2022 IEEE 61st Conference on Decision and Control (CDC)
December 6-9, 2022. Cancún, Mexico

Distributed submodular maximization: trading performance for privacy

Navid Rezazadeh and Solmaz S. Kia, *IEEE member, Senior*

2022 IEEE 61st Conference on Decision and Control (CDC) | 978-1-6654-6761-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/CDC51059.2022.9992452

Abstract— **This paper considers a multi-agent submodular set function maximization problem subject to partition matroid in which the utility is shared, but the agents' policy choices are constrained locally. The paper's main contribution is a distributed algorithm that enables each agent to find a sub-optimal policy locally with a guaranteed level of privacy. The submodular set function maximization problems are NP-hard. For agents communicating over a connected graph, this paper proposes a polynomial-time distributed algorithm to obtain a guaranteed near optimal solution. The proposed algorithm is based on a distributed randomized gradient ascent scheme on the multilinear extension of the submodular set function in the continuous domain. Our next contribution is the design of a distributed rounding algorithm that does not need any inter-agent communication. We base our algorithm's privacy preservation characteristic on our proposed stochastic rounding method and tie the level of privacy to the variable $\gamma \in [0, 1]$. That is, the policy choice of an agent can be determined with the probability of at most γ. We show that our distributed algorithm results in a strategy set that when the team's objective function is evaluated in the worst case, the objective function value is in $1 - (1/\mathrm{e})^{h(\gamma)} - O(1/T)$ of the optimal solution, highlighting the interplay between level of optimality gap and guaranteed level of privacy where T is the number of communication rounds between the agents.**

I. Introduction

We consider a set of $\mathcal{A}, |\mathcal{A}| = N$ agents (agents) with communication and computation capabilities, interacting over a connected undirected graph $\mathcal{G}(\mathcal{A}, \mathcal{E})$. Each agent $a \in \mathcal{A}$ has a distinct discrete policy set $\mathcal{P}_a$ and wants to choose $\kappa_a \in \mathbb{Z}_{\geq 1}$ policies from its policy set such that a monotone increasing and submodular utility function $f : 2^{\mathcal{P}} \to \mathbb{R}_{\geq 0}, \mathcal{P} = \bigcup_{a \in \mathcal{A}} \mathcal{P}_a$, evaluated at all the agents' policy selection is maximized[1]. In other words, the agents aim to solve in a distributed manner

$$\max_{\mathcal{R} \in \mathcal{I}} f(\mathcal{R}) \quad \text{s.t.} \tag{1a}$$

$$\mathcal{I} = \big\{ \mathcal{S} \subset \mathcal{P} \,\big|\, |\mathcal{S} \cap \mathcal{P}_a| \leq \kappa_a,\ a \in \mathcal{A} \big\}. \tag{1b}$$

agents' access to the utility function is through a black box that returns $f(\mathcal{R})$ for any given set $\mathcal{R} \in \mathcal{P}$ (value oracle model). Constraint set (1b) is called a *partition matroid*, which restricts the number of policy choices of each agent $a \in \mathcal{A}$ to a prespecified number κ_a. While seeking a distributed solution for (1), each agent wants to have a formal guarantee that its final policy choice stays private. Even though a distributed solution eliminates the necessity of information aggregation in a central location, inter-agent communication can still expose distributed network operations to adversarial eavesdroppers. These adversaries can be other agents in the network or outside eavesdroppers that intercept communication messages. Because in problem (1) the agents have joint utility function, privacy preservation is particularly a challenging problem. In this paper, the privacy preservation is defined in the following sense.

Definition 1: Let $\bar{\mathcal{R}}_a \in \mathcal{P}_a$ be the policy selected by agent $a \in \mathcal{A}$ when the agents use a distributed algorithm to solve the policy selection problem (1). We say $\bar{\mathcal{R}}_a \in \mathcal{P}_a$ is γ-Private where $\gamma \in [0, 1]$, if an intelligent entity other than agent a, which has access to the inter-agent communication messages, knows the network topology and also the distributed algorithm routine, is only able to estimate $p \in \bar{\mathcal{R}}_a$ with probability of at most γ for all $p \in \mathcal{P}_a$

Problem (1) falls in the so-called distributed-constraint submodular maximization class of problems in networked systems. In a distributed-constraint problem, there is a shared utility, but each agent has to choose its strategy from a local constraint set that is disjoint from other agents and is only known to the agent [1]–[6]. An example is the heterogeneous coverage problem where each agent has a set of heterogeneous sensors while the area to cover is shared among them [7]. This is different than the distributed-utility problems such as Welfare problem [8] where the team's utility function is the sum of the separable local utilities, and agents choose their strategies from a shared strategy set [9]–[11].

Submodular maximization subject to matroid constraint is an NP-hard problem [12]. However, thanks to the inherent properties of submodular functions, suboptimal solutions with quantifiable approximation factors have been successfully proposed in the literature. The most well-known result is the *sequential greedy algorithm* that dates back to the 1970s by [12], guaranteeing $1/2$-approximation factor solution for problem (1) when it is solved in a centralized manner. The sequential greedy algorithm presets a sequence, and each agent chooses its own best local policy given the choices of the preceding agents in the sequence. The sequential greedy algorithm can be implemented via sequential message-passing over a connected graph. However, this comes with routing overhead to identify the shortest path that visits every agent. More importantly, in the sequential greedy algorithm, the agents' privacy is breached as each agent passes its local policy set to those proceeding it in the message-passing sequence.

