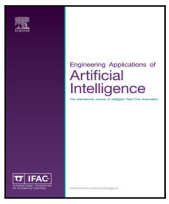
Engineering Applications of Artificial Intelligence 166 (2026) 113706

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Research paper

Cooperative patrol routing: Optimizing urban crime surveillance through multi-agent reinforcement learning

Juan Palma-Borda, Eduardo Guzmán *, María-Victoria Belmonte

Dpto. de Lenguajes y Ciencias de la Computación. E.T.S. de Ingeniería Informática. Universidad de Málaga, Málaga, Spain

ARTICLE INFO

Keywords:
Multi-agent reinforcement learning
Cooperative routing optimization
Police patrolling
Crime hotspots

ABSTRACT

Patrolling can be defined as the act of visiting locations of interest at regular intervals for either surveillance, control, protection, or monitoring purposes. The effective design of patrol strategies is a difficult and complex problem, especially in medium and large areas. The objective is to plan, in a coordinated manner, the optimal routes for a set of patrols in a given area, in order to achieve maximum coverage of the area, while also trying to minimize the number of patrols. In this paper, we propose a multi-agent reinforcement learning (MARL) model, based on a decentralized partially observable Markov decision process, to plan unpredictable patrol routes within an urban environment represented as an undirected graph. The model attempts to maximize a target function that characterizes the environment within a given time frame. Our model has been tested to optimize police patrol routes in three medium-sized districts of the city of Málaga. The aim was to maximize surveillance coverage of the most crime-prone areas, based on actual crime data in the city. To address this problem, several MARL algorithms have been studied, and among these the *Value Decomposition Proximal Policy Optimization* (VDPPO) algorithm exhibited the best performance. We also introduce a novel metric, the *coverage index*, for the evaluation of the coverage performance of the routes generated by our model. This metric is inspired by the *predictive accuracy index* (PAI), which is commonly used in criminology to detect hotspots. Using this metric, we have evaluated the model under various scenarios in which the number of agents (or patrols), their starting positions, and the level of information they can observe in the environment have been modified. Results show that the coordinated routes generated by our model achieve a coverage of more than 90% of the 3% of graph nodes with the highest crime incidence, and 65% for 20% of these nodes; 3% and 20% represent the coverage standards for police resource allocation. The source code of our implementation is available in this public repository (https://github.com/iacomlab/marl-patrol-routing). The data cannot be provided for confidentiality reasons, as they contain sensitive information.

1. Introduction

Patrolling can be defined as the act of visiting locations of interest at regular intervals for surveillance, control, protection, or monitoring purposes (Machado et al., 2002b; Guo et al., 2023). This persistent monitoring process is repetitive in nature and takes prolonged periods of time (Hari et al., 2020). It can be applied to solve a wide range of problems in virtual worlds, such as strategic games, or in the real world: smart cities, smart defense, monitoring an area with drones, patrolling water resources, identifying objects or people in dangerous situations that should be rescued by humans or robots, etc. Machado et al. (2002b), Luis et al. (2022), Soliman et al. (2023). This is a multi-agent problem, since by nature it is conveniently well suited to be shared in space and time by several agents, leading to what is often called *multi-agent patrolling* (Othmani-Guibourg et al., 2017). A patrolling strategy that schedules visits to different areas by agents, is essential for the effective execution of the patrolling task. However, the design of patrol strategies is a notoriously difficult and complex problem (Guo et al., 2023).

Patrolling strategies can be decomposed into three subtasks or problems (Samanta et al., 2022): environment clusterization or district design, resource allocation, and route design. Concerning district design, it focuses on setting different disjunct areas within the environment where patrolling takes place. This clusterization is often developed using expert knowledge in terms of number of crimes committed, population density, rapid incident response, and other relevant socio-economic factors. Resource allocation consists in the identification of the optimal quantity of resources necessary to cover each design area, including manpower, vehicles, and other equipment. Finally,

* Corresponding author.
E-mail addresses: juanpalmaborda@uma.es (J. Palma-Borda), eguzman@uma.es (E. Guzmán), mvbelmonte@uma.es (M.-V. Belmonte).

https://doi.org/10.1016/j.engappai.2025.113706
Received 14 January 2025; Received in revised form 23 October 2025; Accepted 30 December 2025
Available online 31 December 2025

route design requires planning the optimal route for each patrol, in terms of the selected area and the available resources.

Regarding the design of the route, different types of patrolling could be required depending on the characteristics of the domain Machado et al. (2002b). In this sense, the objectives may vary if the aim is to minimize response times to certain emergencies, maximize the number of hotspots visited, or minimize the cost of the route in terms of resources or delay time between two passes through the same place (Machado et al., 2002b). In addition, many situations demand that the routes be unpredictable, so that an adversary capable of observing the patrolling behavior of the agents cannot accurately predict the future locations (Guo et al., 2023), making the route design much more complex. Finally, it is important to bear in mind that there may be several patrols operating simultaneously and that the real benefit will result from the combination of these routes rather than from each route separately. Routes should therefore be planned in a coordinated manner.

In this paper, we will focus on the problem of unpredictable route design optimization in urban environments, aiming to maximize a target function without knowing the routes' final destinations and within a specific time frame. To tackle this problem, we propose a multi-agent reinforcement learning (MARL) model based on a decentralized partially observable Markov decision process. This problem is studied following a cooperative approach with a discrete space action. The agents are homogeneous, i.e., there are no additional agents other than the patrols in the model, so the goal is shared by all the agents in the model. The observability of the model can be set as a partial or total, depending on whether full information about the area to be monitored or only a portion of this information is provided to the patrol agents. Our model is designed for environments where agents may have partial information about the environment. In this particular situation, the only information available to agents is the position of their peers prior to choosing an action. For this reason, we introduce the concept of agents' *line of sight* as a way to determine the part of the surrounding environment that is known to them.

The environment in which patrols perform their activity strongly conditions this type of problem. The most complex scenario is the continuous patrolling of the terrain since the mobility space is large (Machado et al., 2002b). We follow the commonly used approach of digitizing the real terrain through a grid that is later transformed into a graph that models all the possible paths. This approach, called *skeletonization*, leads to an abstract representation of the environment that can be applied to different types of problems (Machado et al., 2002b).

To evaluate the suitability and performance of our model, we have applied it to the concrete problem of police patrolling in a real urban environment. In this case, we start with different urban areas between 1.8 km^2 and 3 km^2 and a limited number of police patrols. The allocation of resources by public administrations is a topic that is subject to constant review because there is a general shortage of resources, and their misuse directly impacts the lives of citizens. There is not always an adequate number of human resources available to carry out effective policing, which leads to having to prioritize policing strategies in some areas of the city to the detriment of others.

The contributions of this work can be summarized as follows:

- A new MARL-based model for the optimized and coordinated design of time-bound routes in urban environments without predetermining the specific nodes that will be part of the routes.
- The model can be used to estimate the optimal number of patrols, helping to avoid over-patrolling in high-tension areas and reducing the waste of public resources.
- Unlike most of the existing proposals in the literature, the model has been applied to a real urban problem: the optimization of police patrol routes in the city of Málaga, maximizing the number of high-crime areas monitored. To the best of our knowledge, this is the first approach designed to maximize the number of crimes monitored in a coordinated manner, given a set of patrols during a single shift.
- Most metrics evaluating route design are based on the concept of idleness, which is however meaningless in our domain. For this reason, to evaluate the performance of our model, we introduce a novel metric, the *coverage index*, to rate the coverage performance of the routes. We have analyzed the model performance in terms of agents' number, observability, and starting position.

The paper is organized as follows: Section 2 describes the state of the art in the field of patrol route design, focusing on those approaches that use MARL solutions. Section 3 formalizes our approach to the problem as a decentralized partially observable Markov decision process. Section 4 outlines how our model can be applied to optimize urban crime surveillance in three urban environments, also highlighting our studies on the most appropriate MARL algorithm for this problem. Section 5 presents the results of the experiments in these urban areas, and Section 6 discusses them. Finally, Section 7 presents some conclusions and future work.

2. Related works

In this section, we review the state of the art in the field of patrol routing design, focusing on those approaches that use MARL. Multi-agent solutions for patrolling areas have a well-established presence in the literature, but the introduction of new algorithms of MARL in these solutions is not widely considered due to their high training costs in terms of time and resources and their relatively recent development. In the first subsection, we introduce basic features of MARL and then go on to describe some of the most prominent works in the literature, most of which belong to the domain of multi-robot patrolling.

2.1. Multi-agent reinforcement learning

MARL stands as a relevant framework for acquiring the agent policies required to accomplish specific tasks across diverse domains, encompassing disaster response (Parker et al., 2016), autonomous vehicles (Shalev-Shwartz et al., 2016), or multiplayer games (Wijaya and Maulidevi, 2019; Samvelyan et al., 2019). To complete this task, several algorithms can be found in the literature, principally grouped into 4 families: Proximal Policy Optimization (PPO) (Schulman et al., 2017), Deep Q-Network (DQN) (Mnih et al., 2015) or Deep Deterministic Policy Gradient (DDPG) (Lillicrap et al., 2015), Advantage Actor Critic (A2C) (Mnih et al., 2016), and Trust Region Policy Optimization (TRPO) (Schulman et al., 2015a).

The selection of the most suitable algorithm for a certain problem to be solved is generally conditioned by the limitations of each model. Some algorithms may not be applicable to continuous or discrete action spaces or may not be designed for competitive or for cooperative problems. For cooperative problems, the approaches and constraints of MARL algorithms are even more important, primarily due to the necessity of coordinating various agents to achieve a certain task or several tasks, and many of them are only achievable when agents perform various tasks in a coordinated manner. This coordination of agents is not trivial, especially in problems with homogeneous agents, where there are no associated roles, which would allow decomposing the problem into subtasks that agents in each role could learn. Finally, some recent approaches have introduced MARL algorithms for solving problems with incomplete information, as a way to train agents in particular environments such as natural disasters. These approaches perform well compared to usual solving techniques, especially if they are complemented with expert knowledge of the subject (Lee and Lee, 2021).

2.2. Patrol route design

The route design problem focuses on planning the path of one or more patrols, depending on the selected area of action and the available resources. This task aims to achieve maximum area coverage at minimal cost in a coordinated manner between assigned patrols, thus giving coordination an essential role. In addition, in some cases it may be necessary to introduce a certain degree of randomness Yin et al. (2012).

In this context, classical routing problems such as the *Covering Salesman Problem* (CSP) (Current and Schilling, 1989), alongside more recent formulations like the *Budget Constrained Traveling Salesman Problem* (BC-TSP) (Mak et al., 2024), have proven valuable for modeling patrol scenarios under realistic operational constraints. The CSP aims to minimize the total cost of a tour while ensuring that all locations are either directly visited or lie within a predefined coverage distance of visited nodes—making it well-suited for surveillance applications where complete visitation is unnecessary, but effective coverage is essential. Several studies have applied CSP-based formulations in related contexts. For example, Salari and Naji-Azimi (2012), Salari et al. (2015) propose integer programming and ant colony optimization approaches, while Oliveira et al. (2015) explore multi-vehicle covering tours for urban patrolling. On the other hand, the BC-TSP, as a more contemporary model, focuses on maximizing the number or value of visited nodes within a limited travel budget (e.g., time or energy). It addresses modern challenges where resource constraints are critical. Together, these formulations provide flexible and complementary frameworks that extend traditional route planning by explicitly incorporating pragmatic limitations such as constrained budgets and coverage requirements.

Notably, one of the main areas of application of the patrol route design problem is robotics, more specifically multi-robot patrolling, which is perhaps the most explored field in terms of the study of routes within a certain area. In the literature, various techniques for planning these robots using centralized or distributed coordination can be found (Huang et al., 2019). Centralized coordination includes those methods managed by a central coordinator (Machado et al., 2002a; Almeida et al., 2003; Othmani-Guibourg et al., 2017) or cyclic strategies (Chevaleyre, 2004) where robots are arranged on a closed path. For instance, Pasqualetti et al. (2012a) used this technique to generate a route to visit all important locations. Partition-based strategy (Chevaleyre, 2004) is another centralized coordination strategy, where the area is divided into disjoint regions, similarly to the environment clusterization but in smaller regions. Using this strategy, each agent covers one region of the divided area, having as many regions as agents. Portugal and Rocha (2010) establishes a *multilevel subgraph patrolling* (MSP) algorithm by modifying the partitioning phase, thus achieving a reduction in redundant work compared to cyclic approach. On another note, Sea et al. (2018) uses k-means clustering to divide the regions and find the shortest path using a simulated annealing algorithm. Lastly, Stranders et al. (2013) presents several environments based on the elaboration of patrol routes on graphs using both multi- and single-agent approaches. This divides the entire graph into different groups called atomic clusters, for which a solution is sought that determines the travel time and establishes the entry and exit points of each cluster.

