LE MATEMATICHE
Vol. LXXIX (2024) – Issue II, pp. 555–583
doi: 10.4418/2024.79.2.17

AN AGENT FRAMEWORK TO EXPLORE PATHFINDING STRATEGIES IN MAZE NAVIGATION PROBLEM

C. CRESPI - V. CUTELLO - M. PAVONE - F. ZITO

The planning of paths in complex, interconnected, and unknown structures, such as mazes, is a crucial topic in various fields, including artificial intelligence and robotics. Agents capable of making independent decisions require efficient navigation through mazes, and their performance can be influenced by various dynamics and features. Understanding these factors is essential not only for developing more efficient and robust navigation algorithms but also for gaining deeper insights into which attributes to prioritize in the design and implementation of autonomous agents. In this article, we propose an agent framework to analyze various navigation strategies based on the concepts of memory and visibility. Our goal is to identify the parameters that impact the agents' performance the most and how variations on these key parameters influence agents' efficiency on complex maze-solving.

1. Introduction

Planning paths in complex, interconnected, and unknown structures, namely mazes, is a topic of significant importance across different areas such as artificial intelligence and robotics [12, 29]. In artificial intelligence, mazes are often used as benchmarks to test the efficiency of algorithms in determining

Received on September 29, 2024

Keywords: multi-agent system, maze navigation problems, path finding strategies

optimal paths [1, 28]. This helps in evaluating and improving algorithms designed for pathfinding and decision-making in complex environments. Similarly, in robotics, mazes serve as models for virtual environments where optimizing robot movements is crucial for tasks such as autonomous navigation and obstacle avoidance [2, 14, 18, 34].

Within these broader contexts, the study of pathfinding for autonomous agents—intelligent entities capable of making independent decisions—has become a crucial area of research [27]. These agents need to navigate efficiently through mazes, and their performance can be influenced by various characteristics. Understanding these factors is essential not only for developing more efficient and robust navigation algorithms but also for gaining deeper insights into which attributes to prioritize in the design and implementation of autonomous agents.

In light of this importance, this research work proposes a framework to explore and analyze the impact that some features of these agents have on maze-solving tasks and how their variations affect overall performance. Specifically, visibility, memory size, and exploration tendency have been selected as key features for this investigation. For clarity, *visibility* refers to the agent's perception range within the maze; *memory size* to the agent's capacity to store crossed paths; and *exploration tendency* to the agent's preference for prioritizing new and unexplored paths or previously crossed ones. Each agent is characterized by a unique combination of these parameters, which govern its navigation strategies and capabilities and is tasked with identifying the maze exit from a given entrance.

The aim of this research work is to propose a new framework for investigating and understanding the influence of visibility and memory on the performance of autonomous agents in pathfinding problems in mazes. This framework has been developed with potential extensions to multi-agent systems in mind. While the current study focuses on individual agent features like visibility and memory, the approach provides a foundation for exploring how these features can scale or adapt in systems involving multiple agents. This adaptability positions the framework as a versatile tool for studying both isolated agent behavior and broader dynamics in multi-agent contexts.

The paper is organized as follows. Section 2 provides an overview of recent trends and advancements in the field of multi-agent systems. Section 3 describes the key features of our proposed model and the strategies employed by the agents. The experimental protocol used to test our model, including the maze instances, is described in Section 4. Section 5 outlines the experimental setup, presents the results, and provides a critical discussion on the investigation carried out. Finally, Section 6 presents our concluding remarks.

2. Recent Trends and Advancements

The analysis of different navigation strategies for efficient maze-solving is an interesting and well-studied area in literature, with several proposed algorithms. In this section, we analyze different perspectives that take into account various factors such as the cooperative and competitive behavior of intelligent agents, the use of memory, and visibility. Earlier investigations focused on analyzing the effects of cooperative and competitive behaviors on agent performance within maze-like structures, leveraging the Ant Colony Optimization framework [8] to model agents' movement within the environment. Specifically, it was developed an agent-based model featuring two types of agents both aiming to reach the exit of a virtual environment. The two types of agents were competitive agents, which could damage random parts of the environment, and collaborative agents, which could repair the damaged parts while providing information about the cost of a crossed section [4, 7].

Different versions of such model have been proposed in the last years. In a first model version, a profit function was used as an evaluation metric, which showed that predominantly, but not entirely, collaborative groups of agents achieve the highest profit function values [6]. Subsequently, the model was upgraded into a new version including the number of agents which had exited the maze, exit times, and path costs as evaluation metrics. Then, a sensitivity analysis of the parameters of the model has been conducted, discovering that predominantly collaborative sets of agents provide the best performance according to these evaluation metrics [4, 5]. However, these outcomes emerge only under specific circumstances related to the interplay between the parameters. Indeed, in those cases where the available information is high, then, the presence of competitive agents turns out to be crucial to achieve efficient performances; otherwise, entirely collaborative groups reach best results [7].

Building on these insights, our currently research focuses on investigating the role of visibility and memory as key parameters of the aforementioned model. Then, first goal of this paper is to analyze how these characteristics are implemented and studied within the context of pathfinding problems. In order to do that, several research works published over the past five years have been taken into account. After that, the goal is to design and investigate these two features, i.e. visibility and memory, in order to understand their impact on the performance in pathfinding maze tasks.

2.1. The role of Visibility in Navigation Strategies

One of the widely used techniques to integrate visibility in path planning problems is the *visibility-graph method*. For instance, the authors in [16] developed a fast path planning algorithm for UAVs (Unmanned Aerial Vehicles) in 3D environments with obstacles, utilizing visibility graphs (VG) and sparse visibility graphs (SVG) to reduce computational complexity while maintaining path quality. Their results showed that SVGs could significantly improve computational efficiency without compromising path accuracy. Similarly, [35] explored the impact of spatial attributes and physical design elements in emergency departments on staff satisfaction and performance, using visibility graphs to analyze and visualize visibility patterns. The study found that improved visibility within the department correlated with higher staff satisfaction and better performance.

In the marine context, the authors in [15] employed visibility graphs and quadtree representation for path planning of USVs (Unmanned Surface Vehicle), aiming to minimize voyage time. Their methodology proved effective in optimizing navigation routes in complex maritime environments. Likewise, [13] presented an algorithm for pathfinding in confined environments using visibility graphs to create optimal movement paths for autonomous vessels, demonstrating that visibility graphs can significantly improve navigational efficiency.

Another interesting approach to modeling visibility was the use of Voronoi diagrams, as demonstrated by the authors in [17], who introduced a Voronoi-based navigation mesh for video games that integrates tactical properties such as visibility to enhance path planning. Their findings indicated that incorporating visibility into the navigation mesh led to more efficient and realistic pathfinding behavior in game environments.

Further research papers focused on the role of visibility in pathfinding efficiency. For example, the authors in [30] explored the performance of Multipath Adaptive A* (MPAA*) in goal-directed navigation in unknown terrain, finding that visibility played a crucial role in enhancing the algorithm's performance. The authors in [24] applied reinforcement learning for pathfinding in grid environments with static and stochastic obstacles, leveraging visibility to enhance the learning algorithms. Their results showed that reinforcement learning with visibility considerations outperformed traditional algorithms in dynamic environments. The same authors continued this line of research in [23] by coordinating multiple agents in complex scenarios using a decentralized multi-agent pathfinding approach, demonstrating superior performance due to the incorporation of visibility in their models.

Finally, the authors in [11] investigated wayfinding in multi-level buildings using agent-based modeling and VR experiments, highlighting the importance of visibility for efficient navigation. Their study concluded that better visibility improves wayfinding efficiency in complex indoor environments.

As seen from the analyzed papers, one of the most commonly used methods to address the concept of visibility in pathfinding problems is through spatial representations such as visibility graphs and Voronoi diagrams. These methods, however, are not typically implemented within agents or robots to model vision directly. Instead, they serve as indirect approaches for environment modeling. As detailed in [33], in the context of environment modeling the goal is to create a simplified representation of the space, which requires prior environmental information acquisition. Visibility graphs and Voronoi diagrams are among the techniques used for this purpose. Our interpretation is that these methods consider visibility as a given for the agents rather than an inherent capability to be developed within them. Consequently, they necessitate prior knowledge of the environment, making them less suitable for scenarios where the topology is unknown.

