2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Cross-Observability Learning for Vehicle Routing Problems

Ruifan Liu[1], Hyo-Sang Shin[2] and Antonios Tsourdos[3]

Abstract— **This study seeks towards a better understanding of multi-vehicle routing problems (VRPs) under restricted observability. Unlike most prior research that assumes full knowledge of tasks and vehicles, this paper addresses VRPs where each vehicle's observation is confined to the k-nearest neighbourhood. Vehicles make decisions based on localized policies in a decentralized manner. We theoretically demonstrate that for the imitation policy, the upper bound of the optimality gap diminishes as the neighbourhood range expands. Subsequently, we employed a multi-agent cross-observability policy optimization (MACOPO) algorithm to solve the VRPs with restricted observability. The algorithm optimizes a cross-entropy term by leveraging a fully observable expert to guide the training. Empirical results supported both the theoretical findings and the effectiveness of the multi-agent learning algorithm.**

I. Introduction

Over the last several years, significant efforts have been devoted to adapting the reinforcement learning (RL) methodology to NP-hard combinatorial optimization challenges [1] [2], such as the travel salesman problems (TSPs) or more general VRPs. Addressing these issues produces practical benefits across many industries, such as logistics, finance, and energy. In contrast to conventional solvers [3] [4], RL offers the capability to prescribe actions on evolving observations, adapting to the dynamic and stochastic environment. For instance, a sequential decision process is proposed in [5] where routes are constructed via appending the next visit node for each vehicle. To optimize the global rewards, vehicles communicate with each other and continuously update the states of all systems entries. Complete observability is assumed in most existing studies for VRPs, where vehicles can access the global state at any time of decision-making, such as in [5] and [6]. Nevertheless, transmitting large amounts of information is undesirable, especially for scaled-up scenarios or communication with limited bandwidth. It thus claims the demand for planning routes only upon localized observations to accommodate practical constraints on communication and also to facilitate scalability.

Several research studies have shed light on using multi-agent RL to solve VRPs with restricted observing ranges. In recent work [7], a substantiation of VRPs on advanced air mobility was modelled as a partially observable stochastic game (POSG), where communication limitations are considered as an observation uncertainty in the mobility network. It indicates that if the policy network is well designed, fleet rewards are positively correlated with the percentage of observed depots. Similarly, Junyoung *et al* [8] employed RL to solve the min-max TSP problem via a type-aware graph attention (TGA)-based neural network. The proposed policy was validated on a practical mTSP variant, where agents act based on local observations due to communication constraints. The paper shows that as the number of observable agents decreases, the timespan increases due to reduced communication, which hinders coordination. To conclude, while the impact of observation range on performance has been acknowledged, little research has been conducted to explore how this range affects suboptimal outcomes.

Besides VRPs, the partial observability (PO) challenge extends to many other real-world robotics problems, motivating researchers to explore various strategies for decision-making in PO environments. Techniques like recurrent neural networks [9], belief-state MDPs [10], and gradient estimators [11] are proposed to tackle this issue. A practical approach often preferred involves using a fully observable (FO) expert to offer guidance whenever possible, as highlighted in [12] [13]. Furthermore, the concept of cross-observability learning, introduced in [14], which leverages an offline FO policy outcome when calculating descent gradients, has shown superior performance compared to the original reinforcement learning algorithm.

Delving into the reasoning behind the impact of observable ranges on VRPs can help practitioners determine the optimal communication range, striking a balance between scalability and the quality of routes. In this study, we particularly look into the following three research questions within the framework of the PO-VRP problem: (1) How does the observation range affect the performance? (2) Is there any boundary for the sub-optimality of a PO policy? (3) Can we take advantage of an off-line FO policy to improve the PO performance?

To answer these questions, this paper constructs a $\boldsymbol{k}$-restricted VRP to model restricted observability, where $\boldsymbol{k} = (k_1, k_2)$ describe the limited communication between vehicle-vehicle and vehicle-task. The restricted VRP is then addressed using a MACOPO method where an FO expert is used to guide the policy training in PO settings. The impact of $\boldsymbol{k}$ values on the performance of PO policies is then theoretically analyzed and empirically investigated.

[1]Ruifan Liu is with the School of Aerospace, Manufacturing and Transport, Cranfield University, Cranfield, United Kingdom ruifan.liu@cranfield.ac.uk

[2]Hyo-Sang Shin is with the School of Aerospace, Manufacturing and Transport, Cranfield University, Cranfield, United Kingdom and Korea Advanced Institute of Science and Technology, Republic of Korea h.shin@cranfield.ac.uk

[3]Antonios Tsourdo is with the School of Aerospace, Manufacturing and Transport, Cranfield University, Cranfield, United Kingdom a.tsourdos@cranfield.ac.uk

II. Related Work

A. MARL for Solving Routing Problems

In recent years, great potential has been found to employ deep RL to resolve routing problems. To use deep RL, the route is constructed by appending next visit nodes, formulated as a sequential Markov decision-making process. Graph-based policy networks and sequence-to-sequence networks are found in the literature to map the state space to action probabilities. The Pointer Network is the first seminar work to cope with the routing problem via learning, which solves TSP via recurrent neural networks [15] [2]. Furthermore, the work in [16] generalizes the application of PtrNet to a wider range of CO such as CVRP, where element embeddings could be dynamic. After that, more effective DL architectures are proposed to represent problem statements by combining the transformer-based attention model [17], [5] or graph embedded structure [18], [19], showing outperforming results in comparison with heuristic methods. A stage has arrived where deep neural networks can extract useful information from customer configurations and obtain high-quality policies for typical routing problems through reinforcement learning [20].