Distributed coordination includes three groups: (1) Reactive approaches (Machado et al., 2002a), where each agent selects the nodes with the most idleness (this metric quantifies the duration for which a node remains unvisited) aiming to reduce it. For example, Yan and Zhang (2016) proposes a distributed algorithm where each robot estimates the idleness of each vertex using information shared with the other agents, without the need for centralized planning or control, and Chen et al. (2015) develops a new police planning strategy that mixes Bayesian and ant colony methods and tries to minimize the average time lag between consecutive visits to hotpots. (2) Auction-based approaches implement negotiation mechanisms among the robots. For example, Hwang et al. (2009) explores the coordination of patrols and develops a system where each patrol selects the points to visit through an auction system. (3) Learning-based approaches, where the agents adapt their strategies to the environment having only partial information (Santana et al., 2004; Othmani-Guibourg et al., 2018, 2019; Portugal et al., 2013; Portugal and Rocha, 2016; Guo et al., 2023). Santana et al. (2004) models this problem as a semi-Markov decision problem using reinforcement learning as a way to coordinate all agents behaviors; Othmani-Guibourg et al. (2018, 2019) proposes an LSTM (*long-short term memory*) architecture using the *heuristic pathfinder cognitive coordinated* strategy combined with small randomness to train the agents; Portugal et al. (2013), Portugal and Rocha (2016) propose a *concurrent Bayesian learning strategy* (CBLS) and develop a probabilistic model to represent the availability of robots to move to a neighboring point from their current location. Additionally, they also adopt a reward-based technique to influence the robots' future movements during patrolling; Guo et al. (2023) have recently addressed another key factor, i.e., balancing unpredictability and efficiency along with a MARL approach, where the agents are trained using HAPPO (*heterogeneous-agent proximal policy optimization*); lastly, Chen et al. (2023) establishes a distributed model combined with the cyclic strategy often found in centralized approaches. In this model, agents select their routes based on the crime values assigned to the roads in question.

Table 1 summarizes the main features of the aforementioned works, highlighting the differentiating characteristics of our approach, which are detailed in the last row of the table. The first column of this table shows the reference of each proposal; the second one summarizes the main purpose of the proposal; and the third shows the type of agent for which the proposal was developed, robot or human. The fourth column indicates whether the environment used is real or artificial. In this sense, if an environment used in the simulation is designated as an existing city center, neighborhood, institution, building, or vehicle, this environment is labeled as real. The model behind the proposal is described in the fifth column, and column six registers if coordination between agents is centralized or distributed. The seventh column classifies the size of the studied area. The term "Tiny" is used to describe a street or a floor of a building; "Small" to describe a campus, a housing estate, or a single building; "Medium" to describe an entire neighborhood or group of neighborhoods; and "Large" to characterize a medium-sized city or district of a metropolis. The last two columns indicate whether the nodes or points to be monitored are predetermined within the selected area, or whether each node is of equal importance to be surveilled, and the metrics used to evaluate the performance of the proposal.

Next, we outline the main differences between our approach and those presented in the table. These differences highlight how our work diverges in terms of assumptions and objectives, addressing challenges in urban patrolling for human patrols that have been overlooked or simplified in related studies, primarily due to the use of robots instead of humans as agents in the model.

- Most of the approaches (Pasqualetti et al., 2012a,b; Stranders et al., 2013; Portugal and Rocha, 2016; Othmani-Guibourg et al., 2017, 2018, 2019) focus on reducing the idleness of all nodes in the environment or a preselection of them without prioritizing visits to more problematic areas or those of greater interest. For example, to achieve this goal, Othmani-Guibourg et al. (2017, 2018, 2019) generates a multi-agent LSTM model on different types of graphs, while Portugal et al. (2013), Portugal and Rocha (2016) apply Bayesian learning in order to achieve the same goal. In our work, nodes are not preselected, neither is it feasible to cover all of them within a single shift, making this metric inapplicable. Consequently, our objective is to identify the optimal route that ensures comprehensive coverage of the most interesting areas, which differs fundamentally from classical models like the CSP or the BC-TSP. Unlike these *Traveling Salesman Problem*

Table 1
Main features of the literature review on patrol route design.

Reference	Objective	Agent	Environment	Model	Coordination	Area size	Preselected or homogeneous nodes	Metrics
Machado et al. (2002a)	Minimize revisit time of all nodes	—	Artificial	Algorithms for multi-agent systems	Both	—	Yes	Idleness and exploration time
Almeida et al. (2003)	Minimize revisit time of all nodes	–	Artificial	Control algorithm	Centralized	–	Yes	Idleness
Santana et al. (2004)	Minimize revisit time of all nodes	–	Artificial	MARL	Distributed	–	Yes	Idleness
Hwang et al. (2009)	Minimize total length path of all agents and minimize revisit time of patrol points	Robot	Artificial	Cooperative auction system	Distributed	—	Yes	MINIMAX
Portugal and Rocha (2010)	Minimize revisit time of selected positions	Robot	Artificial	MSP algorithm	Centralized	Tiny	Yes	Average Node Frequency
Pasqualetti et al. (2012a)	Minimize revisit time of selected positions	Robot	Real	Control algorithm	Centralized	Small	Yes	—
Pasqualetti et al. (2012b)	Minimize revisit time of selected positions	Robot	Real	Control algorithm	Centralized	Small	Yes	–
Stranders et al. (2013)	Minimize revisit time of all nodes	Robot	Artificial	Divide and conquer and greedy algorithms	Centralized	—	Yes	Reward similar to idleness
Portugal et al. (2013)	Minimize revisit time of selected positions	Robot	Artificial	Bayesian learning	Distributed	Tiny	Yes	Idleness
Chen et al. (2015)	Minimize revisit time of hotspots	Human	Real	Ant colony	Distributed	Large	Yes	Idleness
Yan and Zhang (2016)	Minimize revisit time to selected positions	Robot	Real	Distributed algorithm	Distributed	Tiny	Yes	Idleness
Portugal and Rocha (2016)	Minimize revisit time of selected positions	Robot	Real	Bayesian learning	Distributed	Tiny	Yes	Idleness
Othmani-Guibourg et al. (2017)	Minimize revisit time of all nodes	Robot	Artificial	LSTM	Centralized	–	Yes	Idleness
Othmani-Guibourg et al. (2018)	Minimize revisit time of all nodes	Robot	Artificial	LSTM	Distributed	–	Yes	Idleness
Othmani-Guibourg et al. (2019)	Minimize revisit time of all nodes	Robot	Artificial	LSTM	Distributed	–	Yes	Idleness
Guo et al. (2023)	Balance between unpredictability in the routes and minimizing revisit time of selected positions	Robot	Artificial	MARL	Distributed	—	Yes	Idleness and time entropy
Chen et al. (2023)	Maximize coverage of hotspot roads and total roads	Human	Artificial	Deep reinforcement learning	Distributed	Small	No	Coverage
Our work	Maximize crime surveillance during a single shift	Human	Real	MARL	Distributed	Medium	No	Coverage index

(TSP) variants, our model is a decentralized, partially observable MARL system, where agents learn policies through interaction rather than explicit route computation. There is no requirement for full graph traversal, cyclic tours, or predefined return points, agents optimize behavior based on evolving rewards and local observations. Furthermore, while BC-TSP focuses on maximizing collected rewards within a budget and CSP emphasizes covering all or nearby nodes in a cycle, our method aims to maximize effective coverage during a working shift, where it is not possible to perform repeated patrols over a very small set of nodes or to visit all of them effectively. This makes it more adaptable to real-world policing scenarios, where patrols do not follow fixed cycles or return to a central depot.

- Many works are designed for continuous monitoring of an area, regardless of work shifts or a finite time frame (Stranders et al., 2013; Yan and Zhang, 2016). In our approach, human patrols are carried out in 8-hour shifts and typically involve the monitoring of medium-sized areas. While many of the aforementioned techniques can be easily adapted, it is not feasible to simultaneously monitor an area carefully and repeatedly survey all the points of a graph whose nodes are sometimes hundreds of meters apart. Chen et al. (2015), however, focus on human patrols and a real city with the aim of establishing daily patrol routes that minimize the global average idleness of hotspots in a real environment. This work also does not yet address the joint restriction of finite time and effective surveillance time of an area, which involves being present at a specific node for a given period of time.
- Other works focus on the surveillance of smaller environments, such as the floor of a building (e.g., Portugal et al., 2013; Portugal and Rocha, 2016; Yan and Zhang, 2016), or are tested in artificial environments that could not be completely translated into a real case (e.g., Machado et al., 2002a; Almeida et al., 2003; Santana et al., 2004; Hwang et al., 2009; Chen et al., 2023). These approaches are therefore not directly comparable with the surveillance of a substantial part of a city. In addition, as is often the case, they do not include a differentiating factor between nodes or vertices in the environment. This differentiation is emphasized in our work by aiming to cover certain points while ignoring others with little interest in coverage.

Finally, to sum up, our proposal focuses on maximizing crime surveillance without the preselection of nodes to visit within a single work shift of human patrols. In our view, for optimal surveillance work, patrols should spend a relative amount of time at each location. In addition, they rarely can complete multiple cycles in an extensive area within the same working day. In this sense, Chen et al. (2015) confirms that in order to achieve a significant decrease in the average node idleness, it is necessary to significantly increase the number of patrols in the system. Most other approaches are more suitable for other types of vigilance where the patrols have a smaller area to cover or when the number of patrols is enough to cover a determined area really well.

By assuming that surveillance time will not be infinite and that the generated route is not intended to be cyclical, discussing the idleness of the nodes becomes less relevant. This is because some nodes will not be visited due to time constraints or lack of interest, and certain conflict points may not warrant visitation on every occasion if they are too isolated from the other hotspots. For this reason, a new metric has been developed to measure the effectiveness of a generated route in terms of area coverage.

3. Methodology

3.1. Definition of the problem

The patrolling problem in an urban environment can be classified as a *decentralized partially observable Markov decision process* (dec-POMDP), also generalizable to *multi-agent markov decision processes* (MMDPs), depending on whether the agents are allowed to know the entire state of the environment in their observations. In our model, the problem is formulated as a dec-POMDP with a tuple $< I, S, A, \Gamma, R, \Omega, O, \gamma >$ (Oliehoek et al., 2016), where $I = \{a_1, a_2, \dots, a_N\}$ is the society of N patrolling agents; S is the representation of the environment through which agents are able to move, represented as a set of states; A is a function modeling the set of actions an agent can perform; Γ is the set of conditional transitions between states, defined as $\Gamma(s_{t+1}|s_t, a_t)$; R is the reward function, defined as $R(s_t, a_t)$; Ω is the set of observations; O is the probability of these observations; and γ is the discount factor.

3.2. The environment representation

The monitoring area of our proposal could be an urban environment that is initially transformed into a grid and subsequently converted into an undirected graph. The urban space is thus parceled into nodes, connected among them in terms of their walkability. This approach is one of the most common options in the field of spatial representation of urban environments (Devia and Weber, 2013; Birks et al., 2008). Consequently, each node will represent a cell of the grid, with edges connecting neighboring cells that share a common road. The use of a grid as the basis of our graph also implies that the distance between the nodes in the real environment is the same, simplifying any consideration of choosing one path over another.

Formally, the environment graph can be represented as a tuple $S =< V, E, C, \rho, \sigma >$, where $V = \{v_1, v_2, \dots, v_M\}$ is the set of the

M nodes of the graph. $E : V \times V \rightarrow \{0,1\}$ is the *mobility function*, which is also commutative, i.e., $E(v_i, v_j) = E(v_j, v_i)$. $E(v_i, v_j) = 1$ if an agent can transit from the node v_i to the node v_j and vice versa; otherwise, $E(v_i, v_j) = 0$. C is the subset of nodes, $C \subseteq V$, that have to be monitored.[1] $\rho_t : I \rightarrow V$ is the *location function* relating each agent with their position at time t, and $\sigma : V \rightarrow \mathbb{R}$ a *target function*. The higher the value of the target function for a node, the more important it is for agents to transit through it. Therefore, the goal of the model will be to maximize the coverage of those nodes with higher target values.

With regard to the temporal constraints of the environment, the objective is to reproduce a shift of a group of human patrols (represented by the agents) at each episode t, $1 \leq t \leq T$, where T is the total number of episodes of the simulation. Therefore, there is no penalty target at the conclusion of the episode t, i.e., agents are not rewarded for going faster. The number of steps that an agent can take in the environment is determined by the real surface that each node represents and by the time that we consider to be dedicated to monitoring each one of them. So, the maximum number of steps in the environment is something that is not fixed and depends on the area to be patrolled. We believe that for surveillance to be effective, it is necessary that a patrol remains in a given area for a certain period of time.