More recently, another suboptimal solution for submodular maximization subject to matroid constraints with an

The authors are with the Mechanical and Aerospace Eng. Dept. of University of California Irvine, Irvine, USA. {nrezazad, solmaz}@uci.edu.
This work was supported by NSF award ECCS-1653838.

[1]For clarity, we provide a brief description of the notation and the definitions in Section II.

improved approximation factor of $(1-1/\mathrm{e})$ was proposed [8], [13]–[16]. This method relies on continuous relaxation using a multilinear extension of submodular set functions and matroid polytope. The continuous relaxation is solved using a gradient ascent algorithm, and the integer solution then is rounded using appropriate rounding procedures such as Pipage rounding [17] or randomized Pipage rounding [13]. Besides its improved optimality gap, this approach has been shown to be amenable for distributed implementations that use synchronous inter-neighbor communication both for distributed-constraint problems [1]–[3] and distributed-utility problems [9]–[11].

However, none of the existing work in distributed submodular maximization formally addresses the privacy preservation of the agents. Our privacy preservation mechanism differs from existing methods for distributed algorithms such as differential privacy [18], [19], which rely on adding additive noises to the inter-agent communication messages. Differential privacy has also been the main approach in centralized submodular maximization problems subject to cardinality [20], [21] and matroid constraints [22]. Instead of adding noise to data or inter-agent communications, the innovation in our work is to base our privacy preservation mechanism on a stochastic rounding method and tie the level of privacy to a variable $\gamma \in [0,1]$. We show that our distributed algorithm results in a strategy set such that when the team's objective function is evaluated at worst case, the objective function value is in $1-(1/\mathrm{e})^{1-\sqrt[\kappa_{\max}]{1-\gamma}} - O(1/T)$, $\kappa_{\max} = \underset{a\in\mathcal{A}}{\max}\kappa_a$ of the optimal solution in value oracle model, highlighting the interplay between level of optimality gap and guaranteed level of privacy.

II. Notation

For a vector $\mathbf{x} \in \mathbb{R}^n$, the pth element of the vector is returned as $[\mathbf{x}]_p$. For $x \in \mathbb{R}$, its absolute value is $|x|$. By overloading the notation, we also use $|\mathcal{R}|$ as the cardinality of set $\mathcal{R}$. Given a set $\mathcal{P} = \{1, \cdots, n\}$ the vector $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ with $0 \leq [\mathbf{x}]_p \leq 1$, is referred to as *belief vector*, and the set $\mathcal{R}_{\mathbf{x}}$ is a random set where $p \in \mathcal{P}$ is in $\mathcal{R}_{\mathbf{x}}$ with the probability $[\mathbf{x}]_p$. Furthermore, for $\mathcal{R} \subset \mathcal{P}$, $\mathbf{1}_{\mathcal{R}} \in \{0,1\}^n$, referred to as *belief indicator vector*, is the vector whose p^{th} element is 1 if $p \in \mathcal{R}$ and 0 otherwise. Lastly, $\Delta_f(p|\mathcal{R}) = f(\mathcal{R}\cup\{p\}) - f(\mathcal{R})$ for any $\mathcal{R} \subset \mathcal{P}$ and $p \in \mathcal{P}$. We denote a graph by $\mathcal{G}(\mathcal{A}, \mathcal{E})$ where $\mathcal{A}$ is the node set and $\mathcal{E} \subset \mathcal{A}\times\mathcal{A}$ is the edge set. $\mathcal{G}$ is undirected if and only $(i,j) \in \mathcal{E}$ means that agents i and j can exchange information. An undirected graph is connected if there is a path from each node to every other node in the graph. We denote the set of the neighboring nodes of node i by $\mathcal{N}_i = \{j \in \mathcal{A} | (i,j) \in \mathcal{E}\}$. We also use $d(\mathcal{G})$ to show the diameter of the graph.

III. Problem ReFormulation

A distributed solution to problem (1) should enable each agent $a \in \mathcal{A}$ to choose κ_a policies from its local policy set $\mathcal{P}_a$. To propose our distributed algorithm, we assert a problem reformulation by introducing virtual agents to the network. We split each agent $a \in \mathcal{A}$ into κ_a fully connected local sub-agents embedded in agent a, see Fig 1, and design

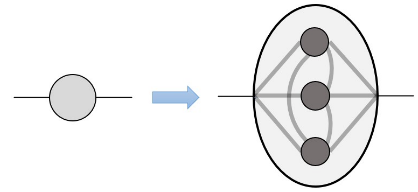

Fig. 1: The extension of an agent with $\kappa_i = 3$ to the sub-agents.