B. VRPs with Restricted Observability

Despite of the fact that solving VRPs under restricted observation has practical implications, it has received little attention in research. Fernando *et al* [7] considered the practical case in air mobility where the observation range is restricted by specifying the number of nearest nodes, and adopted centralized training decentralized execution (CTDE) multi-agent RL approach to address the non-stationarity. Beyond restricted observation, the inherent uncertainty and the occurrence of dynamic events can also cause partial observability in VRPs [21] [22]. To address dynamic and stochastic VRPs, Bono *et al.* [5] developed an online representation of problem states enabling real-time planning via RL based on the latest observation of vehicles. Pan *et al.* [23] took a step further, and formulated the PO Markov decision process for VRPs subject to uncertainties regarding customer locations and demands.

A similar problem arises in the field of multi-robot path planning where individual observation is restricted by the local field of view. To resolve conflicted decisions caused by partial observation, a key-query-like attention mechanism is proposed in [24] to map the inter-robot communication. Likewise, a communication-based MA-G-PPO algorithm is designed in [25] for visibility-based persistent monitoring, where a graph attention layer integrates local features from neighbours. Different observation types are also investigated regarding local maps, mini-global maps, and both. Nevertheless, both studies consider neighbours including all other agents. A large amount of communication is still desired to obtain information from others. Therefore, queries remain regarding the necessity of acquiring all states for decision-making in VRPs, the feasibility of disregarding certain states to enhance efficiency without compromising performance, and the method for identifying which states can be omitted.

III. Problem Formulation

A. Decentralized MDPs of VRP-TWs

In a VRP with time windows (VRP-TW), a fleet of heterogeneous vehicles $v \in \mathcal{V}$ is sent from a central depot d to serve a list of customers $c \in \mathcal{C}$. For the VRP-TW, optimising the vehicle flow indicates routing the vehicles along all the customers while maximizing the collected rewards and minimizing the penalties for missing time windows. For its dec-MDPs, each agent $i \in \mathcal{I}$ corresponds to one vehicle. Individual state s^i for each vehicle (position, and next destination), and joint mission state s^m including the features and status of each client (reward, time window, and pending or assigned) constitute the global state $s_t \in \mathcal{S}$.

A vehicle v will make an action $a_t^i \in \mathcal{A}^i$ when $1_{avail}(v^t) = 1$, choosing the next client to visit from the pending list or returning to the depot. Upon the assignment, the vehicle v switches to unavailable status $1_{avail}(v^{t+\Delta t}) = 0$. It then travels to the destination and serves the client, at which point the availability switches back to $1_{avail}(v^{t+\tau+\Delta t}) = 1$. $\tau = \tau_1 + \tau_2$ denotes the sum of the travel time it takes to the client c and the serving time at c.

Upon completing the travel $t + \tau$, the vehicle collects a payoff from the customer $r(c)$. Some requests may want to be served during specific time windows. In this case, when the arrival time of the vehicle goes beyond the client's designated service window (specifically, when $t + \tau_1 > t_c^d$, where t_c^d is the latest time window), no reward is earned. Instead, a constant penalty is applied. If a task is not visited by any vehicle within its time window, a pending penalty is also imposed. The serving process is considered stochastic, meaning the element defining the problem, such as the travel speed or the serving time, does not have to be constant and can be described as a random variable. Also as in the real world, not all clients are known in advance. New requests are constantly appearing, and the vehicles need to make decisions based on demand.

B. Partial Observability

Each vehicle in the system communicates with its neighbouring agents and clients to acquire its local observation, such that $\mathbb{O} : \mathcal{S} \to \mathcal{O}_i$ maps the joint state of the system to the local observation of agent i. Different observation ranges can be applied to clients and other vehicles due to their distinct update frequencies, that is, the locations of the vehicles are continuously changing while clients are relatively static. Thus, we define a time-varying neighbourhood for a vehicle v, $\mathcal{N}_v^t$ containing its observable vehicle $\mathcal{V}_v^t$ and the clients $\mathcal{C}_v^t$ at the time step t, such that $\mathcal{N}_v^t \in \mathcal{V} \cup \mathcal{C}$, such that the vehicle observation is $o_v = s^{\mathcal{N}_v^t}$. Specifically, a $\boldsymbol{k}$-nearest neighbour graph is used to model partially observable VRP environments. $\boldsymbol{k} = (k_1, k_2)$, where k_1 is how many nearest neighbours are included in agent-agent graphs and k_2 is how many nearest neighbours are included in task-agent graphs. Fig. 1 gives a toy demonstration of how observation shifts when $k_1 = 1, k_2 = 2$.

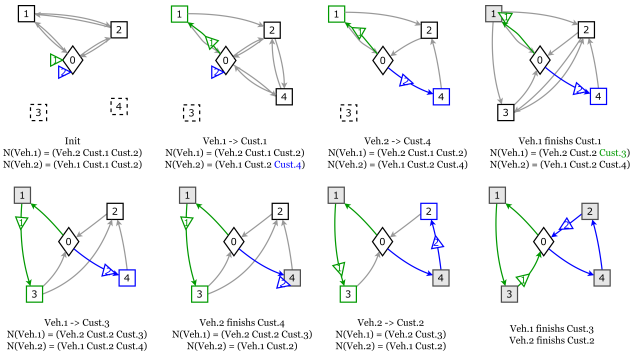


Fig. 1: A toy example of PO-VRP. It is adapted from [5], but added with the PO feature. The observation range of both vehicles towards customers is restricted to 2. Observation updates whenever customers are finished or assigned. Dashed customers are not observed by any of the vehicles. Gray edges indicate the valid travel.

C. Sparse Timescale

A sparse timescale is specifically customized for VRP-TWs, accounting for 1) the curse of dimensionality and 2) asynchronous actions. The multiple autonomous vehicle system in VRPs has a sparse timescale nature. After one vehicle commits to a high-level decision (i.e., choosing the next client in the context), its intermediary actions have little impact on system state transitions and thus can be tailored to reduce the problem dimensionality. As vehicles do not complete their journeys simultaneously most of the time, these macro actions happen asynchronously, especially when agents act in a decentralized manner. Considering these properties, decision-making for the multiple-vehicle system calls for a sparse/macro action modelling while being capable of handling the asynchronicity.