2.2. Exploration in Unknown Environments through the use of Memory

Recent studies have explored the use of memory and its implications in agent navigation within unknown environments. For instance, in [26], it is studied how agents navigate unknown environments while incorporating memory decay into the wayfinding procedures. Memory decay reflects the gradual fading or weakening of memory traces associated with routes, influencing decision-making processes. In a different context, [19] proposes a model which uses attention-augmented memory (AAM) to interpret the decision-making process in long-horizon tasks, showing how AAM facilitates more stable decision-making. Furthermore, the work in [25] explores how external memory can aid decision-making in complex visual reinforcement learning tasks, by incorporating into the model both short-term recurrent memory and long-term external memory.

In [3], memory-enhanced ant colonies have been proposed to investigate the influence of colony division in a maze navigation problem. While investigating the role of memory was beyond the scope of the paper, memory was implemented by allowing the ants to remember the cost of an already crossed path. This enabled ants to compute and estimate the cost of a path yet to be traversed path when its cost was not directly visible.

In contrast to the approaches discussed in the previous paragraphs, the literature presents an alternative approach that involves a multi-agent system based on Reinforcement Learning [21]. The main idea is to develop an agent that can learn autonomously by using rewards to determine the optimal strategy in unknown environments. These agents are based on Recurrent Neural Networks that incorporate Long and Short-Term Memory Units to make decisions based on past experiences. An enhanced version of this approach is described in [32], where multiple agents cooperate by sharing information in a multi-agent system.

2.3. On the role of visibility and memory in pathfinding problems

During our analysis of the literature on maze navigation strategies, we observed that there is a lack of research specifically aimed at studying the roles of visibility and memory in pathfinding problems. Indeed, this gap in research presents an opportunity for further investigation into the potential benefits of incorporating visibility and memory into pathfinding algorithms. By studying the impact of these factors on agent performance in maze-solving tasks, it is possible to gain a better understanding of how to optimize navigation strategies in unknown environments. Additionally, the findings from such research could have practical applications in fields such as robotics, autonomous vehicles, and video game development.

The incorporation of exploration tendencies into the model represents a novel aspect to be investigated. While memory and visibility, as mentioned above, have been extensively studied in different contexts, in this research we decided to add the exploratory tendencies in the aforementioned model to introduce an additional layer of complexity in the analysis of behavioral dynamics. This choice aims not only at making agents more similar to humans in their behavior, which often involves a blend of exploration and caution in navigation decisions, but also at exploring how these tendencies can positively influence maze-solving capabilities.

Unlike existing works that primarily focus on multi-agent frameworks, our research isolates and investigates key features at the individual agent level, such as visibility and memory. By bridging this gap, our approach complements studies on multi-agent dynamics, offering new insights into agent-specific behaviors.

3. Agent Modeling

Building upon the analysis conducted in the previous section, we present here a model for the maze navigation problem. In this research work, we will discuss different navigation strategies used by intelligent agents as they navigate through the maze with the aim of reaching the exit. These strategies will be based on two key agent features: visibility and memory.

3.1. Maze Representation

In general, a maze is a structure that is characterized by having only one entrance and only one exit: agents enter the maze through the entrance and must find their way to the exit based on their strategy used. In our model, each intelligent agents acts according to different visibility, memory and exploration strategies. To each

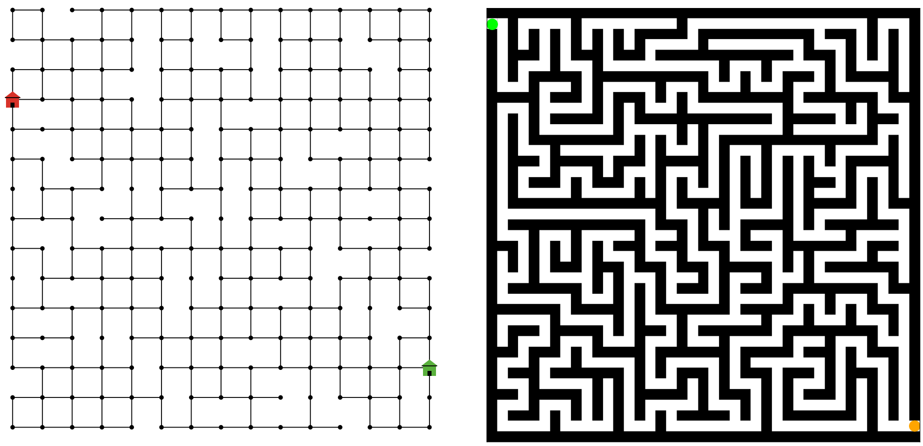


Figure 1: Two different ways to model a maze: as a *graph* in left plot, and as a *matrix* in right plot.

agent is assigned a determined starting energy whose consumption depends on the paths taken. The goal of each agent is then to find the exit consuming as less energy as possible, that is, to reach the exit before its energy runs out. If this happens, the agent will not be able to escape the maze and will perish.

In one of our previous work [3], we modeled a maze as a graph $G = (V, E)$, where V is the set of vertices and E is the set of edges. Each vertex in V represents a point in the maze, and each edge in E represents a path between two points. Two vertices, s and t, are designated as the starting and ending points, respectively. The left plot in Fig. 1 shows a representation of a labyrinth based on a graph. However, in this work, we consider a model of a maze from a different point of view. Indeed, here, a maze is modeled as a square matrix of size $N \times N$. An agent can navigate through the maze by moving cell by cell in the four available directions (Up, Down, Left, Right), without diagonal movements. The matrix contains two types of cells: walkable cells, which an agent can traverse, and non-walkable cells, which represent walls. To simulate a real-world scenario, to each walkable cell is associated a cost required to cross it. This cost represents the difficulty or energy required for crossing a specific cell, which could be due to an obstacle or a steep incline, for instance. The right plot of Fig. 1 shows a representation of a maze based on a matrix.

3.2. Agent Framework

As briefly described above, the strategy employed by an agent to escape from a maze is determined by three key characteristics: visibility, memory, and ex-

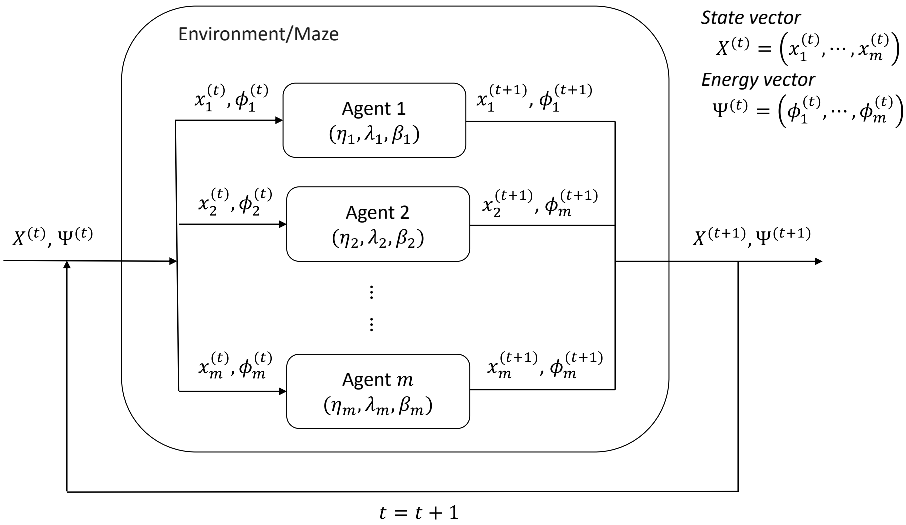


Figure 2: This figure illustrates the movement of agents within the maze environment. Each agent (i) has unique properties, including visibility (η_i), exploration tendency (β_i), and memory size (λ_i). Based on these characteristics, each agent evaluates its current position ($X_i^{(t)}$) and remaining energy ($\Phi_i^{(t)}$) to select the next cell to move into ($X_i^{(t+1)}$). After each move, the agent updates its energy level ($\Phi_i^{(t+1)}$) accordingly. This process continues until either the agent runs out of energy or successfully finds the exit of the maze.

ploratory tendencies. Visibility, denoted by $\eta \in [\eta_{\min}, \eta_{\max}]$ determines an agent's ability to perceive distant paths. Memory denoted using the parameter $\lambda \in [\lambda_{\min}, \lambda_{\max}]$, indicates the number of paths crossed by the agent and which it is able to store. Exploratory tendency is identified by the parameter β and we incorporated it into the rule of the probabilistic function that governs the agents' movement decisions. In other words, to capture the decision-making process of autonomous agents navigating maze environments, we developed a probabilistic function.

