Routing a fleet of vehicles for decentralized reconnaissance with shared workload among regions with uncertain information

Yan Xia [1], Rajan Batta[2]
Department of Industrial and Systems Engineering
The State University of New York at Buffalo
Buffalo, NY 14260 USA

Rakesh Nagi[3]
Department of Industrial and Enterprise Systems Engineering,
University of Illinois at Urbana-Champaign,
Urbana, IL 61801,

Abstract

This paper studies the problem of controlling a fleet of vehicles to search and collect information reward within a specified mission time from a set of regions containing uncertain information. We seek a decentralized time-allocation policy using pre-calculated routes to maximize the total reward. We demonstrate that sharing regions among vehicles is beneficial. However, shared regions make the decentralized time-allocation problem computationally intractable. To overcome this, we develop an approximate formulation using an independency assumption. This approximate model allows us to decompose, by vehicle, the time-allocation problem, and obtain an easily implementable policy that takes on a Markovian form. We derive a tight upper bound for the decentralized time-allocation policy using the obtained Markovian policy. We also develop a sufficient condition under which the approximate formulation becomes exact. A numerical study establishes the computational efficiency of the method–only a few CPU seconds are needed for problems with a planning horizon of 300 time units and 40 regions, and demonstrates the benefit of using a region-sharing strategy. The numerical study also examines the fleet's workload sharing behavior with respect to the cooperation factor (which measures the fused information reward gained from sharing), the mission duration and the search sequence.

Keywords: Search Theory; Decentralized Control; Resource Allocation; Markovian Policy; Multi-agent System

[1]E-mail: yanxia@buffalo.edu
[2]E-mail: batta@buffalo.edu
[3]E-mail: nagi@illinois.edu

1. Introduction, Motivation and Contribution

Information collection is a prerequisite in various real world operations. In military operations, a mission commander's decisions are typically made based on information about the enemy positions and assets, which usually refer to the existence of military targets such as arsenal, radar station, and airport. Collection of the information can, for example, correspond to a fixed period of video surveillance. In disaster relief operations, the relief allocation is usually decided based on information about the damage to the area where information can be the existence of casualties in a region and collection of the information can be identifying the size of the casualty group (Gong and Batta, 2007). In rescue management, prior information is collected to facilitate rescue planning. For instance, in the search of the Malaysia missing plane MH370 (BBC, 2014), the search team attempts to localize as many pieces of suspicious debris as possible over the sea. The debris is then checked by rescue ships in hope of finding the plane. In forest fire fighting, besides knowing whether a wildfire exists, information about the spreading direction and the level of the fire is also important for fighting wildfires (Merino et al., 2006). In border control (Pietz and Royset, 2013), it is important to identify smugglers (information) and track them (collection) so that smuggling can be stopped by the coast guard. In space exploration, one important task is to collect environmental information from different sites of a planet (Becker et al., 2004).

Information collection is traditionally performed by manned aircraft such as helicopters or ground vehicles, which can be very expensive. Recent development of automatic agents such as unmanned aerial vehicles (UAVs) provides a more economical solution to collect information from a large area with disjoint regions of interest (Romesh, 2013; Wayne, 2014). This paper considers a reconnaissance problem of controlling a fleet of vehicles (agents) to collect information from a set of regions. In each region, information may or may not exist, and if the information exists, it takes a vehicle a random amount of time to detect it. After detecting it, the vehicle can decide whether to collect the information, which takes a given amount of time and provides a reward (a measurement of the information's value) to the fleet. The goal is to maximize the total reward collected by the fleet within a given mission time.

Two commonly used control policies are centralized and decentralized. Our focus is on a decentralized policy. Clearly, a centralized policy, by definition, provides better performance than a decentralized policy; however, it requires that a central agent through online communication collects real-time observations from individual members, processes the observations, and returns the control decisions to individual member. These requirements create several drawbacks. First, the centralized method lacks robustness due to communication loss (Seiler and Sengupta, 2001), under which scenario some individual members cannot contact the central agent. Second, transfer of each individual member's real-time observations to the central agent may be restricted by limited bandwidth. Under either situation, the central agent cannot collect all the necessary observations to make a decision for each member and centralized control fails. Third, deciding a centralized control policy has high computational complexity and is difficult to implement efficiently (Shima and Rasmussen, 2009). Finally, security is a concern since the communication network can be hacked and sensitive messages may be revealed to the enemy, which creates a risk to the mission (Howard, 2013). Even when centralized

policies can be implemented, strategies need to be developed for operations over a period of time when communication is lost between vehicles and the central agent. Decentralized strategies provide a way to operate in such a situation, with the understanding that once communications are restored a switch is made back to a centralized policy.

The decentralized control problem studied in this paper can be viewed as a finite horizon partially observable decentralized Markov decision process (DEC-POMDP), for which Seuken and Zilberstein (2008) provide an excellent review. Optimizing such a process is proved to be NEXP-hard (Bernstein et al., 2002). Several exact and approximate methods are designed to solve a DEC-POMDP problem. Szer et al. (2005) develop a search heuristic based on the widely applied A^* algorithm (Hart et al., 1968), and in Szer and Charpillet (2006) they propose a dynamic programming based approach, which can be executed exactly or approximately. Oliehoek et al. (2008b) apply approximate Q-learning to a general DEC-PODMP problem, which is a classical approximate dynamic programming method (Powell, 2007). As a meta-heuristic to the problem, Oliehoek et al. (2008a) implement the cross-entropy method (Kroese, 2010) and compare the method's performance with another heuristic called joint equilibrium search for policies (JESP), which is designed for solving general DEC-PODMP problems (Nair et al., 2003). JESP starts from any feasible policy and iteratively improves the policy to an equilibrium where no agent can improve its policy by itself. Aras and Dutech (2010) study the problem using a mixed integer linear programming (MILP) approach, which can explore the high-efficiency computation provided by commercial MILP solvers such as CPLEX (IBM, 2014). These reviewed methods provide policies with good qualities but their applicability is limited to small problems with fewer than 10 planning horizons but we attempt to solve realistic size problem far beyond these methods' computational capability.

General DEC-POMDPs require policies with history-dependent actions. An implementation in our context will require exponential space. Therefore, we seek an alternative method and use a policy whose action is only dependent on the vehicle's remaining mission time. As illustrated later such a policy only takes bilinear space, and generates a sequence of look-up tables, one for each region-vehicle pair. In each look-up table, a decision is stored for each decision epoch (introduced in the next section) and the remaining time. The policy is straightforward to implement: Whenever a vehicle reaches a decision epoch, it simply executes the action listed in the look-up table. We develop a two-stage solution procedure to obtain such a policy. We plan routes in the first stage to determine the regions for each vehicle and the sequence to visit them. Multiple *route families* are generated in this stage, each of which is composed of one route for each fleet member. In the second stage, we evaluate each route family using a decentralized time-allocation policy, which provides entries to each lookup table. To obtain the time-allocation policy for a route family, we need to solve a DEC-POMDP problem. The final policy is established by finding the time-allocation policy that provides the maximal expected reward to the fleet.

The first stage of the solution procedure relates to a stream of literature that uses deterministic models to analyze the problem of controlling a fleet of UAVs in reconnaissance missions, e.g., Chao et al. (1996), Schumacher et al. (2006), Rathinam et al. (2007), Kress and Royset (2008), Murray and Karwan (2010), Mufalli et al. (2012), and Pietz and Royset (2013). The models established in these papers consider different constraints and objectives but their final solution all assign a route to

each vehicle, which is composed of a sequence of way-points. Depending on the objectives, some of the models also decide the amount of time spent in each of the way-points for each vehicle to collect "reward", e.g., Kress and Royset (2008), Mufalli et al. (2012), and Pietz and Royset (2013). The main difference between these models and our work is the following: The objective value (e.g., the expected total reward collected by the fleet) is automatically calculated in a deterministic model once the routes are decided; however, in our problem we need to further solve a decentralized time-allocation problem to know how much reward the fleet can collect from a route family. As opposed to deterministic models, our approach creates a demonstrated need for sharing regions. It is illustrated in our numerical studies that the extent to when sharing occurs and which regions get selected for sharing is a function of the cooperation factor, the mission duration, and the search sequence for each vehicle.

We also make methodological contributions to solve the DEC-POMDP problem required by the second stage of the solution procedure, which are:

- Design of an approximate formulation to obtain an efficient decentralized time-allocation policy for each route family under an independency assumption.
- Derivation of a tight upper bound for the decentralized time-allocation policy.
- Development of a sufficient condition for the approximate formulation to be exact.

The rest of this paper is organized as follows: §2 presents the model description. §3 explains how to create route families. §4 formulates and solves the decentralized time-allocation problem for a given route family. For an arbitrary route family, a closed form upper bound for the decentralized time-allocation policy is developed in §5. The numerical study is presented in §6. Finally, we make our concluding discussion and propose future research directions in §7.

To enhance readability and conciseness of the paper, much of the material is presented in the appendix: §A provides integer programming formulations for the routing models introduced in §3 of the main article; §B provides an algorithm that extracts two types of quantities from a time-allocation policy, which are used in §4 of the main article; §C contains the proofs of some theorems and propositions that are omitted in the main article. §D shows the tightness of the bound developed in §5.

2. Model Description

Consider a reconnaissance problem in which a fleet of vehicles $\mathcal{U} = \{1, 2, ..., n\}$ is assigned to search and collect information from a set of regions $\mathcal{A} = \{1, 2, ..., L\}$. Each vehicle $i \in \mathcal{U}$ is assigned a start depot and an end depot, and needs to visit a subset of regions in $\mathcal{A}$ and arrive at its end depot within a given mission time T. We use the *remaining time* (t) of a vehicle to represent the time that a vehicle has left to reach its end depot. Time is modeled as a discrete entity.

At most one piece of information exists in region $\alpha \in \mathcal{A}$ with *a priori* probability e_α. Consider a vehicle that arrives at region α and reserves $\tilde{x}$ units of time to search for information. The actual time it consumes to detect the information is a discrete random variable whose value is x with probability $e_\alpha p_\alpha(x),\ x = 1, 2, ..., \tilde{x}$ where $p_\alpha(x)$ is the conditional probability of finding the information in the

x^{th} unit time given that the information exists. The vehicle may also fail to detect any information after spending $\tilde{x}$ units of time with probability $1 - e_\alpha P_\alpha(\tilde{x})$, where we call $P_\alpha(\tilde{x}) = \sum_{x=1}^{\tilde{x}} p_\alpha(x)$ as the *detection function* of a region. An example of how to obtain $P_\alpha(x)$ is provided in our numerical study (§6). If the information is found, the vehicle can either collect it (which takes $s_\alpha \geq 0$ units of time) or leave the region without collecting the information (which is instantaneous).

For region α, we define a random variable θ_α that represents whether information exists ($\theta_\alpha = 1$) or not ($\theta_\alpha = 0$). We also define a random variable w_α^i that indicates the time that vehicle i needs to detect the information in region α. If vehicle i schedules $\tilde{x}$ units of time to search for information in region α, the actual time it spends on searching is $\min\{w_\alpha^i, \tilde{x}\}$. If $\tilde{x} \geq w_\alpha^i$, the vehicle will detect the target; otherwise, it will not. Let $f_\chi(x)$ be the probability that an arbitrary random variable χ takes the value x and $f_\chi(x|\cdot)$ be the corresponding conditional probability given any condition specified in "$\cdot$".

When $\theta_\alpha = 1$ (information exists), we have

$$f_{w_\alpha^i}(w|\theta_\alpha = 1) = p_\alpha(w), \text{ for } w = 0, 1, 2, ...$$

When $\theta_\alpha = 0$ (information does not exist), we have

$$f_{w_\alpha^i}(w|\theta_\alpha = 0) = \begin{cases} 1 & \text{if } w = T + 1, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Under (1), the vehicle can never detect the information when it does not exist and consumes all its reserved search time since the vehicle can never schedule more than T units of time to search in a region. To simplify the notation, throughout the reminder of this paper we use w_α^i and θ_α to represent both the random variable and its realization.

If the vehicle does not detect any information after spending x units of time in searching, the posterior probability that $\theta_\alpha = 1$ can be obtained using Bayes' rule. We define an observation variable o_α^i given that $\tilde{x}$ units of time are reserved for searching for information in region α, which satisfies

$$o_\alpha^i = \begin{cases} \tilde{x} & \text{if } \tilde{x} < w_\alpha^i, \\ -1 & \text{if } \tilde{x} \geq w_\alpha^i. \end{cases} \tag{2}$$

In (2), $o_\alpha^i \geq 0$ means that the information is not found and o_α^i units of time have been spent in searching and $o_\alpha^i = -1$ means that the information is found. The conditional probability that $\theta_\alpha = 1$ given o_α^i is

$$f_{\theta_\alpha}(1|o_\alpha^i) = \begin{cases} \frac{e_\alpha(1-P_\alpha(o_\alpha^i))}{e_\alpha(1-P_\alpha(o_\alpha^i))+1-e_\alpha} & \text{if } o_\alpha^i \geq 0, \\ 1 & \text{if } o_\alpha^i = -1. \end{cases} \tag{3}$$

We make the following assumption for assigning regions to vehicles.

Assumption 1 *A region can be assigned to at most two vehicles.*

If a region α is searched by a single vehicle, a fixed reward g_α is collected. If two vehicles successfully collect the information, a reward of $(1 + \gamma_\alpha)g_\alpha$ is obtained by the fleet, where $\gamma_\alpha \geq 0$ is the *cooperation*

factor of the vehicles in region α. Information fusion (a technique that merges the information from heterogeneous sources) can be used to obtain an appropriate value of γ_α (Nakamura et al., 2007). The cooperation factor provides a quantitative measure of the extra reward gained by fusing the information.

We use $P(\cdot)$ to represent the probability that event "$\cdot$" will happen. For the search process of each vehicle in each of its assigned regions, we make the following assumption.

Assumption 2 *For any subset $\mathcal{U}' \subset \mathcal{U}$ and $\mathcal{A}^i \subset \mathcal{A}, i \in \mathcal{U}'$, we have*

$$P(\{w_\alpha^i\}_{\alpha\in\mathcal{A}^i, i\in\mathcal{U}'}, \{\theta_\alpha\}_{\alpha\in\bigcup_{i\in\mathcal{U}'}\mathcal{A}^i}) = \prod_{\alpha\in\bigcup_{i\in\mathcal{U}'}\mathcal{A}^i} f_{\theta_\alpha}(\theta_\alpha) \prod_{i\in\mathcal{U}', \alpha\in\mathcal{A}^i} f_{w_\alpha^i}(w_\alpha^i|\theta_\alpha). \tag{4}$$

Equation (4) implies three types of independency. Existence of information in a region is independent of existence of information in other regions. A similar independency applies to the time needed to detect information in a region. Lastly, for any shared region $\alpha \in \mathcal{A}^i \bigcap \mathcal{A}^j$, the detection time is independent for both assigned vehicles given θ_α.

Our decentralized policy is constructed under the following considerations:

1. Each region is visited by a vehicle at most once.
2. Each vehicle follows a specified route, composed of a sequence of regions from lower order to higher order. During the mission, the vehicle is only allowed to travel from a lower order region to a higher order region. (Note that the order of a region can be different for different vehicles.)

With these considerations, we represent vehicle i's route as $H^i = \{h_0^i, h_1^i, h_2^i, ..., h_{L_i}^i, h_{L_i+1}^i\}$, $i \in \mathcal{U}$. $h_l^i, 1 \le l \le L_i$, indicates the index of a region and l is the *order index* of the region where a larger l means a higher order. h_0^i and $h_{L_i+1}^i$ are the start and end depots, respectively. The routes in a route family are created such that each region is either visited by one or by two vehicles. This implies that for each vehicle $i \in \mathcal{U}$, we can divide the regions that it visits into two sets, $\mathcal{S}_i$ (shared) and $\mathcal{O}_i$ (not shared).

Given the route, a vehicle has at most three decision epochs (in each region): when the vehicle arrives, when the information is detected, and when the vehicle decides to leave. Consider a region α that vehicle i is assigned, $x_\alpha^i(\cdot)$ is an integer that specifies the amount of search time to reserve for searching; $y_\alpha^i(\cdot)$ is a binary variable that specifies whether the information should be collected ($y_\alpha^i(\cdot) = 1$) or not ($y_\alpha^i(\cdot) = 0$); $z_\alpha^i(\cdot)$ is an integer which indicates the order index of the next region to visit. At the start depot we only have $z_\alpha^i(T)$ which determines the first region to visit if the fleet is given T units of mission time.

3. Routing with Shared Regions

The first stage of the solution procedure is to create route families. An initial route family can be obtained by using a suitable deterministic model. We provide two examples in §A. For each initial route family, we use a *Minimal Insertion Rule* to select and assign shared regions according to a

threshold δ. Let $H^i = \{h_0^i, h_1^i, h_2^i, ..., h_L^i, h_{L_i+1}\}$ be vehicle i's route assigned by the initial route family, and $d(\alpha_1, \alpha_2)$ represents the travel time between region α_1 and region α_2. The travel cost to insert a region α to route H^i is $d_M(\alpha, H) = \min_{k=0,1,...,L_i}\{d(h_k, \alpha) + d(\alpha, h_{k+1}) - d(h_k, h_{k+1})\}$. Let $\widetilde{\mathcal{O}}^i$ be the set of non-shared regions in vehicle i's initial route. We define the selection rule:

Definition 1 *(Minimal Insertion Rule) For $i \in \mathcal{U}$, assign region $\alpha \in \widetilde{\mathcal{O}}^i$ as a shared region to vehicle $j \neq i$ if $d_M(\alpha, H^j) < \delta$ and $d_M(\alpha, H^j) = \min_{j' \neq i}\{d_M(\alpha, H^{j'})\}$, where ties are broken arbitrarily.*

Varying the threshold δ of the *Minimal Insertion Rule*, different sets of shared regions can be created. To enumerate all possible combinations, we can increase δ one unit at a time from zero to an upper bound, under which all regions are shared.

Observation 1 *If $d_{max} = \max_{\alpha_i, \alpha_j \in \mathcal{A}} d(\alpha_i, \alpha_j)$, $d_M(\alpha, H^i) \leq 2d_{max}$ for $\forall i \in \mathcal{U}, \alpha \in \mathcal{A}$.*

Since for any regions $\alpha_1, \alpha_2, \alpha_3$, we have $d(\alpha_1, \alpha_2) + d(\alpha_2, \alpha_3) - d(\alpha_1, \alpha_3) \leq 2d_{max} - 0$, Observation 1 holds. According to Observation 1, any region in the area will be shared if $\delta \geq d_{max}$.

For each set of shared regions that a vehicle receives, we re-optimize the vehicle's route. To do so, we consider two methods: Unconstrained and constrained re-optimization. The unconstrained re-optimization method seeks a minimal travel time route for each vehicle to visit all its assigned regions. The constrained re-optimization pursues a minimal travel time route but retains the orders of the initial regions, i.e., if a region has a lower order than another region in the initial route, it must retain a lower order than that region in the re-optimized route. We will explain the reason to perform constrained re-optimization in §4.3. The integer programming formulations of these two re-optimization methods are presented in §A.

4. Finding a decentralized time-allocation policy

To streamline presentation of the material, we divide this section into four parts. §4.1 formulates the decentralized time-allocation problem. §4.2 develops a decomposable approximation for the formulation in §4.1. §4.3 contains an algorithm to find a locally optimal solution. Finally, §4.4 develops a sufficient condition for the approximation in §4.2 to be exact, and also explains calculation of the exact value of a Markovian policy. All of these elements are integral to our approach for finding a decentralized time-allocation policy.

4.1 The decentralized time-allocation problem

To simplify notation, we use a two-vehicle setting to establish the results in the following two sections. The model and its results can be naturally extended to the multi-vehicle scenario, for which we will provide an explanation in the conclusion. Note that when there are two vehicles, we have $\mathcal{U} = \{1, 2\}$ and $\mathcal{S} = \mathcal{S}_1 = \mathcal{S}_2$ and we use the notation interchangeably.

Assume that each vehicle i has been assigned a route $H^i = \{h_0^i, h_1^i, h_2^i, ..., h_{L_i}^i, h_{L_i+1}^i\}$. The time-allocation problem for a given route family is to determine a decentralized time-allocation policy that

provides the maximum expected reward. We represent vectors in bold and define $\mathbf{o}^i_{h^i_l} = (o^i_{h^i_1}, o^i_{h^i_2}, ..., o^i_{h^i_l})$ for $1 \leq l \leq L_i$. We also define $\mathbf{o}^i_{h^i_0} \equiv 0$ for the consistency of the notation, which has no practical meaning. The form of the optimal time-allocation policy is given in the following proposition.

Proposition 1 *(Oliehoek et al., 2008b) There exists an optimal decentralized time-allocation policy $\pi = \{\pi_i\}_{i\in\mathcal{U}}$ such that $\pi_i,\ i \in \mathcal{U}$, can be represented as*

$$\pi_i = \{z^i_{h^i_0}(T);\ x^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}), y^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}), z^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}) : t = 0, 1, 2, ..., T, l = 1, 2, ..., L_i\}. \tag{5}$$

Proposition 1 is a direct result of Proposition 2.1 in Oliehoek et al. (2008b).

The feasibility of a time-allocation policy is restricted by the vehicle's remaining time and the orders of the regions specified by the given route. More explicitly, for a policy to be feasible, the vehicle should always have enough time to reach the end depot. In addition, when the vehicle decides to leave a region, it can only travel to the region that has a higher order in the given route.

Observation 2 *A policy $\pi = \{\pi_i\}_{i\in\mathcal{U}}$ in the form of (5) is feasible if and only if for $i \in \mathcal{U}$*

$$T - d(h^i_0, h^i_{z^i_{h^i_0}(T)}) \geq d(h^i_{z^i_{h^i_0}(T)}, h^i_{L_i+1}),\ L_i + 1 \geq z^i_{h^i_0}(T) > 0, \tag{6a}$$

$$t - x^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}) \geq d(h^i_l, h^i_{L_i+1}),\ l = 1, ..., L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}), \forall \mathbf{o}^i_{h^i_{l-1}}, \tag{6b}$$

$$t - y^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}) s_{h^i_l} \geq d(h^i_l, h^i_{L_i+1}),\ l = 1, ..., L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}), \forall \mathbf{o}^i_{h^i_{l-1}}, \tag{6c}$$

$$t - d(h^i_l, h^i_{z^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}})}) \geq d(h^i_{z^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}})}, h^i_{L_i+1}),\ l = 1, ..., L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}), \forall \mathbf{o}^i_{h^i_{l-1}}, \tag{6d}$$

$$z^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}) > l,\ l = 0, 1, ..., L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}), \forall \mathbf{o}^i_{h^i_{l-1}}. \tag{6e}$$

We use $\widehat{\Pi}_i$ to represent the set of all feasible policies of vehicle i in the form of (5).

Constraint (6a) states the initial condition that the vehicle can only start with a region if it still has enough time to reach the end depot after arriving at the region. Constraints (6b) and (6c) state that the vehicle has to reserve enough time to travel to the end depot when it decides to search for or collect information in a region. Constraint (6d) ensures that if the vehicle decides to travel to region h^i_k where $k = z^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}})$, it must have enough time to reach the end depot after arriving at the region. Constraint (6e) requires that the vehicle can only travel to a region that has a higher order than its current region.

Throughout the paper, a vehicle succeeds in a region means that the vehicle detects information in a region and collects it. For a feasible policy π_i of vehicle i, we use $\tau^i_\alpha(\pi_i)$ to represent the conditional probability that the vehicle succeeds in region α given that information exists in the region. The expected reward collected from a non-shared region $\beta \in \mathcal{O}_i$ is

$$R_\beta(\pi_i) = e_\beta g_\beta \tau^i_\beta(\pi_i).$$

For a decentralized policy π we let $\tau^1_\alpha(\pi|1)$ be the conditional probability that vehicle 1 succeeds in region α given that information exists and vehicle 2 also succeeds, and $\tau^1_\alpha(\pi|0)$ be the conditional

probability that vehicle 1 succeeds in region α given that information exists but vehicle 2 does not succeed. The expected reward that the fleet collects from a shared region $\alpha \in \mathcal{S}$ under policy π is

$$R_\alpha(\pi) = e_\alpha g_\alpha[\tau_\alpha^1(\pi|0)(1-\tau_\alpha^2(\pi_2)) + (1-\tau_\alpha^1(\pi|1))\tau_\alpha^2(\pi_2) + (1+\gamma_\alpha)\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2)].$$

For each vehicle $i \in \mathcal{U}$, we define $R^{\pi_i} = \sum_{\beta \in \mathcal{O}_i} R_\beta(\pi_i)$ as the expected reward that vehicle i collects from all its non-shared regions under policy π_i. The expected total reward collected by the fleet under policy π is

$$R(\pi_i) = \sum_{i \in \mathcal{U}} R^{\pi_i} + \sum_{\alpha \in \mathcal{S}} R_\alpha(\pi). \tag{7}$$

Then we can formulate the time-allocation problem as

$$\begin{aligned} &\max R(\pi_i) \\ s.t. \quad &\pi_1 \in \widehat{\Pi}_1, \pi_2 \in \widehat{\Pi}_2. \end{aligned} \tag{8}$$

To solve the problem defined by (8), we need to optimize a DEC-POMDP, which is impractical considering the size of the problem that we attempt to solve. Furthermore, a policy in the form of (5) is not only difficult to compute but also expensive to store since the number of possible realizations of $\mathbf{o}^i_{h^i_l}$ is exponential in l and we need to determine an action for each combination of the realization $\mathbf{o}^i_{h^i_l}$ and the corresponding remaining time t. With these difficulties associated with a general policy in the form of (5), we focus on the implementation of *Markovian* policies.