We adopt the technique of discrete-event simulation (DES) to achieve the evolvement of the autonomous multi-vehicle system. In DES, time directly advances to the occurrence of discrete events as opposed to continuous or fixed-timestep models. The autonomous vehicle system involves discrete events such as the assignment of clients and the arrival of new requests. Specifically, consider an identity function that indicates a vehicle v's availability at time t such that $1_{avail}(v^t) = 1$ when v is free from any client, or $1_{avail}(v^t) = 0$ when it is committed to a client and unavailable. Every change in the vehicle's availability function constitutes an event in the system. Additionally, changes on the client list because of the arrival of new tasks, i.e., $\mathcal{C}^{t+\Delta t} = \mathcal{C}^t \cup c'$, will also trigger an event. Throughout the DES, the vehicle's action selection and observations only happen in timesteps associated with events, leaving the local trajectory and control to take place intermediary.

IV. Method

In this session, we start by theoretically analysing the rationale behind the performance of PO policies. Following that, we present a MACOPO-based multi-agent reinforcement learning algorithm to solve the VRPs with restricted observability.

A. Theoretical Analysis for PO-VRPs

According to partial observability defined in Session III-B, the information used in generating policies is restricted to the local observation $o_i = s^{\mathcal{N}_i^k}$. Recall that $\mathcal{N}_i^k$ denotes the set of k-nearest neighborhood of vehicle i, and define $s^{\mathcal{N}_{-i}^k} = s_i / s^{\mathcal{N}_i^k}$ as the state beyond the observability. It means the value of $\pi_i(s)$ depends on s only through o_i, i.e., $\pi_i(s) = \pi_i(s')$ implies $o_i = o_i'$. For simpler notation, this restricted process is described by a filtration function: $f^{\mathcal{N}} : \mathcal{S} \to \mathcal{O}$, and the space of feasible policies is limited to those $\mathcal{N}$-restricted policies, for which we write as $\pi_i^{\mathcal{N}}(o)$ to mean $\pi_i(s)$ if $f^{\mathcal{N}}(s) = o$.

The first part of the analysis focuses on the impact of the policy observation range k on value function $V^\pi(s)$. Inspired by the abstract MDPs proposed in [26], we expand the near-optimal behaviour of the value function to the policy aggregation. In [26], approximation abstract in MDPs is performed by aggregating the states within a certain proximity, which exhibits a polynomially near-optimal performance. Similarly, the key idea here is that by defining the upper bound of the policy function gap, the sub-optimal value function gap is also preserved. The result is formally given by Theorem 1.

Theorem 1 Suppose the reward is upper bounded as $0 \leq r(s,a) \leq \bar{r}, \forall s, a$, and the total variation distance between the FO policy μ and the PO policy π is $d_{TV}(\mu, \pi) = \sup_s |\mu(s) - \pi(s)|$. Then, for all $s \in \mathcal{S}$,

$$V^\mu(s) - V^\pi(s) \leq \frac{\bar{r} d_{TV}(\mu, \pi)}{(1-\gamma)^2} \tag{1}$$

Proof of Theorem 1 Given a multi-agent MDP with a set of non-stationary policy $\pi(s,a)$, its state-value function is $V^\pi(s) = \sum_{a \in A} \pi(s,a) Q^\pi(s,a)$. Thus, value function under a policy $\pi(a \mid o_i)$ with the partial observation o_i is:

$$V^\pi(s) = \sum_{a_i \in A_i} \pi(a_i \mid o_i = f_i^{\mathcal{N}}(s)) Q^\pi(s, a_i) \tag{2}$$

where $f_i^{\mathcal{N}}(s)$ denote the observation function of agent i regarding its neighorbood $\mathcal{N}$. If fully observed, the value function is:

$$V^\mu(s) = \sum_{a_i \in A_i} \mu(a_i \mid s) Q^\mu(s, a_i) \tag{3}$$

where $\mu(a_i \mid s)$ presents the policy of full observability.

Now consider a policy π_t of following the PO policy π for t steps and then following the optimal ground policy μ:

$$\pi_t(s) = \begin{cases} \mu(s) & \text{if } t = 0 \\ \pi(s) & \text{if } t > 0 \end{cases} \tag{4}$$

For $t > 0$, the value of this policy for $s \in S$ is

$$V^{\pi_t}(s) = E_{a \sim \pi_t(s)} \Big[r(s,a) + \gamma \sum_{s' \in S} \mathcal{T}(s,a,s') V^{\pi_{t-1}}(s') \Big] \tag{5}$$

For $t = 0$, $V^{\pi_t}(s) = V^\mu(s)$.

We now proceed by induction on t to show that

$$\forall_{T,s\in S} V^{\mu}(s) - V^{\pi_t}(s) \leq \sum_{i=0}^{t} \gamma^i \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma} \tag{6}$$

1) *Base case* $t = 0$: By definition, when $t = 0$, $V^{\pi_t}(s) = V^{\mu}(s)$, the bound trivially holds.