3.3. Actions

The actions in the model consist mainly of the movements of the agents. An agent can only move to any of the nodes directly connected to the one on which it is. In addition, it can remain on the same node. Let $\delta : V \times V \rightarrow \mathbb{N}^+$ be a *distance function* between nodes that can be denoted as:

$$\delta(v_i, v_j) = \begin{cases} 0 & : v_i = v_j \\ 1 & : E(v_i, v_j) = 1 \\ 1 + \min_{v_k \in V, v_k \neq v_i} \delta(v_k, v_j) & : E(v_i, v_j) = 0 \wedge \exists v_k \in V, \\ & \quad E(v_i, v_k) = 1 \wedge v_i \neq v_k \end{cases} \tag{1}$$

δ measures the distance between two nodes of the graph. If the nodes are adjacent, its value is 1; otherwise, this function will calculate the shortest distance between these nodes. The action function of the model, $A : I \times V \rightarrow V$, can also be denoted, as can be seen in Eq. (2).

$$A(a_i, v_j) = v_k, \delta(v_j, v_k) \leq 1 \tag{2}$$

The action function represents the movements that an agent can make. Thus, the agent can remain at its current node or move only to neighboring nodes, i.e., those with which it is connected via a path.

3.4. Observations

Our model is designed for environments where agents may have partial and individual information about the environment, i.e., we focus on problems with partially observable environments. Observations determine the information available to an agent when making a decision. The existence of a single type of agent in our model entails that all of them share the same design and decision rules. However, at any given moment, each agent may receive different information depending on its location and the local environmental state it perceives. The reason to withhold environmental information from patrols is largely in line with the argument put forward by Santana et al. (2004). Firstly, it simplifies the problem, accelerating the training process and facilitating the convergence of the model. Secondly, it sets up the environment more realistically as agents may not have complete information about its state. Thirdly, restricting information allows subsequent modifications of the training environment without increasing the size of the observation space. This makes the model more scalable and flexible. Moreover, partial observability reflects the reality of patrol scenarios, where agents cannot access complete information about the environment. This limitation directly affects how they learn to make decisions and coordinate effectively. For these reasons, we introduce the concept of agents' *line of sight* as a way to determine what information of the environment state is known by the patrols. As will be seen, this line of sight of the agents affects their training and thus the model performance in the resolution of the problem. A comparison of results has been sought by providing different numbers of patrols with different amounts of information.

$$\begin{aligned} \Omega_t^i = < & \{\rho_t(a_1), \rho_t(a_2), \ldots, \rho_t(a_N)\}, \\ & \{v_t(v_1), v_t(v_2), \ldots, v_t(v_M)\}, \\ & \{\sigma(v_1), \sigma(v_2), \ldots, \sigma(v_M)\} > \end{aligned} \tag{3}$$

The information contained in the observations, Ω_t^i, of agent i at time t (Eq. (3)), can be divided into three parts: (1) The locations, $\rho_t : I \rightarrow V$, of N patrols in the model, represented by the node identifier where they are located at time t. (2) The number of visits agents have made to each one of the M nodes in the line of sight, at time t, where $v_t : V \rightarrow \mathbb{N}^+$ is the function that computes this information. (3) The value of the target function, σ, of each node. The values calculated from v_t are significant because the goal is to ensure that patrols monitor areas at a higher target value, with the intention that the value of revisiting an area in the short term will decrease once this area has been visited.

3.5. Rewards

In our model, the objective function can represent any value calculated for a given cell. Aggregated crime data is utilized as this value, with the goal of ensuring that the reward function reflects statistical measures of criminal activity. This enables agents to prioritize patrolling areas based on the intensity and recurrence of incidents rather than distributing their efforts uniformly across the environment. However, designing a reward function that effectively guides agents to maximize coverage of high-crime areas is probably the main challenge. Unlike conventional reinforcement learning scenarios, where agents receive clearly defined signals for success or failure, this problem lacks an obvious, reliable indicator of desirable behavior. Agents must learn to balance between focusing on high-value zones and exploring less-frequented regions, thereby ensuring comprehensive and significant coverage.

This challenge situates our task within the realm of cooperative problem solving, where the outcome of an action depends not only on the individual agent but also on the behavior of the team. The reward is shared and shaped by group dynamics, and agents must coordinate their actions to maximize collective utility. Developing a reward function that enables this coordination while avoiding behaviors such as overcrowding high-value nodes or ignoring isolated yet important areas has been one of the key difficulties in shaping a patrol policy aligned with real-world surveillance needs.

For this reason, we explored different approaches to the reward function in terms of its effect on agent training. First, we considered the sum of the objective values of all nodes monitored by the agents at each point in time, without taking into account the number of visits to a node. This resulted in all agents moving to a node with a high objective value and remaining there, since it was the node that gave them the highest reward. The goal of the problem, if we were to translate it to the real world, would be to coordinate the agents to search for the highest target value nodes in the environment rather than the optimal routes. Then, we modified that reward value by dividing it by the number of visits. This resulted in all agents moving to an area (a group of neighbor nodes) with a high target value and staying there. The two problems with this behavior are, firstly, that they may not mind passing many

[1] It should be noted that there is no preselection of nodes within the designated area. However, auxiliary nodes that do not belong to the selected area are utilized to facilitate the movements of agents.

Table 2
Model parameters.

Input parameters			
Parameter	Meaning	Domain	Value
Zone ID	Identifier of the zone	$\mathbb{N}$	{3, 9, 10}
# agents	Agents to be deployed	$\mathbb{N}$	{2, 5, 10}
Line of sight	Size of observation box	$\mathbb{N}$	{1, 3, 6}
Starting position	Initial distribution of the agents	Enum	{Random, Best}
RL algorithm	Algorithm chosen to train the agents	Enum	{VDPPO}
Reward parameters			
Parameter	Meaning	Domain	Value
η	Normalization factor	$\mathbb{R}$	10
ϕ	Relevance threshold	$\mathbb{R}$	10
ν	Coverage factor	$\mathbb{R}$	{-25, -10}
α^-	Exploration reward	$\mathbb{R}$	{5, 10}
α^+	Optimal exploration reward	$\mathbb{R}$	{50, 100}

nodes outside the area to reach their target faster and, secondly, that they may not cover the most isolated hotspots in the model, resulting in certain areas being overwatched. Finally, the (cooperative) reward function we have used in our model is denoted in Eq. (4). The function setting the final reward for taking an action at a particular moment will be the sum of the rewards of all agents at that moment, plus the added reward for the individual agent. This is aimed at minimizing the occurrence of lazy agents (those who wait for others to perform beneficial actions) among the group of agents.

$$R_t(a_i) = R'_t(a_i) + \sum_{1 \le j \le N} R'_t(a_j) \tag{4}$$

An individual reward, defined as $R'_t(a_i)$ (Eq. (5)), incorporates mechanisms to encourage efficient and focused patrolling. First, it includes a penalty ν, the *coverage factor*, for agents that traverse nodes either outside the surveillance area or of low interest, specifically, those whose reward values are below 1. Additionally, to discourage redundant visits, the reward function lowers the contribution of each successive visit, using the value produced by the target function σ, divided by the number of visits υ_t the node has received. This means that repeated visits yield diminishing rewards, discouraging agents from revisiting the same location unless strategically justified. To balance the overall reward structure after introducing these penalties, a *normalization factor* η was added to the equation to partially harmonize rewards and penalties. Altogether, this design promotes coordinated and non-repetitive exploration, guiding agents to prioritize unvisited or high-crime areas within the limited duration of each episode. During each episode, there is no mechanism for the relevance of the nodes to increase again. Although this concept could be considered, our model is based on eight-hour shifts. We understand that, at the end of each shift, the model's values would be reset, returning to their original state.

Finally, an *exploration function*, $\tau_t : V \to \mathbb{N}$ (Eq. (6)), was defined to reward patrolling more nodes in the model and to allow a better exploration phase during training. In that equation, α^+ is a reward parameter in our model, granted when the target value of a node exceeds a relevance threshold ϕ, a parameter which determines the important nodes in the whole area. Conversely, if the relevance threshold is not met, α^- is given. These rewards are only granted when a node is visited for the first time without another patrol being present. As noted in Eq. (5), if the exploration reward is too strong, it may lead agents to over-prioritize novel areas at the expense of critical zones. To mitigate this issue, the reward parameters α^+ and α^- were adjusted to strike a balance between effective exploration and sustained coverage of high-crime areas. This last addition is crucial for training since, as has been pointed out in other studies such as (Zhang et al., 2021), without it agents tend to stay in local minima of the problem and fail to identify targets that are difficult to reach.

$$R'_t(a_i) = \begin{cases} \nu & : \rho_t(a_i) \notin C \\ \frac{\sigma(\rho_t(a_i))}{\eta[\upsilon(\rho_t(a_i))]} + \tau_t(\rho_t(a_i)) & : \rho_t(a_i) \in C \wedge \frac{\sigma(\rho_t(a_i))}{\eta[\upsilon(\rho_t(a_i))]} \ge 1 \\ \frac{\sigma(\rho_t(a_i))}{\eta[\upsilon(\rho_t(a_i))]} + \tau_t(\rho_t(a_i)) + \frac{\nu}{2} & : \rho_t(a_i) \in C \wedge \frac{\sigma(\rho_t(a_i))}{\eta[\upsilon(\rho_t(a_i))]} < 1 \end{cases} \tag{5}$$

$$\tau_t(a_i) = \begin{cases} 0 & : \upsilon(\rho_t(a_i)) \neq 1 \\ \alpha^+ & : \upsilon(\rho_t(a_i)) = 1 \wedge \sigma(\rho_t(a_i)) \ge \phi \\ \alpha^- & : \upsilon(\rho_t(a_i)) = 1 \wedge \sigma(\rho_t(a_i)) < \phi \end{cases} \tag{6}$$

3.6. Model parameters

The specific parameters used in the training of the model can be seen in Table 2. Those not labeled as input parameters are calibrated by maximizing the reward function and minimizing the loss function. The first part of the table lists the general input parameters of the model, more specifically, the number of agents, the line of sight of each agent, and the strategy used to place each agent in the first node of the model that is being trained.

The second section of the table shows all those parameters related to the reward function that have been explained in the previous section. Finally, note that the last column of the table contains the values used during the evaluation of the model, as will be explained in Section 5.

4. Urban crime surveillance optimization

The allocation of resources by public administrations is a matter of ongoing concern due to the lack of resources, and their misuse may directly impact the lives of citizens. In the case of the police, there is not always an adequate number of officers or resources available to carry out effective surveillance. Thus, it is necessary to prioritize surveillance in different areas of the city over others. In addition, it is also important to note in this domain that patrol routes are slightly different each day (Yin et al., 2012; Sherman et al., 2014). This is intended to prevent offenders from identifying surveillance patterns that they would learn to avoid in the real world.

Through this section, we particularize our model to the case of urban crime surveillance, and more concretely, to crime prevention in three areas of the city of Málaga (Spain) selected based on data availability. Two of these areas are similar in terms of crime rates and size, while the third area is larger and has a significantly lower crime density due to its size. The main goal is to collectively monitor the cells with the highest crime rate within that urban environment by setting different routes that complement each other for all police patrols. Fig. 1 illustrates a flowchart describing all the stages of optimizing urban crime surveillance and thus how the methodology is applied to this problem. The input data to the flow are the geographical coordinates of the city's urban environment and information about its

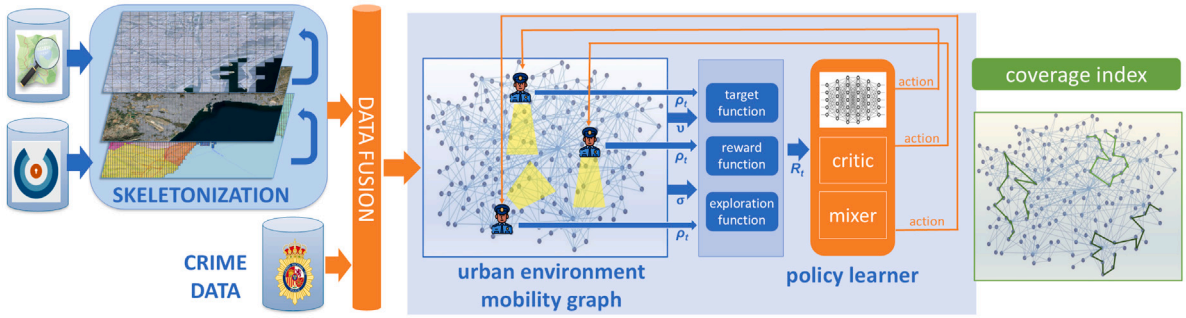


Fig. 1. Flowchart of the urban crime surveillance with our model.