Definition 2 *A decentralized policy $\pi = \{\pi_i\}_{i \in \mathcal{U}}$ is a* Markovian *policy, if π_i, $i \in \mathcal{U}$, can be represented as*

$$\pi_i = \{z^i_{h^i_0}(T);\ x^i_{h^i_l}(t), y^i_{h^i_l}(t), z^i_{h^i_l}(t) : t = 0,1,2,...,T, l = 1,2,...,L_i\}.$$

Similar to (6), for a Markovian policy to be feasible, we have the following observation.

Observation 3 *A Markovian policy $\pi = \{\pi_i\}_{i \in \mathcal{U}}$ is feasible if and only if for $i \in \mathcal{U}$*

$$\begin{aligned}
&T - d(h^i_0, h^i_{z^i_{h^i_0}(T)}) \geq d(h^i_{z^i_{h^i_0}(T)}, h^i_{L_i+1}),\ L_i + 1 \geq z^i_{h^i_0}(T) > 0, \\
&t - x^i_{h^i_l}(t) \geq d(h^i_l, h^i_{L_i+1}),\ l = 1,...,L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}), \\
&t - y^i_{h^i_l}(t) s_{h^i_l} \geq d(h^i_l, h^i_{L_i+1}),\ l = 1,...,L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}), \\
&t - d(h^i_l, h^i_{z^i_{h^i_l}(t)}) \geq d(h^i_{z^i_{h^i_l}(t)}, h^i_{L_i+1}),\ l = 1,...,L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}), \\
&L_i + 1 \geq z^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}) > l,\ l = 1,...,L_i, T \geq t \geq d(h^i_l, h^i_{L_i+1}).
\end{aligned}$$

We use Π_i to represent the set of all feasible Markovian policies of vehicle i.

The size of a Markovian policy is bilinear on L_i and T but it is still difficult to find the optimal Markovian policy since it requires a forward induction over the entire Markovian policy space given by (9). Nevertheless, a sub-optimal Markovian policy can be obtained by replacing the objective function of (8) with an approximation and decomposing the optimization problem into sub-problems each of which can be considered as a single vehicle problem.

4.2 Formulation of a Decomposable Approximation

The approximation formulation is developed by using an approximate objective function. The expected reward collected from a shared region α under a policy π is approximated as

$$\widehat{R}_\alpha(\pi) = e_\alpha g_\alpha[\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1)) + (1+\gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)]. \tag{10}$$

The approximation (10) is created based on the assumption that *under any decentralized policy π in the form of (5), whether a vehicle succeeds in a shared region α is conditionally independent of the other vehicle given that information exists in the region.* We should note that this assumption generally does not hold but we will provide a sufficient condition in §4.4 under which this assumption is satisfied. With (10) the expected reward of applying policy π is approximated as

$$\widehat{R}(\pi) = \widehat{R}(\pi_1, \pi_2) = R^{\pi_1} + R^{\pi_2} + \sum_\alpha \widehat{R}_\alpha(\pi). \tag{11}$$

Under (11) the fleet's time-allocation problem is re-formulated as:
(IAP): Independent approximation problem.

$$\begin{aligned} &\max \widehat{R}(\pi) \\ s.t. \quad &\pi_1 \in \widehat{\Pi}_1, \pi_2 \in \widehat{\Pi}_2. \end{aligned}$$

We first consider a subproblem of problem (IAP), (IAP^i): $\max\{\widehat{R}(\pi) : \pi_i \in \widehat{\Pi}_i | \pi_j, j \neq i\}$, which optimizes vehicle i's policy by fixing the policy applied by the other vehicle. To prove our main theorem we need the following lemma, which shows the fact that if the policy played by one vehicle is fixed, the maximal expected reward that the other vehicle can collect for the fleet at any decision epoch is only determined by the decision epoch and the vehicle's remaining time given the objective function (11).

Lemma 1 *Assume that vehicle i always collects a reward of $(1-\tau_\alpha^j(\pi_j)) + \gamma_\alpha \tau_\alpha^j(\pi_j)$ in a shared region α given that the other vehicle applies policy π_j. For $l = 1, 2, ..., L_i$, $T \geq t \geq d_{h_l^i, h_{L_i+1}^i}$ there exist:*

1. *$G_{h_l^i}^s(t)$, $T \geq t \geq d_{h_l^i, h_{L_i+1}^i}$: The maximal expected (future) reward that vehicle i can collect if it arrives at region h_l^i with t units of remaining time.*

2. *$G_{h_l^i}^c(t)$, $T \geq t \geq d_{h_l^i, h_{L_i+1}^i}$: The maximal expected (future) reward that vehicle i can collect if it detects the information in region h_l^i with t units of remaining time.*

3. *$G_{h_l^i}^d(t)$, $T \geq t \geq d_{h_l^i, h_{L_i+1}^i}$: The maximal expected (future) reward that vehicle i can collect if it leaves region h_l^i with t units of remaining time.*

Proof. Since no information reward can be collected at the end depot, we can define $G_{h_{L_i+1}^i}^s(t) = 0$ for $t = 0, 1, 2, ..., T$, which is also the maximal expected reward that vehicle i can collect when it arrives at the end depot with t units of remaining time. We will prove the lemma using induction. As the

induction hypothesis, we assume that for $L_i + 1 \geq l > k$, $G^s_{h^i_l}(t),\ d(h^i_l, h^i_{L_i+1}) \leq t \leq T$ is well defined. We will show that $G^s_{h^i_l}(t)$, $G^c_{h^i_l}(t)$ and $G^d_{h^i_l}(t),\ d(h^i_l, h^i_{L_i+1}) \leq t \leq T$ are also well defined for $l = k$.

When vehicle i decides to leave region h^i_k with t units of remaining time, it can travel to region $h^i_{k'}$ if $k' > k$ and $t - d(h^i_k, h^i_{k'}) \geq d(h^i_{k'}, h^i_{L_i+1})$. If the vehicle decides to travel to region $h^i_{k'}$, according to the induction hypothesis, it will collect a maximal expected reward of $G^s_{h^i_{k'}}(t)$. Because the vehicle has to travel to one of the regions in $\{h^i_{k+1}, ..., h^i_{L^i_i+1}\}$, the maximal expected future reward that the vehicle can collect is

$$G^d_{h^i_k}(t) = \max_{k'} \left\{ G^d_{h^i_{k'}}(t - d(h^i_k, h^i_{k'})) : L_{i+1} \geq k > l,\ t - d(h^i_k, h^i_{k'}) \geq d(h^i_{k'}, h^i_{L_i+1}) \right\}. \tag{12}$$

For $T \geq t \geq d(h^i_k, h^i_{L_i+1})$, since $t - d(h^i_k, h^i_{L_i+1}) \geq 0 = d(L_i + 1, L_i + 1)$, $G^d_{h^i_k}(t)$ is well defined.

For $G^c_{h^i_k}(t)$, we consider two possible cases:

1. $h^i_k \in \mathcal{O}_i$ (region h^i_k is a non-shared region): If the vehicle decides to collect the information, the vehicle collects an immediate reward of $g_{h^i_k}$ and consumes $s_{h^i_l}$ units of time.

2. $h^i_k \in \mathcal{S}$ (region h^i_k is a shared region): If the vehicle decides to collect the information, due to the lemma's assumption, the vehicle collects a reward of $g_{h^i_k}(\gamma_{h^i_k} \tau^j_{h^i_k}(\pi_j) + 1 - \tau^j_{h^i_k}(\pi_j))$ and consumes $s_{h^i_k}$ units of time.

Under both cases, if the vehicle decides to leave without collecting, the vehicle collects no immediate reward and consumes no time. After either decision, the vehicle leaves region h^i_k, and the maximal expected reward that the vehicle can collect afterwards is given by $G^d_{h^i_k}(\cdot)$. We define

$$\hat{\tau}^i_{h^i_k} = \begin{cases} 0 & \text{if } h^i_k \in \mathcal{O}_i, \\ \tau^j_{h^i_k}(\pi_j) & \text{if } h^i_k \in \mathcal{S}. \end{cases}$$

Since the vehicle can only collect the information if $t - s_{h^i_k} \geq d(h^i_k, h^i_{L_i+1})$, the maximal expected future reward that the vehicle can collect is

$$G^c_{h^i_k}(t) = \begin{cases} G^d_{h^i_k}(t) & \text{if } d(h^i_k, h^i_{L_i+1}) \leq t < d(h^i_k, h^i_{L_i+1}) + s_{h^i_k}, \\ \max\left\{ G^d_{h^i_k}(t), G^d_{h^i_k}(t - s_{h^i_k}) + g_{h^i_k}(\gamma_{h^i_k} \hat{\tau}^i_{h^i_k} + (1 - \hat{\tau}^i_{h^i_k})) \right\} & \text{if } d(h^i_k, h^i_{L_i+1}) + s_{h^i_k} \leq t \leq T. \end{cases} \tag{13}$$

Since $G^d_{h^i_k}(t)$ is well defined for $T \geq t \geq d(h^i_k, h^i_{L_i+1})$, $G^c_{h^i_l}(t)$ is well defined for $T \geq t \geq d(h^i_k, h^i_{L_i+1})$.

When the vehicle arrives at region h^i_k, it decides the maximal amount of time ($\tilde{x}$) to search for the information. The information can be found after spending $t = 1, 2, .., \tilde{x}$ units of time and then the vehicle needs to decide whether to collect the information, under which scenario the maximal expected reward that the vehicle can collect is given by $G^c_{h^i_l}(\cdot)$. The vehicle may also fail to find the information under which scenario zero reward is collected from region h^i_k and it will leave the region with the expected future reward given by $G^d_{h^i_k}(\cdot)$. Since the vehicle cannot schedule more than $t - d(h^i_k, h^i_{L_i+1})$

units of time to search for the information, the maximal expected reward that the vehicle can collect is

$$G^s_{h^i_l}(t) = \max_{0\le\tilde{x}\le t-d(h^i_k,h^i_{L_i+1})} \left\{ \sum_{x=1}^{\tilde{x}} e_{h^i_k} p_{h^i_k}(x) G^c_{h^i_k}(t-x) + (1-e_{h^i_k}P_{h^i_{\hat{k}}}(\tilde{x}))G^d_{h^i_k}(t-\tilde{x}) \right\}. \tag{14}$$

Since $\tilde{x}=0$ is always a feasible decision when $t \ge d(h^i_k, h^i_{L_i+1})$, $G^s_{h^i_l}(t)$ is well defined for $T \ge t \ge d(h^i_k, h^i_{L_i+1})$. The induction is complete and the lemma is proved. □

Our main result establishes a Markovian policy that solves problem (IAP^i).

Theorem 1 *There is a Markovian policy $\pi_i \in \Pi_i$ that solves problem (IAP^i) given $\pi_j \in \widehat{\Pi}_j$.*

Proof. Given $G^s_{h^i_l}(t)$, $G^c_{h^i_l}(t)$, and $G^d_{h^i_l}(t)$ defined for $l = 1, 2, ..., L_i+1$, we can design a Markovian policy as follows: For $l = 1, 2, ..., L_i$ and $T \ge t \ge d(h^i_k, h^i_{L_i+1})$:

$$x^i_{h^i_l}(t) = \arg \max_{0\le\tilde{x}\le t-d(h^i_l,h^i_{L_i+1})} \left\{ \sum_{x=1}^{\tilde{x}} e_{h^i_l} p_{h^i_l}(x) G^c_{h^i_l}(t-x) + (1-e_{h^i_l}P_{h^i_{\hat{l}}}(\tilde{x}))G^d_{h^i_l}(t-\tilde{x}) \right\}, \tag{15a}$$

$$y^i_{h^i_l}(t) = \begin{cases} 1 & \text{if } G^c_{h^i_l}(t) = G^d_{h^i_l}(t-s_{h^i_l}) + g_{h^i_l}(\gamma_{h^i_l}\hat{\tau}^i_{h^i_l} + (1-\hat{\tau}^i_{h^i_l})), \\ 0 & \text{otherwise.} \end{cases} \tag{15b}$$

$$z^i_{h^i_l}(t) = \arg\max_k \left\{ G^d_{h^i_k}(t-d(h^i_l,h^i_k)) : L_{i+1} \ge k > l,\ t-d(h^i_l,h^i_k) \ge d(h^i_k,h^i_{L_i}) \right\}. \tag{15c}$$

Under the policy specified by (15), the maximal expected rewards defined by $G^s_{h^i_l}(t)$, $G^c_{h^i_l}(t)$ and $G^d_{h^i_l}(t)$ can be achieved for each decision epoch and the corresponding remaining time. When $l=0$, similar to what we did for $l>0$, the maximal expected reward that the vehicle can collect from regions $h^i_1, h^i_2, ..., h^i_{L^i_i+1}$ given that vehicle i leaves region h^i_0 with T units of remaining time is

$$G^d_{h^i_0}(T) = \max_k \left\{ G^d_{h^i_k}(T-d(h^i_0,h^i_k)) : L_{i+1} \ge k > 0,\ T-d(h^i_0,h^i_k) \ge d(h^i_k,h^i_{L_i}) \right\}. \tag{16}$$

The decision $z^i_{h^i_0}(T)$ that achieves the maximal expected reward is

$$z^i_{h^i_0}(T) = \arg\max_k \left\{ G^s_{h^i_k}(T-d(h^i_0,h^i_k)) : L_{i+1} \ge k > 0,\ T-d(h^i_0,h^i_k) \ge d(h^i_k,h^i_{L_i}) \right\}. \tag{17}$$

Now we show that the policy specified by (15) and (17) solves problem (IAP^i) for any given π_j in the form of (5). Let π_i be the policy specified by (15) and (17). We consider any policy $\hat{\pi}_i$ in the form of (5). According to the definition of $G^s_{h^i_0}(T)$, we have

$$\begin{aligned} &\sum_{\alpha\in\mathcal{O}_i} e_\alpha g_\alpha \tau^i_{h^i_k}(\pi_i) + \sum_{\alpha\in\mathcal{S}} e_\alpha g_\alpha \tau^i_\alpha(\pi_i)[(1-\tau^j_\alpha(\pi_j)) + \gamma_\alpha \tau^j_\alpha(\pi_j)] \\ \ge &\sum_{\alpha\in\mathcal{O}_i} e_\alpha g_\alpha \tau^i_{h^i_k}(\hat{\pi}_i) + \sum_{\alpha\in\mathcal{S}} e_\alpha g_\alpha \tau^i_\alpha(\hat{\pi}_i)[(1-\tau^j_\alpha(\pi_j)) + \gamma_\alpha \tau^j_\alpha(\pi_j)]. \end{aligned}$$

Given π_j, we consider $\widehat{R}(\pi_i, \pi_j) - \widehat{R}(\hat{\pi}_i, \pi_j)$. We have

$$\widehat{R}(\pi_i, \pi_j) - \widehat{R}(\hat{\pi}_i, \pi_j) = \sum_{\alpha \in \mathcal{S}} e_\alpha g_\alpha [\tau_\alpha^1(\pi_1)(1 - \tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi_2)(1 - \tau_\alpha^1(\pi_1)) + (1 + \gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)]$$
$$+ R^{\pi_i} + R^{\pi_j} - R^{\hat{\pi}_i} - R^{\pi_j} - \sum_{\alpha \in \mathcal{S}} e_\alpha g_\alpha [\tau_\alpha^1(\hat{\pi}_1)(1 - \tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi_2)(1 - \tau_\alpha^1(\hat{\pi}_1)) + (1 + \gamma_\alpha)\tau_\alpha^1(\hat{\pi}_1)\tau_\alpha^2(\pi_2)]$$
$$= \sum_{\alpha \in \mathcal{O}_i} e_\alpha g_\alpha \tau_{h_k^i}^i(\pi_i) - \sum_{\alpha \in \mathcal{O}_i} e_\alpha g_\alpha \tau_{h_k^i}^i(\hat{\pi}_i) + \sum_{\alpha \in \mathcal{S}} e_\alpha g_\alpha \tau_\alpha^i(\pi_i)[(1 - \tau_\alpha^j(\pi_j)) + \gamma_\alpha \tau_\alpha^j(\pi_j)]$$
$$- \sum_{\alpha \in \mathcal{S}} e_\alpha g_\alpha \tau_\alpha^i(\hat{\pi}_i)[(1 - \tau_\alpha^j(\pi_j)) + \gamma_\alpha \tau_\alpha^j(\pi_j)] \geq 0.$$

Since $\widehat{R}(\pi_i, \pi_j)$ is greater than $\widehat{R}(\hat{\pi}_i, \pi_j)$ for any $\hat{\pi}_i \in \hat{\Pi}_i$, the policy established by (15) and (17) solves problem (IAP^i), which is a Markovian policy by its form. □

Remark 1 *When no regions are shared, the solution of problem (IAP^i) maximizes R^{π_i}. Since we also have $R(\pi) = R^{\pi_1} + R^{\pi_2}$, the policies obtained by solving problems (IAP^1) and (IAP^2) compose an optimal time-allocation policy in this case.*

For readability and conciseness of the paper, we delay the proofs of the remaining theorems and propositions to §C of the appendix.

4.3 Obtaining a Local Optimum

We first define a local optimum of problem (IAP).

Definition 3 *A Markovian policy $\pi^{L,*} = (\pi_1^{L,*}, \pi_2^{L,*})$ is a local optimum of problem (IAP) if*

$$\pi_1^{L,*} \in \arg\max\{\widehat{R}(\pi_1, \pi_2^{L,*}) : \pi_1 \in \widehat{\Pi}_1\},$$
$$\pi_2^{L,*} \in \arg\max\{\widehat{R}(\pi_1^{L,*}, \pi_2) : \pi_2 \in \widehat{\Pi}_2\}.$$

Algorithm 1 implements the idea of JESP algorithm (Nair et al., 2003), which iteratively improves one vehicle's policy while fixing the policy applied by the other vehicle. Since we solve the approximate problem (IAP) in the algorithm instead of the true problem, our algorithm runs much more efficiently.

Theorem 2 *Algorithm 1 converges to a local optimum of problem (IAP) in a finite number of steps.*

Each restart of Algorithm 1 may converge to a different local optimum since the initial vehicle i is randomly selected and the values of $\hat{\tau}_\alpha^i, \alpha \in \mathcal{S}$ are also randomly generated. To interpret this, we can regard Algorithm 1 as using a local-search method to find a local optimum of a non-convex optimization problem. Starting from a different initial point, the local-search method may converge to a different local optimum.

Now we explain why constrained re-optimization is performed. Any feasible time-allocation policy for the initial route family is still feasible to the route family obtained by constrained re-optimization

Algorithm 1

1: Initialization: Select a random $i \in \{1,2\}$ and generate a random $\hat{\tau}^i_\alpha \in [0,1]$, for $\forall \alpha \in \mathcal{S}$. Set $R \leftarrow 1, R^* \leftarrow 0$ and $j \in \{1,2\} \setminus \{i\}$.
2: Solve (IAP^i)'s using the $\hat{\tau}^i_\alpha, \alpha \in \mathcal{S}$ and obtain the policy π_i. For $\alpha \in \mathcal{S}$: $\hat{\tau}^j_\alpha \leftarrow \tau^i_\alpha(\pi_i)$.
3: **while** $R \neq R^*$ **do**
4: $R \leftarrow R^*$.
5: Solve (IAP^j) and use the obtained policy to update π_j. (Do not update if the current π_j solves (IAP^j).)
6: For $\alpha \in S$: $\hat{\tau}^i_\alpha \leftarrow \tau^j_\alpha(\pi_j)$.
7: Solve (IAP^i) and use the obtained policy to update π_i. (Do not update if the current π_i solves (IAP^i).)
8: For $\alpha \in S$: $\hat{\tau}^j_\alpha \leftarrow \tau^i_\alpha(\pi_i)$.
9: Obtain $\widehat{R}(\pi_1, \pi_2)$ using (10) and update $R^* \leftarrow \widehat{R}(\pi_1, \pi_2)$.
10: **end while**

where the same expected reward is collected by the fleet from the two families. To this end, we consider an arbitrary feasible policy for the initial route family. $z^i_\alpha(\cdot)$ decisions in the policy will not lead the vehicle to a new region and all $z^i_\alpha(\cdot)$ decisions are still feasible since the orders of the initial regions do not change. Also because the $x^i_\alpha(\cdot)$ and $y^i_\alpha(\cdot)$ decisions for each region are the same, each vehicle will behave in the re-optimized route exactly the same as what it does in the initial route. Thus, if we use the optimal policy of the initial route family to generate the $\hat{\tau}^i_\alpha,\ \alpha \in \mathcal{S}$ for Algorithm 1, there is a high chance that we will obtain a better time-allocation policy from the new route family.

Even though Algorithm 1 may still go over all feasible policies before convergence in the worst case, in practice, it only goes over a few policies to converge, which we will illustrate in our numerical study. More importantly, each iteration (lines 4-9) runs in polynomial time of L and T, this is the main reason that the algorithm can solve realistic size problems. To this end, the problems in line 5 and line 7 are solved using backward induction on the recursions defined by (12), (13), (14) and (16) with initial condition $G_{h^i_{L_i+1}}(t) = 0, t = 0, 1, 2, ..., T$. It is easy to verify that the backward induction takes $O(TL^2 + T^2L)$ time. We use forward induction in line 6 and line 8 to update the variables $\hat{\tau}^i_\alpha, \alpha \in \mathcal{S}, i \in \mathcal{U}$, for which we provide an algorithm in §B. The forward induction is performed over the route of each vehicle (from h^i_0 to $h^i_{L_i+1}$ for vehicle $i \in \mathcal{U}$) but the size of the induction tree in each iteration is linear in T. The forward induction takes $O(LT^2)$ time. In line 9, we calculated $\widehat{R}^\pi_\alpha$ using stored values and it takes only $O(L)$ time.

4.4 Calculation of Exact Value of a Markovian Policy and a Sufficient Condition for the Approximation to be Exact

Algorithm 1 provides an efficient method to find a policy; however, it does not offer the true value of the obtained policy since $\widehat{R}(\pi)$ is only an approximation of $R(\pi)$. To calculate $R(\pi)$, we use conditional independency. The method is derived based on the *shared order index* of each shared region for each vehicle, which is defined as follows:

Definition 4 *The* shared order index *of a shared region α for vehicle i is defined as $I^i_\alpha = k$, if α has the k^{th} highest order index among the vehicle's shared regions.*

The maximal and minimal shared order indices of region α are defined as $\overline{I}_\alpha = \max\{I^1_\alpha, I^2_\alpha\}$ and $\underline{I}_\alpha = \min\{I^1_\alpha, I^2_\alpha\}$, respectively. Based on the shared order indices of the shared regions, we define the *dependent set* of each shared region.

Definition 5 $\Phi_\alpha = \{\beta : \overline{I}_\beta \leq \underline{I}_\alpha, \beta \in \mathcal{S}\}$ *is the* dependent set *of a shared region* α.

According to the definition of the dependent set, for any shared region α, if l_1 and l_2 are the corresponding order indices of region α in the two vehicles' routes, it is easy to verify that $\Phi_\alpha = \{h^1_1, ..., h^1_{l_1-1}\} \bigcap \{h^2_1, ..., h^2_{l_2-1}\}$.

For any region set $\mathcal{V} \subseteq \mathcal{A}$, we use $\theta_\mathcal{V} = \{\theta_\alpha\}_{\alpha\in\mathcal{V}}$ to represent a realization of the existence of information in each region that belongs to $\mathcal{V}$.

Theorem 3 *Whether a vehicle succeeds in a shared region* α *is conditionally independent of the other vehicle under a decentralized Markovian policy given* θ_α *and* θ_{Φ_α}.