2) *Inductive case* $t > 0$: Consider a fixed but arbitrary state $s \in \mathcal{S}$, we assume by our inductive hypothesis that

$$V^{\mu}(s) - V^{\pi_{t-1}}(s) \leq \sum_{i=0}^{t-1} \gamma^i \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma} \tag{7}$$

Applying the equation (5) yields

$$\begin{aligned} V^{\pi_t}(s) \geq & E_{a\sim\pi_t(s)}\Big[r(s,a) + \\ & \gamma \sum_{s'\in S} \mathcal{T}(s,a,s')\Big(V^{\mu}(s') - \sum_{i=0}^{t-1} \gamma^i \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma}\Big)\Big] \end{aligned}$$

Since the $\bar{r}$ and $d_{TV}(\mu,\pi)$ are constant values, we have

$$\begin{aligned} V^{\pi_t}(s) \geq & -\gamma \sum_{i=0}^{t-1} \gamma^i \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma} + \\ & E_{a\sim\pi_t(s)}\Big[r(s,a) + \gamma \sum_{s'\in S} \mathcal{T}(s,a,s') V^{\mu}(s')\Big] \end{aligned}$$

Recall the definition of total variance distance $d_{TV}(\mu,\pi)$,

$$\begin{aligned} V^{\pi_t}(s) \geq & -\gamma \sum_{i=0}^{t-1} \gamma^i \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma} + V^{\mu}(s) \\ & - d_{TV}(\mu,\pi)\Big[r(s,a) + \gamma \sum_{s'\in S} \mathcal{T}(s,a,s') V^{\mu}(s')\Big] \end{aligned}$$

According to the definition of value function, we have its upper bound $V^{\mu}(s') \leq \sum_{i=0}^{\infty} \gamma^i \bar{r} = \frac{\bar{r}}{1-\gamma}$. Therefore,

$$\begin{aligned} V^{\mu}(s) - V^{\pi_t}(s) &\leq \gamma \sum_{i=0}^{t-1} \gamma^i \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma} + \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma} \\ &= \sum_{i=0}^{t} \gamma^i \frac{\bar{r} d_{TV}(\mu,\pi)}{1-\gamma} \end{aligned}$$

Since s was arbitrary, we conclude that the bound holds for all states in S for the inductive case. Further, as $t \to \infty$, $\sum_{i=0}^{t} \gamma^t \to \frac{1}{1-\gamma}$ by the sum of infinite geometric series and $\pi_t \to \pi$, Theorem 1 is yielded. ■.

It may not be clear that how to get the upper bound of the total variance distance between FO and PO policies. To demonstrate it, We further define the ρ-decay property of the policy.

Definition 1 For a function $\rho : \mathbb{N} \to \mathbb{R}^+$ that satisfies $\lim_{k\to\infty} \rho(k) = 0$, the ρ-decay property of policy holds if, for any stationary policy π, for any $s_{\mathcal{N}_i^k} \in \mathcal{S}_{\mathcal{N}_i^k}, s_{\mathcal{N}_{-i}^k} \in \mathcal{S}_{\mathcal{N}_{-i}^k}, a_i \in A_i$, if the policy probability satisfies,

$$|\pi(s_{\mathcal{N}_i^k}, s_{\mathcal{N}_{-i}^k}, a_i) - \pi(s_{\mathcal{N}_i^k}, s'_{\mathcal{N}_{-i}^k}, a_i)| \leq \rho(k) \tag{8}$$

Assuming a cross-observability case where we have access to a fully observable policy $\mu(s)$ and use it to guide the training of policy $\pi_i(o_i)$ which can only access the restricted state o_i. FO policy has access to the global state and might be optimal among all policies. To leverage the full observability, PO policy can be trained by minimizing the expected cross entropy $\mathbb{E}[\mathcal{H}(\mu\|\pi)]$ between μ and π over the state space. we denote the policy minimizing the cross entropy as the imitate policy π_{im}. Intuitively, when some of the state is not available in restricted observation, the PO policy tempts to imitate the FO policy by marginalizing unobservable information. This intuition is formally formulated as Lemma1.

Lemma 1 (Policy Averaging [12]) For any $s \in S$ with $o = f(s)$, we have the imitate policy equal to the expectation of FO policy over the state distribution:

$$\pi_{im}(o) = \mathbb{E}[\mu(s) \mid f(s) = o] \tag{9}$$

With Lemma 1, the policies exhibiting ρ-decay property hold the upper bound of $\rho(k)$ as the total variance distance between FO and imitated PO policies: $d_{TV}(\mu, \pi_{im}^{\mathcal{N}^k}) \leq \rho(k)$. In that case, the upper bound of the suboptimality gap defined in Theorem 1 is determined by the decay function $\rho(k)$. More specifically, within the scope of VRPs,

$$\begin{aligned} d_{TV}(\mu, \pi_{im}^{\mathcal{N}^k}) &= \sup_s \Big|\mu(s) - \pi_{im}^{\mathcal{N}^k}(s)\Big| \\ &= \sup_s \Big|\mu(s) - \sum_{s'\in\mathcal{S}} \mu(s') Pr(s' \mid o = f^{\mathcal{N}^k}(s)\Big| \\ &\leq \sup_s |1 - Pr(s \mid o)| \end{aligned}$$

If all customers are uniformly and independently distributed, the conditional distribution is upon the size of unobserved area: $Pr(s \mid o) = (\frac{1}{\pi m^2 - \pi k_2^2})^{m-k_2}$, where $m = |\mathcal{C}|$ is the number of all customers, k_2 is the number of observed nearest customers. It is easy to find out that the upper bound of d_{TV} quickly drops as k increases. It shows we may achieve a good near-optimal performance using a scalable PO policy that only depends on k neighbours.

It is noted that the imitate policy π_{im} is typically not the optimal PO policy. Imitating the FO policy projects the FO behaviours to PO behaviours, which is known to be sub-optimal as the information that the FO policy uses is not available for the PO policy. Further, FO behaviours can be sub-optimal for PO decision-making, e.g., PO agents might need to engage in information gathering beforehand. As a result, an optimality gap exists between the imitate policy π_{im} and the optimal PO policy π^*.

To formalize the gap between the optimal PO policy and the FO policy, we give Theorem 2.