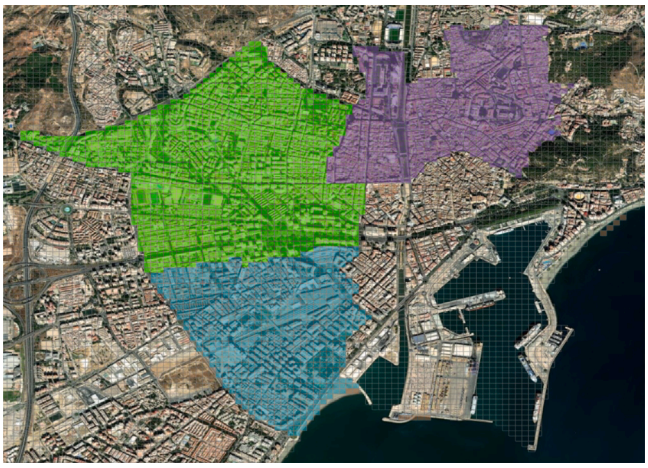


Fig. 2. Urban areas selected for patrol routing, Málaga, Spain: zones 3 (blue), 9 (purple), and 10 (green).

streets. This information is combined, and through a skeletonization process, the urban environment is digitized into a grid on which a graph showing the mobility between its cells is also overlaid. Next, the urban environment is then enriched with crime data that will shape the target function. In the second stage, the MARL algorithm (in our case, VDPPO, i.e., the one with the best performance) is used to learn the agents' policy. The results of the agent's movements in the environment are evaluated through the target and exploration functions, generating the reward that feeds the policy learner iteratively until the learning process converges. As a result of the whole process, the surveillance routes of the agents and the coverage index are generated.

4.1. The urban environment

In order to create the environment, several input data have been used. Firstly, we collect information about the roads in the city of Málaga. This information is extracted and combined from two datasets: the road map of Andalusia, Spain, obtained from OpenStreetMap (OpenStreetMap contributors, 2017) and the list of the information on the roads of Málaga on the open data portal of Málaga City Council (Málaga city council, 2024).

Secondly, we collected various datasets that contain information on crime corresponding to citizen reports. These datasets were provided by the Spanish National Police Force and include all the crimes occurred in Málaga between 2010 and 2018, representing a total of 376,737 cases. These crime reports are manually introduced by the police officers and thus need to be geolocated in terms of the street name where they occurred. Eventually, only 304,125 crimes were successfully geolocated and used in this study.

After collecting and combining all the data, a grid was overlaid on the city map. All the information corresponding to streets and crimes was then added to each cell. In our case, the area of each cell corresponds to 2500 m^2, 50 m per side, and connections between cells correspond to the roads between them. These cell dimensions were chosen in this case to provide a realistic approximation of the movement of an agent, preventing it from taking too long to move from one cell to another. Additionally, this size contributes to a better and more realistic distribution of criminal activity, which with larger grid sizes may be condensed in a single node. Note also that, to our knowledge, our grid size is one of the smallest that can be found in the literature of crime prediction or hotspots analysis. For example, Kadar et al. (2019) and Rummens et al. (2017) use a cell size of 200 m per side, Adepeju et al. (2016) 250 m per side, and Lee et al. (2020) 152 m per side.

From the grid, the area of the city to be patrolled had to be selected. In this case, we focused on three urban areas in the city center (Fig. 2). These three zones (3, 9, and 10) correspond to clusters of adjacent neighborhoods rather than official administrative or police districts, which in Málaga are considerably larger and mainly serve organizational purposes. The boundaries were defined based on spatial continuity and local crime density to ensure that each area represents a coherent and walkable portion of the city, consistent with the scale of a typical work shift conducted on foot. This configuration enhances the model's transferability to other urban contexts, as the selected areas exemplify medium-sized, mixed-use, and primarily residential environments common in many European cities, while also providing meaningful case studies to analyze how patrol strategies adapt to variations in area size, population density, and crime distribution. Given these considerations, it is important to outline the specific characteristics of each zone, as their distinct spatial and criminological features directly influence patrol behavior and the learning dynamics of the model.

Each of these areas differs slightly in size and crime density. The first two areas have an approximate size of 1.8 km^2 and a crime density of 1999 and 1672 offenses per year per km^2 respectively, while the third area is 3 km^2 with a crime density of 1080 offenses per year and per km^2. The areas were chosen because they are all residential, although there are differences in the density of crime, as well as in its typology, as can be seen in Fig. 3. The histogram shows that the most common criminal activities in the city are drug-related crimes and thefts, which are most frequently observed in all three zones. Zone 3 encompasses the city's main railway station, which has become a focus for drug dealing and theft, largely due to the high volume of passengers in the area. This area has the highest crime rate of the three areas surveyed. In contrast, zone 10 is considerably larger than the other two and is mainly residential, with no discernible presence of tourists. The size of the zone translates into crime levels that make it comparable to the other two, and the zone is notable for the total number of reported burglaries. Finally, zone 9, comparable in size to zone 3, is also residential, like zone 10, but is located closer to the city center,

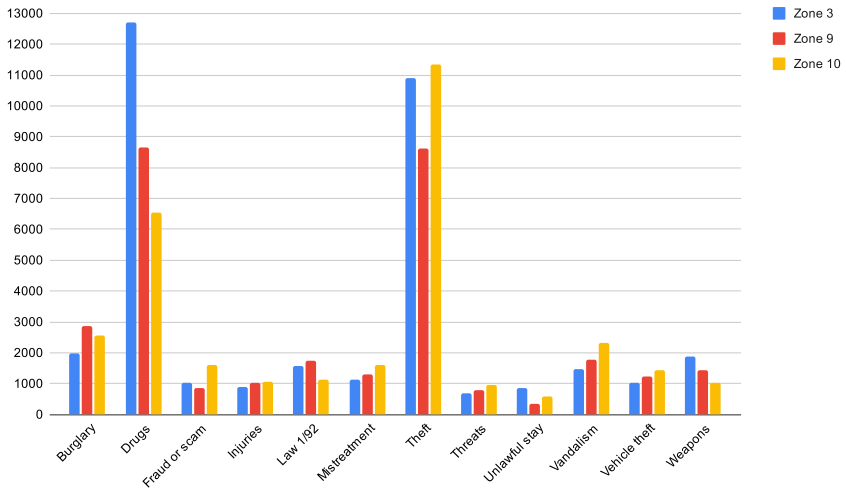


Fig. 3. Number of crimes by typology in each studied area.

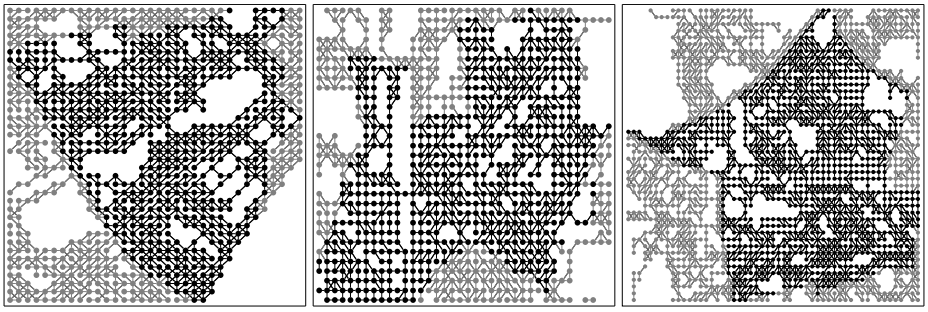


Fig. 4. Graphs generated from the selected areas in Fig. 2: zone 3 (left), 9 (center), and 10 (right).

resulting in a lower but comparable crime rate to zone 3. Nevertheless, it is not the main hotspot for any type of criminal activity. Although crime types vary across zones, our model currently uses aggregate crime density as the target function to remain general-purpose and consistent with local police objectives. However, the framework can be extended to incorporate weighted crime categories or specialized patrol policies, and it also allows for focusing on a single crime type if desired.

Additionally, to ensure that the selection of a particular area has not disrupted the usual flow of people by excluding any connections between the cells in the area that is crossed by external roads, additional cells outside the zone were added to reproduce a perfect rectangle. Finally, every cell not containing at least one road was removed due to its inaccessibility, and each remaining cell was also provided with its own crime rate, which will be our target function σ.

After all this processing, each zone grid was converted into a graph, as illustrated in Fig. 4. These graphs, although derived from a grid, become irregular due to the removal of non-navigable cells, resulting in large gaps or voids in the graphs. For instance, in zone 3, there are two main gaps: the Mediterranean Sea in the bottom right corner and the railway tracks in the middle; and in zone 10, there are hills in the top left corner. Table 3 shows the main characteristics of each patrolling area, i.e., its dimensions, annual crime rates, and the number of nodes and edges of the graph generated from the grid.

Once the environments were designed and configured, three main parameters needed to be configured: the number of agents, their initial positions, and their range of vision (i.e., *line of sight*). Regarding the number of agents, we explored a small range of values, all of them feasible according to the law enforcement resources. This range allows for adequate coverage of the environment while also being achievable with the city's police resources. Similarly, a range of values was used for the agents' *line of sight*. Finally, for the initial agent positions, two possibilities were tested: randomly positioning the patrols and deploying them at the graph nodes with the highest target function (σ) value.

Table 3
Characteristics of each patrolling area.

Zone Id.	Extension	Annual crime rate	# nodes	# edges
zone 3	1.81 km^2	1999 crimes/km^2	921	2095
zone 9	1.83 km^2	1672 crimes/km^2	923	1809
zone 10	2.98 km^2	1080 crimes/km^2	1766	3733

4.2. Assumptions

When representing real-world scenarios using reinforcement learning models, it is essential to make clear the constraints and assumptions about the real problem to be considered. The more complex the problem to be reproduced, the more difficult it will be to define an effective reward function that facilitates an adequate model learning process. Conversely, if the problem is oversimplified, there is a risk that the results of model training may not be applicable (or may be meaningless) in real situations. For this reason, the following assumptions have been made to model this police patrolling scenario:

- Agents are not constrained to pass through any particular cell during the surveillance, meaning there is no requirement to monitor any specific cell within the area on the map.
- Continuous over-surveillance of an area reduces the effectiveness of patrols in that same area for a short period (Eck, 1993).
- Patrols will not have to dynamically deviate from their route. To establish a route, we assume that no supervening factors will force

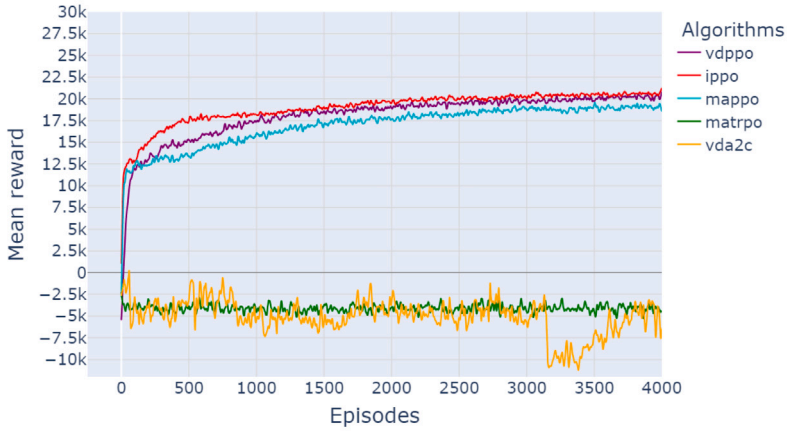


Fig. 5. Comparison of the maximum rewards obtained with IPPO (red), MAPPO (cyan), VDPPO (purple), VD2AC (yellow), MATPRO (green), using a line of sight of 3.

agents to change course. Although this does not fully conform to real-world conditions, we believe that, in most cases, patrols can continue from where they left off without significantly affecting the resolution of the problem.

- All patrols will be equally effective in crime surveillance and will behave similarly, meaning the agents will be homogeneous or interchangeable and, therefore, will share a common policy.
- Each episode of time t corresponds approximately to a 10-minute interval. Consequently, an eight-hour shift is equivalent to 48 steps, which has been rounded up to 50 for the sake of simplicity. This approximation is informed by the spatial resolution of 50 × 50 meter cells and the average human walking speed of 1.3 m per second (Murtagh et al., 2021). This finding indicates that the traversal time for a cell traversing diagonally is approximately 53 s (70 m). The 10-minute duration is designed to encompass time for movement, observation, and engagement, and can be adjusted by law enforcement agencies to align with specific operational objectives.

4.3. Reinforcement learning

To address this police patrol optimization problem, various MARL algorithms were explored. Algorithms that extend or implement *Deep Deterministic Policy Gradient* (DDPG), such as Independent DDPG (IDDPG) (Lillicrap et al., 2015), Multi-Agent DDPG (MADDPG) (Lowe et al., 2017), or *factored multi-agent centralized policy gradients* (FACMAC) (Peng et al., 2021), are designed for continuous action space environments and are not applicable to discrete action space environments, such as our scenario. We also tested the algorithms belonging to the *Advantage Actor Critic* (A2C) family, such as *Multi-Agent A2C* (MAA2C) (Iqbal and Sha, 2019) or *Value Decomposition A2C* (VDA2C) (Su et al., 2021), and the *Trust Region Policy Optimizer* (TRPO) family, such as *Multi-Agent TRPO* (MATRPO) (Li and He, 2023) or *Heterogeneous-Agent Multi-Agent TRPO* (HATRPO) (Kuba et al., 2021). However, they were discarded because they were unable to learn to solve the problem (see Fig. 5). The three algorithms exhibiting the best performance were: *Independent PPO* (IPPO) (De Witt et al., 2020), an adaptation of PPO defined for independent learning; *Multi-Agent PPO* (MAPPO) (Yu et al., 2022), a multi-agent adaptation of PPO derived from IPPO; and *Value Decomposition PPO* (VDPPO) (Ma and Luo, 2022), an extension of IPPO focusing on credit assignment learning.