Using Theorem 3, we can calculate $R_\alpha(\pi)$ using conditional independence given θ_α and θ_{Φ_α}.

$$\begin{aligned} R_\alpha(\pi) = \sum_{\theta_{\Phi_\alpha}\in\{0,1\}^{|\Phi_\alpha|}} P(\theta_{\Phi_\alpha})[\tau^1_\alpha(\pi_1|\theta_{\Phi_\alpha})(1-\tau^2_\alpha(\pi_2|\theta_{\Phi_\alpha})) + \tau^2_\alpha(\pi_2|\theta_{\Phi_\alpha})(1-\tau^1_\alpha(\pi_1|\theta_{\Phi_\alpha})) \\ + (\gamma_\alpha+1)\tau^1_\alpha(\pi_1|\theta_{\Phi_\alpha})\tau^2_\alpha(\pi_2|\theta_{\Phi_\alpha})], \end{aligned} \tag{19}$$

where $\tau^i_\alpha(\pi_i|\theta_{\Phi_\alpha})$ is the conditional probability that vehicle i succeeds in region α given θ_α and θ_{Φ_α}. $\tau^i_\alpha(\pi_i|\theta_{\Phi_\alpha})$, $i \in \mathcal{U}, \alpha \in \mathcal{S}$, can be calculated using the algorithm provided in §B and $P(\theta_{\Phi_\alpha})$ can be calculated using Lemma 2 provided in §C. The following proposition establishes the monotonicity of the dependent set for each shared region of two vehicles with respect to the region's minimal shared order index.

Proposition 2 *The dependent set of a shared region* α *is monotone w.r.t.* $\underline{I}_\alpha$, *i.e., if* $\underline{I}_{\alpha_1} \geq \underline{I}_{\alpha_2}$, $\Phi_{\alpha_1} \supseteq \Phi_{\alpha_2}$.

Using the result of Proposition 2, the complexity of calculating $R(\pi)$ is determined by the size of the largest dependent set. In other words, we only need to enumerate all possible realizations of θ_{Φ_β} where $\beta \in \arg\max_{\alpha\in\mathcal{S}}\{\underline{I}_\alpha\}$ so that we can calculate the expected reward collected in each shared region and the number of possible realizations for θ_{Φ_β} is $2^{|\Phi_\beta|}$. Note that we have $|\Phi_\beta| \leq |\mathcal{S}| - 1$ and $|\Phi_\beta| = |\mathcal{S}| - 1$ if and only if β is the last shared region for both vehicles to search.

Proposition 3 *Whether a vehicle succeeds in* $\alpha \in \mathcal{S}$ *is conditionally independent of the other vehicle given* θ_α *under a decentralized policy* π *in the form of (5) if the shared regions are searched in an exactly opposite order by the two vehicles.*

Proposition 3 proposes a sufficient condition, under which (10) provides the exact expected reward collected by any Markovian policy from a shared region. Under a weaker condition, we are able to prove an important result.

Theorem 4 *If whether a vehicle succeeds in $\alpha \in \mathcal{S}$ is conditionally independent of the other vehicle given θ_α under an optimal decentralized policy π^* in the form of (5), there exists a Markovian policy that solves the time-allocating problem optimally.*

The result of Theorem 4 provides a sufficient condition under which the approximate problem (IAP) is equivalent to the original decentralized time-allocation problem, which, in its general form, is a DEC-POMDP. The theorem also implies that under the condition of Proposition 3, the optimal solution of problem (IAP) provides an optimal solution to the time-allocating problem for a given route family.

We end this section with an illustration of the largest dependent set of a route family in Figure 1. In scenario (L), since both vehicles visit the shared regions in the same order, the size of the largest dependent set reaches its maximum, which is $|\mathcal{S}| - 1$. In scenario (R), since both vehicles visit the shared regions in an exact opposite order, the largest dependent set is still empty.

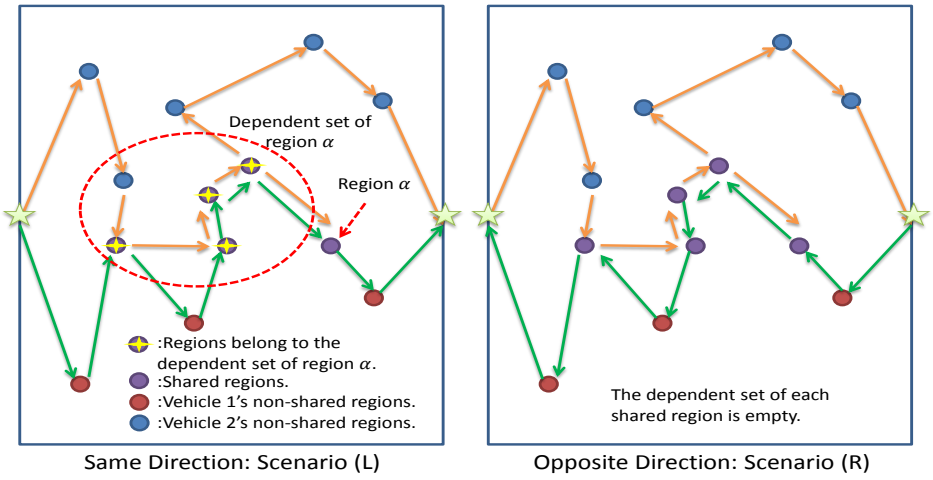

Figure 1: The dependent set of a shared region

5. An upper bound on the maximum reward collected from a group of routes

In this section, we use the value provided by a local optimum of problem (IAP), i.e., $\widehat{R}(\pi^{L,*})$, to develop an upper bound on the maximum expected reward that the fleet can collect from a given route family by following a decentralized time-allocation policy, i.e., $R(\pi^*)$. To establish the upper bound, we derive a lower bound on the ratio $\frac{\widehat{R}(\pi^{L,*})}{R(\pi^*)}$.

The lower bound on the ratio is established with respect to the cooperation factors of the shared regions, each of which can be greater, smaller, or equal to 1. When the cooperation factor is greater than 1, the second vehicle that collects the information gains more reward for the fleet. This implies that the fused information from different sensors provides much more value than the information

from a single source. For instance, multi-sensor image fusion can provide a fused image that is more informative than any of the input images (Haghighat et al., 2011). While it is also common to see that the second vehicle collects less information than the first vehicle since the total information existing in a region can decrease after each successful collection, under which situation we expect the cooperation factor to be smaller than 1. The lower bound of the ratio is found in a closed form of $\overline{\gamma}$ and $\underline{\gamma}$, which satisfy $\overline{\gamma} = \max\{1, \max_{\alpha\in\mathcal{S}} \gamma_\alpha\}$ and $\underline{\gamma} = \min\{1, \min_{\alpha\in\mathcal{S}} \gamma_\alpha\}$.

Theorem 5

$$\frac{\widehat{R}(\pi^{L,*})}{R(\pi^*)} \geq \frac{1}{(2-\underline{\gamma})\overline{\gamma}}. \tag{20}$$

We consider two special scenarios of Theorem 5.

Remark 2 *When $\overline{\gamma} = 1$, (20) degenerates to*

$$\frac{\widehat{R}(\pi^{L,*})}{R(\pi^*)} \geq \frac{1}{(2-\underline{\gamma})}. \tag{21}$$

When $\underline{\gamma} = 1$, (20) degenerates to

$$\frac{\widehat{R}(\pi^{L,*})}{R(\pi^*)} \geq \frac{1}{\overline{\gamma}}. \tag{22}$$

Remark 2 highlights two special cases that are common in practice. The bound (21) refers to the scenario where the value of information in each shared region decreases after the first collection. For this scenario, we have $\frac{1}{(2-\underline{\gamma})} \geq \frac{1}{2}$. This implies that $\widehat{R}(\pi^{L,*})$ achieves at least 50% of the optimal value in the worst case. The bound (22) refers to the scenario where a much larger information reward can be obtained by fuzing the information collected by two vehicles. We can observe that $\widehat{R}(\pi^{L,*})$ can be arbitrarily bad compared to the optimal value if $\overline{\gamma}$ approaches infinity. In §D, we provide two corollaries to show that how the bounds (21) and (22) can be approached from a theoretical point of view. We also notice that when $\underline{\gamma}$ and $\overline{\gamma}$ approach to 1, the general bound (20) converges to 1, which means that $\widehat{R}(\pi^{L,*})$ converges to the same value that the optimal policy provides. Under this situation, bound (20) can be used to estimate the convergence speed.

To examine the worst-case performance, we can first compute an upper bound of $R(\pi^*)$ using the the value $\widehat{R}(\pi^{L,*})$ and the bound (20). Then the worst-case performance ratio $\frac{R(\pi^{L,*})}{R(\pi^*)}$ can be derived. Since we notice that $R(\pi^{L,*})$ and $\widehat{R}(\pi^{L,*})$ are usually close in practice, we can use the bound (20) as an estimate of the true worst-case performance ratio.

6. Numerical Study

The purpose of the numerical study is threefold. §6.2 establishes the efficiency of the algorithm in §4.3 that finds a Markovian time-allocation policy for a given route family. §6.3 explores the benefit of using a region-sharing strategy. §6.4 presents additional insights related to the mission duration, cooperation factor value and search sequence. Tests are all performed on a personal computer with Intel i7 CPU and 8 GB RAM under Window 7 system.

6.1 Parameter Setup and Scenario Generation

A rectangular map of size $W \times L$ has two depots located at the middle points of the left and right edges. Two vehicles are assigned to search N regions, which are uniformly generated at integer points of the map, within T units of mission time. The travel time between two regions is calculated using Euclidean distance rounded down to the nearest integer. Two basic scenarios are tested which are the scenarios (L) and (R) presented in Figure 1 (at the end of §4). In scenario (L), both vehicles travel from the left depot to the right depot. In scenario (R), one vehicle travels from the left depot to the right depot while the other vehicle travels from the right depot to the left depot.

We generate values for parameters associated with a region α as follows:

- e_α, the *a priori* probability that information exists, is uniformly generated from a sub-interval $[\underline{e}, \overline{e}] \subseteq [0, 1]$.
- g_α, the reward of collecting the information, is assumed to be 1.
- s_α, the time to collect the information, is an integer random variable whose value is equally likely to be either 0,1,2,3,4 or 5.
- γ_α, the cooperation factor, is uniformly generated in the range $[\underline{\gamma}, \overline{\gamma}] = [0.0, 2.0]$.
- $P_\alpha(t)$, the detection function, is assumed to follow the random search formula in Koopman (1980) and takes the form
$$P_\alpha(t) = 1 - \rho_\alpha^t, \tag{23}$$
where ρ_α is uniformly generated from $[\underline{\rho}, \overline{\rho}] \subseteq (0, 1)$. According to the development in Koopman (1980), ρ_α is determined by the size of the region, the speed of the vehicle, and the effective range of the sensor in practice and always falls within $(0, 1)$.

For each map, we generate two initial route families using the formulations provided in §A. The first family minimizes the total travel time of the fleet and the second family minimizes the maximal travel time of a vehicle. We obtain the optimal policies for these two families and use NS to represent the better policy. For each initial route, we use the minimum insertion rule to select all possible sets of shared regions for the two vehicles, and for each set of shared regions we obtain two route families respectively using the constrained and unconstrained re-optimization. For each re-optimized route family, a time-allocation policy is generated by Algorithm 1. We select one policy among the policies generated for each initial route family that has the highest approximate value ($\widehat{R}(\pi)$). Then, the exact values of the selected policies are calculated using (7) and (19). We use S to represent the better policy of the two selected policies.

The notation $\widehat{V}^{(i)}(j)$ and $V^{(i)}(i)$ is used to denote, respectively, the approximate and exact values of the policy j for scenario i. For example, $V^{(L)}(S)$ represents the exact value of the policy S obtained under scenario (L).

6.2 The efficiency of finding a decentralized time-allocation policy

We investigate the efficiency of computing a Markovian policy for a given group family using Algorithm 1. We use the following parameter setup: $W \times L = 30 \times 60$, $[\underline{e}, \overline{e}] = [0.3, 0.7]$, $[\underline{\rho}, \overline{\rho}] = [0.15, 0.35]$,

and $[\underline{\gamma}, \overline{\gamma}] = [0.0, 2.0]$. For each map, we randomly select an initial route family from the two initial families. The threshold of the minimal insertion rule is set to $\delta = 25$. After adding the selected shared regions to each vehicle's initial route, we use one of the two re-optimization methods to modify vehicle routes. The objective is to test how much computation time it takes to obtain a local optimum of problem (IAP) and the time needed to calculate the policy's expected value ($R(\pi)$).

We first present the results for scenario (R) under different units of mission time (T) and different numbers of regions (N). For each (T, N) pair, the results are collected from 50 random maps.

Table 1: Computation Complexity: Scenario (R)

(T, N)	Iterations			Time(C)			Time(E)			$\lvert\mathcal{S}\rvert$	$\lvert\Phi_\beta\rvert$
	min	*max*	*avg*	*min*	*max*	*avg*	*min*	*max*	*avg*	*avg*	*avg*
$(100, 20)$	3	10	3.90	0.046	0.396	0.078	0.001	0.014	0.002	17.02	0.60
$(200, 20)$	3	8	4.46	0.276	1.354	0.488	0.004	0.068	0.009	15.24	1.24
$(300, 20)$	3	15	4.76	0.728	3.749	1.243	0.013	0.131	0.028	15.54	0.72
$(100, 30)$	3	11	3.96	0.078	0.390	0.124	0.001	0.505	0.026	24.00	2.08
$(200, 30)$	3	7	4.28	0.403	1.130	0.673	0.004	0.144	0.013	24.30	2.06
$(300, 30)$	3	13	4.70	1.061	4.907	1.791	0.012	0.438	0.040	24.78	1.7
$(100, 40)$	3	8	3.80	0.103	0.754	0.162	0.001	0.076	0.005	32.04	3.22
$(200, 40)$	3	13	4.76	0.528	2.808	0.981	0.005	7.387	0.187	31.78	3.46
$(300, 40)$	3	9	4.82	1.553	4.732	2.578	0.011	18.757	0.464	31.80	2.68

In Table 1, "Iterations" represents the number of iterations for Algorithm 1 to converge, "Time(C)" captures the corresponding CPU seconds consumed, and "Time(E)" records the CPU seconds to calculate $R(\pi)$. In addition, for each quantity, *min*, *max* and *avg* provide the corresponding minimum, maximum and average of the corresponding quantity among the 50 random maps. $|\mathcal{S}|$ represents the number of shared regions and $|\Phi_\beta|$ refers to the size of the largest dependent set, for which we provide their average values in the table.

For the largest problem, i.e., $(300, 40)$, the maximal time of computing a local optimum of problem (IAP) is still small (less than 5 seconds), and the average time is about half of the maximal time. This shows the efficiency of using the approximate formulation to find a decentralized time-allocation policy. However, "Time(E)" has larger variation. Even though, on average, it takes negligible amount of time to calculate $R(\pi)$ under this scenario, it may take several seconds to compute it in the worst-case situation. Under these situations, we observe large dependent sets.

Table 2 provides the test results for scenario (L). We observe large computation time for calculating $R(\pi)$ for the cases $(100, 20)$, $(200, 20)$ and $(300, 20)$, and we do not list the computation time for the other cases since we experienced constant "out-of-memory" problems during the experiments. This is because the size of the dependent set can be much larger when the vehicles travel in the same direction than that when the vehicles travel in an opposite direction, which can be observed by comparing the $|\Phi_\beta|$ columns in the two tables. We use a breadth-first algorithm to calculate $R(\pi)$, which saves computation time but may explode the memory when the size of the dependent size is large. Under such a scenario, we can use a memory saving algorithm, e.g., a depth-first enumeration algorithm or

Table 2: Computation Complexity: Scenario (L)

(T, N)	Iterations			Time(C)			Time(E)			$\lvert\mathcal{S}\rvert$	$\lvert\Phi_\beta\rvert$
	min	max	avg	min	max	avg	min	max	avg	avg	avg
$(100, 20)$	3	6	3.61	0.046	0.520	0.085	0.002	9.480	1.195	16.14	14.60
$(200, 20)$	3	8	4.25	0.277	1.014	0.451	0.054	89.133	10.819	16.48	15.32
$(300, 20)$	3	11	4.26	0.584	2.673	1.050	0.164	126.078	18.457	16.16	14.62
$(100, 30)$	3	11	4.10	0.064	0.446	0.122	-	-	-	23.66	22.10
$(200, 30)$	3	12	4.86	0.380	1.944	0.779	-	-	-	23.96	22.12
$(300, 30)$	3	14	4.96	0.934	5.196	1.993	-	-	-	24.20	22.28
$(100, 40)$	3	8	3.74	0.064	0.500	0.114	-	-	-	32.54	30.42
$(200, 40)$	3	9	4.39	0.447	1.572	0.751	-	-	-	32.00	29.36
$(300, 40)$	3	17	5.37	1.080	7.238	2.213	-	-	-	31.54	29.16

Monte Carlo simulation to estimate $R(\pi)$. We also observe from both tables that the time to compute a policy and the number of iterations to converge do not change much even though the average size of the largest dependent set ($|\Phi_\beta|$) differs significantly in the two tables. This observation implies that *the convergence of Algorithm 1 is not influenced by the size of the dependent set*. It is shown in both tables that the average computation time and iterations of obtaining a policy increase with N and T. This implies that *it takes more iterations to converge for a larger problem*. Since the time to perform an iteration also increases with the problem size, the time to compute a policy also increases with the problem size.

Finally, we do not provide the computation time of solving the routing problems since we use standard routing models and solve them using a standard solver. According to our experiments, the largest routing problems listed above, i.e., the problems generated under $(300, 40)$, can be solved in a few seconds by a standard commercial solver (CPLEX) using the formulations provided in A. When it becomes expensive to solve the routing problem exactly, heuristic methods can be applied to create and re-optimize the routes.

6.3 Benefits of sharing

This section illustrates the influence of the cooperation factor and mission time values on sharing strategies. We use the following parameter setup: $W \times L = 14 \times 20$, $N = 16$, $[\underline{e}, \overline{e}] = [0.3, 0.7]$ and $[\underline{\rho}, \overline{\rho}] = [0.15, 0.35]$. In each map, we use a constant cooperation factor for all regions. The tested values are $\gamma = 0.2, 0.5, 0.8, 1.1, 1.4, 1.7, 2.0$ and 50 random maps are generated for each value. The tests are performed for scenario (L) with two different mission time, $T = 80$ and $T = 100$, and the results are listed in Figures 2 and 3, respectively.

In each figure, the three lines present the maximum, average, and minimum of the percentage improvements through region-sharing, i.e., $\frac{V^{(L)}(S)}{V^{(L)}(NS)} - 1$, among the 50 random maps generated for each cooperation factor value. We have the following observations from comparing the two figures. First, the three quantities increase with the cooperation factor in both figures, which implies that

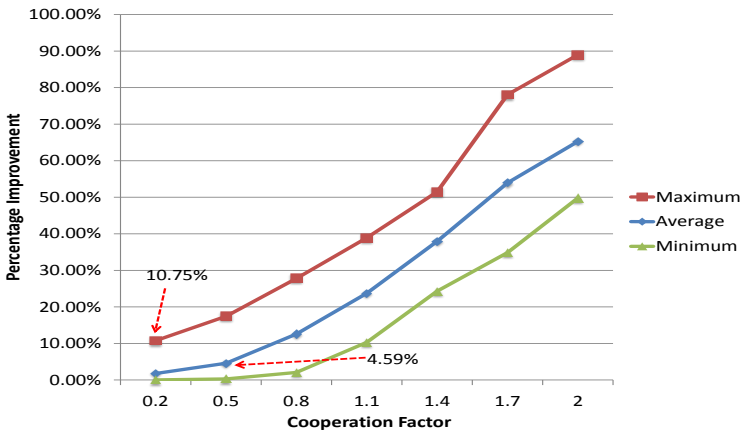

Figure 2: Percentage improvements under scenario (L) for different cooperation factors and $T = 80$

a larger cooperation factor bring a larger benefit through region-sharing. Second, in Figure 2, the average improvements are much smaller than the ones in Figure 3. This implies that a larger benefit is gained through region-sharing when the fleet is given more mission time. In addition, the average and minimal performance improvements also increase much faster with respect to the cooperation factor value in Figure 3 than that in Figure 2, especially when the cooperation factor is low. This indicates that *when the fleet has more mission time, they also benefit more from the increase of the cooperation efficiency (i.e., cooperation factor value).* Third, for $\rho = 0.2, 0.5$ the average improvements in Figure 2 are less than 5%. This means that having two vehicles search one region is generally not economical when the cooperation factor is low and the mission time is insufficient. Nevertheless, we still observe in Figure 2 that the maximal improvement at $\gamma = 0.2$ is more than 10%, which implies the potential benefit of region-sharing even when the fleet have low cooperation efficiency and insufficient mission time. Furthermore, when more mission time is given to each vehicle, i.e., the scenarios presented in Figure 3, the minimal performance improvement through region-sharing has already been 7.98% at $\gamma = 0.5$ and the average improvement is more than 10%. This shows that *we can expect a stable performance improvement through region-sharing at a relatively low cooperation factor if the fleet has a sufficiently large mission time.*

We present how the improvements are distributed among the random cases generated for $\gamma = 0.2, 2$ and $T = 80, 110$ in Figure 4. Except for the setting where $\gamma = 0.2, T = 80$, we observe that the improvement follows a unimodal distribution and the peak is close to the average. This shows that the average improvement curves presented in Figures 2 and 3 generally reflect the benefit of using region-sharing strategy for the corresponding cooperation factor value and mission time. However, for the setting $\gamma = 0.2, T = 80$, the peak of the distribution is to the right of the average and the distribution has a long tail. This shows that *when the fleet has insufficient mission time and the cooperation factor is low, the improvement has a decreasing chance to be larger but the chance of having a large improvement cannot be ignored.*

Finally, we investigate the benefit of sharing under an extreme scenario where we have $\gamma = 0$ for all

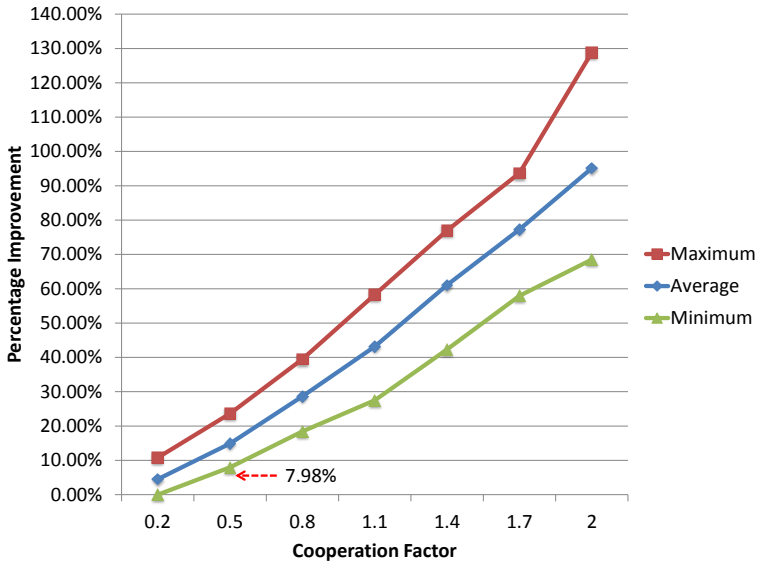

Figure 3: Percentage improvements under scenario (L) for different cooperation factors and $T = 110$

regions. This means that if one vehicle collects the information from a region, the other vehicle gains no extra reward for the fleet even if it also collects the information. This time, we consider scenario (R) and three settings are tested with the corresponding parameters listed in Table 3. For each setting, we generate 25 maps. We present two ratios in the table. Ratio 1 presents the performance improvement through region-sharing according to the policy's approximate value ($\widehat{R}(\pi)$) and Ratio 2 presents the true improvement. We present the maps that have the largest and the second largest Ratio 2 among the 25 maps for the corresponding setting. We observe a performance improvement of 15.37% in map 2. In addition, in maps 1 and 3, we observe the improvements through region-sharing directly from the approximate values of the corresponding policies. In maps 2, 4, 5, and 6, *we do not observe large improvements from the policies' approximate values (Ratio 1) but the true improvements (Ratio 2) are much larger*. This is because a large dependent set can exist when the two vehicles travel in the same direction which can make a significant difference between $\widehat{V}^{(R)}(S)$ and $V^{(R)}(S)$.

6.4 Additional Insights

This section investigates the joint behavior of the fleet through region-sharing with respective to the mission duration, the value of the cooperation factor and the search sequence.