the distributed algorithm such that each sub-agent is responsible to choose a single policy. Without loss of generality, we index the sub-agent set as $\mathcal{A}^{\text{Ex}} = \{1, \cdots, N^{\text{Ex}}\}$ where $N^{\text{Ex}} = \sum_{a\in\mathcal{A}} \kappa_a$ and $\mathcal{A}_a = \{a_1, \cdots, a_{\kappa_a}\} \subset \mathcal{A}^{\text{Ex}}$. Hence $\mathcal{A}^{\text{Ex}} = \bigcup_{a\in\mathcal{A}} \mathcal{A}_a$. Moreover, we define the ground set in the extended space as $\mathcal{P}^{\text{Ex}} = \{1, \cdots, n^{\text{Ex}}\}$, where $n^{\text{Ex}} = \sum_{a\in\mathcal{A}} \kappa_a |\mathcal{P}_a|$ and the policy set of each sub-agent $i \in \mathcal{A}_a \subset \mathcal{A}^{\text{Ex}}$ is defined as $\mathcal{P}_i^{\text{Ex}} = \{p_i^1, \cdots, p_i^{|\mathcal{P}_a|}\} \subset \mathcal{P}^{\text{Ex}}$ and $\mathcal{P}^{\text{Ex}} = \bigcup_{i\in\mathcal{A}^{\text{Ex}}} \mathcal{P}_i^{\text{Ex}}$. The policy set of sub-agent $i \in \mathcal{A}_a$ is defined as a copy of the policy set of the agent a. Hence, given that $p_i^l \in \mathcal{P}_i^{\text{Ex}}$ and $p_j^l \in \mathcal{P}_j^{\text{Ex}}$ are the copies of the l-th policy of agent $a \in \mathcal{A}$ where sub-agents $i, j \in \mathcal{A}_a$ then for any $\mathcal{R} \subset \mathcal{P}^{\text{Ex}}$ the following holds

$$\Delta_f(p_i^l|\mathcal{R}) = \Delta_f(p_j^l|\mathcal{R}) \tag{2a}$$

$$\Delta_f(p_i^l|\mathcal{R}\cup\{p_j^l\}) = \Delta_f(p_j^l|\mathcal{R}\cup\{p_i^l\}) = 0 \tag{2b}$$

Moreover, we define the policy mapping function $\mathsf{PolicMap}(p) = q$, $p \in \mathcal{P}_i^{\text{Ex}}$ to return $q \in \mathcal{P}_a$ where p is a copy of policy q. In this extended space, problem (1) can equivalently be represented as

$$\max_{\mathcal{R}\in\mathcal{I}^{\text{Ex}}} f(\mathcal{R}) \quad \text{s.t.} \tag{3a}$$

$$\mathcal{I}^{\text{Ex}} = \left\{\mathcal{R} \subset \mathcal{P}^{\text{Ex}} \,\middle|\, |\mathcal{R} \cap \mathcal{P}_i^{\text{Ex}}| \leq 1,\ i \in \mathcal{A}^{\text{Ex}}\right\}. \tag{3b}$$

Without loss of generality we assume that $\mathcal{P}^{\text{Ex}}$ is ordered according to $\mathcal{A}^{\text{Ex}}$ in a sense that $1 \in \mathcal{P}_1^{\text{Ex}}$ and $n^{\text{Ex}} \in \mathcal{P}_{N^{\text{Ex}}}^{\text{Ex}}$. $\mathcal{A}^{\text{Ex}}$ is ordered in a way that sub-agents of an agent are ordered sequentially and the sub-agent sets are ordered in accordance with their corresponding agent order in $\mathcal{A}$. With the new formulation of the problem in the context of sub-agents, the optimal solution to problem (3) is equivalent to the optimal solution of the main problem (1). The equivalent problem formulation suggests that instead of each agent $a \in \mathcal{A}$ selecting κ_a policies, they create κ_a sub-agents and each selects only one policy out of $\mathcal{P}_a$.

To solve (3), we use a continuous relaxation method. Notice that the utility set function f assigns values to all the subsets of $\mathcal{P}^{\text{Ex}} = \bigcup_{i\in\mathcal{A}^{\text{Ex}}} \mathcal{P}_i^{\text{Ex}} = \{1, \cdots, n^{\text{Ex}}\}$. Thus, equivalently, we can regard the set value utility function as a function on the Boolean hypercube $\{0,1\}^{n^{\text{Ex}}}$, i.e., $f : \{0,1\}^{n^{\text{Ex}}} \to \mathbb{R}$. For a submodular function $f : 2^{\mathcal{P}^{\text{Ex}}} \to \mathbb{R}_{\geq 0}$, its multilinear-extension $F : [0,1]^{n^{\text{Ex}}} \to \mathbb{R}_{\geq 0}$ is [8]

$$F(\mathbf{x}) = \sum_{\mathcal{R}\subset\mathcal{P}^{\text{Ex}}} f(\mathcal{R}) \prod_{p\in\mathcal{R}} [\mathbf{x}]_p \prod_{p\notin\mathcal{R}} (1-[\mathbf{x}]_p), \quad \mathbf{x} \in [0,1]^{n^{\text{Ex}}}, \tag{4}$$