In general, we can formally define the i-th agent as a tuple $(\eta_i, \beta_i, \lambda_i)$, which represents its visibility, exploration tendency, and memory size, respectively. Each agent utilizes these features to adapt its strategy for exploring the maze. Figure 2 provides a schematic representation of this exploration process. At any given moment, each agent occupies a specific cell within the maze. We define a *vector state* $X^{(t)}$ as a vector that contains all the positions of agents in the maze at time t. This can be formally expressed as:

$$X^{(t)} = \left(x_1^{(t)}, x_2^{(t)}, \ldots, x_m^{(t)}\right), \tag{1}$$

where $x_i^{(t)}$ represents the cell in which agent i is located at time t. On the other hand, we define an *energy vector* $\Psi^{(t)}$ that contains the remaining energy of each agent, which is defined as follows:

$$\Psi^{(t)} = (\phi_1^{(t)}, \phi_2^{(t)}, \ldots, \phi_m^{(t)}). \tag{2}$$

Both the state vector and the energy vector are updated at each time step, specifically when an agent moves from one cell to another. The remainder of this section discusses the process by which an agent selects the cell to which it will move, based on its current position $x_i^{(t)}$ and remaining energy $\phi_i^{(t)}$, which serve as the agent's inputs.

Figure 3 illustrates the flow diagram that the i-th agent follows to navigate the maze. Given its current position $x_i^{(t)}$ and remaining energy $\phi_i^{(t)}$ as inputs, the agent computes the next position $x_i^{(t+1)}$ based on its strategy and updates its current remaining energy $\phi_i^{(t+1)}$. In our context, when agents have to choose between two or more alternative paths, the value σ_j (defined in equation 3) tells us whether agents prefer to explore new paths or retrace previous ones. It quantifies the overlap between the current path and previously explored paths by the ratio of overlapping cells to the total cells in the previous path:

$$\sigma_j = \frac{\pi_{\text{overlap}}}{\pi_{\text{tot}}}, \tag{3}$$

where π_{overlap} denotes the count of overlapping cells, and π_{tot} denotes the total number of cells explored in the agent's preceding path. A high σ_j suggests the agent favors the same paths, while a low σ_j indicates a preference for exploring new paths.

Using the σ values and starting with the Normalized Exponential Function, a common tool for modeling probability distributions, the probability $\bar{p}_j$ which an agent uses to selects its next path j is initially expressed in equation 4:

$$\bar{p}_j = \frac{e^{\sigma_j}}{\sum i = 1^n e^{\sigma_i}}. \tag{4}$$

To refine this function, we introduced the parameter β, which adjusts $\bar{p}_j$ based on agent behavior ranging from a preference for new paths ($\beta = 0.0$ equation 5) to a preference for old paths ($\beta = 1.0$ equation 6).

$$\beta = 0 \Rightarrow \bar{p}_j = 1 - \frac{e^{\sigma_j}}{\sum_{i=1}^{n} e^{\sigma_i}}, \tag{5}$$

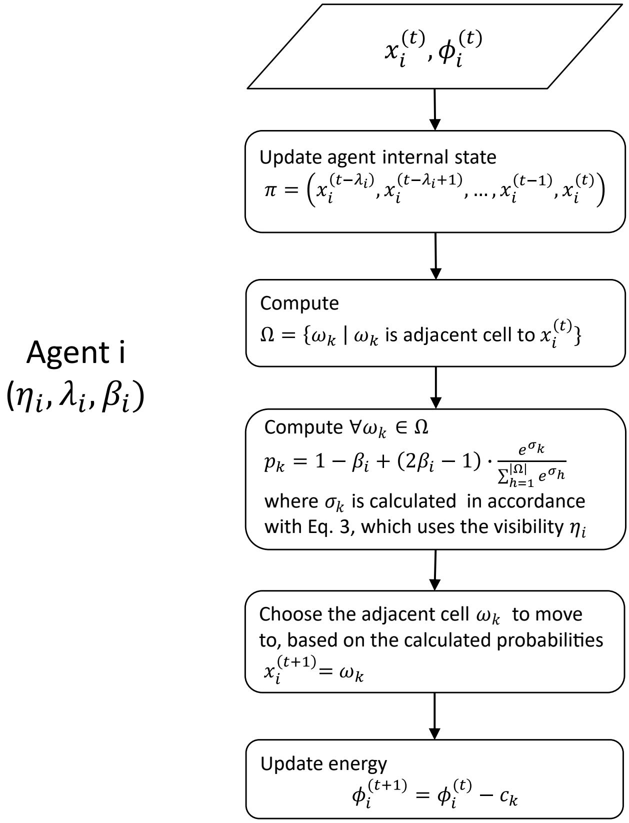


Figure 3: Flow diagram illustrates the operations that the i-th agent performs to select the next cell to move into in a maze.

$$\beta = 1 \Rightarrow \bar{p}_j = \frac{e^{\sigma_j}}{\sum_{i=1}^{n} e^{\sigma_i}}. \tag{6}$$

To derive our final probabilistic function, we used the equation of a line passing through two points because it provides a straightforward method to interpolate between the two extreme behaviors governed by the parameter β. This approach allowed us to smoothly transition from scenarios where agents exclusively prefer new paths ($\beta = 0$) to those where they prefer retracing old paths ($\beta = 1$). Mathematically, the equation of a line is an efficient way to create a linear combination of two conditions. For our model, we identified then the two key points: $\beta = 1$, agents prefer always old paths and $\beta = 0$, agents prefer entirely new paths.

By using the line equation passing through these points, we create a formula that linearly interpolates between these two behaviors, ensuring that for any value of $\beta \in [0,1]$, $\bar{p}_j$ is a weighted average of the two extremes. This interpolation captures the continuous spectrum of agent behaviors from choosing new paths to choosing old paths, based on β. Thus, the final equation is shown in equation 7:

$$\bar{p}_{j,\beta} = 1 - \beta + (2\beta - 1) \cdot \frac{e^{\sigma_j}}{\sum_{i=1}^{n} e^{\sigma_i}}, \tag{7}$$

and it represents a balanced combination of the two probabilities, scaled by β, ensuring that our model accurately reflects the desired agent behavior across the entire range of β values. In particular, it is clear from equation 7 that

$$\bar{p}_{j,1-\beta} = 1 - \bar{p}_{j,\beta}$$

and, for the two borderline cases, $\beta = 0$ and $\beta = 1$ we have:

$$\begin{aligned} \bar{p}_{j,0} &= 1 - \frac{e^{\sigma_j}}{\sum_{i=1}^{n} e^{\sigma_i}} \quad (\beta = 0) \\ \bar{p}_{j,1} &= \frac{e^{\sigma_j}}{\sum_{i=1}^{n} e^{\sigma_i}} \quad (\beta = 1) \end{aligned}$$

Once the agent has selected the next cell (k) to move into based on the probability computation described above, the i-th agent updates its current remaining energy as follows:

$$\phi_i^{(t+1)} = \phi_i^{(t)} - c_k, \tag{8}$$

where c_k is the weight of the cell selected by the agent, and $\phi_i^{(t)}$ represents the energy level prior to the movement.

4. Experimental protocol

As mentioned in the previous section, Eq. 7 governs the strategy used by an agent to select the path to exit from the maze. To prove the practicality of this equation, we designed an agent framework to analyze various strategies for exiting a maze. By using multiple agents, the framework allows for the exploration of different approaches to maze navigation, including those that are based on visibility, memory, and exploratory tendencies.

