With environmental features tied to memory access and perceptual judgments, portions of the wilderness rescue data (about human paths) would be explained by our cognitive model.

We acknowledge that there is little opportunity for performance comparison between different models at this point. Further work will test the integration of perceptual and memory-based path-planning and possibly provide a theory of how humans apply metacognitive reasoning and evaluation to choose visual or topological-map-based strategies in ambiguous situations. Our model proposes to interface those strategies using real-world landmarks as cues to the retrieval of locations stored in memory; operations to rotate and merge known paths through the memory map remain a challenge. Finally, the interaction between the perceptual and memory-based components deserves further scrutiny in the light of the question of how much algorithmic power the human visual system possesses.

The intention of this model is to address path-planning in the most general cases. It is not a task model: Consequently, we have ignored how humans balance different task-specific constraints. For instance, we do not predict the specific waypoints in the multi-robot simulation environment, where the cost for setting a waypoint (a mouse-click) is non-negligible. Similarly, we acknowledge that the robot navigation task also concerns exploration, a constraint that sometimes led subjects away from the most direct route between start and end points.

Our approach is significantly different from the emphasis on optimization that dominates algorithmic path-planning. While it still seeks to achieve the best possible path, it fundamentally operates in real time and factors in the time and effort to find a solution given cognitive constraints and mechanisms. Strictly speaking, the current model satisfices rather than optimizes in that it stops with the first solution that it finds, but it would certainly be possible for it to continue searching for a better solution and use a more complex time-accuracy tradeoff to its anytime-solution approach. One fundamental difference between algorithmic and cognitive solutions to path-planning (and indeed many AI problems) is the differential reliance on knowledge versus computation. Algorithmic solutions typically rely on brute processing speed (the strength of traditional CPUs) at the expense of knowledge, while cognitive solutions eschew computation (limited by a brain processing cycle of 10-20 Hz) in favor of improving performance with practice by accumulation, deployment and generalization of knowledge facilitated by the massively parallel access to post-cortical memory structures. This emphasis on knowledge-based solutions also makes it possible to integrate high-level factors in its solution in a more direct and natural way than an algorithmic approach, which must convert any abstract constraint on solution in terms of the underlying optimization framework (e.g., edge costs). Finally, the structured nature of a cognitive solution makes it easier to flexibly re-plan when the situation changes. Cognitive and algorithmic approaches to path-planning have distinct strengths and weaknesses, which leads us to conclude that a combination of those techniques might provide the best overall solution.

**Acknowledgements** We would like to acknowledge Michael Lewis, Huadong Wang, and Zheng Ma for providing data resulting from their multi-robot control study. This work was funded by the Air Force Office of Scientific Research (MURI grant FA95500810356).

## References

- Anderson JR (1991) The place of cognitive architectures in a rational analysis. In: VanLehn K (ed) *Architectures for Intelligence*, Lawrence Erlbaum Associates, Mahwah, NJ, pp 1–24
- Anderson JR (2007) *How can the human mind occur in the physical universe?* Oxford University Press, Oxford, UK
- Balakirsky S, Carpin S, Kleiner A, Lewis M, Visser A, Wang J, Zipara V (2007) Toward heterogeneous robot teams for disaster mitigation: Results and performance metrics from robocup rescue. *Journal of Field Robotics* 24(11-12):943–967
- Best BJ, Lebiere C (2006) Cognitive agents interacting in real and virtual worlds. In: Sun R (ed) *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation*, Cambridge University Press, New York, NY, pp 186–218
- Byrne MD (2006) A theory of visual salience computation in ACT-R. In: *Presentation at the Thirteenth Annual ACT-R Workshop*, Pittsburgh, PA
- Chandrasekaran B (2006) Multimodal cognitive architecture: Making perception more central to intelligent behavior. In: *Proceedings of the Twenty-First AAAI National Conference on Artificial Intelligence*, AAAI Press, Boston, MA, pp 1508–1512