which expands the function evaluation of the utility function over the space between the vertices of the Boolean hypercube $\{0,1\}^{n^{\text{Ex}}}$. Given $\mathbf{x} \in [0,1]^{n^{\text{Ex}}}$ we can define $\mathcal{R}_{\mathbf{x}}$ to be the random subset of $\mathcal{P}$ in which each element $p \in \mathcal{P}$ is included independently with probability $[\mathbf{x}]_p$ and not included with probability $1-[\mathbf{x}]_p$. Then the multilinear-extension F in (4) is interpreted

$$F(\mathbf{x}) = \mathbb{E}[f(\mathcal{R}_{\mathbf{x}})], \tag{5}$$

where $\mathbb{E}[.]$ indicates the expected value. Then, we obtain [8]

$$\frac{\partial F}{\partial [\mathbf{x}]_p}(\mathbf{x}) = \mathbb{E}[f(\mathcal{R}_{\mathbf{x}} \cup \{p\}) - f(\mathcal{R}_{\mathbf{x}} \setminus \{p\})]. \tag{6}$$

The partition matroid constraint is also extended to continuous space using the *matroid polytope*

$$\mathcal{M} = \{\mathbf{x} \in [0,1]^{n^{\text{Ex}}} \mid \textstyle\sum_{p \in \mathcal{P}_i^{\text{Ex}}} [\mathbf{x}]_p \leq 1, \forall i \in \mathcal{A}^{\text{Ex}}\}, \tag{7}$$

which is the convex hull of the vertices of the hypercube $\{0,1\}^{n^{\text{Ex}}}$ that satisfy the partition matroid constraint (3b). Additionally, note that according to (4), $F(\mathbf{x})$ for any $\mathbf{x} \in \mathcal{M}$ is a weighted average of values of F at the vertices of the matriod polytope $\mathcal{M}$. Then, equivalently, $F(\mathbf{x})$ at any $\mathbf{x} \in \mathcal{M}$ is a normalized-weighted average of f on the strategies satisfying constraint (1b). As such,

$$f(\mathcal{R}^\star) \geq F(\mathbf{x}),\ \ \mathbf{x} \in \mathcal{M}, \ \text{ and } \ f(\mathcal{R}^\star) = F(\mathbf{1}_{\mathcal{R}^\star}),$$

which is equivalent to $f(\mathcal{R}^\star) = \max_{\mathbf{x} \in \mathcal{M}} F(\mathbf{x})$, where $\mathcal{R}^\star$ is the optimizer of problem (3) [8]. Therefore, to find $\mathcal{R}^\star$, we can solve the continuous domain optimization problem

$$\max_{\mathbf{x} \in \mathcal{M}} F(\mathbf{x}). \tag{8}$$

IV. Distributed private policy selection

The continuous relaxation (8) of the set value optimization problem (3) allows us to use a plethora of continuous domain optimization solvers such a gradient-based algorithms. But since problem (8) is not a concave problem, a gradient ascent approach does not necessarily lead to the optimal value. Nevertheless, in this section, we propose Algorithm 1 as a γ-private distributed gradient ascent algorithm base on (8) to find a sub-optimal solution to the problem (1).

Let every sub-sub-agent $i \in \mathcal{A}^{\text{Ex}}$ maintain and evolve a belief state vector as $\mathbf{x}_i(t) \in \mathbb{R}^{n^{\text{Ex}}}$. Each entry $[\mathbf{x}_i(t)]_p$, $p \in \mathcal{P}_j^{\text{Ex}}$, $j \in \mathcal{A}^{\text{Ex}}$ of the belief state corresponds to the estimate of sub-agent i on the confidence of sub-agent j about choosing p as the final selected policy. Since $\mathcal{P}^{\text{Ex}} = \{1, \cdots, n^{\text{Ex}}\}$ is sorted sub-agent-wise, we denote $\mathbf{x}_i(t) = [\hat{\mathbf{x}}_{i1}^\top(t), \cdots, \mathbf{x}_{ii}^\top(t), \cdots, \hat{\mathbf{x}}_{iN^{\text{Ex}}}^\top(t)]^\top \in \mathbb{R}^{n^{\text{Ex}}}$ where $\mathbf{x}_{ii}(t) \in \mathbb{R}_{\geq 0}^{|\mathcal{P}_i^{\text{Ex}}|}$ is the belief vector of sub-agent i's own policy with entries of $[\mathbf{x}_i(t)]_p$, $p \in \mathcal{P}_i^{\text{Ex}}$ at iteration $t \in \{0, 1, \cdots, T\}$, $T \in \mathbb{Z}_{>0}$, while $\hat{\mathbf{x}}_{ij}(t) \in \mathbb{R}_{\geq 0}^{|\mathcal{P}_j^{\text{Ex}}|}$ is the local estimate of the belief vector of sub-agent j by sub-agent i with entries of $[\mathbf{x}_i(t)]_p$, $p \in \mathcal{P}_j^{\text{Ex}}$, $j \in \mathcal{A}^{\text{Ex}} \setminus \{i\}$. Let

$$\lambda = \frac{T}{1 - \sqrt[\kappa_{\max}]{1-\gamma}}, \quad \kappa_{\max} = \max_{a \in \mathcal{A}} \kappa_a. \tag{9}$$