6.4.1 The study of an idealized scenario.

We first investigate the problem using an idealized scenario presented in Figure 5. We have 10 regions located on a straight line and every two adjacent regions have a distance of 1 unit. For each region α, we set $e_\alpha = 0.5$, $\rho_\alpha = 0.25$, and $s_\alpha = 0$. Such a scenario excludes the influence of the regions' location

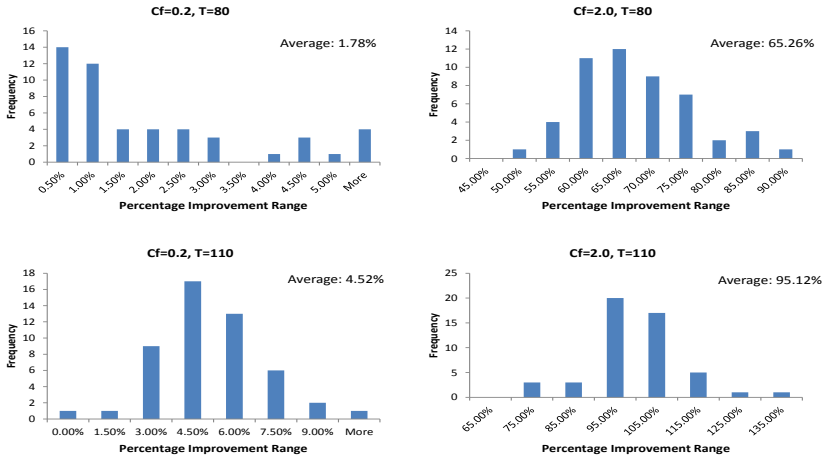

Figure 4: Distribution of the improvements under scenario (L) for $\gamma = 0.2, 2$ and $T = 80, 110$

Table 3: Results under scenario (R) where $\max_{\alpha \in S} \gamma_\alpha = 0$

Map	$W \times L$	N	T	$[\underline{e}, \overline{e}]$	$[\underline{\rho}, \overline{\rho}]$	$V^{(R)}(NS)$	$\widehat{V}^{(R)}(S)$	$V^{(R)}(S)$	Ratio 1	Ratio 2
1	10×15	16	65	$[0.3, 0.7]$	$[0.15, 0.35]$	5.81	6.33	6.34	8.95%	9.12%
2	10×15	16	65	$[0.3, 0.7]$	$[0.15, 0.35]$	5.92	6.06	6.83	2.36%	15.37%
3	14×20	16	70	$[0.3, 0.7]$	$[0.15, 0.35]$	5.58	6.14	6.14	10.04%	10.04%
4	14×20	16	70	$[0.3, 0.7]$	$[0.15, 0.35]$	5.95	5.97	6.31	0.34%	6.05%
5	16×20	16	75	$[0.3, 0.7]$	$[0.15, 0.35]$	6.03	6.10	6.39	1.16%	5.97%
6	16×20	16	75	$[0.3, 0.7]$	$[0.15, 0.35]$	5.22	5.23	5.70	0.19%	9.20%

Ratio 1: $\frac{\widehat{V}^{(R)}(S)}{V^{(R)}(NS)} - 1$, Ratio 2: $\frac{V^{(R)}(S)}{V^{(R)}(NS)} - 1$.

since it is not beneficial to skip a region and the influence of the information collection time in each region since a vehicle will always collect the information if it detects it.

In the given scenario, both vehicles have to spend 11 units of travel time whether or not they skip a region. When $T = 16$, each of them has 5 units of time to allocate. From the results for $\gamma = 0.5$, we observe that all regions are searched, which means that each of the vehicles searches 5 regions by spending 1 unit of time in each of them and none of the regions is shared. For $\gamma = 1.5$, however, only 5 regions are actually searched and all of them are shared by the two vehicles. For $\gamma = 0.5$, when we give the vehicles more mission time, each vehicle tends to allocate the additional time to the regions that are initially searched by the other vehicle instead of their own regions. This can be observed from the increase of the "Shared(No.)" in the upper bracket of the table. This shows that *the vehicles can prefer region-sharing rather than securing their own information reward even if the cooperation factor is below 1 (i.e., with low cooperation efficiency).* For $\gamma = 1.5$, when the vehicles have more mission

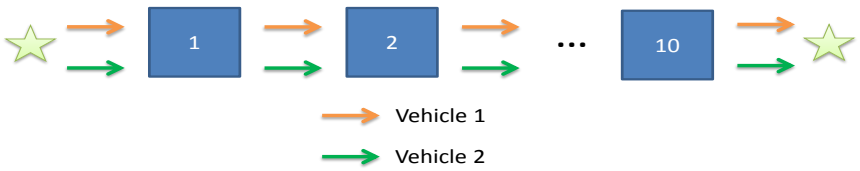

Figure 5: A map with regions on a straight line

Table 4: Percentages of the regions that are searched and shared

$\gamma = 0.5$	Searched(No.)	10	10	10	10	10	10	10
	Shared(No.)	0	2	4	6	8	10	10
	T	16	17	18	19	20	21	22
$\gamma = 1.5$	Searched(No.)	5	6	6	7	8	9	9
	Shared(No.)	5	6	6	7	8	9	9
	T	16	17	18	19	20	21	22

Searched(No.): The number of regions that are actually searched.
Shared(No.): The number of regions that are shared.

time, *the vehicles can prefer allocating more time to secure the joint reward in a shared region that have already been searched than searching new regions (under high cooperation efficiency)*. This can be observed from the "Searched(No.)" in the lower bracket of the table since only 9 regions are actually searched when each vehicle is given 22 units of mission time.

To investigate the influence of the cooperation factor, we now let the 10 regions have different cooperation factors, from 0.2 to 2.0 with an increment of 0.2 per region. Two scenarios are tested. (A): The regions are located so that each region has a lower cooperation factor than the regions on its left. (B): The regions are located so that each region has a lower cooperation factor than the regions on its right. We present the test results for different mission time T in Tables 5 and 6, respectively.

Comparing the rewards (i.e., $V^{(A)}(S)$ and $V^{(B)}(S)$) in the two tables, we find out that the expected reward gained in scenario (A) is slightly higher than the one gained in scenario (B). In addition, the numbers below the cooperation factors are the conditional probabilities that vehicles 1 (left) and 2 (right) succeed in the corresponding region given that the information exists. Since the regions are the same except for their cooperation factors, when regions are shared, intuitively we should expect the vehicles to have larger probabilities to succeed in the regions that have larger cooperation factors. We observe this intuitive behavior in scenario (A) while in scenario (B) we find out that *both vehicles have the largest probabilities to succeed in the second last region ($\gamma = 1.8$)*. This is explained as follows: If both vehicles reserve large amounts of time for the last region they may end up wasting the remaining

Table 5: Test results for scenario (A) with different mission time T

T	$\gamma = 2.0$		$\gamma = 1.8$		$\gamma = 0.4$		$\gamma = 0.2$		$\widehat{V}^{(A)}(S)$	$V^{(A)}(S)$
15	0.438	0.438	0.273	0.273	0.000	0.000	0.000	0.000	1.072	1.072
20	0.578	0.578	0.438	0.438	0.000	0.079	0.079	0.000	2.420	2.420
25	0.578	0.578	0.578	0.578	0.000	0.250	0.250	0.000	3.651	3.651
30	0.684	0.684	0.684	0.684	0.328	0.027	0.051	0.255	4.768	4.767
35	0.763	0.763	0.706	0.706	0.298	0.299	0.186	0.267	5.757	5.755
40	0.822	0.822	0.776	0.776	0.358	0.375	0.299	0.309	6.613	6.610

Table 6: Test results for scenario (B) with different mission time T

T	$\gamma = 2.0$		$\gamma = 1.8$		$\gamma = 0.4$		$\gamma = 0.2$		$\widehat{V}^{(B)}(S)$	$V^{(B)}(S)$
15	0.273	0.273	0.438	0.438	0.000	0.000	0.000	0.000	1.050	1.060
20	0.455	0.455	0.458	0.458	0.000	0.250	0.000	0.000	2.360	2.360
25	0.497	0.484	0.486	0.481	0.000	0.250	0.250	0.000	3.571	3.573
30	0.521	0.521	0.633	0.633	0.000	0.437	0.437	0.000	4.644	4.647
35	0.605	0.602	0.711	0.711	0.437	0.000	0.000	0.437	5.601	5.603
40	0.696	0.693	0.745	0.742	0.578	0.250	0.000	0.578	6.432	6.435

time if they detect the information early in the last region. As a result, they reserve the largest amounts of time for the second last region. The sequence also influences the vehicles' behaviors in the region with low cooperation factors. In scenario (A), we observe that *both vehicles have large probabilities to succeed in the regions with low cooperation factors* ($\gamma = 0.2, 0.4$) when more mission time is given ($T = 35, 40$). This is because that the low cooperator regions are searched at the end and if any vehicle spends less time in its early stages of the search it will search these regions. However, in scenario (B) the low cooperator regions are searched in the beginning. As a result, only one vehicle will commit to each of them since the cooperation factor is too low, i.e., it is not economical for both vehicles to search them without knowing how much time they will need in the future.

To further investigate the influence of the sequence, we take scenario (A) and then change the position (counted from the left) of the region with the highest cooperation factor ($\gamma = 2.0$). The results for $T = 40$ are presented in Table 7.

Two observations are obtained from Table 7. When the region is located in a later position, both vehicles have smaller probabilities to succeed in the region. More importantly, the expected reward that the fleet can collect also decreases. Combined with the former observations, the observations suggest that *it is more beneficial to search regions with higher cooperation factors before searching regions with lower cooperation factors.*

Table 7: Results for different positions of the region with the highest cooperation factor

τ^1	0.822	0.822	0.799	0.799	0.799	0.789	0.785	0.785	0.770	0.699
τ^2	0.822	0.822	0.799	0.799	0.799	0.789	0.785	0.785	0.770	0.685
Reward	6.611	6.610	6.609	6.608	6.605	6.602	6.597	6.586	6.560	6.488
Position	1	2	3	4	5	6	7	8	9	10

τ^i: The conditional probability for vehicle $i \in \{1, 2\}$ to succeed in the region given that the information exists.

6.4.2 The study of random scenarios.

In Table 3, we observe some cases where $\widehat{V}^{(R)}(S)$ is very different from $V^{(R)}(S)$ (maps 2,4,5,6) while in Tables 5 and 6, $\widehat{V}^{(R)}(S)$ and $V^{(R)}(S)$ are very close. To further investigate this observation, we use three different amounts of mission time $T = 65, 80, 100$, and randomly generate 25 maps for each with the parameter setups listed in Table 8. The tests are performed for scenarios (L) and (R). As illustrated in Figure 1 in §4 and the results presented in Tables 1 and 2, when the two vehicles travel in the same direction, we expect large dependent sets; on the contrary, when they travel in opposite directions, we expect the dependent sets to be small or empty.

We select the maps where we observed a considerable difference (more than 1.5% or smaller than -1.5%) between the approximate and exact values of the best policy obtained after sharing for each scenario (Ratio 1 and Ratio 2), or between the exact values of the best policies obtained for both scenarios (Ratio 3). We found 8 out of 75 random maps, which are listed in Table 8. Comparing Ratio 1 and Ratio 2 in the table, we find that the exact and approximate values of the policy obtained for the scenario (L) is more likely to be different. In maps 7, 8, 10, 11, 12, 13, and 14, we observe a large difference between $\widehat{R}(\pi)$ and $R(\pi)$. On the contrary, the difference in scenario (R) can be negligible except for map 22. This shows that a larger dependent set more likely leads to a difference between $\widehat{R}(\pi)$ and $R(\pi)$. However, *the influence of the dependency can be either beneficial or harmful.* In maps 8, 10, 13, 14, $R(\pi)$ is greater than $\widehat{R}(\pi)$ but in maps 7, 11, 12, $R(\pi)$ is smaller. Combining this observation with Ratio 3, we find that when the dependency influences in a positive way, the policy obtained in scenario (L) can be better than the one obtained in scenario (R) where the influence of the dependency is negligible. We also notice that 7 maps are selected for $T = 80, 100$ but only 1 map is select for $T = 60$. This shows that *the dependency has a larger impact when the fleet is given more mission time.*

Finally, for both scenarios presented in Table 8, $R(\pi)$ and $\widehat{R}(\pi)$ are close in map 9 and $R(\pi)$ is greater than $\widehat{R}(\pi)$ in map 14. However, in both maps we observe that the policy found in scenario (R) is better than the one found in scenario (L). This means that *regardless of the dependency, the travel direction can make an significant influence on the performance of the policy obtained through region-sharing.* This observation is consistent with the observation obtained from Tables 5 and 6 that the search sequence of the regions can make an impact to the expected reward that the fleet can collect.

Table 8: Random maps for testing the influence of travel directions

Map	$W \times L$	T	$\widetilde{V}^{(L)}(S)$	$V^{(L)}(S)$	$\widetilde{V}^{(R)}(S)$	$V^{(R)}(S)$	Ratio 1	Ratio 2	Ratio 3
7	14×20	65	6.84	6.66	6.87	6.89	-2.63%	0.29%	-3.34%
8	14×20	80	10.26	11.05	10.41	10.41	7.70%	0.00%	6.15%
9	14×20	80	8.73	8.73	9.32	9.32	0.00%	0.00%	-6.33%
10	14×20	80	10.78	11.42	10.79	10.80	5.94%	0.09%	5.74%
11	14×20	100	12.72	12.44	12.65	12.66	-2.20%	0.08%	-1.74%
12	14×20	100	11.98	11.04	11.98	11.98	-7.85%	0.00%	-7.85%
13	14×20	100	10.51	11.09	10.58	10.58	5.52%	0.00%	4.82%
14	14×20	100	11.16	11.65	11.19	12.02	4.39%	7.42%	-3.08%

Ratio 1: $\frac{V^{(L)}(S)}{\widetilde{V}^{(L)}(S)} - 1$, Ratio 2: $\frac{V^{(R)}(S)}{\widetilde{V}^{(R)}(S)} - 1$, Ratio 3: $\frac{V^{(L)}(S)}{V^{(R)}(S)} - 1$.
Unlisted parameters: $[\underline{\gamma}, \overline{\gamma}] = [0.0, 2.0]$, $[\underline{e}, \overline{e}] = [0.3, 0.7]$, $[\underline{\rho}, \overline{\rho}] = [0.15, 0.35]$.

6.5 Multi-vehicle Implementation

Finally, we explain how to implement the method in cases having more than two vehicles under Assumption 1. The approximate formulation (IAP) and the solution algorithm (Algorithm 1) can be modified in the following way: For each vehicle i, problem (IAP^i) is constructed by fixing the policies applied by all vehicles except vehicle i where $\hat{\tau}^i_\alpha(\pi), \alpha \in \mathcal{S}_i$ may be calculated from different vehicles' policies. Then we can obtain a Markovian policy by solving problem (IAP^i) using the same formulation defined by (15) and (17). Within each iteration of the algorithm, we solve problem (IAP^i) for all $i \in \mathcal{U}$ and the algorithm continues until no improvement is made during an entire iteration where a local optimum of problem (IAP) is reached. To calculate the expected reward of the obtained policy, we first calculate $R^{\pi_i}, i \in \mathcal{U}$, i.e., the expected reward that vehicle i collects from its non-shared regions, for which we can use the same algorithm provided in §B. Since a shared region α is assigned to two vehicles under Assumption 1, we can still use Theorem 3 to identify the dependent set of this region with respect to the two vehicles. Then (19) can be applied to calculate the expected reward collected from the region. However, as explained in §4.4, calculating the expected reward from a shared region requires enumerating $2^{|\mathcal{S}|-1}$ possible scenarios in the worst case, where $|S|$ is the number of shared regions between the two vehicles. If every vehicle shares regions with each of the other vehicles, we need to enumerate a total number of $\sum_{i \neq j} 2^{|\mathcal{S}_i \bigcap \mathcal{S}_j|-1}$ possible scenarios in the worst case, which is computationally expensive. This is the reason that we do not perform the numerical tests for scenarios having more than two vehicles. Nevertheless, the computation time to obtain a Markovian policy that establishes a local optimum of problem (IAP) does not increase significantly since the computation time that each iteration of Algorithm 1 takes increases linearly with respect to the number of vehicles. Finally, we should note that the bound in Theorem 5 still holds in the multi-vehicle case as long as Assumption 1 holds. The corresponding proof can be mimicked using the steps presented in §C.

7. Conclusion and Future Research

This paper studies how to route a fleet of vehicles to search and collect information from a set of regions with uncertain information. We demonstrate the benefit of using a region-sharing strategy under a decentralized environment and develop a method that conquers the computational difficulty of the associated time-allocation problem. A Markovian policy is derived to guide the vehicles' decentralized decisions, which is obtained through a decomposable approximation of the original problem. We propose the concept of dependent set to calculate the exact value of such a policy using conditional independency. A sufficient condition is developed under which there exists an optimal Markovian policy that solves the time-allocation problem, which also implies that the approximation is exact given the condition. To examine the performance loss of using a Markovian policy, we develop a tight upper bound on the performance of a decentralized time-allocation policy. Through a numerical study we show that region-sharing is beneficial even in the scenarios where the regions' cooperation factors are low. In addition, when the cooperation factors increase, we observe an increasing trend of the performance improvement through region-sharing. These results establish the value of the method developed in this paper. Insights are gained for understanding how mission time, cooperation factor and search sequence influence the vehicles' behaviors under a region-sharing strategy.

The strength and simplicity of a Markovian policy makes it a valuable alternative solution for other DEC-POMDPs, which involve multi-agent sequential resource-allocation. The resource to allocate is not restricted to time. For instance, we may have a team of agents, each of which has a fixed budget to fund several assigned projects. Each project may consume a different amount of money to continue in each period and the success of a (some) project(s) provides a (joint) reward to the team. A Markovian policy can be obtained using a similar iterative method under an independency assumption.

The study in this paper has several limitations. The theories only hold for the scenario where a region is shared by no more than two vehicles since we do not provide a reward model to determine the joint reward collected in a region that is searched by more than two vehicles. Given a proper reward model, we can still find a Markovian policy using the iterative algorithm designed in the paper under the same independency assumption but the upper bound will not hold. We leave the investigation for future research considering that many missions may require cooperation among several vehicles for a single task (region). Another extension is to allow a region to have multiple pieces of information. Under this situation, a vehicle might not leave a region after detecting a single piece of information. Moreover, our model is established by assuming that the existence of information and the corresponding detection time in different regions are independent, and the detection time of two vehicles in a shared region is conditionally independent given that information exists. Relaxation of these independencies is suggested for future work. In short, our study is only an initial attempt to model and solve the decentralized multi-vehicle resource-allocation for information searching and collecting. With future relaxations, we believe that the model can cover a broader class of DEC-POMDP problems. Our hope is that the proposed method can turn into a powerful tool to conquer those problems which are known for their notorious computational complexities.

References

Aras, R., A. Dutech. 2010. An investigation into mathematical programming for finite horizon decentralized POMDPs. *Journal of Artificial Intelligence Research* **37**(1) 329–396.

BBC. 2014. The search for flight MH370. `http://www.bbc.com/news/world-asia-26514556`; accessed 5-May-2014.

Becker, R., S. Zilberstein, C. V. Goldman. 2004. Solving transition independent decentralized Markov decision processes **22**(1) 423–455.

Bernstein, D.S., R. Givan, N. Immerman, S. Zilberstein. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research* **27**(4) 819–840.

Chao, I., B. L. Golden, E. A. Wasil. 1996. The team orienteering problem. *European Journal of Operational Research* **88**(3) 464–474.

Gong, Q., R. Batta. 2007. Allocation and reallocation of ambulances to casualty clusters in a disaster relief operation. *IIE Transactions* **39**(1) 27–39.

Haghighat, M. B. A., A. Aghagolzadeh, H. Seyedarabi. 2011. A non-reference image fusion metric based on mutual information of image features. *Computers & Electrical Engineering* **37**(5) 744–756.

Hart, P. E., N. J. Nilsson, B. Raphael. 1968. A formal basis for the heuristic determination of minimum cost paths. *Systems Science and Cybernetics, IEEE Transactions on* **4**(2) 100–107.

Howard, C. 2013. Uav command, control & communications. `http://www.militaryaerospace.com/articles/print/volume-24/issue-7/special-report/uav-command-control-communications.html`; accessed 1-May-2014.

IBM. 2014. CPLEX optimizer. `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`; accessed 10-August-2014.

Koopman, B. O. 1980. *Search and Screening: General Principles with Historical Applications*. Pergamon Press, Amsterdam, Netherlands.

Kress, M., J. O. Royset. 2008. Aerial search optimization model (ASOM) for UAVs in special operations. *Military Operations Research* **13**(1) 23–33.

Kroese, D. P. 2010. *Cross-Entropy Method*. John Wiley & Sons, Hoboken, New Jersey, USA.

Merino, L., F. Caballero, J. R. Martínez-de Dios, J. Ferruz, A. Ollero. 2006. A cooperative perception system for multiple UAVs: Application to automatic detection of forest fires. *Journal of Field Robotics* **23**(3-4) 165–184.

Mufalli, F., R. Batta, R. Nagi. 2012. Simultaneous sensor selection and routing of unmanned aerial vehicles for complex mission plans. *Computers & Operations Research* **39**(11).

Murray, C. C., M. H. Karwan. 2010. An extensible modeling framework for dynamic reassignment and rerouting in cooperative airborne operations. *Naval Research Logistics* **57**(7) 634–652.

Nair, R., M. Tambe, M. Yokoo, D. Pynadath, S. Marsella. 2003. Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. *Proceedings of the 18th International Joint Conference on Artificial Intelligence*. 705–711.

Nakamura, E. F., A. A. F. Loureiro, A. C. Frery. 2007. Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys* **39**(3).

Oliehoek, F. A., J. F. P. Kooij, N. Vlassis. 2008a. The cross-entropy method for policy search in decentralized POMDPs. *Informatica* **32**(4) 341–357.

Oliehoek, F. A., M. T. J. Spaan, N. A. Vlassis. 2008b. Optimal and approximate Q-value functions for decentralized POMDPs. *Journal of Artificial Intelligence Research* **32**(1) 289–353.

Pietz, J., J. O. Royset. 2013. Generalized orienteering problem with resource dependent rewards. *Naval Research Logistics* **60**(4) 294–312.

Powell, W.B. 2007. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. John Wiley & Sons, Hoboken, New Jersey, USA.

Rathinam, S., R. Sengupta, S. Darbha. 2007. A resource allocation algorithm for multivehicle systems with nonholonomic constraints. *Automation Science and Engineering, IEEE Transactions on* **4**(1) 98–104.

Romesh, R. 2013. Five reasons why drones are here to stay. `http://www.businessweek.com/articles/2013-05-23/five-reasons-why-drones-are-here-to-stay#p1`; accessed 15-July-2014.

Schumacher, C., P. R. Chandler, M. Pachter, L. S. Pachter. 2006. Optimization of air vehicles operations using mixed-integer linear programming. *Journal of the Operational Research Society* **58**(4) 516–527.

Seiler, P., R. Sengupta. 2001. Analysis of communication losses in vehicle control problems. *Proceedings of the 2001 American Control Conference*, vol. 2. 1491–1496.

Seuken, S., S. Zilberstein. 2008. Formal models and algorithms for decentralized decision making under uncertainty. *Autonomous Agents and Multi-Agent Systems* **17**(2) 190–250.

Shima, T., S. J. Rasmussen. 2009. *UAV Cooperative Decision and Control: Challenges and Practical Approaches*. Society for Industrial Mathematics, Philadelphia, Pennsylvania, USA.

Szer, D., F. Charpillet. 2006. Point-based dynamic programming for DEC-POMDPs. *Proceedings of the 21st National Conference on Artificial Intelligence*, vol. 2. 1233–1238.

Szer, D., F. Charpillet, S. Zilberstein. 2005. MAA*: A heuristic search algorithm for solving decentralized POMDPs. *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence*. 576–583.

Wayne, M. 2014. Drones are cheap, soldiers are not: A cost-benefit analysis of war. `http://theconversation.com/`; accessed 15-July-2014.

A. Integer Programming Formulation for Routing Models

In this section, we provide two examples to illustrate how to create initial route families. The first example minimizes the total travel time of the fleet under the condition that each region has to be visited by exactly one vehicle. This model seeks the maximum total amount of remaining time for the fleet to invest in searching and information collecting; however, it may assign unbalanced workload to the fleet members. For instance, a vehicle may incur a large travel time to cover many regions so that it has little time to invest in searching. On the contrary, another may have a large amount of remaining time to spend but only have a few regions to search. To help circumvent this issue, the second model minimizes the maximal travel time of a vehicle. This model seeks a solution under which each vehicle will have a reasonable amount of time to spend in searching and information collecting after the assignment. We define the following notation.