4.1. Agent Simulation

The framework was developed in Python and designed as an an agent framework These agents are equipped with an initial energy charge, whose consumption depends on the crossed paths, and are tasked with reaching the maze's exit in the shortest time while preserving as much energy as possible before two termination criteria are met. The first termination criterion is the maximum simulation time. The second termination criterion is the maximum possible path cost that an agent incurs. This maximum cost corresponds to the total sum of all maze weights and is interpreted as follows: each agent has a finite amount of energy to navigate the maze, which decreases proportionally with the weight of each traversed cell. Thus, as an agent traverses cells with higher weights, its energy decreases proportionally. Consequently, we set the agent's available energy not to exceed the total sum of all maze weights. Therefore, simulation termination occurs when either of the two termination criteria is met. Specifically, the simulation ends if an agent fails to find the exit within the allocated maximum time or if the agent depletes its energy.Upon meeting either termination criterion, the respective agent is killed.

4.2. Maze instances

To evaluate the agents' performance, we generated instances of mazes representing different configurations. These instances were created using the same seed and categorized into two types: large (L) of size 81×81 and medium (M) of size 41×41. Each type includes a set of ten mazes with different degrees of density. The density parameter determines the number of possible paths present in the maze, that is the count of navigable cells within it. Mathematically is defined as the ratio between the traversable cells and the total number of cells (traversable cells and walls). All ten mazes share the same dimensions but the number of navigable cells, i.e. the space agents can cross, might change and if it increases, the number of walls decreases proportionally.

In table Tab.1 are reported the features of each generated instance, and considered for presented investigation. In particular, are reported, respectively: instance name (No); total number of traversable cells (N_c) in the squared matrix with which the maze is represented (see section 3.1); total number of walls (N_w) in the maze; and finally the relative density (Δ) of each instance. Then, L1 and M1 represent the least dense mazes, while L10 and M10 are the ones with highest densities.

In all mazes, the weight distribution is uniform within the interval $w_{ij} \in [0;0.7]$, except for the entrance that has an undefined weight (NaN) to prevent agents from mistakenly selecting it as an exit and the exit that weights 0 to facilitate the agents' exit.

Figures Fig.4 and Fig.5 show, respectively, four examples of generated mazes for medium (Fig.4) and large (Fig.5) types. For brevity, we have decided to display in increasing order of density (from top-left to bottom-right), only 4 out of the 10 generated mazes, namely M1, M3, M7 and M10 for medium size, and

Table 1: Details about the maze instances by considering two different types: large (L) and medium (M). For each kind of instances, are displayed instance name (No); total number of traversable cells (N_c); total number of walls (N_w); and density ($\Delta = N_c/(N_c + N_w)$) of the maze.

(a) Large (L)

No	N_c	N_w	Δ
L1	3519	3042	0.53635
L2	3806	2755	0.58009
L3	4042	2519	0.61606
L4	4276	2285	0.65173
L5	4514	2047	0.688
L6	4664	1897	0.71087
L7	4832	1729	0.73647
L8	4976	1585	0.75842
L9	5110	1451	0.77884
L10	5221	1340	0.79576

(b) Medium (M)

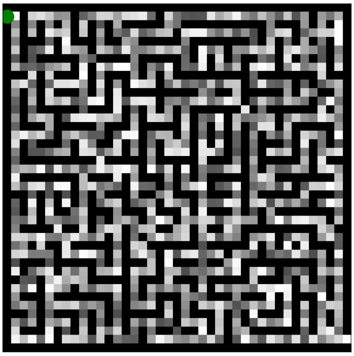
No	N_c	N_w	Δ
M1	878	803	0.52231
M2	945	736	0.56217
M3	1013	668	0.60262
M4	1068	613	0.63534
M5	1114	567	0.6627
M6	1150	531	0.68412
M7	1189	492	0.70732
M8	1228	453	0.73052
M9	1261	420	0.75015
M10	1283	398	0.76324

L1, L3, L7 and L10 for the large size. The entrance of the mazes is always located in the top left corner (represented by a green tile), while the exit is always in the bottom right corner (simply represented as a hole in the wall).

In addiction, the walls are denoted by the color black, while white cells and cells with different degrees of gray are the traversable ones. Color shades denote the different traversability weight of each cell: from the smallest weight (white color) to the highest one (dark gray).

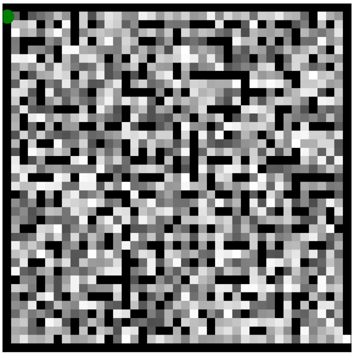
We recall that we defined density as the ratio between the traversable cells over the total number of cells. Therefore, as it easily visible from both figures Fig.4 and Fig.5, the less dense is the maze the more complex it is to solve it, since fewer paths lead to the exit and there is a higher number of walls that the agent can encounter which, in turn, increases the presence of dead-ends. Conversely, if the maze has a high density then the number of walls/obstacles is low and, consequently, there is a higher number of paths leading to the exit. Such an inference is also confirmed by inspecting figure Fig.6, where we report the success rate (SR) for the different maze instances, specifically (6a) for medium and (6b) for large maze. SR is a classic evaluation metric that represents, in our investigation, the percentage of agents that successfully reach the exit, and in these plots are displayed versus those who instead run out of energy before reaching the exit.

In particular, in the X-axis the ten maze instances are displayed in order from least dense to most dense; the Y-axis represents the percentage of agents that,

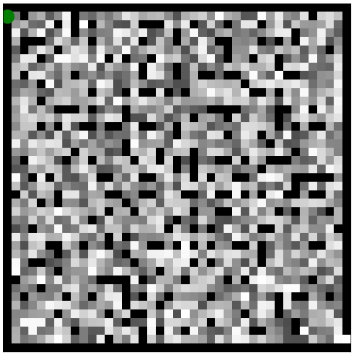


(a) M1

(b) M3



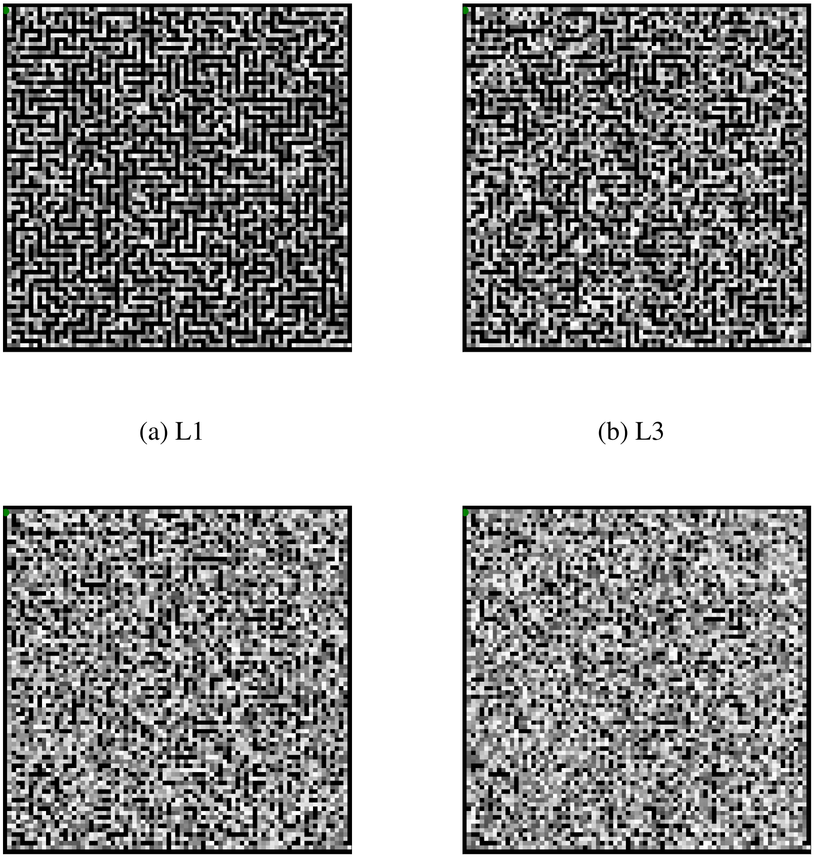
(c) M7



(d) M10

Figure 4: Examples of medium-sized mazes at different values of the density parameter: M1, M3, M7, and M10.