Every sub-agent $i \in \mathcal{A}^{\text{Ex}}$ initializes at $\mathbf{x}_i(0) = \mathbf{0}$ and implements the *propagation* and *update* steps

$$\mathbf{x}_i^-(t+1) = \mathbf{x}_i(t) + \frac{1}{\lambda}\widetilde{\mathbf{v}}_i(t), \tag{10a}$$

$$\mathbf{x}_i(t+1) = \max_{j \in \mathcal{N}_i \cup \{i\}} \mathbf{x}_j^-(t+1), \tag{10b}$$

where

$$\widetilde{\mathbf{v}}_i(t) = \underset{\mathbf{w} \in \mathcal{M}_i}{\operatorname{argmax}}\, \mathbf{w}.\widetilde{\nabla F}(\mathbf{x}_i(t)) \tag{11}$$

$$\mathcal{M}_i = \Big\{\mathbf{w} \in [0,1]^{n^{\text{Ex}}} \Big| \, \mathbf{1}.\mathbf{w} \leq 1\ , [\mathbf{w}]_p = 0,\ \forall p \in \mathcal{P}^{\text{Ex}} \backslash \mathcal{P}_i^{\text{Ex}}\Big\}. \tag{12}$$

The value of $\widetilde{\nabla F}(\mathbf{x}_i(t))$ is the empirical estimate of $\nabla F(\mathbf{x}_i(t))$ calculated locally by each agent by using K_i samples. The distributed sampling and empirical calculation of $\widetilde{\nabla F}(\mathbf{x}_i(t))$ are the same as the method introduced in [2, Section III.C] and are omitted here for brevity. In the propagation step (10a) sub-agent i takes a step along a feasible gradient ascent direction in its own local polytope (12). But because the propagation is only based on the local information, in the update step (10b), the propagated $\mathbf{x}_i^-(t+1)$ of each sub-agent $i \in \mathcal{A}$ is updated by element-wise maximum seeking among its neighbors. Because f is monotone increasing, we have $\frac{\partial F}{\partial [\mathbf{x}]_p} \geq 0$, which leads to $\mathbf{1}.\tilde{\mathbf{v}}_i(t) = 1$ where $\tilde{\mathbf{v}}_i(t) \in \mathcal{M}_i$. Therefore, given λ by equation (9), at the end of the propagation and update process at time T, we have

$$\mathbf{1}.\mathbf{x}_{ii}(T) = 1 - \sqrt[\kappa_{\max}]{1-\gamma}. \tag{13}$$

For the rounding procedure it is essential that each sub-agent has a local belief state vector on its own policy set that sums up to 1. Therefore, given (13), each sub-agent $i \in \mathcal{A}^{\text{Ex}}$ applies a γ-private operation on its local belief state vector and generates the local belief vector $\mathbf{z}_i$ such that

$$\mathbf{z}_i = [\mathbf{0}^\top, \cdots, \mathbf{x}_{ii}^\top(T) + \mathbf{y}_{ii}^\top, \cdots, \mathbf{0}^\top]^\top \in \mathbb{R}^{n^{\text{Ex}}}, \tag{14}$$

where $\mathbf{y}_{ii}$ is generated privately by sub-agent i such that

$$\mathbf{1}.\mathbf{y}_{ii} = \sqrt[\kappa_{\max}]{1-\gamma}. \tag{15}$$

We are now ready to propose our distributed rounding scheme that enables each agent $a \in \mathcal{A}$ to choose its local policy set. Our proposed rounding scheme does not require any inter-agent communication because our design process leads to $\mathbf{1}.(\mathbf{x}_{ii}(T)+\mathbf{y}_{ii}) = 1$ and consequently $\mathbf{1}.\mathbf{z}_i = 1$. Our proposed rounding method is cooperative in the level of sub-agents of an agent $a \in \mathcal{A}$. This is an acceptable assumption since the sub-agents of each agent $a \in \mathcal{A}$ are virtual agents created by agent a. Given that $\mathcal{A}_a = \{a_1, \cdots, a_{\kappa_a}\} \subset \mathcal{A}^{\text{Ex}}$, and defining from $\mathcal{T}(0) = \emptyset$, each sub-agent $i = a_k \in \mathcal{A}_a$ of agent $a \in \mathcal{A}$, uses the vector $\mathbf{z}_i$ and a uniformly random generated variable $\zeta \in [0,1]$ to choose a single policy $p \in \mathcal{P}_i^{\text{Ex}} \subset \mathcal{P}^{\text{Ex}}$ satisfying

$$\textstyle\sum_{l=1}^{p} [\mathbf{z}_i]_l \leq \zeta \leq \sum_{l=1}^{p+1} [\mathbf{z}_i]_l \tag{16}$$