$\mathcal{U}$	set of vehicles.
$\mathcal{A}$	set of regions to search, $\mathcal{A} = \{1, 2, ..., L\}$.
m	index of a vehicle, $m \in \mathcal{U}$.
i, j	indices of regions, $i, j \in \mathcal{A}$.
$0, L+1$	indices represent a vehicle's start and end depots, respectively.
$x^m_{i,j}$	a binary variable represents whether vehicle m travels from region i to j, $i, j \in \mathcal{A}$.
$x^m_{0,i}$	a binary variable represents whether vehicle m travels from its start depot to region i, $i \in \mathcal{A}$.
$x^m_{i,L+1}$	a binary variable represents whether vehicle m travels from region i to its end depot, $i \in \mathcal{A}$.
$x^m_{0,L+1}$	a binary variable represents whether vehicle m travels from its start depot to its end depot
$d(i, j)$	a given integer provides the travel time between region i and j, $i, j \in \mathcal{A}$.
$d^m(0, i)$	a given integer provides the time for vehicle m to travel from its start depot to region i, $i \in \mathcal{A}$.
$d^m(i, L+1)$	a given integer provides the time for vehicle m to travel from region i to its end depot, $i \in \mathcal{A}$.
$d^m(0, L+1)$	a given integer provides the time for vehicle m to travel from its start depot to its end depot.
k^m_i	an integer variable represents the order index of region i for vehicle m.
z^m	an integer variable represents the total travel time of vehicle m.
z_{max}	an integer variable represents the maximal travel time of a vehicle among the fleet.

We first present the formulation for the model that minimizes the total travel time of the fleet subject to that each region in the area is visited by exactly one vehicle.

(Rout$_1$): Minimize the total travel time.

$$\min \sum_{m\in\mathcal{U}}[\sum_{i,j\in\mathcal{A}} d(i,j)x^m_{i,j} + \sum_{i\in\mathcal{A}}(d^m(0,i)x^m_{0,i} + d^m(i,L+1)x^m_{i,L+1}) + d^m_{0,L+1}x^m_{0,L+1}]$$

$$s.t. \quad \sum_{i\in\mathcal{A}} x^m_{0,i} + x^m_{0,L+1} = 1,\ m\in\mathcal{U}, \tag{24a}$$

$$\sum_{i\in\mathcal{A}} x^m_{i,L+1} + x^m_{0,L+1} = 1,\ m\in\mathcal{U}, \tag{24b}$$

$$x^m_{0,i} + \sum_{j\in\mathcal{A}} x^m_{j,i} = x^m_{i,L+1} + \sum_{j\in\mathcal{A}} x^m_{i,j},\ i\in\mathcal{A},\ m\in\mathcal{U}, \tag{24c}$$

$$\sum_{m\in U}[\sum_{j\neq i} x^m_{j,i} + x^m_{0,i}] = 1,\ i\in\mathcal{A},\ m\in\mathcal{U}, \tag{24d}$$

$$k^m_j - k^m_i \geq x^m_{i,j} - (1-x^m_{i,j})(L-1),\ i,j\in\mathcal{A},\ m\in\mathcal{U}, \tag{24e}$$

$$x^m_{i,j}\in\{0,1\},\ i,j\in\mathcal{A},\ m\in\mathcal{U}, \tag{24f}$$

$$x^m_{0,i}\in\{0,1\},\ i\in\mathcal{A},\ m\in\mathcal{U}, \tag{24g}$$

$$x^m_{i,L+1}\in\{0,1\},\ i\in\mathcal{A},\ m\in\mathcal{U}, \tag{24h}$$

$$x^m_{0,L+1}\in\{0,1\},\ m\in\mathcal{U}. \tag{24i}$$

In (Rout$_1$), the objective function is the total travel time of the fleet. (24a)-(24c) compose the flow balance from the start depot to the end depot for each vehicle. (24d) requires that each region is visited by exactly one vehicle. (24e) enforces a strict order of each region visited by a vehicle. (24f)-(24i) are binary constraints.

Then we present the formulation for minimizing the maximal travel time of a fleet member while each region still has to be assigned to at least one vehicle.
(Rout$_2$): Minimize the maximal travel time of each vehicle.

$$\min z_{max}$$

$$s.t. \quad \text{(24a)-(24i)}$$

$$z_{max} \geq \sum_{i,j\in\mathcal{A}} d(i,j)x^m_{i,j} + \sum_{i\in\mathcal{A}}(d^m(0,i)x^m_{0,i} + d^m(i,L+1)x^m_{i,L+1}) + d^m_{0,L+1}x^m_{0,L+1},\ m\in\mathcal{U}. \tag{25}$$

In (Rout$_2$), the right hand side of (25) is the total travel time of vehicle m. Since z_{max} is greater than or equal to each vehicle's total travel time, minimizing z_{max} is equivalent to minimizing the maximal travel time of a fleet member.

Next, we present our re-optimization methods. Consider that vehicle m receives a new assignment $H^m = \{h^m_1, h^m_2, ..., h^m_{L_m}, h^m_{L_m+1}, ..., h^m_{L_m+v_m}\}$. Here $h^m_1, h^m_2, ..., h^m_{L_m}$ are the indices of the originally assigned regions and $1, 2, ..., L_m$ are their corresponding order indices in the original route. $h^m_{L+1}, ..., h^m_{L+v_m}$ are the indices of the shared regions added to the route. We first introduce the formulation for the unconstrained re-optimization method.

(Rout$_3$): Minimize the total travel time of vehicle m.

$$\min \sum_{i,j\in H^m} d(i,j)x^m_{i,j} + \sum_{i\in H^m} (d^m(0,i)x^m_{0,i} + d^m(i,L+1)x^m_{i,L+1}) + d^m_{0,L+1}x^m_{0,L+1}$$

$$s.t. \quad \sum_{i\in H^m} x^m_{0,i} = 1, \tag{26a}$$

$$\sum_{i\in H^m} x^m_{i,L+1} = 1, \tag{26b}$$

$$\sum_{j\in H^m} x^m_{j,i} + x^m_{0,i} = \sum_{j\in H^m} x^m_{i,j} + x^m_{i,L+1},\ i \in H^m, \tag{26c}$$

$$\sum_{j\in H^m} x^m_{j,i} + x^m_{0,i} = 1,\ i \in H^m, \tag{26d}$$

$$t^m_j - t^m_i \geq x^m_{i,j} - (1 - x^m_{i,j})(L-1),\ i,j \in H^m \tag{26e}$$

$$x^m_{i,j} \in \{0,1\},\ i,j \in \mathcal{A}, \tag{26f}$$

$$x^m_{0,i} \in \{0,1\},\ i \in \mathcal{A}, \tag{26g}$$

$$x^m_{i,L+1} \in \{0,1\},\ i \in \mathcal{A}, \tag{26h}$$

$$x^m_{0,L+1} \in \{0,1\}. \tag{26i}$$

In (Rout$_3$), the objective function is the total travel time of vehicle m. Constraints (26a)-(26c) establish the flow balance. (26d) requires that each assigned region has to be visited by the vehicle. (26e) enforces a strict order of the assigned regions to be visited. (26f)-(26i) are binary constraints.

Finally, we present the formulations for the constrained re-optimization method.
(Rout$_4$): Minimize the total travel time of vehicle m but retrain the orders of its original regions.

$$\min z_{max}$$

$$s.t. \quad \text{(26a)-(26i)}$$

$$t^m_{h_{l+1}} \geq t^m_{h_l},\ l = 1,2,...,L_m - 1. \tag{27}$$

(27) is the additional constraint that we put for the constrained re-optimization, which retains the orders of the original regions. Note that region h^m_{l+1} has a higher order than region h^m_l in the original route and (27) enforces that it must be visited later than region h^m_l in the re-optimized route.

B. An Algorithm to Calculate the Expected Reward and Success Probabilities

For a suggested route $H_i = \{h^i_0, h^i_1, h^i_2, ..., h^i_{L_i}, h^i_{L_i+1}\}$ and a Markovian policy π_i, Algorithm 2 provides two outputs: $\tau^i_{h^i_l}(\pi_i|\theta_{\mathcal{V}^i_{h^i_l}})$ for $h^i_l \in \mathcal{A}$ and R^{π_i} (only when $\mathcal{V} = \emptyset$). For any given $\mathcal{V} \subset \mathcal{S}$, we define $\mathcal{V}^i_{h^i_l} = \{h^i_1, h^i_2, ..., h^i_{l-1}\} \bigcap \mathcal{V}$. $\tau^i_{h^i_l}(\pi_i|\theta_{\mathcal{V}^i_{h^i_l}})$ is the conditional probability that vehicle i succeeds in region h^i_l given $\theta_{\mathcal{V}^i_{h^i_l}}$ and $\theta_{h^i_l} = 1$.

Algorithm 2

Require: $H^i = \{h_0^i, h_1^i, h_2^i, ..., h_{L_i}^i, h_{L_i+1}^i\}$, $\pi_i = \{z_{h_l^i}^i(T); x_{h_l^i}^i(t), y_{h_l^i}^i(t), z_{h_l^i}^i(t) : t = 0, 1, 2, ..., T, l = 1, 2, ..., L_i\}$ and $\theta_{\mathcal{V}}$.

Initialization: $R^{\pi_i} \leftarrow 0$; $l_c \leftarrow 0$; $f_{h_l^i}^s(t|\theta_{\mathcal{V}^i_{h_l^i}}), \tilde{f}_{h_l^i}^c(t|\theta_{\mathcal{V}^i_{h_l^i}}), f_{h_l^i}^c(t|\theta_{\mathcal{V}^i_{h_{l+1}^i}}), f_{h_l^i}^d(t|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) \leftarrow 0$ for $l = 1 : L_i$, $t = 0 : T$; $\tau_{h_l^i}^i(\pi_i|\theta_{\mathcal{V}^i_{h_l^i}}) \leftarrow 0$ for $l = 1 : L_i$.
$l_c \leftarrow z_{h_0^i}^i(T)$, $f_{l_c^i}^s(T - d(h_0^i, h_{l_c}^i)|\theta_{\mathcal{V}^i_{h_l^i}}) \leftarrow 1$.
for $l = l_c : L_i$ **do**
  for $t:\ f_{h_l^i}^s(t|\theta_{\mathcal{V}^i_{h_l^i}}) > 0$ **do**
    for $x = 1 : x_{h_l^i}^i(t)$ **do**
      $\tilde{f}_{h_l^i}^c(t - x|\theta_{\mathcal{V}^i_{h_l^i}}) \leftarrow \tilde{f}_{h_l^i}^c(t - x|\theta_{\mathcal{V}^i_{h_l^i}}) + e_{h_l^i} p_{h_l^i}(x)$.
    end for
    if $h_l^i \notin \mathcal{V}$ **then**
      for $x = 1 : x_{h_l^i}^i(t)$ **do**
        $f_{h_l^i}^c(t - x|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) \leftarrow f_{h_l^i}^c(t - x|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) + e_{h_l^i} p_{h_l^i}(x)$.
      end for
      $f_{h_l^i}^d(t - x_{h_l^i}^i(t)|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) \leftarrow f_{h_l^i}^d(t - x_{h_l^i}^i(t)|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) + [1 - e_{h_l^i} P_{h_l^i}(x_{h_l^i}^i(t))] f_{h_l^i}^s(t|\theta_{\mathcal{V}^i_{h_l^i}})$.
    else
      if $\theta_{h_l^i} = 1$ **then**
        for $x = 1 : x_{h_l^i}^i(t)$ **do**
          $f_{h_l^i}^c(t - x|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) \leftarrow f_{h_l^i}^c(t - x|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) + p_{h_l^i}(x)$.
        end for
        $f_{h_l^i}^d(t - x_{h_l^i}^i(t)|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) \leftarrow f_{h_l^i}^d(t - x_{h_l^i}^i(t)|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) + [1 - P_{h_l^i}(x_{h_l^i}^i(t))] f_{h_l^i}^s(t|\theta_{\mathcal{V}^i_{h_l^i}})$.
      else
        $f_{h_l^i}^d(t - x_{h_l^i}^i(t)|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) \leftarrow f_{h_l^i}^d(t - x_{h_l^i}^i(t)|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) + f_{h_l^i}^s(t|\theta_{\mathcal{V}^i_{h_l^i}})$.
      end if
    end if
  end for
  for $t:\ \tilde{f}_{h_l^i}^c(t|\theta_{\mathcal{V}^i_{h_l^i}}) > 0$ **do**
    $\tau_{h_l^i}^i(\pi_i|\theta_{\mathcal{V}^i_{h_l^i}}) \leftarrow \tau_{h_l^i}^i(\pi_i|\theta_{\mathcal{V}^i_{h_l^i}}) + y_{h_l^i}^i(t) \tilde{f}_{h_l^i}^c(t|\theta_{\mathcal{V}^i_{h_l^i}})/(1 - e_{h_l^i})$.
  end for
  for $t:\ f_{h_l^i}^c(t|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) > 0$ **do**
    if $l^i \in \mathcal{O}_i$ **and** $\mathcal{V} = \emptyset$ **then**
      $R^{\pi_i} \leftarrow R^{\pi_i} + y_{h_l^i}^i(t) f_{h_l^i}^c(t|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) g_{h_l^i}$.
    end if
    $f_{h_l^i}^d(t - y_{h_l^i}^i(t) s_{h_l^i}|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) \leftarrow f_{h_l^i}^d(t - y_{h_l^i}^i(t) s_{h_l^i}|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) + f_{h_l^i}^c(t|\theta_{\mathcal{V}^i_{h_{l+1}^i}})$.
  end for
  for $t:\ f_{h_l^i}^d(t|\theta_{\mathcal{V}^i_{h_{l+1}^i}}) > 0$ **do**
    $l_d \leftarrow z_{h_l^i}^i(t)$, $f_{h_{l_d}^i}^s(t - d(h_l^i, h_{l_d}^i)|\theta_{\mathcal{V}^i_{h_{l_d}^i}}) \leftarrow f_{h_l^i}^d(t|\theta_{\mathcal{V}^i_{h_{l+1}^i}})$.
  end for
end for
return R^{π_i} and $\tau_{h_l^i}^i(\pi_i|\theta_{\mathcal{V}^i_{h_l^i}})$ for $l = 1 : L_i$.

We use the following quantities in Algorithm 2: For a given $\theta_{\mathcal{V}^i_{h^i_l}}$, $f^s_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_l}})$ is the conditional probability that vehicle i arrives at region h^i_l with t units of remaining time and $\tilde{f}^c_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_l}})$ is the conditional probability that vehicle i detects the information in region h^i_l with t units of remaining time. For a given $\theta_{\mathcal{V}^i_{h^i_{l+1}}}$, $f^c_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_{l+1}}})$ is the conditional probability that vehicle i detects the information in h^i_l with t units of remaining time and $f^d_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_{l+1}}})$ is the conditional probability that vehicle i leaves region h^i_l with t units of remaining time.

In the algorithm, lines 3-23 correspond to searching. Lines 6-9 update $\tilde{f}^s_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_l}})$. Lines 8-23 update $f^c_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_{l+1}}})$ and $f^d_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_{l+1}}})$: Lines 9-12 are for the scenario $h^i_l \notin \mathcal{V}$; lines 15-18 are for the scenario where $\theta_{h^i_l} = 1$ is given in $\theta_{\mathcal{V}^i_{h^i_{l+1}}}$; line 20 is for the scenario where $\theta_{h^i_l} = 0$ is given in $\theta_{\mathcal{V}^i_{h^i_{l+1}}}$. Lines 24-32 correspond to information collecting. Lines 24-26 update $\tau^i_{h^i_l}(\pi_i|\theta_{\mathcal{V}^i_{h^i_l}})$ using $\tilde{f}^s_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_l}})$ calculated in lines 6-9. Lines 28-30 update the reward R^{π_i}. Line 31 updates $f^d_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_{l+1}}})$. Lines 33-35 correspond to leaving a region where we update $f^s_{h^i_{l_d}}(t|\theta_{\mathcal{V}^i_{h^i_{l_d}}})$ for the regions that will be visited afterwards using $f^d_{h^i_l}(t|\theta_{\mathcal{V}^i_{h^i_{l+1}}})$.

Finally, if we set $\mathcal{V} = \emptyset$, the algorithm outputs R^{π_i} and $\tau^i_{h^i_l}(\pi_i)$. If we set $\mathcal{V} = \Phi_\beta$ where Φ_β is the largest dependent set, $\tau^i_{h^i_l}(\pi_i|\theta_{\mathcal{V}^i_{h^i_l}})$ is the conditional probability required by (19) in §4.4.

C. Proofs

This section presents the proofs of the theorems and propositions proposed in §4 and §5 of the main article. Let $\mathbf{x}$ be an arbitrary k-dimension integer vector, throughout the reminder of the appendix, $\sum_{\mathbf{x}}$ represents that the summation is over the k-dimension integer lattice, i.e., $\mathbb{Z}^k$.

Proof of Theorem 2. The fleet's policy changes only in line 5 and line 7 of Algorithm 1. Without losing generality, we assume that $i = 1, j = 2$. Let (π^k_1, π^k_2) be the policy obtained after k iterations of the loop defined in lines 3-10. R^k and R^{*k} are the corresponding values of R and R^* at the end of the kth iteration (after line 9 is executed). R represents the objective value achieved in the former iteration and R^* is the objective value achieved in the current iteration. At iteration $k > 1$, the problem solved in line 5 is $\pi^k_1 \in \arg\max\{\widehat{R}(\pi_1, \pi^{k-1}_2) : \pi_1 \in \widehat{\Pi}_1\}$ and the problem solved in line 7 is $\pi^k_2 \in \arg\max\{\widehat{R}(\pi^k_1, \pi_2) : \pi_2 \in \widehat{\Pi}_2\}$. We should note that $R^k = \widehat{R}(\pi^{k-1}_1, \pi^{k-1}_2)$ and $R^{*k} = \widehat{R}(\pi^k_1, \pi^k_2)$. First, we have

$$R^k = \widehat{R}(\pi^{k-1}_1, \pi^{k-1}_2) \leq \max\{\widehat{R}(\pi_1, \pi^{k-1}_2) : \pi_1 \in \widehat{\Pi}_1\} \leq \max\{\widehat{R}(\pi^k_1, \pi_2) : \pi_2 \in \widehat{\Pi}_2\} = R^{*k}. \tag{28}$$

If $R^{k*} > R^k$ for $k = 1, 2, ..., K$, we have $\widehat{R}(\pi^K_1, \pi^K_2) > \widehat{R}(\pi^{K-1}_1, \pi^{K-1}_2) > ... > \widehat{R}(\pi^1_1, \pi^1_2)$. Thus, we must go over K different policies. Since there is a finite number of policies, after a finite number of iterations $(\overline{K})$, we must reach $R^* = R$, which is equivalent to $\widehat{R}(\pi^{\overline{K}-1}_1, \pi^{\overline{K}-1}_2) = \widehat{R}(\pi^{\overline{K}}_1, \pi^{\overline{K}}_2)$. According to (28), $\pi^{\overline{K}-1}_1$ must solve $\max\{\widehat{R}(\pi_1, \pi^{k-1}_2) : \pi_1 \in \Pi_1\}$ in line 5 and $\pi^{\overline{K}-1}_2$ must solve $\max\{\widehat{R}(\pi^{k-1}_1, \pi_2) : \pi_2 \in \Pi_2\}$

in line 7 (note that no update is made at line 5); otherwise, R^* must be strictly greater than R. Therefore,

$$\pi_1^{\overline{K}-1} \in \arg\max\{\widehat{R}(\pi_1, \pi_2^{\overline{K}-1}) : \pi_1 \in \widehat{\Pi}_1\},$$
$$\pi_2^{\overline{K}-1} \in \arg\max\{\widehat{R}(\pi_1^{\overline{K}-1}, \pi_2) : \pi_2 \in \widehat{\Pi}_2\},$$

and the algorithm converges to a local optimum of problem (IAP). □

Assumption 2 serves as the key assumption to establish the proof. We first introduce a lemma that develops three direct results from Assumption 2.

Lemma 2 *Under Assumption 2, for any route family we have the following results:*

i). For $\mathcal{V} \subseteq \mathcal{A}$,

$$P(\{\theta_\alpha\}_{\alpha\in\mathcal{V}}) = \prod_{\alpha\in\mathcal{V}} f_{\theta_\alpha}(\theta_\alpha) \tag{29}$$

ii). Consider two region sets $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathcal{A}$. *Let* $\mathcal{S}' = \mathcal{V}_1 \bigcap \mathcal{V}_2$ *and* $\mathcal{O}_i' = \mathcal{V}_i \setminus \mathcal{S}', i \in \{1, 2\}$, *and we have*

$$\begin{aligned} &P(\{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{O}_1'}, \{w_{\alpha_2}^2, \theta_{\alpha_2}\}_{\alpha_2\in\mathcal{O}_2'}, \{w_\alpha^1, w_\alpha^2\}_{\alpha\in\mathcal{S}'} | \{\theta_\alpha\}_{\alpha\in\mathcal{S}'}) \\ =& \prod_{\alpha_1\in\mathcal{O}_1'} f_{w_{\alpha_1}^1}(w_{\alpha_1}^1|\theta_{\alpha_1}) f_{\theta_{\alpha_1}}(\theta_{\alpha_1}) \prod_{\alpha_2\in\mathcal{O}_2'} f_{w_{\alpha_2}^2}(w_{\alpha_2}^2|\theta_{\alpha_2}) f_{\theta_{\alpha_2}}(\theta_{\alpha_2}) \prod_{\alpha\in\mathcal{S}'} f_{w_\alpha^1}(w_\alpha^1|\theta_\alpha) f_{w_\alpha^i}(w_\alpha^2|\theta_\alpha) \\ =& P(\{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{O}_1'}, \{w_\alpha^1\}_{\alpha\in\mathcal{S}'} | \{\theta_\alpha\}_{\alpha\in\mathcal{S}'}) P(\{w_{\alpha_2}^2, \theta_{\alpha_2}\}_{\alpha_2\in\mathcal{O}_2'}, \{w_\alpha^2\}_{\alpha\in\mathcal{S}'} | \{\theta_\alpha\}_{\alpha\in\mathcal{S}'}), \end{aligned} \tag{30}$$

where

$$P(\{w_{\alpha_i}^i, \theta_{\alpha_i}\}_{\alpha_i\in\mathcal{O}_i'}, \{w_\alpha^i\}_{\alpha\in\mathcal{S}'} | \{\theta_\alpha\}_{\alpha\in\mathcal{S}'}) = \prod_{\alpha_i\in\mathcal{O}_i'} f_{w_{\alpha_i}^i}(w_{\alpha_i}^i|\theta_{\alpha_i}) f_{\theta_{\alpha_i}}(\theta_{\alpha_i}) \prod_{\alpha\in\mathcal{S}'} f_{w_\alpha^i}(w_\alpha^i|\theta_\alpha),\ i \in \{1, 2\}. \tag{31}$$

iii). Consider two region sets $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathcal{A}$ *satisfying* $\mathcal{V}_1 \bigcap \mathcal{V}_2 = \emptyset$. *We have*

$$P(\{w_{\alpha_2}^2, \theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2}, \{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}) = P(\{w_{\alpha_2}^2, \theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2}) P(\{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}). \tag{32}$$

Moreover, let $g(\{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1})$ *be a function of* $\{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}$ *and* x *be a reachable value of* $g(\{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1})$. *We have*

$$P(\{w_{\alpha_2}^2, \theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2} | g(\{w_{\alpha_1}^1, \theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}) = x) = \prod_{\alpha_2\in\mathcal{V}_2} f_{w_{\alpha_2}^2}(w_{\alpha_2}^2|\theta_{\alpha_2}) f_{\theta_{\alpha_2}}(\theta_{\alpha_2}). \tag{33}$$

Proof. i). Let $\mathcal{U}' = \{1\}$ in (4) and we have

$$\begin{aligned} P(\{\theta_\alpha\}_{\alpha\in\mathcal{V}}) &= \sum_{\{w_\alpha^1\}_{\alpha\in\mathcal{V}}} P(\{w_\alpha^1, \theta_\alpha\}_{\alpha\in\mathcal{V}}) \\ &= \sum_{\{w_\alpha^1\}_{\alpha\in\mathcal{V}}} \prod_{\alpha\in\mathcal{V}} f_{w_\alpha^1}(w_\alpha^1|\theta_{\alpha_1}) f_{\theta_{\alpha_1}}(\theta_{\alpha_1}) = \prod_{\alpha\in\mathcal{V}} [f_{\theta_{\alpha_1}}(\theta_{\alpha_1}) \sum_{w_\alpha^1} f_{w_\alpha^1}(w_\alpha^1|\theta_{\alpha_1})] = \prod_{\alpha\in\mathcal{V}} f_{\theta_\alpha}(\theta_\alpha). \end{aligned}$$

ii). It is obvious that

$$P(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{O}'_1},\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{O}'_2},\{w^1_\alpha,w^2_\alpha\}_{\alpha\in\mathcal{S}'}|\{\theta_\alpha\}_{\alpha\in\mathcal{S}'}) = \frac{P(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{O}'_1},\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{O}'_2},\{w^1_\alpha,w^2_\alpha,\theta_\alpha\}_{\alpha\in\mathcal{S}'})}{P(\{\theta_\alpha\}_{\alpha\in\mathcal{S}'})}. \tag{34}$$