- Dye HA (2007) Diagrammatic reasoning: Route planning on maps with ACT-R. In: Lewis RL, Polk TA, Laird JE (eds) *Proceedings of the Eighth International Conference on Cognitive Modeling*, Ann Arbor, MI, pp 217–218
- Fu WT (2003) An ACT-R adaptive planner in a simple map-navigation task. In: Detje F, Doerner D, Schaub H (eds) *Proceedings of the Fifth International Conference on Cognitive Modeling*, Bamberg, Germany, pp 99–104
- Fum D, del Missier F (2000) Climbing the mazes: A cognitive model of spatial planning. In: *Proceedings of the Third International Conference on Cognitive Modeling*, Universal Press, Veenendaal, The Netherlands, pp 126–133
- Glasgow J, Papadias D (1998) Computational imagery. In: Thagard P (ed) *Mind Readings*, MIT Press, Cambridge, MA
- Gluck KA, Pew RW (2005) *Modeling Human Behavior With Integrated Cognitive Architectures: Comparison, Evaluation and Validation*. Lawrence Erlbaum Associates, Mahwah, NJ
- Gonzalez C, Lebiere C (2005) Instance-based cognitive models of decision making. In: Zizzo D, Courakis A (eds) *Transfer of knowledge in economic decision making*, Palgrave MacMillan, New York, NY
- Hart SG, Stavenland LE (1998) Development of NASA-TLX (task load index): Results of empirical and theoretical research. In: Hancock PA, Meshkati N (eds) *Human Mental Workload*, Elsevier, Amsterdam, The Netherlands, pp 139–183
- Hiatt LM, Trafton JG, Harrison AM, Schulz AC (2004) A cognitive model for spatial perspective taking. In: *Proceedings of the Sixth International Conference on Cognitive Modeling*, Pittsburgh, PA, pp 354–355
- Johnson T, Wang H, Zhang J, Wang Y (2002) A model of spatio-temporal coding of memory for multidimensional stimuli. In: Gray WD, Schunn C (eds) *Proceedings of the 24th Annual Meeting of the Cognitive Science Society*, Fairfax, VA
- Kelley TD (2006) Developing a psychological inspired cognitive architecture for robotic control: The symbolic and sub-symbolic robotic intelligence control system (SS-RICS). *International Journal of Advanced Robotic Systems* 3(2)
- Kosslyn SM (1980) *Image and Mind*. Harvard University Press, Cambridge, MA
- Kosslyn SM (1994) *Image and Brain*. MIT Press, Cambridge, MA
- Lathrop SD, Laird JE (2007) Towards incorporating visual imagery into a cognitive architecture. In: *Proceedings of the Eighth International Conference on Cognitive Modeling*, Ann Arbor, MI, pp 25–30
- Lebiere C, Wallach D (2001) Sequence learning in the ACT-R cognitive architecture: Empirical analysis of a hybrid model. In: Sun R, Giles L (eds) *Sequence Learning: Paradigms, Algorithms, and Applications*, Springer LNCS/LNAI, Berlin, Germany
- Lewis M, Wang J, Hughes S (2007) USARsim: Simulation for the study of human-robot interaction. *Journal of Cognitive Engineering and Decision Making* 1(1):98–120
- Lin L, Goodrich MA (2010) A bayesian approach to modeling lost person behaviors based on terrain features in wilderness search and rescue. *Computational and Mathematical Organization Theory* (this volume)
- Lyon DR, Gunzelmann G, Gluck KA (2008) A computational model of spatial visualization capacity. *Cognitive Psychology* 57:122–152
- Millán JDR, Torras C (1992) A reinforcement connectionist approach to robot path finding in non-maze-like environments. *Machine Learning* 8(3-4):363–395

- O'Keefe J, Nadel L (1978) *The Hippocampus as a Cognitive Map*. Oxford University Press, Oxford, UK
- Pelz JB, Canosa R (2001) Oculomotor behavior and perceptual strategies in complex tasks. *Vision Research* 41:3587–3596
- Pylyshyn ZW (1989) The role of location indexes in spatial perception: A sketch of the first spatial-index model. *Cognition* 32:65–97
- Reitter D, Lebiere C (2010) Accountable modeling in ACT-UP, a scalable, rapid-prototyping ACT-R implementation. In: *Proceedings of the 10th International Conference on Cognitive Modeling*, Philadelphia, PA
- Salvucci DD (2001) An integrated model of eye movements and visual encoding. *Cognitive Systems Research* 1:201–220
- Sanner S, Anderson JR, Lebiere C, Lovett M (2000) Achieving efficient and cognitively plausible learning in backgammon. In: *Proceedings of the Seventeenth International Conference on Machine Learning*, San Francisco, CA, pp 823–830
- Schunn CD, Harrison AM (2001) ACT-RS: A neuropsychologically inspired module for spatial reasoning. In: *Proceedings of the Fourth International Conference on Cognitive Modeling*, Fairfax, VA, pp 267–268
- Shepard RN, Metzler J (1971) Mental rotation of three-dimensional objects. *Science* 171:701–703
- Simon H (1991) Cognitive architectures in a rational analysis: comment. In: VanLehn K (ed) *Architectures for Intelligence*, Lawrence Erlbaum Associates, Mahwah, NJ, pp 25–40
- Wang H, Lewis M, Velagapudi P, Scerri P, Sycara K (2009) How search and its sub-tasks scale in N robots. In: *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, ACM, New York, NY, pp 141–148
- West RL, Stewart TC, Lebiere C, Chandrasekharan S (2005) Stochastic resonance in human cognition: ACT-R vs. game theory, associative neural networks, recursive neural networks, q-learning, and humans. In: B Bara LBMB (ed) *Proceedings of the Twenty-Seventh Annual Conference of the Cognitive Science Society*, Stresa, Italy, pp 2353–2358
- Wray R, Lebiere C, Weinstein P, Jha K, Springer J, Belding T, Best B, Parunak V (2007) Towards a complete, multi-level cognitive architecture. In: Lewis RL, Polk TA, Laird JE (eds) *Proceedings of the Eighth International Conference on Cognitive Modeling*, Ann Arbor, MI, pp 325–330