Algorithm 1 Distributed γ-Private extension-based algorithm

Init: $\bar{\mathcal{R}} \leftarrow \emptyset,\ \mathbf{x}_i(0) \leftarrow \mathbf{0},\ t \leftarrow 1,$
while $t \leq T$ **do**
  for $i \in \mathcal{A}^{\text{Ex}}$ **do**
    Draw K_i sample policy sets $\mathcal{R}_{\mathbf{x}_i(t)}$.
    for $p \in \mathcal{P}_i^{\text{Ex}}$ **do**
      Calculate $\left[\widetilde{\nabla F}(\mathbf{x}_i(t))\right]_p$ by empirically estimate
    end for
    Solve for $\widetilde{\mathbf{v}}_i(t) = \underset{\mathbf{w}\in\mathcal{M}_i}{\text{argmax}}\, \mathbf{w}.\widetilde{\nabla F}(\mathbf{x}_i(t))$
    Propagate $\mathbf{x}_i^-(t+1) = \mathbf{x}_i(t) + \frac{1}{\lambda}\widetilde{\mathbf{v}}_i(t)$
    Broadcast $\mathbf{x}_i(t)$ to the neighbors $\mathcal{N}_i$.
    Update $\mathbf{x}_i(t+1) = \max\limits_{j\in\mathcal{N}_i\cup\{i\}} \mathbf{x}_j^-(t+1)$
  end for
  $t \leftarrow t+1$.
end while
for $i \in \mathcal{A}^{\text{Ex}}$ **do**
  Generate $\mathbf{y}_{ii}$ and form $\mathbf{z}_i$
end for
for $a \in \mathcal{A}$ **do**
  Sample $\bar{\mathcal{T}}_a$ using $\mathbf{z}_i,\ i \in \mathcal{A}_a$ using Algorithm 2
end for
Map the policies $\bar{\mathcal{R}}_a = \{\mathsf{PolicyMap}(p) | p \in \mathcal{T}_a\}$
Return $\bar{\mathcal{R}} = \bigcup_{a\in\mathcal{A}} \bar{\mathcal{R}}_a$

to get

$$\mathcal{T}(k) = \mathcal{T}(k-1) \cup \{p\}. \tag{17}$$

Setting $\bar{\mathcal{T}}_a = \mathcal{T}(\kappa_a)$, each agent $a \in \mathcal{A}$ chooses its policy set from the policy choices of its sub-agents as

$$\bar{\mathcal{R}}_a = \{\mathsf{PolicyMap}(p) |\, p \in \mathcal{T}_a\}. \tag{18}$$

Defining $\bar{\mathcal{R}} = \bigcup_{a\in\mathcal{A}} \bar{\mathcal{R}}_a$ to be the collective selected policies of the agents $\mathcal{A}$, the following theorems assert the guaranteed optimality bound and privacy preservation guarantee of Algorithm 1.

Theorem 4.1 (Sub-optimality gap of Algorithm 1) *Let $f : 2^{\mathcal{P}} \to \mathbb{R}_{\geq 0}$ be normalized, monotone increasing and submodular set function. Let $\mathcal{R}^\star$ to be the optimizer of problem* (1). *Then, the policy set $\bar{\mathcal{R}}$, the output of distributed Algorithm 1, satisfies*

$$\alpha\big(1 - \frac{1}{\mathrm{e}^{1-\sqrt[\kappa_{\max}]{1-\gamma}}}\big) f(\mathcal{R}^\star) \leq \mathbb{E}[f(\bar{\mathcal{R}})],$$

with the probability of at least $\left(\prod_{i\in\mathcal{A}^{\text{Ex}}}(1 - 2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|\kappa_i}\right)^T$ *and*

$$\alpha = \big(1 - \Big(2{N^{\text{Ex}}}^2 d(\mathcal{G}) + \frac{1}{2}{N^{\text{Ex}}}^2 + N^{\text{Ex}}\Big)\frac{1}{\lambda}\big).$$

The proof of Theorem 4.1 is given in the appendix.

Theorem 4.2 (Privacy characteristics of Algorithm 1) *The distributed Algorithm 1 used by each agent $a \in \mathcal{A}$ to solve the policy selection problem* (1) *to achieve the policy set $\bar{\mathcal{R}}_a \in \mathcal{P}_a$ is γ-Private in the sense of Definition 1. Specifically, each policy in $\mathcal{P}_a$ can be estimated to exist in $\bar{\mathcal{R}}_a$ with probability of at most $1 - (\sqrt[\kappa_{\max}]{1-\gamma})^{\kappa_a} < \gamma$.*

Due to space limitation the proof of Theorem 4.2 appears elsewhere. Intuitively speaking, notice that the privacy guarantees originate from the fact that instead of rounding $\mathbf{X}_{ii}(T)$ each sub-agent $i \in \mathcal{A}^{\text{Ex}}$ rounds $\mathbf{X}_{ii}(T) + \mathbf{y}_{ii}$. Notice that $\mathbf{X}_{ii}(T)$ is exposed due to inter-agent message broadcast, whereas elements of $\mathbf{y}_{ii}$ are chosen locally and randomly, with the only exposed known fact about them being (15).