Using Assumption 2 and (29), (34) can be written as

$$\begin{aligned}
&P(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{O}'_1},\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{O}'_2},\{w^1_\alpha,w^2_\alpha\}_{\alpha\in\mathcal{S}'}|\{\theta_\alpha\}_{\alpha\in\mathcal{S}'})\\
=&\frac{\prod_{\alpha_1\in\mathcal{O}'_1} f_{w^1_{\alpha_1}}(w^1_{\alpha_1}|\theta_{\alpha_1})f_{\theta_{\alpha_1}}(\theta_{\alpha_1})\prod_{\alpha_2\in\mathcal{O}'_2} f_{w^2_{\alpha_2}}(w^2_{\alpha_2}|\theta_{\alpha_2})f_{\theta_{\alpha_2}}(\theta_{\alpha_2})\prod_{\alpha\in\mathcal{S}'} f_{w^1_\alpha}(w^1_\alpha|\theta_\alpha)f_{w^i_\alpha}(w^2_\alpha|\theta_\alpha)f_{\theta_\alpha}(\theta_\alpha)}{\prod_{\alpha\in\mathcal{S}'} f_{\theta_\alpha}(\theta_\alpha)}\\
=&\prod_{\alpha_1\in\mathcal{O}'_1} f_{w^1_{\alpha_1}}(w^1_{\alpha_1}|\theta_{\alpha_1})f_{\theta_{\alpha_1}}(\theta_{\alpha_1})\prod_{\alpha_2\in\mathcal{O}'_2} f_{w^2_{\alpha_2}}(w^2_{\alpha_2}|\theta_{\alpha_2})f_{\theta_{\alpha_2}}(\theta_{\alpha_2})\prod_{\alpha\in\mathcal{S}'} f_{w^1_\alpha}(w^1_\alpha|\theta_\alpha)f_{w^2_\alpha}(w^2_\alpha|\theta_\alpha).
\end{aligned}$$

We only need to show that (31) is true. Using Assumption 2 and (29), we have

$$\begin{aligned}
&P(\{w^i_{\alpha_i},\theta_{\alpha_i}\}_{\alpha_i\in\mathcal{O}'_i},\{w^i_\alpha\}_{\alpha\in\mathcal{S}'}|\{\theta_\alpha\}_{\alpha\in\mathcal{S}'}) = \frac{P(\{w^i_{\alpha_i},\theta_{\alpha_i}\}_{\alpha_i\in\mathcal{O}'_i},\{w^i_\alpha,\theta_\alpha\}_{\alpha\in\mathcal{S}'})}{P(\{\theta_\alpha\}_{\alpha\in\mathcal{S}'})}\\
=&\frac{\prod_{\alpha_i\in\mathcal{O}'_i} f_{w^i_{\alpha_i}}(w^i_{\alpha_i}|\theta_{\alpha_i})f_{\theta_{\alpha_i}}(\theta_{\alpha_i})\prod_{\alpha\in\mathcal{S}'} f_{w^i_\alpha}(w^i_\alpha|\theta_\alpha)f_{\theta_\alpha}(\theta_\alpha)}{\prod_{\alpha\in\mathcal{S}'} f_{\theta_\alpha}(\theta_\alpha)} = \prod_{\alpha_i\in\mathcal{O}'_i} f_{w^i_{\alpha_i}}(w^i_{\alpha_i}|\theta_{\alpha_i})f_{\theta_{\alpha_i}}(\theta_{\alpha_i})\prod_{\alpha\in\mathcal{S}'} f_{w^i_\alpha}(w^i_\alpha|\theta_\alpha).
\end{aligned}$$

iii). We first prove (32). Using Assumption 2,

$$\begin{aligned}
&P(\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2},\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}) = \prod_{\alpha_1\in\mathcal{V}_1} f_{w^1_{\alpha_1}}(w^1_{\alpha_1}|\theta_{\alpha_1})f_{\theta_{\alpha_1}}(\theta_{\alpha_1})\prod_{\alpha_2\in\mathcal{V}_2} f_{w^2_{\alpha_2}}(w^2_{\alpha_2}|\theta_{\alpha_2})f_{\theta_{\alpha_2}}(\theta_{\alpha_2})\\
=&P(\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2})P(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}).
\end{aligned}$$

The last equality also comes from Assumption 2 by setting $\mathcal{U}=\{i\}$ and $\mathcal{A}^i=\mathcal{V}_i$ for $i\in\{1,2\}$.

Now we prove (33). Define an indicator function $\kappa(x,y)$ so that $\kappa(x,y)=1$ if $x=y$; otherwise, $\kappa(x)=0$.

$$\begin{aligned}
&P(\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2}|g(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1})=x)\\
=&\frac{\sum_{\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}}\kappa(g(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}),x)P(\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2},\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1})}{\sum_{\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}}\kappa(g(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1},x)P(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1})}\\
=&\frac{[\sum_{\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}}\kappa(g(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}),x)P(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1})]P(\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2})}{\sum_{\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1}}\kappa(g(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1},x)P(\{w^1_{\alpha_1},\theta_{\alpha_1}\}_{\alpha_1\in\mathcal{V}_1})}\\
=&P(\{w^2_{\alpha_2},\theta_{\alpha_2}\}_{\alpha_2\in\mathcal{V}_2}) = \prod_{\alpha_2\in\mathcal{V}_2} f_{w^2_{\alpha_2}}(w^2_{\alpha_2}|\theta_{\alpha_2})f_{\theta_{\alpha_2}}(\theta_{\alpha_2}).
\end{aligned}$$

Note that the second equality comes from (32). □

We define a random variable $t^{\pi_i}_{h^i_l}$: For a random scenario, if region h^i_l is not skipped by vehicle i under policy π_i, $t^{\pi_i}_{h^i_l}$ is the remaining time of the vehicle when it arrives at region h^i_l; otherwise, $t^{\pi_i}_{h^i_l}$ is equal to

$d(h_l^i, h_{L_i+1}^i)$. Here a random scenario refers to a realization of the random variables $\{w_{h_k^i}^i, \theta_{h_k^i}\}_{k=1:l-1}$. With the assist of $t_{h_l^i}^{\pi_i}$, we can define an indicator function $u_{h_l^i}^{\pi_i}(w_{h_l^i}^i, t_{h_l^i}^{\pi_i})$ so that $u_{h_l^i}^{\pi_i}(w_{h_l^i}^i, t_{h_l^i}^{\pi_i}) = 1$ if vehicle i succeeds in region h_l^i given $w_{h_l^i}^i$ and $t_{h_l^i}^{\pi_i}$; otherwise, $u_{h_l^i}^{\pi_i}(w_{h_l^i}^i, t_{h_l^i}^{\pi_i}) = 0$. Such an indicator function can be represented in the following:

$$u_{h_l^i}^{\pi_i}(w_{h_l^i}^i, t_{h_l^i}^{\pi_i}) = \begin{cases} 1 & \text{if } w_{h_l^i}^i \le x_{h_l^i}^i(t_{h_l^i}^{\pi_i}) \text{ and } y_{h_l^i}^i(t_{h_l^i}^{\pi_i} - w_{h_l^i}^i) = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{35}$$

To show that the indicator function defined by (35) is consistent with the fact, we consider three possible cases for a random scenario:

1. Region h_l^i is skipped under the given scenario: When region h_l^i is skipped, we have $t_{h_l^i}^{\pi_i} = d(h_l^i, h_{L_i+1}^i)$. It is obvious that $x_{h_l^i}^i(t_{h_l^i}^{\pi_i}) = 0 < 1 \le w_{h_l^i}^i$. Thus, we have $u_{h_l^i}^{\pi_i}(w_{h_l^i}^i, t_{h_l^i}^{\pi_i}) = 0$ under this case.

2. Region h_l^i is visited by vehicle i but does not have information: When region h_l^i does not have information, we have $w_{h_l^i}^i = T + 1$. Since $x_{h_l^i}^i(t_{h_l^i}^{\pi_i}) \le T < T + 1$, we also have $u_{h_l^i}^{\pi_i}(w_{h_l^i}^i, t_{h_l^i}^{\pi_i}) = 0$ under this case.

3. Region h_l^i is visited by vehicle i and has information: The search time scheduled by vehicle i in region h_l^i is $x_{h_l^i}^i(t_{h_l^i}^{\pi_i})$ and the information can be detected if and only if $x_{h_l^i}^i(t_{h_l^i}^{\pi_i}) \ge w_{h_l^i}^i$. If the information is detected by the vehicle, the vehicle will have $t_{h_l^i}^{\pi_i} - w_{h_l^i}^i$ units of remaining time. Then, the vehicle will collect the information if and only if $y_{h_l^i}^i(t_{h_l^i}^{\pi_i} - w_{h_l^i}^i) = 1$. Thus, the vehicle will succeed in this case if and only if $x_{h_l^i}^i(t_{h_l^i}^{\pi_i}) \ge w_{h_l^i}^i$ and $y_{h_l^i}^i(t_{h_l^i}^{\pi_i} - w_{h_l^i}^i) = 1$.

Therefore, the indicator function is consistent with the fact.

The proof of Theorem 3 relies on the following lemma, which states that a vehicle's remaining time when it arrives at a region is only dependent on the realizations of the random variables associated with the regions that have lower orders than the current region.

Lemma 3 *Given a Markovian policy π_i, $t_{h_{l_i}^i}^{\pi_i}$ can be written as a function of $\{\theta_{h_k^i}, w_{h_k^i}^i\}_{l=1:l_i-1}$ for any $l_i \ge 1$, i.e.*

$$t_{h_{l_i}^i}^{\pi_i} = g_{h_{l_i}^i}^{\pi_i}(\{\theta_{h_l^i}, w_{h_l^i}^i\}_{l=1:l_i-1}), \tag{36}$$

where $g_{h_{l_i}^i}^{\pi_i}(\cdot)$ maps an integer vector to an integer value.

Proof. If $l_i < z_{h_0^i}^i(T)$, we have $t_{h_{1_i}^i}^{\pi_i} = d(h_{l_i}^i, h_{L_i+1}^i)$. Then we define $g_{h_{l_i}^i}^{\pi_i}(\{\theta_{h_l^i}, w_{h_l^i}^i\}_{l=1:l_i-1}) = d(h_{l_i}^i, h_{L_i+1}^i)$. If $l_i = z_{h_0^i}^i(T)$, we have $t_{h_{l_i}^i}^{\pi_i} = T - d(h_1^i, h_{L_i+1}^i)$. Then we define $t_{h_{1_i}^i}^{\pi_i} = T - d(h_1^i, h_{L_i+1}^i)$. Therefore, the lemma holds for $l_i \le z_{h_0^i}^i(T)$.

We will prove the result for $l_i \ge z_{h_0^i}^i(T)$ using induction. As the induction hypothesis, suppose (36) holds for $l_i \le k$, where $k \ge z_{h_0^i}^i(T)$. To complete the induction we only need to show that (36) holds for $l_i = k + 1$. Consider two possible cases:

1. If region h^i_{k+1} is skipped under the scenario, we have $t^{\pi_i}_{h^i_{k+1}} = d(h^i_{k+1}, h_{L_i+1})$. Then we can define $g^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k}) = d(h^i_{k+1}, h_{L_i+1})$.

2. If the vehicle visits region h^i_{k+1}, let $h^i_{k'}$ be the region that the vehicle comes from, where $k' \leq k$. We have

$$t^{\pi_i}_{h^i_{k+1}} = t^{\pi_i}_{h^i_{k'}} - \min\left\{w^i_{h^i_{k'}}, x^i_{h^i_{k'}}(t^{\pi_i}_{h^i_{k'}})\right\} - u^{\pi_i}_{h^i_{k'}}(w^i_{h^i_{k'}}, t^{\pi_i}_{h^i_{k'}})s_{h^i_{k'}} - d(h^i_{k'}, h^i_{k+1}). \tag{37}$$

In the right hand side of (37), the first term is the vehicle's remaining time when it arrives at region $h^i_{k'}$. The second and the third terms are the time spent in searching and in collecting information from region $h^i_{k'}$, respectively. The last term is the travel time from region $h^i_{k'}$ to region h^i_{k+1}. According to the induction hypothesis, $t^{\pi_i}_{h^i_{k'}}(\pi_i) = g^{\pi_i}_{h^i_{k'}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k'-1})$. Define

$$\begin{aligned} g^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k}) =& g^{\pi_i}_{h^i_{k'}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k'-1}) - \min\left\{w^i_{h^i_{k'}}, x^i_{h^i_{k'}}(g^{\pi_i}_{h^i_{k'}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k'-1}))\right\} \\ & - u^{\pi_i}_{h^i_{k'}}(w^i_{h^i_{k'}}, g^{\pi_i}_{h^i_{k'}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k'-1}))s_{h^i_{k'}} - d(h^i_{k'}, h^i_{k+1}). \end{aligned} \tag{38}$$

Combine the two cases and we have $t^{\pi_i}_{h^i_{l_i}} = g^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:l_i})$ well defined for $l_i = k+1$. It is easy to verify that $g^{\pi_i}_{h^i_{k+1}}(\cdot)$ only provides integer values in (38) since all terms used are integers. Thus, the induction is complete and the lemma holds for $l \geq z^i_{h^i_0}(T)$. $\square$

Proof of Theorem 3. Let l_1 and l_2 be the corresponding orders of a shared region $\tilde{\alpha}$ in vehicle 1's and vehicle 2's routes, i.e., $\tilde{\alpha} = h^1_{l_1} = h^2_{l_2}$. For a random scenario, let $\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l_i}$ be the realization of the corresponding random variables that vehicle i will encounter. According to Lemma 3, $t^{\pi_i}_{h^i_{l_i}} = g^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_k}, w^i_{h^i_k}\}_{k=1:l_i-1})$, $i \in \{1,2\}$ where $g^{\pi_i}_{h^i_{l_i}}(\cdot)$, $i \in \{1,2\}$ are well defined functions. According to the definition of dependent set, we have $\{h^1_1, h^1_2, ..., h^1_{l_1-1}\} \bigcap \{h^2_1, h^2_2, ..., h^1_{l_2-1}\} = \Phi_{\tilde{\alpha}}$. Define $\mathcal{O}'_i = \{h^1_1, h^1_2, ..., h^1_{l_1-1}\} \setminus \Phi_{\tilde{\alpha}}$ for $i \in \{1,2\}$ and $\mathcal{S}' = \Phi_{\tilde{\alpha}} \bigcup \{\tilde{\alpha}\}$.

Let $\tau_\alpha(\pi|\theta_{\Phi_{\tilde{\alpha}}})$ be the conditional probability that both vehicles succeed in region $\tilde{\alpha}$ given $\theta_{\mathcal{S}'} = \{\tilde{\theta}_\alpha\}_{\alpha\in\mathcal{S}'}$ where $\tilde{\theta}_{\tilde{\alpha}} = 1$. Using the indicator functions $u^{\pi_i}_{h^i_{l_i}}(w^i_{h^i_{l_i}}, t^{\pi_i}_{h^i_{l_i}})$, $i \in \{1,2\}$, we have

$$\begin{aligned} &\tau_{\tilde{\alpha}}(\pi|\theta_{\Phi_{\tilde{\alpha}}}) \\ =& \sum_{\{\theta_{h^1_{l_1}}, w^1_{h^1_{l_1}}, t^{\pi_1}_{h^1_{l_1}}\}} \sum_{\{\theta_{h^2_{l_2}}, w^2_{h^2_{l_2}}, t^{\pi_2}_{h^2_{l_2}}\}} \left\{u^{\pi_1}_{h^1_{l_1}}(w^1_{h^1_{l_1}}, t^{\pi_1}_{h^1_{l_1}})u^{\pi_2}_{h^2_{l_2}}(w^2_{h^2_{l_2}}, t^{\pi_2}_{h^2_{l_2}})P(\theta_{h^1_{l_1}}, w^1_{h^1_{l_1}}, \theta_{h^2_{l_2}}, w^2_{h^2_{l_2}}, t^{\pi_1}_{h^1_{l_1}}, t^{\pi_2}_{h^2_{l_2}}|\theta_{\mathcal{S}'})\right\}. \end{aligned} \tag{39}$$

To rewrite (39), we use the same indicator function $\kappa(x,y)$ defined for proving $ii)$ of Lemma 2 and $\tau_\alpha(\pi|\theta_{\Phi_{\tilde{\alpha}}})$ is equal to

$$\begin{aligned} &\sum_{\{\theta_{h^1_l}, w^1_{h^1_l}\}_{l=1:l_1}} \sum_{\{\theta_{h^2_l}, w^2_{h^2_l}\}_{l=1:l_2}} \sum_{t_1}\sum_{t_2} \Big\{u^{\pi_1}_{h^1_{l_1}}(w^1_{h^1_{l_1}}, t_1)\kappa(t_1, g^{\pi_1}_{h^1_l}(\{\theta_{h^1_k}, w^1_{h^1_k}\}_{k=1:l_1-1}))u^{\pi_2}_{h^2_{l_2}}(w^2_{h^2_{l_2}}, t_2) \\ &\kappa(t_2, g^{\pi_2}_{h^2_l}(\{\theta_{h^2_k}, w^2_{h^2_k}\}_{k=1:l_2-1}))P(\{\theta_{h^1_k}, w^1_{h^1_k}\}_{k=1:l_1}, \{\theta_{h^2_k}, w^2_{h^2_k}\}_{k=1:l_2}|\theta_{\mathcal{S}'})\Big\}. \end{aligned}$$

For $P(\{\theta_{h_k^1}, w_{h_k^1}^1\}_{k=1:l_1}, \{\theta_{h_k^2}, w_{h_k^2}^2\}_{k=1:l_2}|\theta_{\mathcal{S}'})$, we have

$$P(\{\theta_{h_k^1}, w_{h_k^1}^1\}_{k=1:l_1}, \{\theta_{h_k^2}, w_{h_k^2}^2\}_{k=1:l_2}|\theta_{\mathcal{S}'})$$
$$= \begin{cases} P(\{\theta_{\alpha_1}, w_{\alpha_1}^1\}_{\alpha_1\in\mathcal{O}_1'}, \{\theta_{\alpha_2}, w_{\alpha_2}^2\}_{\alpha_2\in\mathcal{O}_2'}, \{w_\alpha^1, w_\alpha^2\}_{\alpha\in\mathcal{S}'}|\theta_{\mathcal{S}'}) & \text{if } \theta_{h_{k_1}^1} = \theta_{h_{k_2}^2} = \tilde{\theta}_\alpha,\ \forall h_{k_1}^1 = h_{k_2}^2 = \alpha \in \mathcal{S}', \\ 0 & \text{otherwise.} \end{cases} \tag{40}$$

For $P(\{\theta_{h_k^i}, w_{h_k^i}^i\}_{k=1:l_i}|\theta_{\mathcal{S}'})$, $i \in \mathcal{U}$, we have

$$P(\{\theta_{h_k^i}, w_{h_k^i}^i\}_{k=1:l_i}|\theta_{\mathcal{S}'}) = \begin{cases} P(\{\theta_{\alpha_i}, w_{\alpha_i}^i\}_{\alpha_i\in\mathcal{O}_i'}, \{w_\alpha^i\}_{\alpha\in\mathcal{S}'}|\theta_{\mathcal{S}'}) & \text{if } \theta_{h_k^i} = \tilde{\theta}_\alpha \text{ for } \forall h_k^i = \alpha \in \mathcal{S}', \\ 0 & \text{otherwise.} \end{cases} \tag{41}$$

Using $ii)$ of Lemma 2, (40) and (41), $P(\{\theta_{h_k^1}, w_{h_k^1}^1\}_{k=1:l_1}, \{\theta_{h_k^2}, w_{h_k^2}^2\}_{k=1:l_2}|\theta_{\mathcal{S}'})$ can be further rewritten as

$$P(\{\theta_{h_k^1}, w_{h_k^1}^1\}_{k=1:l_1}, \{\theta_{h_k^2}, w_{h_k^2}^2\}_{k=1:l_2}|\theta_{\mathcal{S}'}) = \prod_{i\in\mathcal{U}} P(\{\theta_{h_k^i}, w_{h_k^i}^i\}_{k=1:l_i}|\theta_{\mathcal{S}'}). \tag{42}$$

Using (42), $\tau_{\tilde{\alpha}}(\pi|\theta_{\Phi_{\tilde{\alpha}}})$ can be written as

$$\begin{aligned} &\left\{ \sum_{\{\theta_{h_l^1}, w_{h_l^1}^1\}_{l=1:l_1}} \sum_{t_1} u_{h_{l_1}^1}^{\pi_1}(w_{h_{l_1}^1}^1, t_1)\kappa(t_1, g_{h_{l_1}^1}^{\pi_1}(\{\theta_{h_k^1}, w_{h_k^1}^1\}_{k=1:l_1-1}))P(\{\theta_{h_k^1}, w_{h_k^1}^1\}_{k=1:l_1}|\theta_{\mathcal{S}'}) \right\} \\ &\times \left\{ \sum_{\{\theta_{h_l^2}, w_{h_l^2}^2\}_{l=1:l_2}} \sum_{t_2} u_{h_{l_2}^2}^{\pi_2}(w_{h_{l_2}^2}^2, t_2)\kappa(t_2, g_{h_{l_2}^2}^{\pi_2}(\{\theta_{h_k^2}, w_{h_k^2}^2\}_{k=1:l_2-1}))P(\{\theta_{h_k^2}, w_{h_k^2}^2\}_{k=1:l_2}|\theta_{\mathcal{S}'}) \right\}. \end{aligned} \tag{43}$$

Since

$$\tau_{\tilde{\alpha}}^i(\pi_i|\theta_{\Phi_{\tilde{\alpha}}}) = \sum_{\{\theta_{h_l^i}, w_{h_l^i}\}_{l=1:l_i}} \sum_{t_i} u_{h_{l_i}^i}^{\pi_i}(w_{h_{l_i}^i}^i, t_i)\kappa(t_i, g_{h_{l_i}^i}^{\pi_i}(\{\theta_{h_k^i}, w_{h_k^i}\}_{k=1:l_i-1}))P(\{\theta_{h_k^i}, w_{h_k^i}^i\}_{k=1:l_i}|\theta_{\mathcal{S}'}),$$

which is the conditional probability that vehicle i succeeds in region $\tilde{\alpha}$ given $\theta_{\mathcal{S}'}$. Therefore, whether a vehicle succeeds in region α is conditionally independent of the other vehicle given $\theta_{\Phi_{\tilde{\alpha}}}$ and the information exists. □

Proof of Proposition 2. We just need to show that if $\alpha \in \Phi_{\alpha_2}$ we must have $\alpha \in \Phi_{\alpha_1}$. To see this, since $\alpha \in \Phi_{\alpha_2}$ we have $\overline{I}_\alpha \le \underline{I}_{\alpha_2}$. According to the proposition's condition, we have $\overline{I}_\alpha \le \underline{I}_{\alpha_2} \le \underline{I}_{\alpha_1}$. Therefore, we have $\alpha \in \Phi_{\alpha_1}$. □

The proof of Proposition 3 requires the following lemma, which considers the policy in the form of (5) and proves a similar result to Lemma 3 but for both the remaining time and the observation variable.