Theorem 2 The optimal value function under partial observability $\mathcal{N}$ is suboptimally bounded by

$$c(\pi_d^*) \leq V^{\mu}(s_0) - V^*(s_0) \leq c(\mu) \tag{10}$$

where μ is the optimal FO policy, $V^*(s_0)$ is the optimal value under PO. π_d^* is the optimal dual policy for the perfect information relaxation problem. c is the ideal penalty of

information relaxation, taking the form of

$$c(\pi) = \sum_{t=0}^{T} \mathbb{E}_{a\sim\pi} \left\{ \mathbb{E}\left[V^*(s_{t+1}) \mid f^{\mathcal{N}}(s_t) \right] - V^*(s_{t+1}) \right\} \tag{11}$$

for both μ and π_d^*, where $f^{\mathcal{N}}(s_t)$ is the filtration of $\mathcal{N}$-restricted process.

Proof of Theorem 2 To prove Theorem 2, we use the technique of information relaxation proposed by [27]. Briefly, information relaxation constructs upper dual bounds for dynamic programming by 1) relaxing the non-anticipativity constraint and 2) including a penalty that punishes the violation of these constraints.

Taking the perfect information relaxation (i.e, FO in our case), we get the upper bound for the PO policy:

$$V^* \leq \max_{\pi} \mathbb{E}\left[\sum_{t=0}^{T} r_t(s_t, \pi_t) - c_t \right] \tag{12}$$

The right-hand side is the dual optimization problem, where c_t is the dual penalties, to punish the actions for using the information that would not be known in the primary filtration $f^{\mathcal{N}}(s_t)$. Specifically for each t, we adopt the ideal penalty

$$c_t := \mathbb{E}\left[V_{t+1}^*(s_{t+1}) \mid f^{\mathcal{N}}(s_t) \right] - V_{t+1}^*(s_{t+1}) \tag{13}$$

for which the equality of (12) will hold. Proofs refer to [27]. Now let us assume the π_d^* is the optimal policy for the dual problem (12), we then have

$$r(\pi_d^*) - c(\pi_d^*) = \sum_{t=0}^{T} r_t(\pi_d^*) - c_t(\pi_d^*) = V^* \tag{14}$$

Obviously, $r(\mu) \geq r(\pi_d^*)$ for the optimal FO policy μ. Thus, we get the lower bound for the gap between PO value and FO value:

$$V^\mu - V^* \geq c(\pi_d^*) \tag{15}$$

For upper bound, we know $r(\mu) - c(\mu) \leq r(\pi_d^*) - c(\pi_d^*)$ due to the maximization operation, we then get:

$$V^\mu - V^* \leq c(\mu) \tag{16}$$

■

The above suboptimal bounds formalize the gap range between the restricted policy and the FO policy. Intuitively, the ideal penalty c_t represents the difference between the real next-step value V_{t+1} with the marginalized value on the restricted filtration $f^{\mathcal{N}}(s_t)$, to eliminate the benefit of relaxed information. Similar to the concept of *Policy averaging*, $\mathbb{E}\left[V_{t+1}(s_{t+1}) \mid f^{\mathcal{N}}(s_t) \right]$ is the value averaged over the unobserved states. The upper bound $c(\mu)$ and the lower bound $c(\pi_d^*)$ accumulate the differences following the policy μ and the policy π_d^* respectively. Unfortunately, the optimal PO value function V^* is generally not known in applications, approximate approaches are often employed to obtain the boundaries [28].

B. Cross-Observability Learning Method

A two-stage MARL algorithm is employed to obtain the policy towards a better performance of partial observations. For the first step, we employ centralized training to obtain the FO state experts $\mu_i(a, s)$, during which we assume all agents can access the global information. Then, MACOPO is employed to obtain a PO policy $\pi_{\theta_i}(a_i, o_i)$ parameterized by θ_i.

We focus on the details of the second stage of implementation. MACOPO extends the single-agent actor and critic policy gradient method. It follows a CTDE structure where a centralized value function $V(s)$ is shared. For the objective function, instead of using the expected reward as in multi-agent policy gradient (MAPG), MACOPO adds a cross-entropy term $\mathcal{H}(\mu_i \| \pi_i)$ that describes the similarity between PO policy and FO expert. Combining these terms, we obtain the following objective, which is (approximately) maximized each iteration:

$$\hat{J}(\theta_i) = \hat{\mathbb{E}}[R_i - \beta_1 \mathcal{H}(\mu_i \| \pi_{\theta_i})] \tag{17}$$

The gradient estimator regarding the above objective function $\hat{J}(\theta_i)$ for agent i is:

$$\nabla_{\theta_i} \hat{J}(\theta_i) = \hat{\mathbb{E}}_{s,a_i\sim\pi_i} \Big[\nabla_{\theta_i} \log \pi_{\theta_i}(a_i, o_i) \hat{A}(s, a) + \beta_1 \mu_i(s, a) \nabla_{\theta_i} \log \pi_{\theta_i}(a_i, o_i) \Big]$$

where $\hat{A}$ is an estimator of the advantage function that is used to reduce variances by using a learned state-value function $\hat{V}_{\theta_v}(s_t)$ parameterized by θ_v, which is:

$$\hat{A}(s_t, a_t) = R(s_t, a_t) - \hat{V}_{\theta_v}(s_t) \tag{18}$$

The $\hat{A}$ and $\hat{V}_{\theta_v}$ are estimated assuming access to an ensemble of local information, including the joint action of all agents $a = (a_1, \cdots, a_N)$ and the state s consisting of the observations from all the agent $s = (o_1, \cdots, o_N)$. $R(s_t, a_t) = \sum_{t=0}^{\infty} [r(s_t, a_t \mid \pi_\theta)]$ is the accumulated reward obtained from the environment according to the joint policy π_θ.