In all cases, the policy has been shared among the group of agents, as they are homogeneous agents. Note also that the implementation of the algorithms used in our work is that provided by the MARLlib library (Hu et al., 2023), which, in turn, encapsulates the functionality of the RLlib library from Ray (Liang et al., 2018).

Fig. 6 shows an overview of the neural network architecture used for the agent policy in the VDPPO implementation provided by the MARLlib library, which serves as the basis for learning and decision-making in our agents. In the figure, the model is instantiated for two agents and a line of sight of 6. It should be mentioned that we have used the names of the layers as they appear in the MARLlib library.

In the neural network, each agent processes its local observation through a dual-encoder structure. The observation vector has a dimensionality of 340, derived from a field of view of six cells and the inclusion of information about two other agents. This input is passed in parallel through two separate modules: one encoder (*p_encoder*) focused on action selection, and another (*vf_encoder*) dedicated to value estimation. Both encoders consist of two fully connected layers with Tanh activation functions and produce 256-dimensional feature representations. The outputs of both encoders are then passed to a shared recurrent layer implemented as a GRU (*Gated Recurrent Unit*) (Cho et al., 2014) with a hidden size of 256, which captures temporal dependencies and facilitates learning under partial observability. This GRU forms part of a learning architecture that includes a preceding encoding layer of 256–256 neurons. The GRU is conceptually similar to LSTM networks (Hochreiter and Schmidhuber, 1997), but features a simpler structure that often enables faster training while maintaining comparable performance (Abbaspour et al., 2020).

The output of the GRU is processed by two distinct linear heads. The *p_branch* generates a 9-dimensional vector of action logits, while the *vf_branch* produces a scalar estimate of the individual value function, denoted as V_i. After that process, individual value estimates from all agents are aggregated into a joint value V_total through a mixing network (mixer) conditioned on the global state. The input to the mixer has a size of 680, resulting from the concatenation of the 340-dimensional local observations for each deployed agent. The mixer uses three hypernetworks (*hyper_w_1*, *hyper_b_1*, and *hyper_w_final*) to dynamically generate the weights and biases of the mixing layers. This architecture ensures the monotonicity constraint between individual and joint values, as required by value decomposition methods such as QMIX (Rashid et al., 2020), which is the mixer we selected for the VDPPO algorithm.

5. Evaluation

As can be seen in the last column of Table 2, the number of patrols used in the evaluation was set at 2, 5, and 10, all these values within the resources available to the police. The *line of sight* range values of 1, 3, and 6 were selected to represent varying levels of information available to the model. Finally, zone 10 differs from the others in terms of crime dispersion and size, requiring specific adjustments to

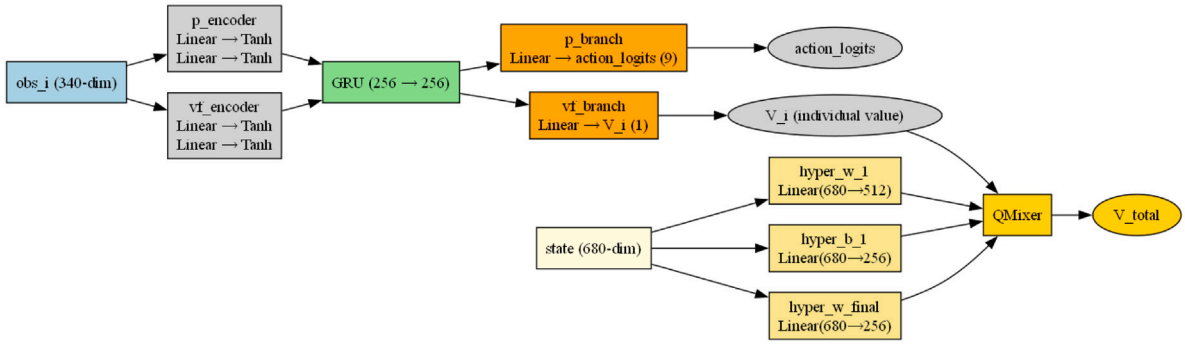


Fig. 6. Neural network model used in VDPPO algorithm.

Table 4
VDPPO hyperparameters.

Parameter	Meaning	Domain	Value
lr	Learning rate	$\mathbb{R}$	0.0005
λ	Lambda	$\mathbb{R}$	0.95
entropy coeff	Entropy coefficient	$\mathbb{R}$	0.01
GAE	Use of Generalized Advantage Estimator	*Boolean*	True
kl coeff	Initial coefficient for KL divergence	$\mathbb{R}$	0.3
Mixer	Neural network that aggregates individual agent value functions into a centralized joint value	Enum	QMIX (Rashid et al., 2020)

train the model effectively. To address these training challenges, the coverage factor for this zone was increased from −25 to -10, and the exploration rewards were doubled to encourage agents to explore the environment thoroughly. These adjustments were necessary because, without increasing the exploration value and reducing penalties, the model had difficulty to train.

The reinforcement learning algorithm used in the evaluation was VDPPO. It is noteworthy that both IPPO and MAPPO were also able to complete the training process and achieve the desired outcomes. However, IPPO required three times longer to train than VDPPO, with an average training time of between eight and twelve hours per model. Additionally, MAPPO consistently performed slightly worse than the IPPO and VDPPO algorithms. Regarding the specific parameters of the VDPPO algorithm (Table 4), a hyperparameter search was conducted to identify the most suitable values within typical ranges. Further information on the parameters, including their meanings and explanations, can be found in Schulman et al. (2015b, 2017), Albrecht et al. (2024).

To evaluate the agents' learning performance, the value of the reward function was initially studied, as well as the effect of the line of sight parameter and the number of patrols deployed on the simulations. As expected, the greater the amount of information about the state of the environment that agents receive, the better the results. Also, we wanted to study the difference in performance depending on the initial node of patrol deployment, i.e., whether patrols are initially deployed in high-crime nodes or, on the contrary, in random positions.

Besides the information provided by the rewards and the loss function of the MARL problem, it is important to check the behavior of the agents to ensure that they have learned the expected behavior and that there is no flaw in the problem setup that causes them to behave in an unexpected way in order to maximize the reward they receive. The only two ways to achieve this are either to have a problem that is simple enough in terms of objectives to be able to define a reward function that simultaneously converges on and perfectly represents the realistic score, or, as in the case of this paper, to set up several metrics to check that, once the model is trained, it behaves as expected.

To study the results, we have not found clear metrics in the literature to help determine which solutions are effective for crime surveillance, mainly because defining what constitutes effective surveillance is not straightforward. In multi-agent patrolling problems, the most used performance criterion is idleness: the time elapsed between consecutive visits to pre-established observation points (Huang et al., 2019). Our model does not preset routes but rather learns or designs coordinated routes that cover the areas with the highest crime rates. So, from our perspective, the idleness criterion was not appropriate to our approach. For this reason, two metrics have been established in order to study the results. The first and simplest is the entropy, which will help us assess how random the routes have been over the 100 runs and especially whether random deployment compensates in terms of randomness compared to the initial deployment in the best nodes. The second metric has been defined to evaluate the coverage of the environment. For this purpose, we define the *coverage index*, an indicator inspired by the *predictive accuracy index* (PAI) (Chainey et al., 2008), a well-known metric within the field of criminology, specifically focusing on crime hotspot detection. The PAI is used to measure the good performance of predictive models of crime. PAI evaluates the quality of prediction of hotspots on a map in terms of a minimum threshold of surface coverage. The accuracy is measured as the proportion of predicted crimes with respect to the chosen coverage area. The values of this area are typically in the range from 3% to 20%. Within the criminology domain, PAI is the only metric that we have found which is both widely used and applicable. Other metrics, such as the forecast accuracy index (FAI) (Zhu and Wang, 2021), are not applicable in our context since we are not actually predicting hotspots. Others, like the predictive efficiency index (PEI) (Du and Ding, 2023), are simply adapted versions of the PAI and are therefore essentially not different enough to generate two different metrics.

Coverage index ($|W_\psi|$): This is a measurement indicator that we have designed based on the coordinated coverage of a surveillance area. Since we have information about crimes that have occurred in these areas, we rank all nodes in the area to be covered by the number of crimes and extract the number of nodes that represent the chosen percentage, ψ, within the range of 3% to 20%. What is measured is the number of nodes with the highest criminal rate, covered by the coordinated routes of the agents. This allows us to evaluate the effectiveness of the selected routes in terms of the coverage of the ψ percentage of nodes with the highest crime rates. Moreover, there is empirical evidence that targeted patrol deployments in identified crime hotspots can lead to a measurable reduction in criminal activity (Basford et al., 2021).

Fig. 7. Comparison of model training in zone 3 in terms of mean reward.

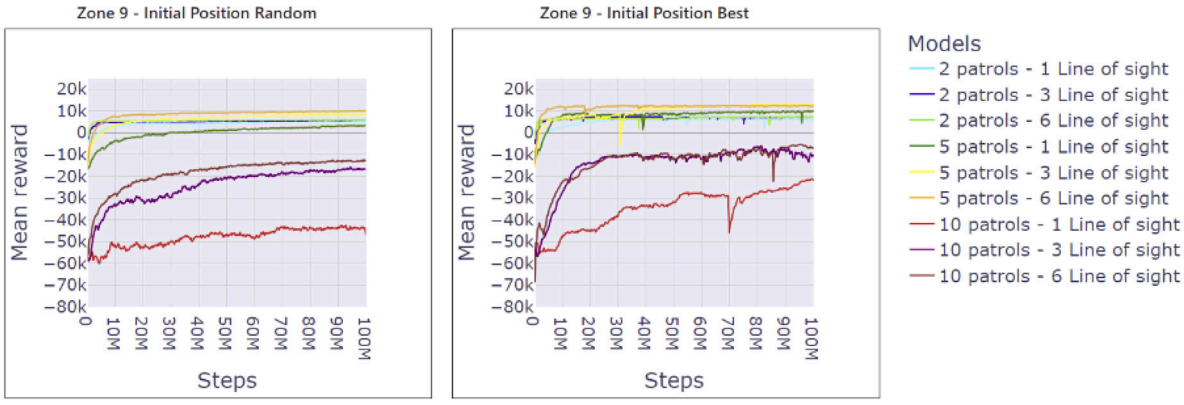


Fig. 8. Comparison of model training in zone 9 in terms of mean reward.

Coverage Index is well-suited for dense urban environments where patrol resources require strategic allocation; however, it may be less appropriate for rural settings, small towns, or sparsely populated areas where spatial assumptions differ. This empirical foundation reinforces the practical relevance of the Coverage Index as a proxy for assessing surveillance effectiveness in realistic policing contexts. Formally, let G be the subset of nodes, $G \subseteq C$, to be monitored, covered in the routes of the agents, and $Z \subseteq G$, a subset that fulfills the following conditions (Eqs. (7) and (8)):

$$\forall v_x \in Z, \forall v_y \notin Z \wedge v_y \in G, \sigma(v_x) > \sigma(v_y) \tag{7}$$

$$|Z| = \frac{\psi |G|}{100} \tag{8}$$

Eq. (7) expresses that all nodes in the subset Z have the highest values of the target function, σ, with respect to G, which is the set of all nodes included in the agents' routes. Eq. (8) states that the cardinality of Z is determined by ψ, having Z therefore the percentage ψ of nodes with the highest values of σ. Finally, the coverage index, which depends on the value of ψ, is the cardinality of the subset W_ψ with the nodes in Z visited by any agent in any simulation episode (Eq. (9)).

$$W_\psi = \{v_i \in Z | \exists t, 1 \leq t \leq T, \exists a_j \in I, \rho_t(a_j) = v_i\} \tag{9}$$

5.1. Comparative study

As mentioned above, our model seeks an optimal selection of coordinated paths by maximizing a target function within a limited time and without having a fixed destination. This means that we cannot compare our results with the state-of-the-art baseline algorithm, CBLS (Portugal et al., 2013; Portugal and Rocha, 2016; Guo et al., 2023), because it works based on the concept of idleness and assumes that all nodes have to be visited. In our problem, it would be impossible to visit all nodes with our limited resources. For this reason, we have developed a greedy algorithm that we believe is comparable to our solution except in terms of communication. This greedy algorithm allows us to determine the minimum threshold of the target function that can be achieved. In this manner, each agent will move to the neighboring cell with the highest score within its reach. All agents will make their decisions simultaneously, similar to the MARL models.

For this greedy algorithm, there is no point in discussing entropy or unpredictability in the case of a fixed initial position of the patrols, such as deployment at the optimal location, because this model is deterministic and, therefore, will always produce the same route with the same starting point. In the case of an initial random position, it will generate something similar; however, if these positions are very close to each other, they may influence the generated routes by moving toward the same cells simultaneously or by one patrol reaching a position before another that was also heading there. This is further exacerbated if the initial position places the patrols in a location that is not particularly relevant on the map.