Lemma 4 *Consider a policy* $\pi = (\pi_1, \pi_2)$ *where* $\pi_i = \{x^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}), y^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}), z^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}) : t = 0, 1, 2, ..., T, l = 0, 1, 2, ..., L_i\}$. *Let* $\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l_i}$ *be a realization of the scenario that vehicle* i *will encounter until leaving region* $h^i_{l_i}$. *Under this scenario, for any* $l_i \geq 1$,

$$\begin{aligned} t^{\pi_i}_{h^i_{l_i}} &= \zeta^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l_i-1}), \\ o^i_{h^i_{l_i}} &= \eta^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l_i}), \end{aligned} \tag{45}$$

where $\zeta^{\pi_i}_{h^i_{l_i}}(\cdot)$ *and* $\eta^{\pi_i}_{h^i_{l_i}}(\cdot)$ *are functions that map an integer vector to an integer value.*

Proof. If $l_i < z^i_{h^i_0}(T)$, we have $t^{\pi_i}_{h^i_{l_i}} = \zeta^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l_i-1}) = d(h^i_{l_i}, h^i_{L_i+1})$ and $o^i_{h^i_{l_i}} = \eta^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l_i}) = 0$. Therefore, the lemma holds for $l_i < z^i_{h^i_0}(T)$. If $l_i = z^i_{h^i_0}(T)$, we have $t^{\pi_i}_{h^i_{l_i}} = \zeta^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l_i-1}) = T - d(h^i_0, h^i_{l_i})$. Given $w^i_{h^i_l}$, define

$$\eta^{\pi_i}_{h^i_{l_i}}(\{\theta_{h^i_l}, w_{h^i_l}\}_{l=1:l_i}) = \begin{cases} x^i_{h^i_{l_i}}(T - d(h^i_0, h^i_{l_i}), \mathbf{0}) & \text{if } x^i_\alpha(T - d(h^i_0, h^i_{l_i}), \mathbf{0}) < w^i_{h^i_{l_i}}, \\ -1 & \text{if } x^i_{h^i_l}(T - d(h^i_0, h^i_{l_i}), \mathbf{0}) \geq w^i_{h^i_{l_i}}. \end{cases}$$

Here $\mathbf{0}$ is a zero vector that represents previous observations. Thus, the lemma also holds for $l_i = z^i_{h^i_0}(T)$. We define an indicator function $u^{\pi_i}_{h^i_l}(w^i_{h^i_l}, t, \mathbf{o}^i_{h^i_{l-1}})$ for $l = 1, 2, ..., L_i$ satisfying $u^{\pi_i}_{h^i_l}(w^i_{h^i_l}, t, \mathbf{o}^i_{h^i_{l-1}}) = 1$ if $x^i_{h^i_l}(t, \mathbf{o}^i_{h^i_{l-1}}) \geq w^i_{h^i_l}$ and $y^i_{h^i_l}(t - w^i_{h^i_l}, \mathbf{o}^i_{h^i_{l-1}}) = 1$, which corresponds to the scenario where vehicle i finds the information in region h^i_l and collects it; otherwise, $u^{\pi_i}_{h^i_l}(w^i_{h^i_l}, t, \mathbf{o}^i_{h^i_{l-1}}) = 0$.

For $l_i \geq z^i_{h^i_0}(T)$, we prove the lemma's result using induction. As the induction hypothesis, we assume (45) holds for $l_i \leq k$ where $k \geq z^i_{h^i_0}(T)$. Now we consider $l_i = k + 1$. Let $h^i_{k'}$ be the region that vehicle i travels from. We have two possible cases:

1. If region h^i_{k+1} is skipped under the scenario, we define:
   $\zeta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k}) = d(h^i_{k+1}, h^i_{L_i+1})$ and $\eta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k+1}) = 0$.

2. If the vehicle visits region h^i_{k+1}, let $h^i_{k'}$ be the region that the vehicle comes from, where $k' \leq k$. We have

   $$t^{\pi_i}_{h^i_{k+1}} = t^{\pi_i}_{h^i_{k'}} - \min\left\{x^i_{h^i_{k'}}(t^{\pi_i}_{h^i_{k'}}, \mathbf{o}^i_{h^i_{k'-1}}), w^i_{h^i_{k'}}\right\} - u^{\pi_i}_{h^i_{k'}}(w^i_{h^i_{k'}}, t^{\pi_i}_{h^i_{k'}}, \mathbf{o}^i_{h^i_{k'}}) s_{h^i_{k'}} - d(h^i_{k'}, h^i_{k+1}).$$

   According to the induction hypothesis, we have:
   $t^{\pi_i}_{h^i_{k'}} = \zeta^{\pi^i}_{h^i_{k'}}(\{\theta_{h^i_l}, w_{h^i_l}\}_{l=1:k'-1})$, $\mathbf{o}^i_{h^i_{k'-1}} = \{\eta^{\pi_i}_{h^i_{l'}}(\{\theta_{h^i_l}, w_{h^i_l}\}_{l=1:l'})\}_{l'=1:k'-1}$. We can define

   $$\begin{aligned} &\zeta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k}) = \zeta^{\pi_i}_{h^i_{k'}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k'-1}) \\ &- \min\left\{w_{h^i_{k'}}, x^i_{h^i_{k'}}(\zeta^{\pi^i}_{h^i_{k'}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k'-1}), \{\eta^{\pi_i}_{h^i_{l'}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l'})\}_{l'=1:k'-1})\right\} \\ &- u^{\pi_i}_{h^i_{k'}}(w^i_{h^i_{k'}}, \zeta^{\pi_i}_{h^i_{k'}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k'-1}), \{\eta^{\pi_i}_{h^i_{l'}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l'})\}_{l'=1:k'-1}) s_{h^i_{k'}} - d(h^i_{k'}, h^i_{k+1}). \end{aligned} \tag{46}$$

For $o^i_{h^i_{k+1}}$ we have

$$o^i_{h^i_{k+1}} = \begin{cases} x^i_{h^i_{k+1}}(t^{\pi_i}_{h^i_{k+1}}, \mathbf{o}^i_{h^i_k}) & \text{if } x^i_{h^i_{k+1}}(t^{\pi_i}_{h^i_{k+1}}, \mathbf{o}^i_{h^i_k}) \geq w^i_{h^i_{k+1}}, \\ w_{h^i_{k+1}} & \text{if } x^i_{h^i_{k+1}}(t^{\pi_i}_{h^i_{k+1}}, \mathbf{o}^i_{h^i_k}) < w^i_{h^i_{k+1}}. \end{cases}$$

Then we can define

$$\eta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k+1}) = \begin{cases} x^i_{h^i_{k+1}}(\zeta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k}), \{\eta^{\pi_i}_{h^i_{l'}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l'})\}_{l'=1:k}) \\ \text{if } x^i_{h^i_{k+1}}(\zeta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k}), \{\eta^{\pi_i}_{h^i_{l'}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l'})\}_{l'=1:k}) \geq w^i_{h^i_{k+1}}, \\ w^i_{h^i_{k+1}} \\ \text{if } x^i_{h^i_{k+1}}(\zeta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_{l'}}, w^i_{h^i_{l'}}\}_{l'=1:k}), \{\eta^{\pi_i}_{h^i_{l'}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:l'})\}_{l'=1:k}) < w^i_{h^i_{k+1}}. \end{cases} \tag{47}$$

Combine the two cases, we have $t^{\pi_i}_{h^i_{k+1}} = \zeta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k})$ and $o^i_{h^i_{k+1}} = \eta^{\pi_i}_{h^i_{k+1}}(\{\theta_{h^i_l}, w^i_{h^i_l}\}_{l=1:k+1})$ well defined. It is easy to verify that $\zeta^{\pi_i}_{h^i_{k+1}}(\cdot)$ and $\eta^{\pi_i}_{h^i_{k+1}}(\cdot)$ defined in (46) and (47) only provide integer values since all terms used are integers. Thus, the induction is complete and the lemma holds for $l_i \geq z^i_{h^i_0}(T)$. $\square$

Proof of Proposition 3. Consider a shared region $h^1_{l_1} = h^2_{l_2} = \tilde{\alpha}$. Let l_1 and l_2 be the orders of region $\tilde{\alpha}$ in vehicle 1's and vehicle 2's routes, respectively. Since the shared regions are searched in an exact opposite order by the two vehicles, it is easy to verify that $\Phi_{\tilde{\alpha}} = \emptyset$. Given $\theta_{\tilde{\alpha}} = 1$, the conditional probability that both vehicles succeed in region $\tilde{\alpha}$ is provided by (48) using the indicator function $u^{\pi}_{h^i_l}(w^i_{h^i_l}, t, \mathbf{o}^i_{h^i_{l-1}})$ defined in the proof of Lemma 4.

$$\begin{aligned} &\tau_{\tilde{\alpha}}(\pi) = \\ &\sum_{\{w^1_{h^1_{l_1}}, w^2_{h^2_{l_2}}, t^{\pi_1}_{h^1_{l_1}}, t^{\pi_2}_{h^2_{l_2}}, \mathbf{o}^1_{h^1_{l_1-1}}, \mathbf{o}^2_{h^2_{l_2-1}}\}} u^{\pi_1}_{h^1_{l_1}}(w^1_{h^1_{l_1}}, t^{\pi_1}_{h^1_{l_1}}, \mathbf{o}^1_{h^1_{l_1-1}}) u^{\pi_2}_{h^2_{l_2}}(w^2_{h^2_{l_2}}, t^{\pi_2}_{h^2_{l_2}}, \mathbf{o}^2_{h^2_{l_2-1}}) \\ &\qquad P(w^1_{h^1_{l_1}}, w^2_{h^2_{l_2}}, t^{\pi_1}_{h^1_{l_1}}, t^{\pi_2}_{h^2_{l_2}}, \mathbf{o}^1_{h^1_{l_1-1}}, \mathbf{o}^2_{h^2_{l_2-1}} | \theta_{\tilde{\alpha}}) \\ &= \sum_{\{\theta_{h^1_l}, w^1_{h^1_l}\}_{l=1:l_1-1}} \sum_{\{\theta_{h^2_l}, w^2_{h^2_l}\}_{l=1:l_2-1}} \sum_{\{t^{\pi_1}_{h^1_{l_1}}, t^{\pi_2}_{h^2_{l_2}}, \mathbf{o}^1_{h^1_{l_1-1}}, \mathbf{o}^2_{h^2_{l_2-1}}\}} [u^{\pi_1}_{h^1_{l_1}}(w^1_{h^1_{l_1}}, t^{\pi_1}_{h^1_{l_1}}, \mathbf{o}^1_{h^1_{l_1-1}}) u^{\pi_2}_{h^2_{l_2}}(w^2_{h^2_{l_2}}, t^{\pi_2}_{h^2_{l_2}}, \mathbf{o}^2_{h^2_{l_2-1}}) \\ &\qquad \times P(\{\theta_{h^1_l}, w^1_{h^1_l}\}_{l=1:l_1-1}, \{\theta_{h^2_l}, w^2_{h^2_l}\}_{l=1:l_2-1}, w^1_{h^1_{l_1}}, w^2_{h^2_{l_2}}, t^{\pi_1}_{h^1_{l_1}}, t^{\pi_2}_{h^2_{l_2}}, \mathbf{o}^1_{h^1_{l_1-1}}, \mathbf{o}^2_{h^2_{l_2-1}} | \theta_{\tilde{\alpha}})]. \end{aligned} \tag{48}$$

Given $\{\theta_{h^1_l}, w^1_{h^1_l}\}_{l=1:l_1-1}$ and $\{\theta_{h^2_l}, w^2_{h^2_l}\}_{l=1:l_2-1}$, using the indicator function $\kappa(x, y)$ defined in the proof

of Theorem 3, we have

$$\begin{aligned}
&\sum_{\{t^{\pi_1}_{h^1_{l_1}},t^{\pi_2}_{h^2_{l_2}},\mathbf{o}^1_{h^1_{l_1-1}},\mathbf{o}^2_{h^2_{l_2-1}}\}} u^{\pi_1}_{h^1_{l_1}}(w^1_{h^1_{l_1}},t^{\pi_1}_{h^1_{l_1}},\mathbf{o}^1_{h^1_{l_1-1}})u^{\pi_2}_{h^2_{l_2}}(w^2_{h^2_{l_2}},t^{\pi_2}_{h^2_{l_2}},\mathbf{o}^2_{h^2_{l_2-1}}) \\
&\quad\times P(\{\theta_{h^1_l},w^1_{h^1_l}\}_{l=1:l_1-1},\{\theta_{h^2_l},w^2_{h^2_l}\}_{l=1:l_2-1},w^1_{h^1_{l_1}},w^2_{h^2_{l_2}},t^{\pi_1}_{h^1_{l_1}},t^{\pi_2}_{h^2_{l_2}},\mathbf{o}^1_{h^1_{l_1-1}},\mathbf{o}^2_{h^2_{l_2-1}}|\theta_{\tilde{\alpha}}) \\
&=\sum_{\{t_1,t_2,\hat{\mathbf{o}}^1_{l_1-1},\hat{\mathbf{o}}^2_{l_2-1}\}}\Bigg\{u^{\pi_1}_{h^1_{l_1}}(w^1_{h^1_{l_1}},\zeta^{\pi_1}_{h^1_{l_1}}(\cdot),\hat{\mathbf{o}}^1_{l_1-1})u^{\pi_2}_{h^2_{l_2}}(w^2_{h^2_{l_2}},\zeta^{\pi_2}_{h^2_{l_2}}(\cdot),\hat{\mathbf{o}}^2_{l_2-1})\kappa(\zeta^{\pi_1}_{h^1_{l_1}}(\cdot),t_1)\kappa(\zeta^{\pi_2}_{h^2_{l_2}}(\cdot),t_2) \\
&\times\left[\prod_{l=1}^{l_1-1}\kappa(\eta^{\pi_1}_{h^1_l}(\cdot),\hat{o}^1_l)\right]\left[\prod_{l=1}^{l_2-1}\kappa(\eta^{\pi_2}_{h^2_l}(\cdot),\hat{o}^2_l)\right]P(w^1_{h^1_{l_1}},w^2_{h^2_{l_2}},\{\theta_{h^1_l},w^1_{h^1_l}\}_{l=1:l_1-1},\{\theta_{h^2_l},w^2_{h^2_l}\}_{l=1:l_2-1}|\theta_{\tilde{\alpha}})\Bigg\}.
\end{aligned}\tag{49}$$

In (49), we use $\eta^{\pi_i}_{h^i_l}(\cdot)$ and $\zeta^{\pi_i}_{h^i_{l_l}}(\cdot)$ as concise forms of their counterparts defined in Lemma 4. $\hat{\mathbf{o}}^i_{l_i-1}$ is an integer vector that has the same dimension as $\mathbf{o}^i_{l_i-1}$, $i\in\mathcal{U}$. Since all shared regions are searched in an exact opposite order, we have

$$\{h^1_1,...,h^1_{l_1-1}\}\bigcap\{h^2_1,...,h^2_{l_2-1}\}=\emptyset.$$

Using *ii*) of Lemma 2, we can further rewrite (49) as

$$\prod_{i\in\mathcal{U}}\left\{\sum_{t_i,\hat{\mathbf{o}}^i_{l_i-1}}u^{\pi_i}_{h^i_{l_i}}(w^i_{h^i_{l_i}},\zeta^{\pi_i}_{h^i_{l_i}}(\cdot),\hat{\mathbf{o}}^i_{l_i-1})\kappa(\zeta^{\pi_i}_{h^i_{l_i}}(\cdot),t_i)\left[\prod_{l=1}^{l_i-1}\kappa(\eta^{\pi_i}_{h^i_l}(\cdot),\hat{o}^i_l)\right]P(w^i_{h^i_{l_i}},\{\theta_{h^i_l},w^i_{h^i_l}\}_{l=1:l_i-1}|\theta_{\tilde{\alpha}})\right\}.\tag{50}$$

Combine (48), (49) and (50), $\tau_{\tilde{\alpha}}(\pi)$ can be written as

$$\begin{aligned}
\prod_{i\in\mathcal{U}}\Bigg\{&\sum_{\{\theta_{h^i_l},w^i_{h^i_l}\}_{l=1:l_i}}\sum_{t_i,\hat{\mathbf{o}}^i_{l_i-1}}u^{\pi_i}_{h^i_{l_i}}(w^i_{h^i_{l_i}},\zeta^{\pi_i}_{h^i_{l_i}}(\cdot),\hat{\mathbf{o}}^i_{l_i-1})\kappa(\zeta^{\pi_i}_{h^i_{l_i}}(\cdot),t_i)\left[\prod_{l=1}^{l_i-1}\kappa(\eta^{\pi_i}_{h^i_l}(\cdot),\hat{o}^i_l)\right] \\
&P(w^i_{h^i_{l_i}},\{\theta_{h^i_l},w^i_{h^i_l}\}_{l=1:l_i-1}|\theta_{\tilde{\alpha}})\Bigg\},
\end{aligned}\tag{51}$$

which is product of the conditional probability that each vehicle will succeed in region $\tilde{\alpha}$ given $\theta_{\tilde{\alpha}}$. Therefore, whether a vehicle succeeds in the region is conditionally independent of the other vehicle given $\theta_{\tilde{\alpha}}$. □

Proof of Theorem 4. A Markovian policy can be viewed as a special case of the policy in the form of (5). To see this, we can assign the same decision to vehicle $i\in\mathcal{U}$ in a policy in the form of (5) as long as the vehicle has the same remaining time at the same decision epoch no matter what history observations the vehicle received. Then a Markovian policy is established. Let $\pi^*=(\pi^*_1,\pi^*_2)$ be an optimal time-allocation policy in the form of (5) and $R(\pi^*)$ be the expected reward provided by π^*. Since whether a vehicle succeeds in a shared region is independent of the other vehicle, we have $\widehat{R}(\pi)=R(\pi)$. Using Theorem 1, let $\hat{\pi}_1$ be a Markovian policy that solves $\max\{\widehat{R}(\pi_1,\pi^*_2):\pi_1\in\widehat{\Pi}_1\}$ and let $\hat{\pi}_2$ be a

Markovian policy that solves $\max\{\widehat{R}(\hat{\pi}_1, \pi_2) : \pi_2 \in \widehat{\Pi}_2\}$. We should have $\widehat{R}(\hat{\pi}_1, \hat{\pi}_2) \geq \widehat{R}(\pi^*) = R(\pi^*)$. Since policy $(\hat{\pi}_1, \hat{\pi}_2)$ can be viewed as a special case of the policy in the form of (5), according to the theorem's condition, we should have $R(\hat{\pi}_1, \hat{\pi}_2) = \widehat{R}(\hat{\pi}_1, \hat{\pi}_2)$. Therefore, we have $R(\hat{\pi}_1, \hat{\pi}_2) \geq R(\pi^*)$. Since π^* is an optimal policy, $(\hat{\pi}_1, \hat{\pi}_2)$ must also be an optimal policy. □

We establish three lemmas to facilitate the proof of Theorem 5. In Lemma 5, we duplicate each shared region α satisfying $\gamma_\alpha < 1$ and assign one to each vehicle. Then we relate the corresponding expected rewards when the same policy is applied. We use $\alpha \in \mathcal{S} : \gamma_\alpha \leq 1$ to represent "for all for $\alpha \in \mathcal{S}$ satisfying $\gamma_\alpha \leq 1$"; similarly, we also have $\alpha \in \mathcal{S} : \gamma_\alpha > 1$ to represent "for all for $\alpha \in \mathcal{S}$ satisfying $\gamma_\alpha > 1$".

Lemma 5 *For any policy $\pi = (\pi_1, \pi_2)$, let $R^O(\pi)$ be the expected reward the fleet will collect by following policy π assuming that for $\forall \alpha \in S$ satisfying $\gamma_\alpha \leq 1$, g_α will be collected by each vehicle if it succeeds in the region. We have*

$$R^O(\pi) \geq R(\pi). \tag{52}$$

Proof. The expected reward collected by the fleet under policy π is

$$\begin{aligned} R(\pi) = &\sum_{\alpha \in \mathcal{S}:\gamma_\alpha \leq 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi|0)(1-\tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi|0)(1-\tau_\alpha^1(\pi_1)) + (1+\gamma_\alpha)\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2)] + R^{\pi_1} \\ &+ R^{\pi_2} + \sum_{\alpha \in S:\gamma_\alpha > 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi|0)(1-\tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi|0)(1-\tau_\alpha^1(\pi_1)) + (1+\gamma_\alpha)\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2)]. \end{aligned} \tag{53}$$

Due to the assumption that the same amount of reward will be collected regardless of the other vehicle's result, an expected reward of $e_\alpha g_\alpha \tau_\alpha^i(\pi_i)$ will be collected by vehicle i in any shared region α satisfying $\gamma_\alpha < 1$. Therefore, we have

$$\begin{aligned} R^O(\pi) = &R^{\pi_1} + R^{\pi_2} + \sum_{\alpha \in \mathcal{S}:\gamma_\alpha \leq 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi_1) + \tau_\alpha^2(\pi_2)] \\ &+ \sum_{\alpha \in \mathcal{S}:\gamma_\alpha > 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi|0)(1-\tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi|0)(1-\tau_\alpha^1(\pi_1)) + (1+\gamma_\alpha)\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2)]. \end{aligned} \tag{54}$$

Combine (53) and (54), we have

$$\begin{aligned} R^O(\pi) - R(\pi) = \sum_{\alpha \in \mathcal{S}} g_\alpha e_\alpha [&\tau_\alpha^1(\pi_1) + \tau_\alpha^2(\pi_2) - \tau_\alpha^1(\pi|0)(1-\tau_\alpha^2(\pi_2)) \\ &- \tau_\alpha^2(\pi|0)(1-\tau_\alpha^1(\pi_1)) - (1+\gamma_\alpha)\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2)]. \end{aligned}$$

Since

$$\begin{aligned} &\tau_\alpha^1(\pi|0)(1-\tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi|0)(1-\tau_\alpha^1(\pi_1)) + (1+\gamma_\alpha)\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2) \\ \leq &\tau_\alpha^1(\pi|0)(1-\tau_\alpha^2(\pi_2)) + \tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2) + \tau_\alpha^2(\pi|0)(1-\tau_\alpha^1(\pi_1)) + \tau_\alpha^2(\pi|1)\tau_\alpha^1(\pi_1) \\ = &\tau_\alpha^1(\pi_1) + \tau_\alpha^2(\pi_2), \end{aligned} \tag{55}$$

we have $R^O(\pi) \geq R(\pi)$. Note that the inequality in (55) comes from the fact that $\gamma_\alpha < 1$ and $\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2) = \tau_\alpha^2(\pi|1)\tau_\alpha^1(\pi_1) = \tau_\alpha(\pi) \geq 0$. □

In Lemma 6, we compensate each vehicle for its reward loss assuming that it is always the second vehicle to collect the information if both vehicles succeed in a shared region and extend inequality (52) to (56). The compensation is calculated using a fixed $\tilde{\tau}$, which provides each vehicle the conditional probability that the other vehicle succeeds in each shared region α given that information exists in the region. Let $\tilde{\tau} = \{\tilde{\tau}_\alpha^1, \tilde{\tau}_\alpha^2\}_{\alpha \in \mathcal{S}: \gamma_\alpha \leq 1}$ where $0 \leq \tilde{\tau}_\alpha^1, \tilde{\tau}_\alpha^2 \leq 1$.

Lemma 6 *For any policy $\pi = (\pi_1, \pi_2)$, let $R^+(\pi|\tilde{\tau})$ be the expected reward that the fleet will receive by applying policy π assuming that each vehicle $i \in \mathcal{U}$ will collect an expected reward of $g_\alpha e_\alpha \tau_\alpha^i(\pi_i)[(1 - \tilde{\tau}_\alpha^i) + \tilde{\tau}_\alpha^i \gamma_\alpha]$ from region $\alpha \in S$ satisfying $\gamma_\alpha \leq 1$. We have*

$$R^O(\pi) \leq R^+(\pi|\tau) + \sum_{i \in U} \sum_{\alpha \in S: \gamma_\alpha \leq 1} e_\alpha g_\alpha (1 - \gamma_\alpha) \tilde{\tau}_\alpha^i. \tag{56}$$

Proof. To compare $R^O(\pi_i)$ and $R^+(\pi_i|\tau)$, we only need to compare the the expected total reward collected from each shared region α satisfying $\gamma_\alpha \leq 1$ since the expected total rewards collected from all the other regions are the same. We use a big M to represent the expected total reward collected from all the other regions and have

$$R^O(\pi) = M + \sum_{\alpha \in S: \gamma_\alpha \leq 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi_1) + \tau_\alpha^2(\pi_2)].$$

$$\begin{aligned}
&R^+(\pi|\tau^i) + \sum_{i \in U} \sum_{\alpha \in S: \gamma_\alpha \leq 1} e_\alpha g_\alpha (1 - \gamma_\alpha) \tilde{\tau}_\alpha^i = M + \\
&\sum_{\alpha \in S: \gamma_\alpha \leq 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi_1)(1 - \tilde{\tau}_\alpha^1) + \tau_\alpha^2(\pi_2)(1 - \tilde{\tau}_\alpha^2) + \gamma_\alpha(\tau_\alpha^1(\pi_1)\tilde{\tau}_\alpha^1 + \tau_\alpha^2(\pi_2)\tilde{\tau}_\alpha^2) + (\tilde{\tau}_\alpha^1 + \tilde{\tau}_\alpha^2)(1 - \gamma_\alpha)] \\
&\geq M + \sum_{\alpha \in S: \gamma_\alpha \leq 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi_1)(1 - \tilde{\tau}_\alpha^1) + \tau_\alpha^2(\pi_2)(1 - \tilde{\tau}_\alpha^2) + \tau_\alpha^1(\pi_1)\tilde{\tau}_\alpha^1 + \tau_\alpha^2(\pi_2)\tilde{\tau}_\alpha^2] \\
&= M + \sum_{\alpha \in S: \gamma_\alpha \leq 1} g_\alpha e_\alpha [\tau_\alpha^1(\pi_1) + \tau_\alpha^2(\pi_2)] = R^O(\pi).
\end{aligned} \tag{57}$$

The inequality in (57) holds since $\tilde{\tau}_\alpha^1 + \tilde{\tau}_\alpha^2 \geq \tau_\alpha^1(\pi_1)\tilde{\tau}_\alpha^1 + \tau_\alpha^2(\pi_2)\tilde{\tau}_\alpha^2$. □

Lemma 7 provides a ratio under which we can extend the inequalities derived in Lemma 5 and Lemma 7 in the proof of Theorem 5. Let $\tilde{\tau}_\alpha^1 = \tau_\alpha^2(\pi_2), \tilde{\tau}_\alpha^2 = \tau_\alpha^1(\pi_1)$ for $\alpha \in \mathcal{S}$ and $\tilde{\tau} = \{\tilde{\tau}_\alpha^1, \tilde{\tau}_\alpha^2\}_{\alpha \in \mathcal{S}}$. Define $\widetilde{R}^+(\pi_i|\tilde{\tau}) = R^{\pi_i} + \sum_{\alpha \in S} e_\alpha g_\alpha [\tau_\alpha^i(\pi_i)(1 - \tilde{\tau}_\alpha^i) + \gamma_\alpha \tau_\alpha^i(\pi_i)\tilde{\tau}_\alpha^i]$, which is the expected total reward that vehicle i will collect if it collects a reward of $[\gamma_\alpha \tilde{\tau}_\alpha^i + (1 - \tilde{\tau}_\alpha^1)]$ when it succeeds in any shared region $\alpha \in \mathcal{S}$.