When sharing parameters between the policy and value function, the loss function is defined by combining the policy gradient and a central value function error term:

$$L(\theta, \theta_v) = \hat{\mathbb{E}} \sum_i - \log \pi_{\theta_i}(a_i, o_i) \Big[\hat{A}(s, a) + \beta_1 \mu_i(s, a) \Big] + \beta_2 \Big[R(s, a) - \hat{V}_{\theta_v}(s) \Big]^2$$

Moreover, instead of using a fixed coefficient for the entropy term β_1, we set up a learning process for β_1, as suggested in [14]. We formulate a minimization objective for β_1:

$$J_{\beta_1}(\beta_1) = \beta_1 \bar{\mathcal{H}} - \beta_1 \hat{\mathbb{E}} \left[\mathcal{H}(\mu_i \| \pi_{\theta_i}) \right] \tag{19}$$

which dynamically adjusts β_1 to preserve a target expected entropy $\bar{\mathcal{H}}$. Similar to the target entropy in soft actor-critic, $\bar{\mathcal{H}}$ implies the expected divergence between π and μ, while the choice of $\bar{\mathcal{H}}$ likely varying by domain.

V. Empirical Simulations

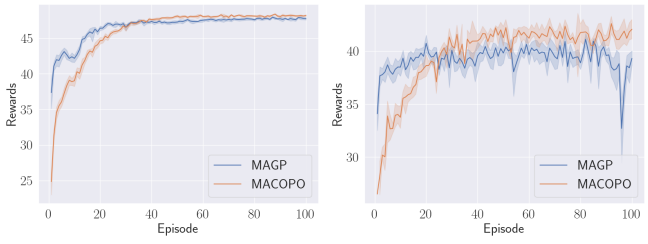
The proposed MACOPO is validated for solving VRP-TWs. The VRP-TW environment is constructed with Python. The policy network and critic network are from [29] and trained with a PyTorch back-end on a Tesla A100 GPU for 20 hours on Cranfield University HILDA high-performance computing system.

A. Vechicle routing problem with time windows

Suppose a scenario where there is a list of customer requests $j \in \mathcal{T}$ and a fleet of vehicles $i \in Ag$ is departed from the depot (x^0, y^0) serving the customers and finally returns to the depot after visiting all the tasks. The corresponding price r_j is collected from the customer j if any of the vehicles arrive at its location (x^j, y^j) during the time window $[t_r^j, t_d^j]$, otherwise a pending cost c_{pen} is imposed. In this logistic problem, optimizing the vehicle flow indicates maximizing the rewards collected from customer requests while minimizing the penalties of missing visiting time windows. For specific parameters see Appendix A.

B. Training

Two policies π_{θ_1} and π_{θ_2} are trained using two sets of scenarios respectively. Both involve 10 vehicles and 50 tasks. π_{θ_1} is trained with scenarios under a restricted agent observation $k_1 \sim \mathcal{U}[1, 10]$. The second policy π_{θ_2} is trained with scenarios under a restricted task observation $k_2 \sim \mathcal{U}[10, 50]$. k_1 and k_2 are randomly generated following the uniform distribution $\mathcal{U}$ for a better generalization ability. The Adam Optimizer is used to train networks with a constant learning rate $\alpha_\theta = 10^{-4}$. The target cross-entropy is set $\bar{\mathcal{H}} = 1$ and the learning rate for entropy weight $\alpha_{\beta_1} = 10^{-4}$. The coefficient for value error is $\beta_2 = 1$. We run training for 100 epochs to get the FO state expert and an additional 100 epochs for MACOPO training. In one epoch, we process 1.28 million instances in 2,500 iterations with a batch size of 512. Each epoch takes up to 13 minutes using the Tesla A100 GPU.



(a) k1 $\sim U[1, 10]$, k2=50 (b) k1=10, k2 $\sim U[10, 50]$

Fig. 2: Training performance of MAPG and MACOPO with varying observation ranges.

The training performance of MACOPO with the MAPG baseline is compared in Fig. 2. MAPG shares the same settings as MACOPO except for the cross-observability entropy term. For both trainings, the proposed MACOPO outperforms MAPG. It implies that by imitating an FO expert, the PO agents will obtain an improved policy, even if the anticipated information used by FO experts is not accessible to PO agents in planning.

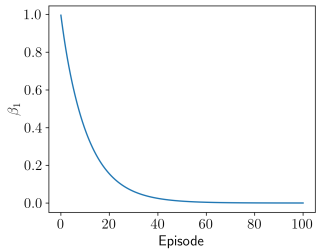
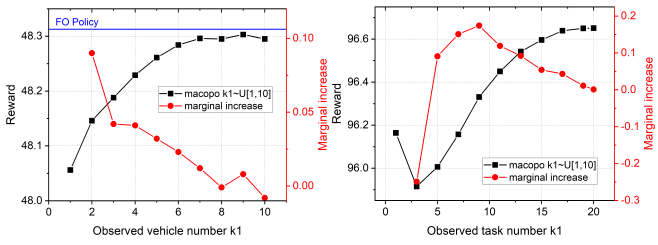


Fig. 3: Training curve of the entropy coefficient β_1.

Fig. 3 depicts the curve of β_1 value throughout the training process. With the coefficient β_1 intially set to 1, the cross-observability entropy guides the learning at the beginning. As training progresses, the value of β_1 diminishes, approaching zero after approximately 60 episodes, at which point its impact on policy learning becomes negligible. Notably, the cross-observability entropy term appears to exert a positive learning effect before 50 episodes, when the MACOPO outperfoms MAPG, as shown in Fig. 2. Furthermore, since β_1 is being adjusted towards a target entropy $\bar{\mathcal{H}}$, the observed deceleration in the decline of β_1 suggests that the entropy between the FO expert and the PO policy converges towards the target value, though it consistently remains above the target.

C. Performance under k_1-restricted observability

We evaluate the PO policy π_{θ_1} with the scenarios where the observability regarding other vehicles is restricted by k_1. Simulation results are depicted in Fig. 4. The two test scenarios involve one with 50 tasks and another with 100 tasks. In the case of 100 tasks, which exceeds the training capability, we generalize the policy trained with 50 tasks to the larger input dimensions. The metric of total rewards is averaged over 1000 random mission instances.