5.2. Results

The model training performance is illustrated in Figs. 7, 8, and 9, where it can be observed that the model converges in all instances. Regarding the areas, in both zones 3 and 9, it can be observed that the configurations with 10 patrols are the least effective in terms of reward value. This occurs because the patrols are penalized for not doing enough individually when there are fewer cells to cover, as the most relevant cells are being covered by other patrols. In configurations

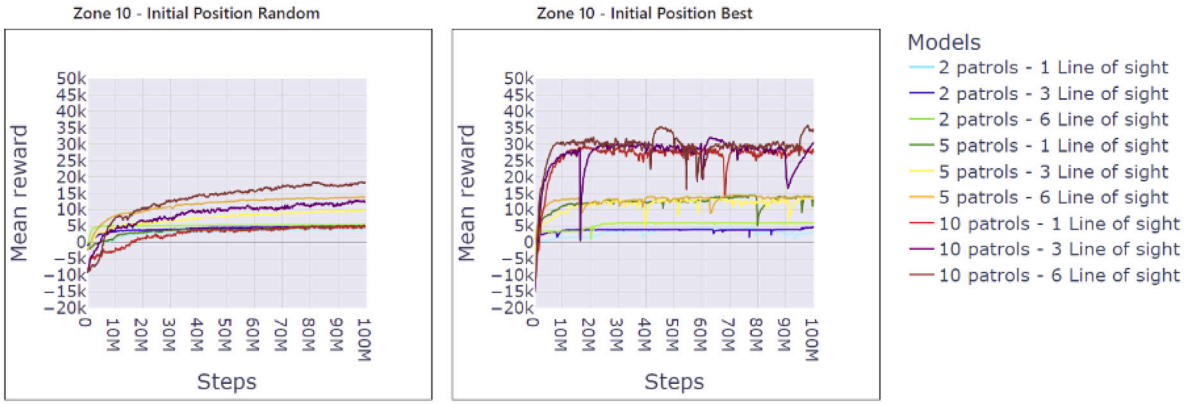


Fig. 9. Comparison of model training in zone 10 in terms of mean reward.

Table 5
Results in zone 3 in terms of coverage index (for values of ψ 3%, 5%, 10%, and 20%) and entropy.

Line of sight	Initial position	# patrols	$\lvert W_3\rvert$	$\lvert W_5\rvert$	$\lvert W_{10}\rvert$	$\lvert W_{20}\rvert$	Entropy
Greedy	Random	2	0.284	0.293	0.279	0.239	5.21
Greedy	Random	5	0.512	0.557	0.550	0.483	5.45
Greedy	Random	10	0.718	0.762	0.767	0.697	5.62
Greedy	Best	2	0.760	0.580	0.350	0.180	0.00
Greedy	Best	5	0.760	0.620	0.490	0.310	0.00
Greedy	Best	10	0.820	0.690	0.570	0.440	0.00
1	Random	2	0.942	0.758	0.633	0.447	4.83
1	Random	5	0.967	0.895	0.773	0.592	5.26
1	Random	10	0.938	0.819	0.732	0.574	5.21
1	Best	2	0.937	0.893	0.748	0.492	4.2
1	Best	5	0.999	0.971	0.889	0.735	5.16
1	Best	10	**1.000**	0.992	0.928	0.850	**5.79**
3	Random	2	0.929	0.756	0.627	0.445	4.77
3	Random	5	0.978	0.95	0.831	0.674	5.22
3	Random	10	0.975	0.924	0.823	0.708	5.59
3	Best	2	0.928	0.907	0.769	0.501	4.23
3	Best	5	0.999	0.984	0.909	0.783	5.06
3	Best	10	**1.000**	0.995	**0.966**	0.890	5.78
6	Random	2	0.908	0.747	0.625	0.459	4.91
6	Random	5	0.977	0.938	0.83	0.675	5.21
6	Random	10	0.996	0.968	0.894	0.772	5.52
6	Best	2	0.868	0.871	0.769	0.551	4.48
6	Best	5	0.999	0.982	0.903	0.764	4.97
6	Best	10	0.998	**0.999**	0.994	**0.916**	5.71

with fewer patrols, each patrol covers more critical areas throughout the simulation, resulting in fewer penalties. For zone 10, the parameters of the reward function were changed (the coverage factor was increased from −25 to −10 and both exploration rewards were doubled) to improve the model exploration capabilities and to allow agents to identify the most relevant areas on the map. By increasing this value, the results with 10 patrols surpass those with 2 and 5 patrols.

The performance of our model in terms of the proposed metrics can be seen in Tables 5, 6, and 7, where the maximum values of each column are marked in bold, and the following ψ values have been used: 3%, 5%, 10%, and 20%. The set of nodes in the 3% of coverage index is the one with the highest incidence of crime; for this reason, the smaller the value of ψ, the more crucial achieving greater coverage becomes. Note also that as the number of nodes expected to be covered increases, the coverage decreases because of the lack of resources to cover them all, given the constraints of a connected path in the graph for each agent.

Finally, it must be pointed out that all simulations were carried out on a machine with 128 GB of RAM, 20 cores of CPU, and an RTX 4090 graphic card. Training time varies depending on the algorithm used and the specific parameter configuration for both termination and weight update of the network. In the final configurations, training times ranged between one and eight hours, for a termination condition of 100,000,000 steps. The training time varied depending on the number of patrols, the line of sight, and the size of the area, but the training time remains between three and six hours. Once trained and loaded, the model is able to generate a coordinated route in less than five minutes or provide the recommended direction in a given situation in real time.

6. Discussion

In the context of patrol routing problems, we have not found metrics that calculate whether the hotspots of the surveillance area have been covered or not. The most common metrics in this field to evaluate the performance are the average duration a node remains unvisited (idle time) or the average visit frequency per node. These metrics are appropriate for timed patrolling, a subcategory within the realm of patrol routing, because the objective is to repeatedly visit all the nodes of the graph or a select group of them that has been previously chosen (Sampaio et al., 2010). Other metrics aim to identify the most

Table 6
Results in zone 9 in terms of coverage index (for values of ψ 3%, 5%, 10%, and 20%) and entropy.

Line of sight	Initial position	# patrols	$\|W_3\|$	$\|W_5\|$	$\|W_{10}\|$	$\|W_{20}\|$	Entropy
Greedy	Random	2	0.232	0.197	0.189	0.179	5.42
Greedy	Random	5	0.496	0.410	0.381	0.345	5.60
Greedy	Random	10	0.664	0.591	0.578	0.535	5.76
Greedy	Best	2	0.570	0.430	0.290	0.170	0.00
Greedy	Best	5	0.630	0.460	0.310	0.200	0.00
Greedy	Best	10	0.890	0.680	0.510	0.350	0.00
1	Random	2	0.784	0.728	0.652	0.416	5.11
1	Random	5	0.851	0.797	0.689	0.519	5.54
1	Random	10	0.389	0.339	0.373	0.372	5.6
1	Best	2	0.811	0.721	0.658	0.437	4.41
1	Best	5	0.941	0.872	0.888	0.692	5.53
1	Best	10	0.900	0.808	0.825	0.654	5.44
3	Random	2	0.801	0.735	0.714	0.441	4.96
3	Random	5	0.918	0.894	0.863	0.622	5.37
3	Random	10	0.925	0.884	0.837	0.678	5.72
3	Best	2	0.894	0.838	0.823	0.524	4.49
3	Best	5	**1.000**	**0.999**	**0.997**	0.782	5.32
3	Best	10	0.987	0.961	0.939	0.764	5.92
6	Random	2	0.816	0.747	0.711	0.439	5.02
6	Random	5	0.965	0.948	0.916	0.662	5.36
6	Random	10	0.958	0.926	0.895	0.731	5.75
6	Best	2	0.933	0.830	0.785	0.480	4.42
6	Best	5	0.946	0.968	0.967	0.753	5.2
6	Best	10	0.999	0.992	0.986	**0.869**	**5.82**

Table 7
Results in zone 10 in terms of coverage index (for values of ψ 3%, 5%, 10%, and 20%) and entropy.

Line of sight	Initial position	# patrols	$\|W_3\|$	$\|W_5\|$	$\|W_{10}\|$	$\|W_{20}\|$	Entropy
Greedy	Random	2	0.141	0.139	0.148	0.125	5.76
Greedy	Random	5	0.296	0.289	0.309	0.274	5.93
Greedy	Random	10	0.460	0.458	0.504	0.469	6.08
Greedy	Best	2	0.290	0.235	0.165	0.097	0.00
Greedy	Best	5	0.548	0.529	0.495	0.329	0.00
Greedy	Best	10	**0.806**	**0.706**	**0.689**	0.502	0.00
1	Random	2	0.424	0.362	0.255	0.165	5.59
1	Random	5	0.263	0.234	0.221	0.205	6.01
1	Random	10	0.287	0.267	0.262	0.268	5.92
1	Best	2	0.249	0.255	0.231	0.163	4.93
1	Best	5	0.502	0.446	0.349	0.291	5.77
1	Best	10	0.507	0.509	0.513	0.464	6.34
3	Random	2	0.56	0.421	0.306	0.198	5.62
3	Random	5	0.519	0.429	0.348	0.27	6.17
3	Random	10	0.535	0.491	0.432	0.367	6.08
3	Best	2	0.441	0.439	0.317	0.235	4.63
3	Best	5	0.576	0.499	0.400	0.326	5.82
3	Best	10	0.591	0.553	0.514	0.495	**6.37**
6	Random	2	0.573	0.43	0.297	0.198	5.70
6	Random	5	0.712	0.594	0.458	0.338	6.24
6	Random	10	0.618	0.544	0.49	0.414	6.30
6	Best	2	0.608	0.500	0.367	0.248	4.82
6	Best	5	0.609	0.570	0.459	0.357	5.76
6	Best	10	0.534	0.552	0.571	**0.538**	6.36

cost-effective routes in terms of agent or fuel cost. However, once again, such problems typically have predetermined objectives to cover. Our problem does not fall within that former subcategory, and our objectives differ from those mentioned for said metrics. Firstly, our model is solved in a single shift duration, and the time spent at each node is also a relevant factor. Therefore, it would be unusual for there to be cycles within the proposed route. Furthermore, the targets to be monitored have not been preselected. Agents must dynamically determine which nodes are worth watching and which ones to exclude from their coordinated route. This differs from other proposals in which the targets, either all the nodes to be monitored or a few of them, have been predetermined in advance (Huang et al., 2019; Pasqualetti et al., 2012a,b; Stranders et al., 2013). For all these reasons, and not having found any suitable metric, we decided to introduce the coverage index as a way of exploring the degree of coverage for the values of the target function of the graph nodes. For the police patrol problem, this metric quantifies whether agents have visited the areas with the highest crime incidence in the scenario tested. The smaller values of the percentage coverage factor, ψ, are the most significant, as they indicate areas with the highest concentration of crime.

In the following, we discuss the performance results of the model in terms of the observability, number of patrols, and starting position.

Observability. Results with a value of 1 line of sight are subject to a large variability in terms of performance. For example, in zone 9 with 10 patrols and a randomly set starting position, the model did not train. This was mainly due to the fact that agents had very limited information to coordinate with, particularly on the status of cells at a moderate distance from them, which can lead to coordination errors. This issue is particularly highlighted in large zones, such as zone 10, where the high-crime nodes are sparse. On the contrary, the performance with a greater line of sight, e.g., 3 and 6, tend to be more stable in all three areas. In zones 3 and 9, with those line-of-sight values, nearly all scenarios achieved a value over 80% in the 3% coverage index, $|W_3|$, with this value increasing as the number of patrols increased. Coverage also exceeded 70% in 5% of cases, with results exceeding 80% when at least 5 patrols were used. The results in these two areas outperform those obtained by the greedy algorithm in all cases and for all values of coverage index tested. Nonetheless, in zone 10, the greedy algorithm performs well, and the results from our model have not been particularly noteworthy, although they continue to outperform the greedy algorithm in almost all cases. This may suggest the need to add more communication pathways for the agents, increase the number of steps, or further refine the parameters to allow better training. It is worth noting that zone 10 has about 40% less crime density because it is 50% larger than the other two zones and has approximately twice as many vertices and edges, which makes training the model much more difficult. While the model is designed to limit agent observability in order to reflect operational constraints, such as limited real-time communication and field of vision, it is realistic to consider that, in actual deployments, some degree of information sharing across adjacent patrol zones does occur. The incorporation of partial information sharing across zone boundaries, particularly in border areas with high crime relevance, has the potential to result in more efficient and coordinated patrol strategies. This remains an interesting direction for future work, as it could help mitigate coordination challenges in both large and sparse environments such as zone 10. It is also important to note that our work focuses on route optimization within predefined patrol zones, whose delimitation is externally decided, under the assumption that adjacent areas are simultaneously covered by other units.