respectively, reach the exit or die before. More in details, each plot contains two lines: the blue line denotes the percentage of agents that exited the maze, while the orange line shows the percentage of agents that stop due to energy exhaustion and consequent death. These two curves are mirror images of each other. Inspecting both figures, it emerges that the percentage of agents successfully exiting increases as the maze density factor rises, resulting, as a conseuqnce, in a decrease of the percentage of killed agents. This analysis suggests and confirms that having more pathways potentially leads to a higher success rate, as stated before, since agents have multiple routes to reach the exit and less likely



(a) L1 (b) L3

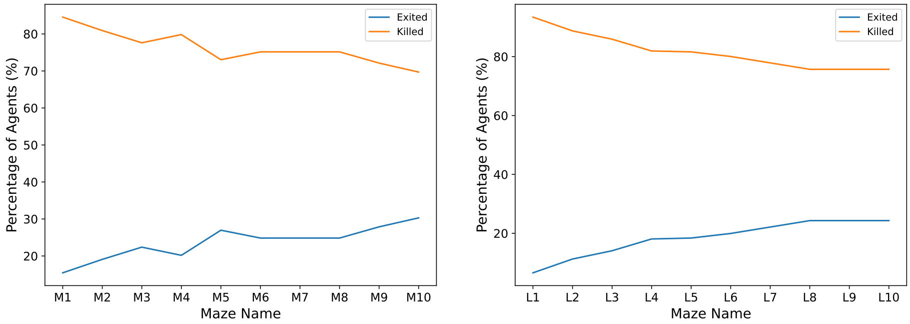
(c) L7 (d) L10

Figure 5: Examples of large-sized mazes at different values of the density parameter: L1, L3, L7, and L10.

to encounter dead ends that may cause them to turn back.

5. Results

To determine which of the three parameters visibility (η), memory (λ), and exploratory tendencies (β) had the most significant impact on agents performance and to identify their best combination, we used the amount of the remaining energy $\bar{E}_r$ of the agent upon exiting the maze as our evaluation metric. Intuitively, an agent that successfully exits the maze with a high energy level can



(a) (b)

Figure 6: Percentage of agents exiting the maze versus percentage of agents killed. Medium 6a and large 6b.

be considered more efficient than others that instead reach the exit with a low energy level. The initial amount of energy assigned to each agent was calculated by summing the weights of all the maze cells and scaling this total by a factor which, as we see in our tests, is either 0.6 or 0.7 or 0.8. The maximum simulation time was set to 1200 seconds for large mazes and 300 seconds for medium ones, based on the empirical consideration that the number of N_c cells in the large case is about 4 times the number of N_c cells in the medium case, for instances with similar density.

Table 2: Range of parameters

Symbol	**Description**	**Range of Value**
β	Exploration Tendency	$[0, 0.2, 0.4, 0.6, 0.8, 1]$
λ	Memory Size	$[0, 10, 20, 30, 40, 50, 60, 70, 80, 90, \text{inf}]$
η	Visibility	$[1, 6, 11, 16, \text{inf}]$

The values considered for the three key parameters are listed in Tab.2, and, we investigated all their possible combinations ($6 \cdot 11 \cdot 5 = 330$) each of which represents the search strategy adopted by an agent. In particular, 10 agents for each combination of η, λ and β have been tested, for a total of 3.300 agents across our experiments. Each configuration was assessed on different maze instances (see Tab.1) to comprehensively evaluate its impact on agent performance. In light of this, we have conducted two types of analyses:

1. *importance of the parameters*: assessing the significance of β, η, and λ in energy consumption;

2. *best parameter configuration*: identifying optimal parameter settings that maximize preserved energy across maze instances.

Discussion and details on each of these types of analyses are provided below. In order to better investigate the importance of these parameters (η, λ, and β), and which is the optimal parameter configuration, three different energy levels were tested, as we mentioned before 60%, 70% and 80% of the total sum of all cell weights.

5.1. Importance of the parameters

Plots displayed in figure Fig.7 report the importance of the three main parameters, namely visibility, memory size, and exploration tendency with respect to the remaining energy, analysed on the two different types of maze (M and L), and at different starting energy level. In particular, it is a percentage measure that quantifies how each of these parameters contributes to conserving or depleting the agent's energy as they navigate the maze.

We considered all possible combinations of parameters to determine the best strategies for an agent in order to escape from a labyrinth. By analyzing these combinations, we can identify which parameters are more important that others and which ones have a significant impact on the energy used during the maze exploration. Indeed, it is possible to define a collection of pairs (θ_i, E_i) where θ_i is a configuration of parameters tested, and E_i is the average energy of the agents that have explored from the maze with the configuration of parameters θ_i. Using these pairs of values, (θ_i, E_i), the parameter that most affects the energy value can be identified by using feature selection techniques, typically employed in machine learning, such as the *Maximum Relevance Minimum Redundancy* (mRMR) algorithm [20]. This algorithm was used in a similar way in [10] to study the importance of parameters of *Dynamic-IA* algorithm, that is an immune-inspired algorithm that dynamically sets the key parameters. An importance score for each parameter is returned by the mRMR algorithm, and this result can be used to determine which parameter has a major effect on the energy required to explore the maze.

By inspecting the impact of these parameters with the lower starting energy level, that is 60% of the total sum of cell weights, displayed at the top of figure Fig.7 (7a and 7b), it is possible to assert that in Medium instances, Memory Size (λ) is the most influential parameter, followed equally by Beta (β) and Visibility (η). A similar trend is observed in Large instances, where Memory Size continues to play a predominant role, but the gap between the parameters is less pronounced compared to Medium instances, indicating a more balanced distribution of importance. Specific peaks in importance show that in Medium