Algorithm 2 Distributed rounding

Input: $a,\ \kappa_a,\ \mathcal{A}_a = \{a_1, \cdots, a_{\kappa_a}\},\ \mathcal{P}_i^{\text{Ex}},\ \mathbf{z}_i,\ i \in \mathcal{A}_a.$
Init: $\mathcal{T}(0) = \emptyset,\ k \leftarrow 1,$
for $k \in \{1, \cdots, \kappa_a\}$ **do**
  Set $i = a_k$
  Generate a random number $\zeta \in [0, 1]$ with a uniform distribution.
  Find $p \in \mathcal{P}_i^{\text{Ex}}$ such that $\sum_{l=1}^{p}[\mathbf{z}_i]_l \leq \zeta \leq \sum_{l=1}^{p+1}[\mathbf{z}_i]_l$
end for
$\mathcal{T}(k) = \mathcal{T}(k-1) \cup \{p\}$
$\mathcal{T}_a = \mathcal{T}(\kappa_a)$
Return $\mathcal{T}_a$

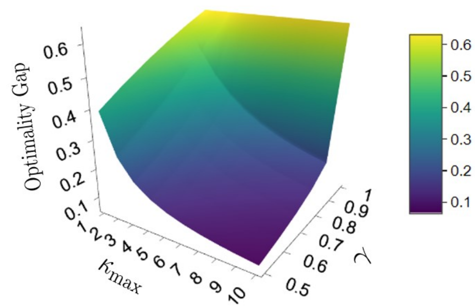

Fig. 2: The trade-off between optimality gap and privacy as a function of $\kappa_{\max}$ when a large value of T is used.

Theorem 4.1 and Theorem 4.2 highlight the trade-off between the size of optimality gap and the guaranteed level of privacy for Algorithm 1. Figure 2 shows the trade-off when T is set to a very large value to eliminate the effect of α. Notice, for example, for $\kappa_{\max} = 1$ and $\gamma = 0.5$ the optimality gap is 0.4, which is a drop from the theoretical best optimality gap of $1 - 1/e \approx 0.632$ corresponding to $\gamma = 1$.

V. Conclusion

We proposed a distributed suboptimal algorithm to solve a distributed-constraint problem of maximizing a monotone increasing submodular set function subject to a partition matroid constraint. The main contribution of this paper was to design a distributed algorithm that enabled each agent to find a suboptimal policy locally with a guaranteed level of privacy. Our algorithm was a distributed solution for the continuous relaxation of the problem of interest followed by a fully distributed loss-less rounding procedure that needed no inter-agent communication. We proposed a novel privacy-preservation framework that tied the level of privacy to a variable $\gamma \in [0, 1]$, which determined the maximum probability that the local policy choice of an agent would be revealed. We conclude by formally establishing the trade-off between the worst-case optimality gap and the guaranteed level of privacy.

References

[1] A. Robey, A. Adibi, B. Schlotfeldt, J. G. Pappas, and H. Hassani, "Optimal algorithms for submodular maximization with distributed constraints," *Learning for Dynamics and Control*, pp. 150–162, 2021.

[2] N. Rezazadeh and S. Kia, "Multi-agent maximization of a monotone submodular function via maximum consensus," in *IEEE Conference on Decision and Control*, pp. 1238–1243, IEEE, 2021.

[3] N. Rezazadeh and S. S.S. Kia, "Distributed strategy selection: A submodular set function maximization approach," *arXiv preprint arXiv:2107.14371*, 2021.

[4] N. Rezazadeh and S. Kia, "A sub-modular receding horizon solution for mobile multi-agent persistent monitoring," *Automatica*, vol. 127, p. 109460, 2021.

[5] B. Gharesifard and S. Smith, "Distributed submodular maximization with limited information," *IEEE transactions on control of network systems*, vol. 5, no. 4, pp. 1635–1645, 2017.

[6] R. Konda, D. Grimsman, and J. Marden, "Execution order matters in greedy algorithms with limited information," *arXiv preprint arXiv:2111.09154*, 2021.

[7] A. Krause and D. Golovin, "Submodular function maximization," in *Tractability: Practical Approaches to Hard Problems* (L. Bordeaux, Y. Hamadi, and P. Kohli, eds.), pp. 71–104, Cambridge, UK: Cambridge University Press, 2014.

[8] J. Vondrák, "Optimal approximation for the submodular welfare problem in the value oracle model," in *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 67–74, 2008.

[9] A. Mokhtari, H. Hassani, and A. Karbasi, "Decentralized submodular maximization: Bridging discrete and continuous settings," in *Int. Conference on Machine Learning*, pp. 3616–3625, 2018.

[10] J. Xie, C. Zhang, Z. Shen, C. Mi, and H. Qian, "Decentralized gradient tracking for continuous dr-submodular maximization," in *Int. Conference on Artificial Intelligence and Statistics*, pp. 2897–2906, 2019.

[11] L. Ye and S. Sundaram, "Distributed maximization of submodular and approximately submodular functions," in *IEEE Conference on Decision and Control*, pp. 2979–2984, 2020.

[12] G. Nemhauser, L. Wolsey, and M. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical programming*, vol. 14, no. 1, pp. 265–294, 1978.

[13] J. Vondrák, "Submodularity and curvature: The optimal algorithm (combinatorial optimization and discrete algorithms)," 2010.

[14] A. A. Bian, B. Mirzasoleiman, J. Buhmann, and A. Krause, "Guaranteed non-convex optimization: Submodular maximization over continuous domains," in *Artificial Intelligence and Statistics*, pp. 111–120, 2017.

[15] A. Mokhtari, H. Hassani, and A. Karbasi, "Stochastic conditional gradient methods: From convex minimization to submodular maximization," *Journal of Machine Learning Research*, vol. 21, no. 105, pp. 1–49, 2020.