Starting position. Regarding the starting node of the agents, the agents can be placed in the best node not yet chosen by any other agent (initial position best) or it can be random (initial position random). The purpose of this comparison is to determine whether changes in the initial positioning lead to greater diversity in the nodes visited in the routes without compromising the primary objective of crime surveillance. The results show that when random deployment is used, the increase in entropy is generally less than 10% and gives worse results in configurations with 10 patrols. This can be understood by the fact that, once the optimal nodes have been initially covered, the model is not compelled to direct multiple patrols to those same locations. As a result, the randomness of the routes increases, since the optimal cell for further movement is less clearly defined (Tables 5, 6, and 7). In addition, these entropy values do not outweigh the better results obtained by placing the patrols in the hotspot areas in advance. It is important to emphasize that randomness, while relevant, is not the main objective. While it is important that the routes differ from each other, the hotspots on the map remain the same, and in each iteration an attempt is made to monitor these areas over those of lesser interest to the police. This results in certain nodes being visited one or more times in each iteration of the model.

Number of patrols. Analyzing the effects of the number of patrols, the use of only two tends to be insufficient—for the three zones, these configurations are outperformed by the configurations with 5 and 10 patrols (Tables 5, 6, and 7). This seems logical, because it is not the performance of individual patrols that is measured but the overall result. A positive aspect to note is that the training with the configurations with two patrols converge more quickly than the others (Figs. 7, 8, and 9)

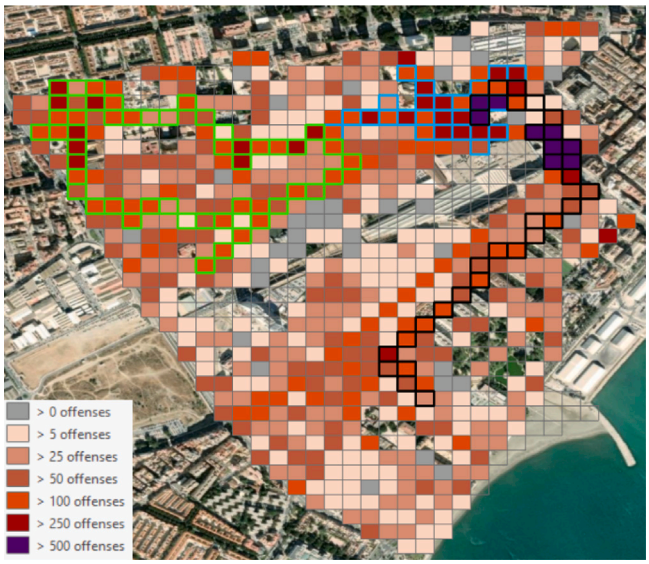


Fig. 10. Three of the five routes designed for our model after a simulation in zone 3.

due to the involvement of fewer agents. Moreover, the efficiency per patrol is maximized, as relatively similar results to configurations with a larger number of resources are achieved with the minimum amount of resources. This allows us to assess the effectiveness of adding a patrol to the surveillance of a zone and, consequently, to determine the optimal number of patrols required to monitor the area effectively. For example, between configurations with 5 or 10 patrols, there is not much difference in most cases. However, on average, coverage decreases by 7% when 10 patrols are deployed randomly compared to initial deployment focused on hotspots, a scenario in which no significant improvement or deterioration is observed. This indicates that the additional five patrols provide no measurable benefit and mostly perform tasks, revealing a saturation point where adding more units becomes unnecessary or even counterproductive, especially when high-priority locations are already well covered. Such redundancy results in inefficient use of public resources and increases the risk of over-patrolling in sensitive areas. This saturation effect is assessed through a redundancy mechanism in the reward function that penalizes revisits to nodes that have already been covered. This approach encourages a more strategic distribution of patrols and facilitates the determination of when it is not worthwhile to add more patrols. Furthermore, while deploying 10 patrols in a single zone may be feasible as a special allocation of resources, it is neither common nor practical for continuous deployment, as it would require maintaining over 100 patrols deployed city-wide at all times.

An alternative modeling approach could involve centralizing the problem, treating patrols as resources controlled by a central agent responsible for training. However, this approach was avoided because the intention is to extend the model over time, giving agents greater autonomy and decision-making capacity. In addition, centralizing the problem may increase training time as the central agent would have to discern which of its multiple decisions is suboptimal relative to the others.

Limitations. In terms of limitations of the proposal, it should be noted that the graphs generated from the test environments are large in size and very irregular as they represent areas of the real world. These characteristics make finding the optimal route from a given starting position prohibitively expensive, as movements are not limited to nodes already visited, there is no designated endpoint, and agents are encouraged to revisit certain nodes. Moreover, our model is not yet sufficiently developed to accommodate responses to incidents that may occur dynamically during a patrol's routine itinerary, such as responding to an accident or moving an offender to the police station.

Furthermore, our environments represent medium-sized urban zones, ranging from approximately 1 to 3 km^2; these are the features of the environments that we believe are suitable for the proposed model. Fig. 10 illustrates an example of the routes designed for three patrols in zone 3, represented through different colors (lime, cyan, and black) of the cell perimeter. Also, a heat map of the entire environment has been used to visualize the crimes occurred in each cell (i.e., the value of the target function). As can be seen, the patrols manage to cover the areas with the highest crime rates. But we are aware that there are techniques more suitable for smaller map sizes, such as the timed patrolling technique (Pasqualetti et al., 2012a,b), primarily because, in very small areas, it may be possible to monitor all the spaces in the zone in a cyclical manner. Our proposal may not be suitable for non-urban environments where law enforcement agencies oversee several villages, farms, or forests within the surveillance area, or for urban environments with extremely sparse data, and we do not believe that the proposed approach will perform adequately in such contexts without substantial modifications to some of its key components. Kadar et al. (2019) establishes that crimes in rural areas are largely determined by geographic points rather than other factors such as criminality data (our case) or population data. The graph resulting from that rural area would involve numerous nodes without information, which would imply a shift in the training environment from dense rewards to sparse rewards. Finally, although we believe our approach can be applied to other similar problems, in each case, it will be necessary to adjust the reward function to enable effective training for the respective problem. In addition, once implemented in real operational contexts, the proposed patrol strategies could complement current metrics such as mean reward, entropy, and coverage index with real policing indicators including response times, detection ratios, or crime reduction trends. This integration would allow validating and refining the model through measurable outcomes, with its real impact lying in the potential deployment by the Málaga Police Department to plan and test patrol strategies in specific districts or special events.

7. Conclusions and future works

This paper proposes a MARL-based model for designing patrol routing strategies, based on a decentralized partially observable Markov decision process and in terms of a target function. The model designs unpredictable and optimized routes for patrols in urban environments represented as undirected graphs, providing area coverage cooperatively, without minimal overlaps, and with partial or full observability. It generates routes suitable for medium-sized urban areas, feasible for foot patrols with a time limited working day. The model has been tested in three districts of the city of Málaga with the aim of optimizing crime surveillance. For this purpose, spatial graphs with nodes representing areas of 50 x 50 m were automatically generated from the street map, and we relied on real data on crimes committed in the city, which were used as a target function of the model.

We have also introduced a novel metric, the *coverage index*, to evaluate the coverage performance of the routes. This index is inspired by the PAI, a well-known metric in criminology that focuses on detection of crime hotspots. Since our model does not preset routes but learns coordinated paths that cover the areas with the highest crime rates, the most commonly used performance criterion, idleness, was not appropriate for our approach. Therefore, we have performed the evaluation of the model in terms of the proposed metric. We have analyzed the impact of the varying information levels observed by agents, the number of agents, and their starting positions. Our findings show that there is no difference in performance between using five and ten patrols. This proves that such a high number of patrols is unnecessary for effective area coverage. Furthermore, results show that starting with a moderate line of sight yields results comparable to those with a larger range. This implies that agents do not need extensive information about the system to operate effectively. Random deployment has proven ineffective at generating significantly more route variety compared to deployment in optimal areas of the map, yielding poorer results on the target metric. Results suggest that the most effective approach is to deploy patrols initially at hotspots within the monitored area. A comparative study with a greedy algorithm proves that the model outperforms the greedy approach in most cases, except in one of the zones. This highlights the need to tailor parameter values to each area to ensure both convergence and task accomplishment.

In future works, we would like to explore the possibility of providing agents with a larger number of steps, thereby reducing the monitoring time at each node. This would enable agents to visit a greater number of nodes in larger urban areas. Additionally, we plan to implement a mechanism for temporal recovery of node importance within a shift, so that areas previously visited can regain priority over time. Moreover, while the current version of the model operates under static crime distribution assumptions, one of our main objectives for future work is to extend it toward dynamic and temporally evolving environments. We plan to incorporate adaptive learning mechanisms capable of updating patrol strategies in real time as crime patterns shift across both space and time. This will allow the framework to more accurately reflect real-world policing contexts, where environmental conditions and criminal activity are inherently dynamic. Furthermore, the incorporation of differentiated weighting for crime categories is of interest, with the potential to allow patrol strategies to focus more explicitly on specific threats, such as violent or drug-related offenses. From an architectural perspective, we plan to compare the results obtained using GRU-based networks with those using transformer-based policies, which have shown promising results in sequential decision-making tasks. We will also study the dynamic modification of the number of agents deployed in an area during the simulation. This scenario is quite common, as certain incidents or emergencies that could occur during surveillance may force patrols to deviate from their planned routes to attend to thefts, accidents, and other events. Finally, we intend to explore overlapping coverage strategies between patrol zones, particularly in high-risk border areas. While our current approach restricts information sharing across zones, future versions may relax this assumption to better reflect coordinated deployments at the system level.

CRediT authorship contribution statement

Juan Palma-Borda: Writing – original draft, Validation, Software, Methodology, Investigation, Data curation, Conceptualization. **Eduardo Guzmán:** Writing – review & editing, Writing – original draft, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization. **María-Victoria Belmonte:** Writing – review & editing, Writing – original draft, Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research is partially supported by the Spanish Ministry of Science and Innovation and by the European Regional Development Fund (FEDER), Junta de Andalucía, and Universidad de Málaga through the research projects with reference PID2021-122381OB-I00 and UMA20-FEDERJA-065. The authors want to express special thanks to the Territorial Intelligence-Analysis Group of the National Police Force (Unidad Territorial de Inteligencia del Cuerpo Nacional de Policía, UTI-CNP) and to the main head for this LEA in the province of Málaga for their support and advice for this work. We also thank the anonymous reviewers for their comments and constructive suggestions, which greatly improved the quality of this paper.

Data availability

The authors do not have permission to share data.

References

Abbaspour, S., Fotouhi, F., Sedaghatbaf, A., Fotouhi, H., Vahabi, M., Linden, M., 2020. A comparative analysis of hybrid deep learning models for human activity recognition. Sensors 20 (19), 5707.

Adepeju, M., Rosser, G., Cheng, T., 2016. Novel evaluation metrics for sparse spatio-temporal point process hotspot predictions-a crime case study. Int. J. Geogr. Inf. Sci. 30 (11), 2133–2154.

Albrecht, S.V., Christianos, F., Schäfer, L., 2024. Multi-Agent Reinforcement Learning: Foundations and Modern Approaches. MIT Press.

Almeida, A., Castro, P., Menezes, T., Ramalho, G., 2003. Combining idleness and distance to design heuristic agents for the patrolling task. In: II Brazilian Workshop in Games and Digital Entertainment. pp. 33–40.

Basford, L., Sims, C., Agar, I., Harinam, V., Strang, H., 2021. Effects of one-a-day foot patrols on hot spots of serious violence and crime harm: A randomised crossover trial. Camb. J. Evidence-Based Polic. 5 (3), 119–133.

Birks, D.J., Donkin, S., Wellsmith, M., 2008. Synthesis over analysis: Towards an ontology for volume crime simulation. In: Artificial Crime Analysis Systems: Using Computer Simulations and Geographic Information Systems. IGI Global, pp. 160–192.

Chainey, S., Tompson, L., Uhlig, S., 2008. The utility of hotspot mapping for predicting spatial patterns of crime. Secur. J. 21, 4–28.

Chen, H., Cheng, T., Wise, S., 2015. Designing daily patrol routes for policing based on ant colony algorithm. ISPRS Ann. Photogramm. Remote. Sens. Spat. Inf. Sci. 2, 103–109.

Chen, H., Wu, Y., Wang, W., Zheng, Z., Ma, J., Zhou, B., 2023. A risk-aware multi-objective patrolling route optimization method using reinforcement learning. In: 2023 IEEE 29th International Conference on Parallel and Distributed Systems. ICPADS, IEEE, pp. 1637–1644.

Chevaleyre, Y., 2004. Theoretical analysis of the multi-agent patrolling problem. In: Proceedings. IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2004.(IAT 2004).. IEEE, pp. 302–308.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

Current, J.R., Schilling, D.A., 1989. The covering salesman problem. Transp. Sci. 23 (3), 208–213.

De Witt, C.S., Gupta, T., Makoviichuk, D., Makoviychuk, V., Torr, P.H., Sun, M., Whiteson, S., 2020. Is independent learning all you need in the starcraft multi-agent challenge?. arXiv preprint arXiv:2011.09533.

Devia, N., Weber, R., 2013. Generating crime data using agent-based simulation. Comput. Environ. Urban Syst. 42, 26–41.