(a) 50 tasks scenarios (b) 100 tasks scenarios

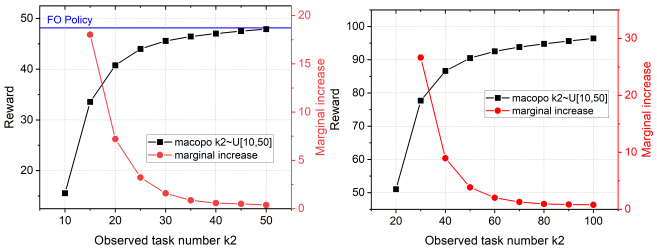
Fig. 4: Total reward when agent observation k_1 varies.

As seen from both plots, the incremental increase of reward shows a diminishing trend as the observation range k_1 increases. It implies that when the current observation set is larger, the newly joined observation element has less impact on the result. Though a few points (the second to last in Fig. 4a and the first 3 in Fig. 4b) show a reverse

trend, it can be justified by the suboptimality of the RL policy. Overall, the differences between results of different k_1 are surprisingly small. According to Theorem 2, the gap between an optimal PO policy and the optimal FO policy is lower bounded by the variable $c(\pi_d^*)$, and upper bounded by $c(\mu)$. Even though the trained policies only approximate the optimal policies, we can still say that the gap between the real state value and the averaged state value is very likely to be quite small, combining the results shown in Fig. 4 and the definition of the boundary (13). Therefore, we conjecture that the awareness of other agents' statuses does not significantly influence the decision-making results in VRPs.

D. Performance under k_2-restricted observability

Similarly, the PO policy π_{θ_2} is evaluated with scenarios where the observability regarding tasks is restricted by k_2. Simulation results are depicted in Fig. 5. The metric of total rewards is averaged over 1000 random mission instances.



(a) 50 tasks scenarios (b) 100 tasks scenarios

Fig. 5: Total reward when task observation k_2 varies.

Fig. 5 demonstrates the improvement of performance for both problem sizes when the task observation range becomes larger. Both policies in evaluation are not exactly the optimal imitate policy π_{im}, which means Lemma 1 cannot be directly applied here. However, from the empirical results, it still shows a clear trend of diminishing gain especially when the observation range is getting larger, especially for k_2.

The diminishing reward increment implies that enlarging the observation range in VRPs does not necessarily bring equivalent benefits to the planning results. As seen from Fig. 5a, observing the nearest 20 tasks achieves 85% of optimality; observing the nearest 40 tasks leads to the results of 89% optimality as shown in Fig.5b. It empirically shows that less than 40% observation range provides solutions of better than 85% optimality. This property features practical benefits, especially in scaled-up cases, or with the communication of limited bandwidth. It indicates that a balance point exists at which limiting the size of the communication neighbourhood achieves better scalability without unduly affecting its optimality.

VI. Conclusion

Our focus in this paper is to investigate the VRP under restricted observability, where each agent is only informed by k-nearest neighbours. We first constructed a sparse-timescale Dec-MDP model of PO-VRPs, followed by a solution using the multi-agent cross-observability RL algorithm. The proposed algorithm shows a superior performance than the multi-agent policy gradient baseline. Furthermore, we provided the near-optimal bounds for the PO policy as a function of its total variance distance from the optimal policy. On the condition of imitation learning, the Theorem justifies the intuition that enlarging the observation range implies a diminishing marginal gain in VRPs. Simulation results show that even if the policy is not trained with pure imitation learning, the decaying pattern still exists. Another observation from simulation results is that the information of other agents has negligible influence on the results, which implies that the awareness of other agents' states may not be necessary in constructing a route. For further work, we will explore further deciding the optimal range of observation optimizing the multiple objectives of communication cost and planning performance. Also, a more efficient communication topology within the bounded neighbourhood is expected to be explored.

APPENDIX

A. Parameters in VRP-TWs

Mission parameters are shown in Table I and Table II. $\mathcal{U}(a,b)$ denotes a discrete uniform distribution on integer interval $[a,b]$; $\mathcal{U}\{x_1,\cdots,x_n\}$ denotes a discrete uniform distribution on finite set $\{x_1,\cdots,x_n\}$; and $\mathcal{B}(p)$ is a Bernoulli distribution of parameter p. Some of the mission settings are adapted from [5].

TABLE I: Generic parameter settings of VRPs

Parameter notation	Value or distribution
Planning horizon (min)	$H = 480$
Expected flight speed (km/min)	$V = 1$
Depot location (km)	$x^0, y^0 \sim \mathcal{U}(0,100)$
Customer locations (km)	$x^j, y^j \sim \mathcal{U}(0,100)$
Customer price	$r^j = 1$
Penalty for failed delivery	$c_{pen} = 1$

Due to the impact of airflows, the time cost on arcades between delivery sites is shown to be stochastic considering the wind-sensitive property of drones. We simulate the stochastic travel times by sampling the drones' speed from a normal distribution $v \sim \mathcal{N}\left(\mu_V, \sigma_V^2\right)$ where $\mu_V = 1km/min$, $\sigma_V = 0.2km/min$.

TABLE II: Parameter settings related to time windows.

Parameter notation	Value or distribution
Is the customer constrained by time window?	$\delta_{TW}^j = \mathcal{B}(p_{TW})$ where $p_{TW} \sim \mathcal{U}(0.25, 0.5, 0.75, 1.0)$
Customer ready time (min)	$t_r^j \sim \mathcal{U}\left(0, t_{\max}^j\right)$ where $t_{\max}^j = H - \frac{dist(0,j)}{V} - \tau^j$
Customer time window width (min)	$w_{TW}^j \sim \mathcal{U}(30,90)$
Customer service duration (min)	$\tau^j \sim \mathcal{U}(10,30)$
Customer due time (min)	$t_d^j = t_r^j + w_{TW}^j$

ACKNOWLEDGMENT

This work is supported by the Innovative UK-funded Future Flight project HADO, reference number 10024815.