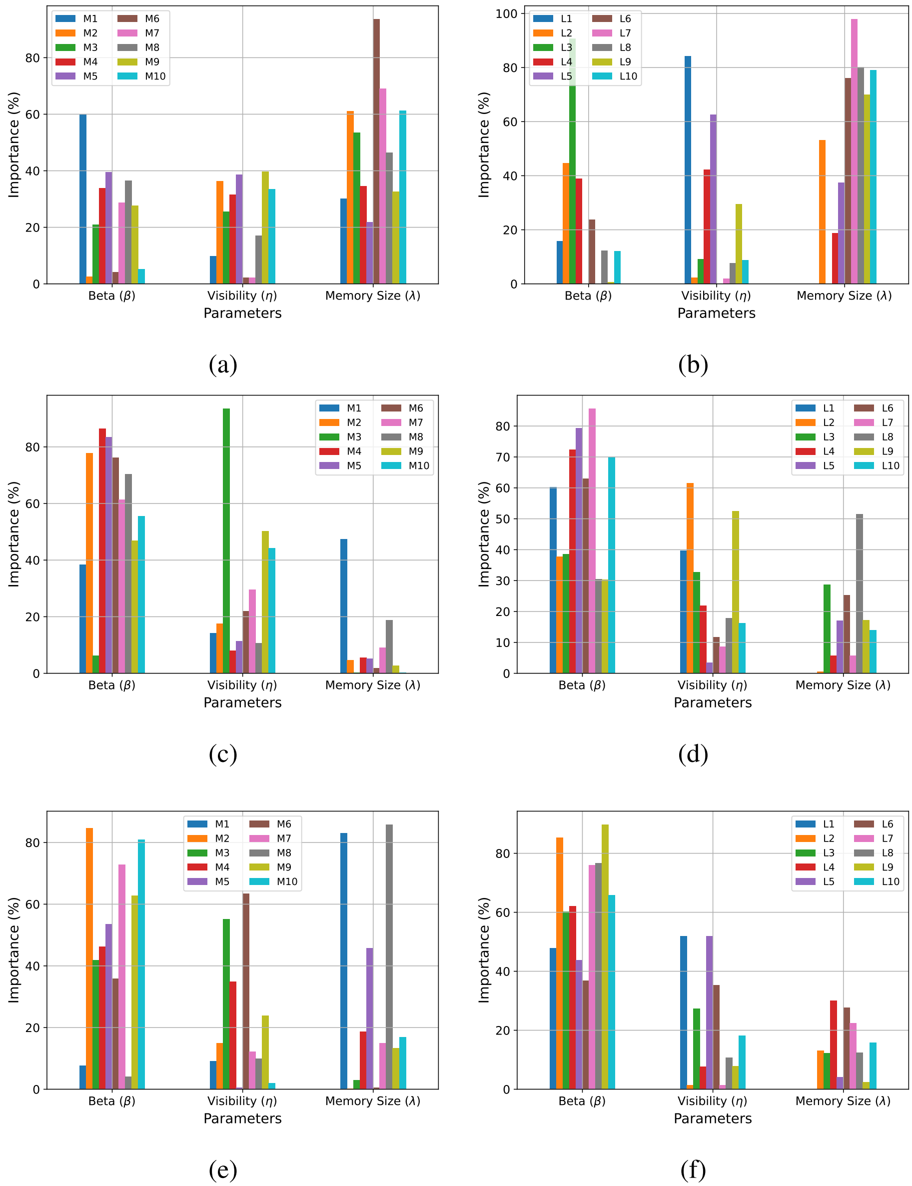


Figure 7: Importance of β, η and λ in Medium (7a, 7c, and 7e) and Large instances (7b, 7d, and 7f), investigated at different starting energy levels: 60% (top), 70% (middle), and 80% (bottom) of the total sum of cell weights.

instances, β is most influential in M1, Visibility holds slightly more importance in M9, and Memory Size is most significant in M6. Similarly, in Large instances, Beta has the greatest impact in L3, Visibility in L1, and Memory Size in L7. Notably, no importance is recorded for β in instances L5, L7, and L8, for η in L6, and for λ in L2. There is a slight positive correlation between maze density and the importance of Memory Size, particularly in Large instances. Conversely, these instances exhibit a slightly negative correlation for β. These correlations are less pronounced in Medium instances, suggesting that the complexity of Large mazes increases the importance of memory, while exploratory behavior becomes less relevant as the mazes become denser.

If, instead, we provide the agents with a slightly higher starting energy level, such as 70% of the total sum of cell weights (plots in the middle, 7c and 7d), it emerges that β plays a dominant role in energy conservation, followed by Visibility (η) and, lastly, Memory Size (λ). This trend persists in Large instances, where β remains the most influential parameter.

For what concerns possible correlations with the density of the maze, no clear patterns are evident. However, a bell-shaped distribution is observed for the importance of β in both Medium and Large instances: β's influence starts low in mazes with fewer available paths, it increases with the maze density, and then it decreases again. For Visibility in Medium instances, a slightly positive correlation appears with respect to maze density, while in Large instances, an inverse bell-shaped trend is observed: Visibility's importance decreases as the maze density increases and then rises again. For Memory Size, Medium instances follow a bell curve similar to β's, while in Large instances, Memory Size aligns with the inverted bell-shaped trend observed for Visibility. Specific peaks of importance indicate that in Medium instances, β is most important in M4, Visibility in M3, and Memory Size in M1. In Large instances, β peaks in L7, Visibility in L2, and Memory Size in L8. These results suggest that there is no evident correlation between what we call parameter importance and maze density, with importance varying based on the specific characteristics of each maze.

Finally, in the bottom plots of Fig.7 (7e and 7f) when the starting energy is higher (80% of total sum of cell weights), for both medium and large maze instances, the parameter β (exploration tendency) has the greatest impact on energy conservation. In medium instances, there appears to be a slight positive correlation with maze density, suggesting that β becomes more influential as the maze density increases. Overall, for both types of instances, visibility and memory have a much smaller impact. Memory, in particular, plays a significantly lesser role in energy conservation, especially in large instances. Regarding the correlation with maze density, neither visibility nor memory show a clear trend. Instead, their influence seems to follow a bell-shaped pattern. In less dense mazes, both visibility and memory are less important. Their significance increases in moderately dense mazes but then it decreases again in the most dense mazes. This pattern indicates that in extremely dense or sparse mazes, these parameters have a minimal impact, while their role is more pronounced in mazes of intermediate density.

Overall, it is evident that the importance of the parameters in energy conservation shifts depending on the initial energy available to the agents. Memory Size plays a more significant role in scenarios where agents start with 60% of the total sum of cell weights, but its importance decreases as the available energy increases. Conversely, the exploratory behavior β becomes more critical as agents have more initial energy. The influence of Visibility also tends to increase with available energy, but this effect is more pronounced in Medium instances, while in Large instances, the opposite trend is observed. This could suggest that in more complex environments (Large instances), as agents are provided with more energy, their reliance on direct visual cues decreases, possibly due to the increased importance of strategic exploration (as indicated by β) in navigating denser mazes. In simpler environments (Medium instances), however, visibility remains crucial as energy increases, maybe because the mazes are less complex, allowing agents to make better use of visual information.

Notably, when comparing extreme cases—such as Medium instances with 60% starting energy and Large instances with 80% starting energy—a clear inversion in parameter importance is observed. In the former, Memory Size is the most influential parameter in energy conservation, while β is the least. In the latter, the situation is reversed: β dominates in importance, while Memory Size plays a lesser role. This suggests a shift in the relevance of these parameters based on the energy context, reflecting different strategies agents may adopt in conserving energy across varying maze complexities and initial energy levels. The decreased reliance on memory in high-energy scenarios may indicate that with sufficient energy, agents are more willing to explore rather than rely on memorized paths, especially in more complex environments where exploration can lead to discovering shorter or more efficient routes.

5.2. Best parameter configuration

A second analysis carried out in this research work, as previously anticipated, is to determine the optimal parameter settings in order to maximize preserved energy. The optimal configurations obtained for the three parameters β, η and λ are reported per each instance in tables Tab.3, Tab.4 and Tab.5, performed considering 60%, 70% and 80% of the total sum of cell weights, respectively. By an overall inspection of all three tables, it is possible to claim that:

- the remaining energy of the agents appears to be highest when their initial energy is set at 70%. While one might expect the remaining energy to increase linearly with the initial energy, this is not the case, as agents with 80% of the total cell weight as initial energy actually end up with less remaining energy. This trend is consistent across both Medium and Large instances. Overall, there seems to be a slight linear correlation between the best values of remaining energy and the density of the maze. This suggests that higher maze density, which provides more available paths, encourages a more exploratory behavior in the agents, allowing them to make more efficient choices;

- in the medium mazes, the best memory values are all finite and tend to increase with the available energy, as agents need more memory to explore effectively. In contrast, in the large mazes, infinite memory values appear only in specific cases, particularly with 60% energy and 70%. As energy increases to 80%, memory values often decrease, suggesting that in larger mazes, agents rely less on memory and more on visibility or exploration strategies to navigate successfully;
- when agents have 60% of the initial energy, the best visibility values are finite, with only one instance where infinite visibility is optimal. The occurrence of infinite visibility as the best configuration tends to increase with the agents' available energy and is more common in large mazes. The correlation between infinite visibility and maze density is slightly positive, but not strong. This might suggest that as maze size and density increase, having infinite visibility becomes more advantageous, allowing agents to see as far as possible to navigate more complex environments;
- across all three energy configurations, the best β-values, which we recall denote the exploratory tendency, are generally 0.0, with a few exceptions where the best results were achieved with a beta of 0.2 and just one result with 1.0.

6. Conclusions and future directions

This research work is focused on two different objectives: (*i*) investigate how visibility and memory are nowadays within pathfinding contexts; and, afterwards, (*ii*) design and develop a new model which allows to analyse and understand how these two features affect the performance of agents in pathfinding maze tasks.

To reach the first objective, we carried out a review on research works of the last five years, reported in Sect. 2, where a short discussion on the current lacks in literature from our point of view has been also included (2.3).