Du, Y., Ding, N., 2023. A systematic review of multi-scale spatio-temporal crime prediction methods. ISPRS Int. J. Geo-Information 12 (6), 209.

Eck, J.E., 1993. The threat of crime displacement. In: Criminal Justice Abstracts, vol. 25, (3), pp. 527–546.

Guo, L., Pan, H., Duan, X., He, J., 2023. Balancing efficiency and unpredictability in multi-robot patrolling: A MARL-based approach. In: 2023 IEEE International Conference on Robotics and Automation. ICRA, IEEE, pp. 3504–3509.

Hari, S.K.K., Rathinam, S., Darbha, S., Kalyanam, K., Manyam, S.G., Casbeer, D., 2020. Optimal UAV route planning for persistent monitoring missions. IEEE Trans. Robot. 37 (2), 550–566.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Hu, S., Zhong, Y., Gao, M., Wang, W., Dong, H., Liang, X., Li, Z., Chang, X., Yang, Y., 2023. MARLlib: A scalable and efficient multi-agent reinforcement learning library. J. Mach. Learn. Res..

Huang, L., Zhou, M., Hao, K., Hou, E., 2019. A survey of multi-robot regular and adversarial patrolling. IEEE/CAA J. Autom. Sin. 6 (4), 894–903.

Hwang, K.S., Lin, J.L., Huang, H.L., 2009. Cooperative patrol planning of multi-robot systems by a competitive auction system. In: 2009 ICCAS-SICE. IEEE, pp. 4359–4363.

Iqbal, S., Sha, F., 2019. Actor-attention-critic for multi-agent reinforcement learning. In: International Conference on Machine Learning. PMLR, pp. 2961–2970.

Kadar, C., Maculan, R., Feuerriegel, S., 2019. Public decision support for low population density areas: An imbalance-aware hyper-ensemble for spatio-temporal crime prediction. Decis. Support Syst. 119, 107–117. http://dx.doi.org/10.1016/j.dss.2019.03.001.

Kuba, J.G., Chen, R., Wen, M., Wen, Y., Sun, F., Wang, J., Yang, Y., 2021. Trust region policy optimisation in multi-agent reinforcement learning. arXiv preprint arXiv:2109.11251.

Lee, H.R., Lee, T., 2021. Multi-agent reinforcement learning algorithm to solve a partially-observable multi-agent problem in disaster response. European J. Oper. Res. 291 (1), 296–308. http://dx.doi.org/10.1016/j.ejor.2020.09.018.

Lee, Y., SooHyun, O., Eck, J.E., 2020. A theory-driven algorithm for real-time crime hot spot forecasting. Police Q. 23 (2), 174–201.

Li, H., He, H., 2023. Multiagent trust region policy optimization. IEEE Trans. Neural Networks Learn. Syst..

Liang, E., Liaw, R., Nishihara, R., Moritz, P., Fox, R., Goldberg, K., Gonzalez, J.E., Jordan, M.I., Stoica, I., 2018. RLlib: Abstractions for distributed reinforcement learning. In: International Conference on Machine Learning. ICML.

Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D., 2015. Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971.

Lowe, R., Wu, Y.I., Tamar, A., Harb, J., Pieter Abbeel, O., Mordatch, I., 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. Adv. Neural Inf. Process. Syst. 30.

Luis, S.Y., Peralta, F., Córdoba, A.T., del Nozal, Á.R., Marín, S.T., Reina, D.G., 2022. An evolutionary multi-objective path planning of a fleet of ASVs for patrolling water resources. Eng. Appl. Artif. Intell. 112, 104852.

Ma, Y., Luo, J., 2022. Value-decomposition multi-agent proximal policy optimization. In: 2022 China Automation Congress. CAC, IEEE, pp. 3460–3464.

Machado, A., Almeida, A., Ramalho, G., Zucker, J.D., Drogoul, A., 2002a. Multi-agent movement coordination in patrolling. In: Proceedings of the 3rd International Conference on Computer and Game. pp. 155–170.

Machado, A., Ramalho, G., Zucker, J.D., Drogoul, A., 2002b. Multi-agent patrolling: An empirical analysis of alternative architectures. In: International Workshop on Multi-Agent Systems and Agent-Based Simulation. Springer, pp. 155–170.

Mak, K.T., Gonzalez, C., Magnaye, Z., Gonzalez, J., Chen, Y., Tang, B., 2024. Budget-constrained traveling salesman problem: a cooperative multi-agent reinforcement learning approach.

Málaga city council, 2024. Sistema de información cartográfica - callejero - datos abiertos ayto. málaga — datosabiertos.malaga.eu. (Accessed 08 April 2024).

Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., Kavukcuoglu, K., 2016. Asynchronous methods for deep reinforcement learning. In: International Conference on Machine Learning. PMLR, pp. 1928–1937.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., et al., 2015. Human-level control through deep reinforcement learning. Nature 518 (7540), 529–533.

Murtagh, E.M., Mair, J.L., Aguiar, E., Tudor-Locke, C., Murphy, M.H., 2021. Outdoor walking speeds of apparently healthy adults: A systematic review and meta-analysis. Sports Med. 51, 125–141.

Oliehoek, F.A., Amato, C., et al., 2016. A Concise Introduction to Decentralized POMDPs, vol. 1, Springer.

Oliveira, W.A.d., Moretti, A.C., Reis, E.F., 2015. Multi-vehicle covering tour problem: building routes for urban patrolling. Pesqui. Oper. 35 (3), 617–644.

OpenStreetMap contributors, 2017. Planet dump. Retrieved from https://planet.osm.org, https://www.openstreetmap.org.

Othmani-Guibourg, M., El Fallah-Seghrouchni, A., Farges, J.L., 2018. Path generation with LSTM recurrent neural networks in the context of the multi-agent patrolling. In: 2018 IEEE 30th International Conference on Tools with Artificial Intelligence. ICTAI, pp. 430–437. http://dx.doi.org/10.1109/ICTAI.2018.00073.

Othmani-Guibourg, M., El Fallah-Seghrouchni, A., Farges, J.L., Potop-Butucaru, M., 2017. Multi-agent patrolling in dynamic environments. In: 2017 IEEE International Conference on Agents. ICA, IEEE, pp. 72–77.

Othmani-Guibourg, M., Farges, J.L., Seghrouchni, A., 2019. LSTM path-maker: a new LSTM-based strategy for the multi-agent patrolling. In: Hawaii International Conference on System Sciences 2019. HICSS-52, http://dx.doi.org/10.24251/HICSS.2019.076.

Parker, J., Nunes, E., Godoy, J., Gini, M., 2016. Exploiting spatial locality and heterogeneity of agents for search and rescue teamwork. J. Field Robot. 33 (7), 877–900.

Pasqualetti, F., Durham, J.W., Bullo, F., 2012a. Cooperative patrolling via weighted tours: Performance analysis and distributed algorithms. IEEE Trans. Robot. 28 (5), 1181–1188.

Pasqualetti, F., Franchi, A., Bullo, F., 2012b. On cooperative patrolling: Optimal trajectories, complexity analysis, and approximation algorithms. IEEE Trans. Robot. 28 (3), 592–606.

Peng, B., Rashid, T., Schroeder de Witt, C., Kamienny, P.A., Torr, P., Böhmer, W., Whiteson, S., 2021. Facmac: Factored multi-agent centralised policy gradients. Adv. Neural Inf. Process. Syst. 34, 12208–12221.

Portugal, D., Couceiro, M.S., Rocha, R.P., 2013. Applying Bayesian learning to multi-robot patrol. In: 2013 IEEE International Symposium on Safety, Security, and Rescue Robotics. SSRR, IEEE, pp. 1–6.

Portugal, D., Rocha, R., 2010. MSP algorithm: multi-robot patrolling based on territory allocation using balanced graph partitioning. In: Proceedings of the 2010 ACM Symposium on Applied Computing. pp. 1271–1276.

Portugal, D., Rocha, R.P., 2016. Cooperative multi-robot patrol with Bayesian learning. Auton. Robots 40 (5), 929–953.

Rashid, T., Samvelyan, M., De Witt, C.S., Farquhar, G., Foerster, J., Whiteson, S., 2020. Monotonic value function factorisation for deep multi-agent reinforcement learning. J. Mach. Learn. Res. 21 (178), 1–51.

Rummens, A., Hardyns, W., Pauwels, L., 2017. The use of predictive analysis in spatiotemporal crime forecasting: Building and testing a model in an urban context. Appl. Geogr. 86, 255–261. http://dx.doi.org/10.1016/j.apgeog.2017.06.011.

Salari, M., Naji-Azimi, Z., 2012. An integer programming-based local search for the covering salesman problem. Comput. Oper. Res. 39 (11), 2594–2602.

Salari, M., Reihaneh, M., Sabbagh, M.S., 2015. Combining ant colony optimization algorithm and dynamic programming technique for solving the covering salesman problem. Comput. Ind. Eng. 83, 244–251.

Samanta, S., Sen, G., Ghosh, S.K., 2022. A literature review on police patrolling problems. Ann. Oper. Res. 316 (2), 1063–1106.

Sampaio, P.A., Ramalho, G., Tedesco, P., 2010. The gravitational strategy for the timed patrolling. In: 2010 22nd IEEE International Conference on Tools with Artificial Intelligence, vol. 1, IEEE, pp. 113–120.

Samvelyan, M., Rashid, T., De Witt, C.S., Farquhar, G., Nardelli, N., Rudner, T.G., Hung, C.M., Torr, P.H., Foerster, J., Whiteson, S., 2019. The starcraft multi-agent challenge. arXiv preprint arXiv:1902.04043.

Santana, H., Ramalho, G., Corruble, V., Ratitch, B., 2004. Multi-agent patrolling with reinforcement learning. In: Autonomous Agents and Multiagent Systems, International Joint Conference on, vol. 4, IEEE Computer Society, pp. 1122–1129.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., Moritz, P., 2015a. Trust region policy optimization. In: International Conference on Machine Learning. PMLR, pp. 1889–1897.

Schulman, J., Moritz, P., Levine, S., Jordan, M., Abbeel, P., 2015b. High-dimensional continuous control using generalized advantage estimation. arXiv preprint arXiv:1506.02438.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

Sea, V., Sugiyama, A., Sugawara, T., 2018. Frequency-based multi-agent patrolling model and its area partitioning solution method for balanced workload. In: Integration of Constraint Programming, Artificial Intelligence, and Operations Research: 15th International Conference, CPAIOR 2018, Delft, the Netherlands, June 26–29, 2018, Proceedings 15. Springer, pp. 530–545.

Shalev-Shwartz, S., Shammah, S., Shashua, A., 2016. Safe, multi-agent, reinforcement learning for autonomous driving. arXiv preprint arXiv:1610.03295.

Sherman, L.W., Williams, S., Ariel, B., Strang, L.R., Wain, N., Slothower, M., Norton, A., 2014. An integrated theory of hot spots patrol strategy: implementing prevention by scaling up and feeding back. J. Contemp. Crim. Justice 30 (2), 95–122.

Soliman, A., Al-Ali, A., Mohamed, A., Gedawy, H., Izham, D., Bahri, M., Erbad, A., Guizani, M., 2023. AI-based UAV navigation framework with digital twin technology for mobile target visitation. Eng. Appl. Artif. Intell. 123, 106318.

Stranders, R., Munoz de Cote, E., Rogers, A., Jennings, N., 2013. Near-optimal continuous patrolling with teams of mobile information gathering agents. Artificial Intelligence 195, 63–105. http://dx.doi.org/10.1016/j.artint.2012.10.006.

Su, J., Adams, S., Beling, P., 2021. Value-decomposition multi-agent actor-critics. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 11352–11360.

Wijaya, R.I., Maulidevi, N.U., 2019. Multiagent system development for cooperative multiplayer video game using deep Q-network. In: 2019 International Conference of Advanced Informatics: Concepts, Theory and Applications. ICAICTA, IEEE, pp. 1–5.

Yan, C., Zhang, T., 2016. Multi-robot patrol: A distributed algorithm based on expected idleness. Int. J. Adv. Robot. Syst. 13 (6), 1729881416663666.

Yin, Z., Jiang, A.X., Tambe, M., Kiekintveld, C., Leyton-Brown, K., Sandholm, T., Sullivan, J.P., 2012. TRUSTS: Scheduling randomized patrols for fare inspection in transit systems using game theory. AI Mag. 33 (4), 59–59.

Yu, C., Velu, A., Vinitsky, E., Gao, J., Wang, Y., Bayen, A., Wu, Y., 2022. The surprising effectiveness of ppo in cooperative multi-agent games. Adv. Neural Inf. Process. Syst. 35, 24611–24624.

Zhang, Q., Lu, C., Garg, A., Foerster, J., 2021. Centralized model and exploration policy for multi-agent RL. arXiv preprint arXiv:2107.06434.

Zhu, H., Wang, F., 2021. An agent-based model for simulating urban crime with improved daily routines. Comput. Environ. Urban Syst. 89, 101680.