Lemma 7 *For any Markovian policy $\pi = (\pi_1, \pi_2)$, we have*

$$(2 - \underline{\gamma})\widehat{R}(\pi) \geq \frac{1 + \overline{\gamma}}{2\overline{\gamma}} \sum_{i \in U} \widetilde{R}^+(\pi_i|\tilde{\tau}) + \sum_{i \in U} \sum_{\alpha \in S: \gamma_\alpha \leq 1} e_\alpha g_\alpha (1 - \gamma_\alpha) \tilde{\tau}_\alpha^i. \tag{58}$$

Proof. We first write $\widehat{R}(\pi)$ as

$$\begin{aligned}
\widehat{R}(\pi) = &\sum_{i \in \mathcal{U}} R^{\pi_i} + \sum_{\alpha \in S: \gamma_\alpha \leq 1} e_\alpha g_\alpha [\tau_\alpha^1(\pi_1)(1 - \tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi_2)(1 - \tau_\alpha^1(\pi_1)) + (1 + \gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)] \\
&+ \sum_{\alpha \in S: \gamma_\alpha > 1} e_\alpha g_\alpha [\tau_\alpha^1(\pi_1)(1 - \tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi_2)(1 - \tau_\alpha^1(\pi_1)) + (1 + \gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)].
\end{aligned} \tag{59}$$

For any $\alpha \in S$ where $\gamma_\alpha \leq 1$, we have

$$\begin{aligned}
&(2-\underline{\gamma})[\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))+(1+\gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)]\\
&\geq (2-\gamma_\alpha)[\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))+(1+\gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)]\\
&=\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\gamma_\alpha\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)+(1-\gamma_\alpha)\tau_\alpha^2(\pi_2)\\
&+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))+\gamma_\alpha\tau_\alpha^2(\pi_2)\tau_\alpha^1(\pi_1)+(1-\gamma_\alpha)\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))\\
&+(1-\gamma_\alpha)(1+\gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)\\
&\geq \tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\gamma_\alpha\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)+(1-\gamma_\alpha)\tau_\alpha^2(\pi_2)\\
&+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))+\gamma_\alpha\tau_\alpha^2(\pi_2)\tau_\alpha^1(\pi_1)+(1-\gamma_\alpha)\tau_\alpha^1(\pi_1)\\
&=\sum_{i\in\mathcal{U}}[\tau_\alpha^i(\pi_i)(1-\tilde{\tau}_\alpha^i)+\gamma_\alpha\tau_\alpha^i(\pi_i)\tilde{\tau}_\alpha^i]+\sum_{i\in\mathcal{U}}(1-\gamma_\alpha)\tilde{\tau}_\alpha^i.
\end{aligned} \tag{60}$$

Using (60), we have

$$\begin{aligned}
&(2-\underline{\gamma})\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha[\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))+(1+\gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)]\\
&\geq \sum_{i\in\mathcal{U}}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha[\tau_\alpha^i(\pi_i)(1-\tilde{\tau}_\alpha^i)+\gamma_\alpha\tau_\alpha^i(\pi_i)\tilde{\tau}_\alpha^i]+\sum_{i\in U}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha(1-\gamma_\alpha)\tilde{\tau}_\alpha^i\\
&\geq \frac{1+\overline{\gamma}}{2\overline{\gamma}}\sum_{i\in\mathcal{U}}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha[\tau_\alpha^i(\pi_i)(1-\tilde{\tau}_\alpha^i)+\gamma_\alpha\tau_\alpha^i(\pi_i)\tilde{\tau}_\alpha^i]+\sum_{i\in U}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha(1-\gamma_\alpha)\tilde{\tau}_\alpha^i.
\end{aligned} \tag{61}$$

For $\alpha \in S : \gamma_\alpha > 1$,

$$\begin{aligned}
&\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))+(1+\gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)\\
=&\frac{1+\overline{\gamma}}{2\overline{\gamma}}\{\frac{2\overline{\gamma}}{1+\overline{\gamma}}[\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))]+\frac{2\overline{\gamma}(1+\gamma_\alpha)}{1+\overline{\gamma}}\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)\}\\
\geq&\frac{1+\overline{\gamma}}{2\overline{\gamma}}[\tilde{\tau}_\alpha^1(1-\tilde{\tau}_\alpha^2)+\tilde{\tau}_\alpha^2(1-\tilde{\tau}_\alpha^1)+2\gamma_\alpha\tilde{\tau}_\alpha^1\tilde{\tau}_\alpha^2]=\frac{1+\overline{\gamma}}{2\overline{\gamma}}\sum_{i\in\mathcal{U}}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha[\tau_\alpha^i(\pi_i)(1-\tilde{\tau}_\alpha^i)+\gamma_\alpha\tau_\alpha^i(\pi_i)\tilde{\tau}_\alpha^i].
\end{aligned}$$

This implies that

$$\begin{aligned}
&(2-\underline{\gamma})\sum_{\alpha\in S:\gamma_\alpha>1} e_\alpha g_\alpha[\tau_\alpha^1(\pi_1)(1-\tau_\alpha^2(\pi_2))+\tau_\alpha^2(\pi_2)(1-\tau_\alpha^1(\pi_1))+(1+\gamma_\alpha)\tau_\alpha^1(\pi_1)\tau_\alpha^2(\pi_2)]\\
&\geq \frac{1+\overline{\gamma}}{2\overline{\gamma}}\sum_{i\in\mathcal{U}}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha[\tau_\alpha^i(\pi_i)(1-\tilde{\tau}_\alpha^i)+\gamma_\alpha\tau_\alpha^i(\pi_i)\tilde{\tau}_\alpha^i].
\end{aligned} \tag{62}$$

Since $\frac{1+\overline{\gamma}}{2\overline{\gamma}}\sum_{i\in\mathcal{U}} R^{\pi_i} \leq \sum_{i\in\mathcal{U}} R^{\pi_i}$, combining (59), (61) and (62), we have (58) hold. □

With the assistance of the three lemmas, we can complete the proof of Theorem 5.

Proof of Theorem 5. Let $\pi^{*,L}$ be an arbitrary local optimum of problem (IAP) and π^* be an optimal policy in the form of (5). Set $\tilde{\tau}_\alpha^i = \tau_\alpha^i(\pi_i^{*,L})$ for $i \in \mathcal{U}, \alpha \in \mathcal{S}$ and we first show

$$R^+(\pi^*|\tilde{\tau}) \leq \frac{1+\overline{\gamma}}{2}\sum_{i\in U}\widetilde{R}^+(\pi_i^{*,L}|\tilde{\tau}). \tag{63}$$

According to Definition 3, $\pi_i^{L,*}$ optimizes $\widetilde{R}^+(\pi_i^{L,*}|\tilde{\tau})$ in $\widehat{\Pi}_i$. We have

$$\widetilde{R}^+(\pi_i^{L,*}|\tilde{\tau}) \geq \widetilde{R}^+(\pi_i^*|\tilde{\tau}). \tag{64}$$

Now we consider $R^+(\pi^*|\tilde{\tau})$.

$$\begin{aligned} R^+(\pi^*|\tilde{\tau}) &= \overbrace{\sum_{i\in U}[R^{\pi_i^*} + \sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha(\tau_\alpha^i(\pi_i^*)(1-\tilde{\tau}_\alpha^i) + \gamma_\alpha\tau_\alpha^i(\pi_i^*)\tilde{\tau}_\alpha^i)]}^{=M^*} \\ &+ \sum_{\alpha\in S:\gamma_\alpha>1} e_\alpha g_\alpha[\tau_\alpha^1(\pi_1^*|0)(1-\tau_\alpha^2(\pi_2^*)) + \tau_\alpha^2(\pi_2^*|0)(1-\tau_\alpha^1(\pi_1^*)) \\ &+ \frac{1+\gamma_\alpha}{2}(\tau_\alpha^1(\pi_1^*|1)\tau_\alpha^2(\pi_2^*) + \tau_\alpha^2(\pi_2^*|1)\tau_\alpha^1(\pi_1^*))] \leq M^* + \frac{1+\overline{\gamma}}{2}\sum_{\alpha\in S:\gamma_\alpha>1} e_\alpha g_\alpha(\tau_\alpha^1(\pi_1^*) + \tau_\alpha^2(\pi_2^*)) \\ &\leq \frac{1+\overline{\gamma}}{2}[M^* + \sum_{\alpha\in S:\gamma_\alpha>1} e_\alpha g_\alpha(\tau_\alpha^1(\pi_1^*) + \tau_\alpha^2(\pi_2^*))]. \end{aligned} \tag{65}$$

Note that the first inequality in (65) holds since $\tau_\alpha^1(\pi_1^*) = \tau_\alpha^1(\pi_1^*|0)(1-\tau_\alpha^2(\pi_2^*)) + \tau_\alpha^1(\pi_1^*|1)\tau_\alpha^2(\pi_2^*)$. On the other hand, we have

$$\begin{aligned} \sum_{i\in U}\widetilde{R}^+(\pi_i^*|\pi^{L,*}) &= M^* + \sum_{\alpha\in S:\gamma_\alpha>1} e_\alpha g_\alpha[\tau_\alpha^1(\pi_1^*)(1-\tilde{\tau}_\alpha^1) + \gamma_\alpha\tau_\alpha^1(\pi_1^*)\tilde{\tau}_\alpha^1 \\ &+ \tau_\alpha^2(\pi_2^*)(1-\tilde{\tau}_\alpha^2) + \gamma_\alpha\tau_\alpha^2(\pi_2^*)\tilde{\tau}_\alpha^2] \geq M^* + \sum_{\alpha\in S:\gamma_\alpha>1} e_\alpha g_\alpha(\tau_\alpha^1(\pi_1^*) + \tau_\alpha^2(\pi_2^*)). \end{aligned} \tag{66}$$

Combining (64), (65) and (66), we have (63) hold. Using Lemma 5 and Lemma 6, we have

$$R^+(\pi^*|\tilde{\tau}) + \sum_{i\in U}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha(1-\gamma_\alpha)\tilde{\tau}_\alpha^i \geq R^O(\pi^*) \geq R(\pi^*). \tag{67}$$

Finally, combining (58), (63) and (67), we have

$$\begin{aligned} R(\pi^*) &\leq R^+(\pi^*|\tilde{\tau}) + \sum_{i\in U}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha(1-\gamma_\alpha)\tilde{\tau}_\alpha^i \\ &\leq [\frac{1+\overline{\gamma}}{2}\sum_{i\in U}\widetilde{R}^+(\pi_i^*|\pi^{L,*}) + \sum_{i\in U}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha(1-\gamma_\alpha)\tilde{\tau}_\alpha^i] \\ &\leq \overline{\gamma}[\frac{1+\overline{\gamma}}{2\overline{\gamma}}\sum_{i\in U}\widetilde{R}^+(\pi_i^*|\pi^{L,*}) + \sum_{i\in U}\sum_{\alpha\in S:\gamma_\alpha\leq 1} e_\alpha g_\alpha(1-\gamma_\alpha)\tilde{\tau}_\alpha^i] \\ &\leq \overline{\gamma}(2-\underline{\gamma})\widehat{R}(\pi^{L,*}). \end{aligned}$$

□

D. Tightness of the upper bound

This section illustrates how the ratios highlighted in Remark 2 can be approached. We use ϵ to represent a very small positive number.

Corollary 1 *The ratio in (21) can be approached arbitrarily close when* $\underline{\gamma} = 0$ *or* $\underline{\gamma} = 1$.

Before proving the corollary, we first study a special case of the time-allocation problem.

Lemma 8 *If* $\gamma_\alpha = 1$, $\forall \alpha \in \mathcal{S}$, *for any policy* $\tilde{\pi} = (\tilde{\pi}_1, \tilde{\pi}_2)$ *in the form of (5),* $(\hat{\pi}_1, \hat{\pi}_2)$ *is an optimal policy if*

$$\hat{\pi}_1 \in \arg\max\{\widehat{R}(\pi_1, \tilde{\pi}_2) : \pi_1 \in \widehat{\Pi}_1\}, \tag{68a}$$

$$\hat{\pi}_2 \in \arg\max\{\widehat{R}(\tilde{\pi}_1, \pi_2) : \pi_2 \in \widehat{\Pi}_2\}. \tag{68b}$$

Proof. For $\alpha \in \mathcal{S}$ and any policy $\pi = (\pi_1, \pi_2)$, we have

$$\begin{aligned} R_\alpha(\pi_1, \pi_2) =& g_\alpha e_\alpha[\tau_\alpha^1(\pi|0)(1 - \tau_\alpha^2(\pi_2)) + \tau_\alpha^2(\pi|0)(1 - \tau_\alpha^1(\pi_2)) + (1 + \gamma_\alpha)\tau_\alpha^1(\pi|1)\tau_\alpha^2(\pi_2) \\ =& g_\alpha e_\alpha[\tau_\alpha^1(\pi_1) + \tau_\alpha^2(\pi_2)] = \widehat{R}_\alpha(\pi_1, \pi_2). \end{aligned} \tag{69}$$

Let (π_1^*, π_2^*) be an optimal policy. Combining (68a) and (69), we have

$$\begin{aligned} R(\hat{\pi}_1, \tilde{\pi}_2) &= R^{\hat{\pi}_1} + R^{\tilde{\pi}_2} + \sum_{\alpha \in \mathcal{S}} g_\alpha e_\alpha[\tau_\alpha^1(\hat{\pi}_1) + \tau_\alpha^2(\tilde{\pi}_2)] \geq R(\pi_1^*, \tilde{\pi}_2) \\ &= R^{\pi_1^*} + R^{\tilde{\pi}_2} + \sum_{\alpha \in \mathcal{S}} g_\alpha e_\alpha[\tau_\alpha^1(\pi_1^*) + \tau_\alpha^2(\tilde{\pi}_2)]. \end{aligned} \tag{70}$$

From (70), we can obtain

$$\begin{aligned} R(\hat{\pi}_1, \pi_2^*) &= R^{\hat{\pi}_1} + R^{\pi_2^*} + \sum_{\alpha \in \mathcal{S}} g_\alpha e_\alpha[\tau_\alpha^1(\hat{\pi}_1) + \tau_\alpha^2(\pi_2^*)] \\ &\geq R^{\pi_1^*} + R^{\pi_2^*} + \sum_{\alpha \in \mathcal{S}} g_\alpha e_\alpha[\tau_\alpha^1(\pi_1^*) + \tau_\alpha^2(\pi_2^*)] = R(\pi_1^*, \pi_2^*). \end{aligned} \tag{71}$$

Therefore, $(\hat{\pi}_1, \pi_2^*)$ is also an optimal policy. Using a similar method to replace π_2^* with $\hat{\pi}_2$, we can easily show that $(\hat{\pi}_1, \hat{\pi}_2)$ is also an optimal policy. □

Lemma 8 shows that a single iteration of Algorithm 1 finds an optimal time-allocation policy when the cooperation factor is equal to one for all shared regions.

Proof of Corollary 1. Given $\overline{\gamma} = 1$, if $\underline{\gamma} = 1$, then $\gamma_\alpha = 1$ for all $\alpha \in \mathcal{S}$. According to Lemma 8, a local optimum of problem (IAP) is also an optimal time-allocation policy. Therefore, $\frac{R(\pi^{L,*})}{R(\pi^*)} = 1$.

We use an example to prove the rest of Corollary 1. Consider the example illustrated in Figure 6 where two vehicles are assigned to search three regions. Since we have only one shared region in the example, $R(\pi) = \widehat{R}(\pi)$ for any Markovian policy π according to Proposition 3. Combining Proposition 3 and Theorem 4, the optimal Markovian policy is also an optimal policy to the time-allocation problem. Note that we also have the same result for the example that we will create to prove Corollary 2.

We have the following parameter setups: For regions 1,2,3, we set $e_1 = 0.5, e_2 = 0.5, e_3 = 0.5$; $g_1 = 1-\epsilon$, $g_2 = \epsilon$, $g_3 = 1+\epsilon$; $p_1(1) = 1-\epsilon$, $p_2(1) = 1-\epsilon^2$, $p_3(1) = 1-\epsilon$; $\gamma_3 = \underline{\gamma} = 0$; $s_1 = s_2 = s_3 = 0$. Note that under the last condition, it is always optimal for a vehicle to collect the information since it takes

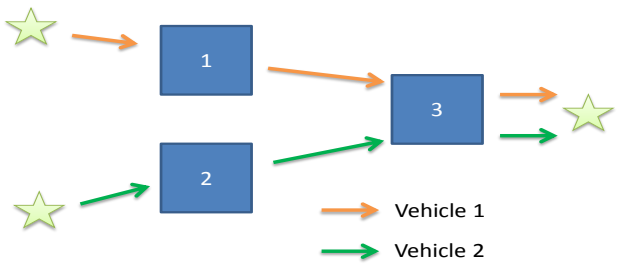

Figure 6: The example to show that $\frac{\widehat{R}(\pi^{L,*})}{R(\pi^*)}$ can approach 1/2 arbitrarily close when $\underline{\gamma} = 0$

zero amount of time and provides non-negative reward to the fleet. In addition, each vehicle's route is a straight line, and therefore there is no benefit to skip a region for both vehicles. Since the travel time does not influence the solution, we assume zero travel between each pair of regions to simplify the representation of the policy. Each vehicle is given one unit of mission time and there are two feasible policies for each vehicle under this scenario: $\pi_1^a = \{x_1^1(1) = 0, x_3^1(1) = 1\}, \pi_1^b = \{x_1^1(1) = 1, x_3^1(1) = 0\}$ and $\pi_2^a = \{x_2^2(1) = 0, x_3^2(1) = 1\}, \pi_2^b = \{x_2^2(1) = 1, x_3^2(1) = 0\}$ for vehicles 1 and 2, respectively. We first show that (π_1^a, π_2^b) is a local optimum of problem (IAP).

First, we have $R(\pi_1^a, \pi_2^b) > R(\pi_1^b, \pi_2^b)$. Since for vehicle 1, region 3 provides a larger reward than region 1 if the vehicle succeeds while the vehicle has the same probability to succeed in both regions. We also have $R(\pi_1^a, \pi_2^a) = e_3 g_3 [1 - (1 - p_3(1))^2] = 0.5(1 - \epsilon^2)$. Since $R(\pi_1^a, \pi_2^b) = e_3 g_3 p_3(1) + e_1 g_1 p_1(1) = 0.5(1 - \epsilon) + 0.5\epsilon(1 - \epsilon^2) = 0.5(1 - \epsilon^3) > 0.5(1 - \epsilon^2) = R(\pi_1^a, \pi_2^a)$, (π_1^a, π_2^b) is a local optimum.

However, it is easy to verify that the optimal policy should be (π_1^b, π_2^a), where $R(\pi_1^b, \pi_2^a) = e_3 g_3 p_3(1) + e_2 g_2 p_2(1) = 0.5(1 - \epsilon)^2 + 0.5(1 - \epsilon)$. Since

$$\lim_{\epsilon \to 0} \frac{0.5(1 - \epsilon^3)}{0.5(1 - \epsilon)^2 + 0.5(1 - \epsilon)} = \frac{1}{2} = \frac{1}{2 - \underline{\gamma}},$$

the ratio provided in (21) can be approached arbitrarily close when $\underline{\gamma} = 0$. □

Corollary 2 *For any $\overline{\gamma} \geq 1$, (22) can be approached arbitrarily close.*

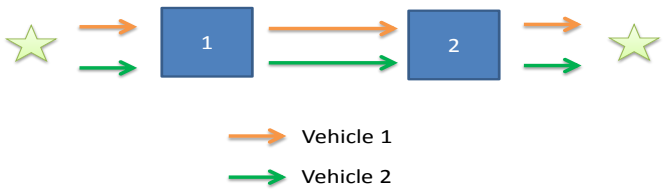

Figure 7: The example to show that $\frac{\widehat{R}(\pi^{L,*})}{R(\pi^*)}$ can approach $1/\bar{\gamma}$ arbitrarily close

Proof. Consider the example in Figure 6, where two vehicles share two regions on the straight line that connects the start depot and the end depot. For regions 1 and 2, we have $e_1 = 0.5, e_2 = 0.5$; $g_1 = \gamma(1-\epsilon)$, $g_2 = 1$; $p_1(1) = 1-\epsilon$, $p_2(1) = 1-\epsilon$; $\gamma_1 = \gamma_2 = \bar{\gamma} \geq 1$; $s_1 = s_2 = 0$.

Similar to the former example, each vehicle has only one unit of mission time and the vehicles always collect the information when they detect it. Note that we also ignore the travel time in this example and there are two feasible policies for each vehicle in this case: $\pi_1^a = \{x_1^1(1) = 0, x_2^1(1) = 1\}, \pi_1^b = \{x_1^2(1) = 1, x_2^2(1) = 0\}$ and $\pi_2^a = \{x_1^1(1) = 0, x_2^1(1) = 1\}, \pi_2^b = \{x_1^2(1) = 1, x_2^2(1) = 0\}$ for vehicles 1 and 2, respectively. We first show that (π_1^a, π_2^a) is a local optimum of problem (IAP).

Since the two vehicles are the same, we just need to show that $R(\pi_1^a, \pi_2^a) > R(\pi_1^b, \pi_2^a)$. To verify this, we have $R(\pi_1^a, \pi_2^a) = 2e_2p_2(1)(1-p_2(1)) + e_2p_2(1)^2(1+\bar{\gamma}) = \epsilon(1-\epsilon) + 0.5(1-\epsilon)^2(1+\bar{\gamma}) = 0.5(1-\epsilon^2) + 0.5\bar{\gamma}(1-\epsilon)^2$ and $R(\pi_1^b, \pi_2^a) = e_2g_2p_2(1) + e_1g_1p_1(1) = 0.5\bar{\gamma}(1-\epsilon)^2 + 0.5(1-\epsilon) < 0.5(1-\epsilon^2) + 0.5\bar{\gamma}(1-\epsilon)^2$. Therefore, (π_1^a, π_2^a) is a local optimum. It is easy to find that the optimal policy is (π_1^b, π_2^b), for which we have $R(\pi_1^b, \pi_2^b) = 2e_1p_1(1)(1-p_1(1)) + e_1p_1(1)^2(1+\bar{\gamma}) = \epsilon(1-\epsilon)^2\bar{\gamma}(1-\epsilon) + 0.5(1-\epsilon)^2(1+\bar{\gamma})(1-\epsilon)$. We have

$$\lim_{\epsilon \to 0} \frac{0.5(1-\epsilon^2) + 0.5\bar{\gamma}(1-\epsilon)^2}{\epsilon(1-\epsilon)^2\bar{\gamma}(1-\epsilon) + 0.5(1-\epsilon)^2(1+\bar{\gamma})\bar{\gamma}(1-\epsilon)} = \frac{1}{\bar{\gamma}}.$$

Thus, (22) can be approached arbitrarily close. □