Table 3: Remaining Energy for the best parameter configuration for $E(0) = 60\%$

Maze	$\bar{E}_r$	β	η	λ
M1	61.5424	0.0	16.0	50.0
M2	51.0223	0.0	11.0	70.0
M3	63.0841	0.0	16.0	70.0
M4	65.8154	0.0	16.0	60.0
M5	62.9886	0.0	inf	30.0
M6	64.6485	0.0	11.0	30.0
M7	72.6486	0.0	16.0	20.0
M8	65.6526	0.2	6.0	80.0
M9	66.6223	0.0	11.0	90.0
M10	71.4604	0.0	11.0	20.0
L1	40.1573	0.0	inf	50.0
L2	57.6738	0.0	6.0	60.0
L3	63.5465	0.0	inf	80.0
L4	63.2109	0.0	6.0	80.0
L5	64.445	0.0	16.0	inf
L6	64.7217	0.0	6.0	40.0
L7	68.9362	0.0	inf	30.0
L8	67.1214	0.0	6.0	70.0
L9	71.1575	0.0	16.0	80.0
L10	68.0527	0.0	11.0	inf

Regarding the second objective, we developed a new model through which to investigate the impact and interplay of visibility, memory, and exploratory tendencies of agents in maze-solving tasks. We analysed how variations in these characteristics affect agent performances. We recall that visibility refers to the agent's perception range within the maze, memory size denotes the agent's capacity to store crossed paths, and exploratory tendency reflects the agent's preference for exploring new or previously crossed paths. Each agent is characterized by a unique combination of these parameters that govern its navigation strategies and capabilities, aiming at finding the exit of the maze starting from a given entrance. The exploratory tendency was integrated into the probabilistic function that rules the agents' movement decisions, capturing the decision-making process of autonomous agents navigating maze environments. The remaining energy was used as the evaluation metric.

For our investigation, we conducted experiments on two types of maze instances: Medium (M) and Large (L). For each type, we used ten mazes with increasing density, where density represents the number of cells the agents can

Table 4: Remaining Energy for the best parameter configuration for $E(0) = 70\%$

Maze	$\bar{E}_r$	β	eta	λ
M1	83.6509	0.0	inf	70.0
M2	80.1404	0.0	inf	80.0
M3	95.6449	0.0	11.0	40.0
M4	97.0003	0.0	6.0	20.0
M5	90.3285	0.2	6.0	80.0
M6	92.9826	0.0	inf	80.0
M7	98.6011	0.0	11.0	40.0
M8	99.6959	0.0	inf	70.0
M9	98.2649	0.0	16.0	90.0
M10	98.6027	0.0	6.0	60.0
L1	73.7722	0.0	inf	60.0
L2	83.8487	0.0	16.0	60.0
L3	87.5786	0.0	6.0	60.0
L4	88.0305	0.0	16.0	90.0
L5	94.0121	0.0	6.0	60.0
L6	94.679	0.0	11.0	60.0
L7	89.624	0.0	inf	50.0
L8	89.7274	0.2	6.0	80.0
L9	96.78	0.0	inf	inf
L10	95.2501	0.0	inf	inf

traverse: high density corresponds in having more available paths and fewer obstacles i.e. dead ends; low density, on the contrary, represents fewer crossed paths and a greater presence of dead ends. Furthermore, three different starting energy per agent were considered for each experiment, that is 60%, 70% and 80% of the total sum of the cell weights. Two different types of analyses were conducted: (*i*) importance of the parameters, that is assessing the significance of visibility, memory, and exploratory tendency in energy conservation; and (*ii*) best parameter configuration, that is determining the optimal setting of the parameters for both types of instances.

From the investigation on the parameter importance emerges that their effect on energy conservation is strictly related with the initial energy available to the agents. In particular, memory size λ plays a more significant role in scenarios where agents start with a lower energy and its influence begins to decrease as the available energy increases. Instead, the exploratory behavior β becomes more crucial when agents have higher initial energy. Regarding the visibility parameter η, its importance tends to increase with available energy, but it appears more

Table 5: Remaining Energy for the best parameter configuration for $E(0) = 80\%$

Maze	$\bar{E}_r$	β	η	λ
M1	73.9552	0.0	11.0	60.0
M2	72.6133	0.0	11.0	70.0
M3	70.845	0.0	11.0	70.0
M4	79.6918	0.0	inf	30.0
M5	79.2694	0.0	6.0	60.0
M6	79.11	0.0	11.0	90.0
M7	81.3269	0.0	6.0	40.0
M8	81.6511	1.0	1.0	30.0
M9	82.8119	0.0	inf	80.0
M10	78.8536	0.0	inf	80.0
L1	71.5567	0.0	16.0	60.0
L2	65.2119	0.0	inf	40.0
L3	80.8658	0.0	16.0	80.0
L4	76.289	0.2	6.0	50.0
L5	74.7903	0.0	inf	10.0
L6	74.3046	0.2	6.0	30.0
L7	76.3296	0.0	6.0	30.0
L8	84.4148	0.0	16.0	20.0
L9	80.1261	0.0	inf	70.0
L10	87.283	0.0	16.0	90.0

pronounced on Medium instances than Large ones. Finally, regarding the best parameter configuration:

- for both medium and large maze instances, $\beta = 0.0$ consistently minimized energy consumption, proving what intuitively one can assume that exploring is an energy consuming task;

- about the memory size λ, on larger mazes and high energy agents explore more and rely less on memory, but they need more memory on the medium mazes because they need more memory to explore effectively the environment;

- there appears to be a positive correlation between global visibility η and maze density, i.e. as maze size and density increase, having global visibility becomes more advantageous, allowing agents to see as far as possible to navigate more complex environments.

The presented model is still in the early stages of development and it partially confirms what is claimed in existing literature that visibility and memory significantly impact agent performance. In our context, these parameters influence the agents' ability to solve mazes and identify optimal paths in unknown environments. Specifically, we used the remaining energy as a performance metric, reflecting the agents' efficiency in navigating and exiting the maze. However, preliminary data suggest that exploration tendencies may be even more crucial in determining agent performance.

This finding highlights the potential importance of incorporating exploration strategies in future enhancements of our model. Indeed, a future direction is to use a reinforcement learning approach to model an agent by considering the three parameters under consideration in this article, such as visibility, memory, and exploration tendencies. In this case, an agent will be able to automatically set the parameters of the model during its exploration and therefore change its strategy based on the payoff. To implement this agent model, we plan to use a recently developed methodology to design machine learning models, see [36, 37]. Subsequently, we plan to test this model on larger maze instances and transportation networks to scale it for real-world applications. Another possible future research direction is to use a multiobjective decision-making perspective, using a reliable metaheuristic such as immune-inspired algorithm [31], which has proven to be a robust and efficient search methodology [9] even in large datasets [22].

REFERENCES

[1] Alaguna, C., Gomez, J.: Maze benchmark for testing evolutionary algorithms. In: Proceedings of the genetic and evolutionary computation conference companion. pp. 1321–1328 (2018)

[2] Alamri, S., Alshehri, S., Alshehri, W., Alamri, H., Alaklabi, A., Alhmiedat, T.: Autonomous maze solving robotics: Algorithms and systems. International Journal of Mechanical Engineering and Robotics Research **10**(12), 668–675 (2021)

[3] Cavallaro, C., Crespi, C., Cutello, V., Pavone, M., Zito, F.: Group dynamics in memory-enhanced ant colonies: The influence of colony division on a maze navigation problem. Algorithms **17**(2), 63 (2024)

[4] Crespi, C., Fargetta, G., Pavone, M., Scollo, R.A.: An agent-based model for crowd simulation. In: Italian Workshop on Artificial Life and Evolutionary Computation. pp. 15–26. Springer (2022)

[5] Crespi, C., Fargetta, G., Pavone, M., Scollo, R.A.: An agent-based model to investigate different behaviours in a crowd simulation. In: Bioinspired Optimization Methods and Their Applications: 10th International Conference, BIOMA 2022,

Maribor, Slovenia, November 17–18, 2022, Proceedings. pp. 1–14. Springer (2022)

[6] Crespi, C., Fargetta, G., Pavone, M., Scollo, R.A., Scrimali, L.: A game theory approach for crowd evacuation modelling. In: Filipič, B., Minisci, E., Vasile, M. (eds.) Bioinspired Optimization Methods and Their Applications (BIOMA2020). Lecture Notes in Computer Science, vol. 12438, pp. 228–239. Springer, Cham (2020).