REFERENCES

[1] E. Khalil, H. Dai, Y. Zhang, B. Dilkina, and L. Song, "Learning combinatorial optimization algorithms over graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[2] I. Bello, H. Pham, Q. V. Le, M. Norouzi, and S. Bengio, "Neural combinatorial optimization with reinforcement learning," *arXiv preprint arXiv:1611.09940*, 2016.

[3] K. Braekers, K. Ramaekers, and I. Van Nieuwenhuyse, "The vehicle routing problem: State of the art classification and review," *Computers & industrial engineering*, vol. 99, pp. 300–313, 2016.

[4] K. Gao and J. Yu, "Capacitated vehicle routing with target geometric constraints," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 7925–7930.

[5] G. Bono, J. S. Dibangoye, O. Simonin, L. Matignon, and F. Pereyron, "Solving multi-agent routing problems using deep attention mechanisms," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 12, pp. 7804–7813, 2020.

[6] B. Park, C. Kang, and J. Choi, "Cooperative multi-robot task allocation with reinforcement learning," *Applied Sciences*, vol. 12, no. 1, p. 272, 2021.

[7] M. Fernando, R. Senanayake, H. Choi, and M. Swany, "Graph attention multi-agent fleet autonomy for advanced air mobility," *arXiv preprint arXiv:2302.07337*, 2023.

[8] J. Park, C. Kwon, and J. Park, "Learn to solve the min-max multiple traveling salesmen problem with reinforcement learning," in *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, 2023, pp. 878–886.

[9] D. Wierstra, A. Foerster, J. Peters, and J. Schmidhuber, "Solving deep memory pomdps with recurrent policy gradients," in *Artificial Neural Networks–ICANN 2007: 17th International Conference, Porto, Portugal, September 9-13, 2007, Proceedings, Part I 17*. Springer, 2007, pp. 697–706.

[10] M. T. Spaan, "Partially observable markov decision processes," in *Reinforcement learning: State-of-the-art*. Springer, 2012, pp. 387–414.

[11] Y. Zhang and M. M. Zavlanos, "Cooperative multi-agent reinforcement learning with partial observations," *IEEE Transactions on Automatic Control*, 2023.

[12] L. Weihs, U. Jain, I.-J. Liu, J. Salvador, S. Lazebnik, A. Kembhavi, and A. Schwing, "Bridging the imitation gap by adaptive insubordination," *Advances in Neural Information Processing Systems*, vol. 34, pp. 19 134–19 146, 2021.

[13] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Overcoming exploration in reinforcement learning with demonstrations," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 6292–6299.

[14] H. H. Nguyen, A. Baisero, D. Wang, C. Amato, and R. Platt, "Leveraging fully observable policies for learning under partial observability," in *6th Annual Conference on Robot Learning*, 2022.

[15] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," *Advances in neural information processing systems*, vol. 28, 2015.

[16] M. Nazari, A. Oroojlooy, L. Snyder, and M. Takác, "Reinforcement learning for solving the vehicle routing problem," *Advances in neural information processing systems*, vol. 31, 2018.

[17] W. Kool, H. Van Hoof, and M. Welling, "Attention, learn to solve routing problems!" *arXiv preprint arXiv:1803.08475*, 2018.

[18] J. James, W. Yu, and J. Gu, "Online vehicle routing with neural combinatorial optimization and deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3806–3817, 2019.

[19] B. Lin, B. Ghaddar, and J. Nathwani, "Deep reinforcement learning for the electric vehicle routing problem with time windows," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11 528–11 538, 2021.

[20] J. Li, L. Xin, Z. Cao, A. Lim, W. Song, and J. Zhang, "Heterogeneous attentions for solving pickup and delivery problem via deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2306–2315, 2021.

[21] F. D. Hildebrandt, B. W. Thomas, and M. W. Ulmer, "Opportunities for reinforcement learning in stochastic dynamic vehicle routing," *Computers & operations research*, vol. 150, p. 106071, 2023.

[22] W. Joe and H. C. Lau, "Deep reinforcement learning approach to solve dynamic vehicle routing problem with stochastic customers," in *Proceedings of the international conference on automated planning and scheduling*, vol. 30, 2020, pp. 394–402.

[23] W. Pan and S. Q. Liu, "Deep reinforcement learning for the dynamic and uncertain vehicle routing problem," *Applied Intelligence*, vol. 53, no. 1, pp. 405–422, 2023.

[24] Q. Li, W. Lin, Z. Liu, and A. Prorok, "Message-aware graph attention networks for large-scale multi-robot path planning," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5533–5540, 2021.

[25] J. Chen, A. Baskaran, Z. Zhang, and P. Tokekar, "Multi-agent reinforcement learning for visibility-based persistent monitoring," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 2563–2570.

[26] D. Abel, D. Hershkowitz, and M. Littman, "Near optimal behavior via approximate state abstraction," in *International Conference on Machine Learning*. PMLR, 2016, pp. 2915–2923.

[27] D. B. Brown, J. E. Smith *et al.*, "Information relaxations and duality in stochastic dynamic programs: A review and tutorial," *Foundations and Trends® in Optimization*, vol. 5, no. 3, pp. 246–339, 2022.

[28] M. B. Haugh and O. R. Lacedelli, "Information relaxation bounds for partially observed markov decision processes," *IEEE Transactions on Automatic Control*, vol. 65, no. 8, pp. 3256–3271, 2019.

[29] R. Liu, H.-S. Shin, B. Yan, and A. Tsourdos, "An auction-based coordination strategy for task-constrained multi-agent stochastic planning with submodular rewards," *arXiv preprint arXiv:2212.14624*, 2022.