[7] Crespi, C., Scollo, R.A., Fargetta, G., Pavone, M.: A sensitivity analysis of parameters in an agent-based model for crowd simulations. Applied Soft Computing **146**, 110684 (2023)

[8] Crespi, C., Scollo, R.A., Pavone, M.: Effects of different dynamics in an ant colony optimization algorithm. In: 2020 7th International Conference on Soft Computing & Machine Intelligence (ISCMI). pp. 8–11 (2020).

[9] Cutello, V., Oliva, M., Pavone, M., Scollo, R.A.: An immune metaheuristics for large instances of the weighted feedback vertex set problem. In: 2019 IEEE Symposium Series on Computational Intelligence (SSCI). pp. 1928–1936 (2019).

[10] Cutello, V., Pavone, M., Zito, F.: Improving an immune-inspired algorithm by linear regression: A case study on network reliability. Knowledge-Based Systems **299**, 112034 (2024).

[11] Gath-Morad, M., Thrash, T., Schicker, J., Hölscher, C., Helbing, D., Aguilar Melgar, L.E.: Visibility matters during wayfinding in the vertical. Scientific reports **11**(1), 18980 (2021)

[12] Kaur, N.K.S.: A review of various maze solving algorithms based on graph theory. IJSRD **6**(12), 431–434 (2019)

[13] Kulbiej, E.: Autonomous vessels' pathfinding using visibility graph. In: 2018 Baltic Geodetic Congress (BGC Geomatics). pp. 107–111. IEEE (2018)

[14] Kumar, R., Jitoko, P., Kumar, S., Pillay, K., Prakash, P., Sagar, A., Singh, R., Mehta, U.: Maze solving robot with automated obstacle avoidance. Procedia Computer Science **105**, 57–61 (2017)

[15] Lee, W., Choi, G.H., Kim, T.w.: Visibility graph-based path-planning algorithm with quadtree representation. Applied Ocean Research **117**, 102887 (2021)

[16] Majeed, A., Lee, S.: A fast global flight path planning algorithm based on space circumscription and sparse visibility graph for unmanned aerial vehicle. Electronics **7**(12), 375 (2018)

[17] Makarov, I., Polyakov, P., Karpichev, R.: Voronoi-based path planning based on visibility and kill/death ratiotactical component. In: AIST (Supplement). pp. 129–140 (2018)

[18] Mohammed, I.M., Al-Dabagh, M.Z.N., Rashid, S.J., Isa, N.A.M.: Path discovering in maze area using mobile robot. TELKOMNIKA (Telecommunication Computing Electronics and Control) **20**(2), 416–425 (2022)

[19] Qu, J., Miwa, S., Domae, Y.: Interpretable navigation agents using attention-augmented memory. In: 2022 IEEE International Conference on Systems, Man,

and Cybernetics (SMC). pp. 2575–2582. IEEE (2022)

[20] Radovic, M., Ghalwash, M., Filipovic, N., Obradovic, Z.: Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. BMC Bioinformatics **18**(1) (Jan 2017).

[21] Schlichting, M.R., Notter, S., Fichter, W.: Long short-term memory for spatial encoding in multi-agent path planning. Journal of Guidance, Control, and Dynamics **45**(5), 952–961 (May 2022).

[22] Scollo, R.A., Spampinato, A.G., Fargetta, G., Cutello, V., Pavone, M.: Discovering entities similarities in biological networks using a hybrid immune algorithm. Informatics **10**(1) (2023).

[23] Skrynnik, A., Andreychuk, A., Nesterova, M., Yakovlev, K., Panov, A.: Learn to follow: Lifelong multi-agent pathfinding with decentralized replanning. In: PRL Workshop Series {\textendash} Bridging the Gap Between AI Planning and Reinforcement Learning (2023)

[24] Skrynnik, A., Andreychuk, A., Yakovlev, K., Panov, A.: Pathfinding in stochastic environments: learning vs planning. PeerJ Computer Science **8**, e1056 (2022)

[25] Smith, R.J., Heywood, M.I.: A model of external memory for navigation in partially observable visual reinforcement learning tasks. In: Genetic Programming: 22nd European Conference, EuroGP 2019, Held as Part of EvoStar 2019, Leipzig, Germany, April 24–26, 2019, Proceedings 22. pp. 162–177. Springer (2019)

[26] Sohn, S., De Stefani, S., Kapadia, M.: Investigating the role of memory in navigating unseen environments using cognitive maps. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018 (2018)

[27] Stern, R.: Multi-agent path finding–an overview. Artificial Intelligence: 5th RAAI Summer School, Dolgoprudny, Russia, July 4–7, 2019, Tutorial Lectures pp. 96–115 (2019)

[28] Sturtevant, N.R.: Benchmarks for grid-based pathfinding. IEEE Transactions on Computational Intelligence and AI in Games **4**(2), 144–148 (2012)

[29] Tjiharjadi, S., Razali, S., Sulaiman, H.A.: A systematic literature review of multi-agent pathfinding for maze research. Journal of Advances in Information Technology Vol **13**(4) (2022)

[30] Ulloa, C.H., Baier, J.A., Asin-Acha, R.: Multipath adaptive a*: Factors that influence performance in goal-directed navigation in unknown terrain. IEEE Access **8**, 116724–116732 (2020)

[31] Vitale, A., Di Stefano, A., Cutello, V., Pavone, M.: The influence of age assignments on the performance of immune algorithms. In: Lotfi, A., Bouchachia, H., Gegov, A., Langensiepen, C., McGinnity, M. (eds.) Advances in Computational Intelligence Systems. pp. 16–28. Springer International Publishing, Cham (2019)

[32] Wang, Y., Sartoretti, G.: Fcmnet: Full communication memory net for team-level cooperation in multi-agent systems. In: Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems. p. 1355–1363. AAMAS '22, International Foundation for Autonomous Agents and Multiagent Systems,

Richland, SC (2022)

[33] Xu, X., Yang, L., Meng, W., Cai, Q., Fu, M.: Multi-agent coverage search in unknown environments with obstacles: A survey. In: 2019 Chinese Control Conference (CCC). pp. 2317–2322. IEEE (2019)

[34] Youssefi, K.A.R., Rouhani, M.: Swarm intelligence based robotic search in unknown maze-like environments. Expert Systems with Applications **178**, 114907 (2021)

[35] Zamani, Z.: Effects of emergency department physical design elements on security, wayfinding, visibility, privacy, and efficiency and its implications on staff satisfaction and performance. HERD: Health Environments Research & Design Journal **12**(3), 72–88 (2019)

[36] Zito, F., Cutello, V., Pavone, M.: A machine learning approach to simulate gene expression and infer gene regulatory networks. Entropy **25**(8) (2023).

[37] Zito, F., Cutello, V., Pavone, M.: A general-purpose neural architecture search algorithm for building deep neural networks. In: Sevaux, M., Olteanu, A.L., Pardo, E.G., Sifaleras, A., Makboul, S. (eds.) Metaheuristics. pp. 126–141. Springer Nature Switzerland, Cham (2024)

C. CRESPI
Department of Mathematics and Computer Science
University of Catania
viale Andrea Doria 6, 95125 Catania, Italy
e-mail: `carolina.crespi@phd.unict.it`

V. CUTELLO
Department of Mathematics and Computer Science
University of Catania
viale Andrea Doria 6, 95125 Catania, Italy
e-mail: `cutello@unict.it`, `mpavone@dmi.unict.it`,
`francesco.zito@phd.unict.it`

M. PAVONE
Department of Mathematics and Computer Science
University of Catania
viale Andrea Doria 6, 95125 Catania, Italy
e-mail: `mpavone@dmi.unict.it`

F. ZITO
Department of Mathematics and Computer Science
University of Catania
viale Andrea Doria 6, 95125 Catania, Italy
e-mail: `francesco.zito@phd.unict.it`