[16] O. Sadeghi and M. Fazel, "Online continuous dr-submodular maximization with long-term budget constraints," in *Int. Conference on Artificial Intelligence and Statistics*, pp. 4410–4419, 2020.

[17] A. A. Ageev and M. I. Sviridenko, "Pipage rounding: A new method of constructing algorithms with proven performance guarantee," *Journal of Combinatorial Optimization*, vol. 8, no. 3, pp. 307–328, 2004.

[18] Z. Huang, S. Mitra, and N. Vaidya, "Differentially private distributed optimization," in *Proceedings of Int. Conference on Distributed Computing and Networking*, p. 4, 2015.

[19] Y. Wang, "Privacy-preserving average consensus via state decomposition," *IEEE Transactions on Automatic Control*, vol. 64, no. 11, pp. 4711–4716, 2019.

[20] M. Mitrovic, M. Bun, A. Krause, and A. Karbasi, "Differentially private submodular maximization: Data summarization in disguise," in *Int. Conference on Machine Learning*, pp. 2478–2487, 2017.

[21] S. Perez-Salazar and R. Cummings, "Differentially private online submodular maximization," in *Int. Conference on Artificial Intelligence and Statistics*, pp. 1279–1287, 2021.

[22] A. Rafiey and Y. Yoshida, "Fast and private submodular and k-submodular functions maximization with matroid constraints," in *Int. Conference on Machine Learning*, pp. 7887–7897, 2020.

[23] N. Rezazadeh and S. S. Kia, "Multi-agent maximization of a monotone submodular function via maximum consensus," *arXiv preprint arXiv:2011.14499*, 2020.

VI. Appendix

Proof: [Proof of Theorem 4.1] Let $\bar{\mathbf{x}}(t) = \max_{i\in\mathcal{A}^{\text{Ex}}} \mathbf{x}_i(t)$. It follows from [23, Lemma 7.3] and [23, Proposition 5.1] that

$$F(\bar{\mathbf{x}}(t+1)) - F(\bar{\mathbf{x}}(t)) \geq \nabla F(\bar{\mathbf{x}}(t)).(\bar{\mathbf{x}}(t+1) - \bar{\mathbf{x}}(t)) - \frac{1}{2}{N^{\text{Ex}}}^2\frac{1}{\lambda^2}f(\mathcal{R}^\star),$$

which, further from [23, Proposition 5.1] we get

$$F(\bar{\mathbf{x}}(t+1)) - F(\bar{\mathbf{x}}(t)) \geq \frac{1}{\lambda}\sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \widetilde{\mathbf{v}}_i(t).\nabla F(\bar{\mathbf{x}}(t)) - \frac{1}{2}{N^{\text{Ex}}}^2\frac{1}{\lambda^2}f(\mathcal{R}^\star). \quad (19)$$

Next, we note that by definition, $\bar{\mathbf{x}}(t) \geq \mathbf{x}_i(t)$ for any $\forall i \in \mathcal{A}^{\text{Ex}}$. Therefore, given [23, Proposition 5.1] and [23, Lemma 7.3], for any $i \in \mathcal{A}^{\text{Ex}}$, and $p \in \{1, \cdots, n^{\text{Ex}}\}$, we can write

$$\left|\frac{\partial F}{\partial[\mathbf{x}]_p}(\bar{\mathbf{x}}(t)) - \frac{\partial F}{\partial[\mathbf{x}]_p}(\mathbf{x}_i(t))\right| \leq N^{\text{Ex}}\frac{1}{\lambda}d(\mathcal{G}))f(\mathcal{R}^\star). \quad (20)$$

knowing that $\mathbf{1}.\widetilde{\mathbf{v}}_i(t) = 1$, $i \in \mathcal{A}^{\text{Ex}}$. Consequently, using (20) we can write

$$\sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \widetilde{\mathbf{v}}_i(t).\nabla F(\bar{\mathbf{x}}(t)) \geq \sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \widetilde{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) - {N^{\text{Ex}}}^2\frac{1}{\lambda}d(\mathcal{G}))f(\mathcal{R}^\star). \quad (21)$$

Next, we let

$$\bar{\mathbf{v}}_i(t) = \underset{\mathbf{w}\in\mathcal{M}_i}{\text{argmax}}\ \mathbf{w}.\nabla F(\bar{\mathbf{x}}(t)),$$

and

$$\hat{\mathbf{v}}_i(t) = \underset{\mathbf{w}\in\mathcal{M}_i}{\text{argmax}}\ \mathbf{w}.\nabla F(\mathbf{x}_i(t)).$$

Because f is monotone increasing, we have $\frac{\partial F}{\partial[\mathbf{x}]_p} \geq 0$, and we conclude that $\mathbf{1}.\bar{\mathbf{v}}_i(t) = 1$, $i \in \mathcal{A}^{\text{Ex}}$ and also $\mathbf{1}.\hat{\mathbf{v}}_i(t) = 1$, $i \in \mathcal{A}$. Therefore, using $\hat{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) \geq \bar{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t))$ and $\hat{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) \geq \tilde{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)), i \in \mathcal{A}^{\text{Ex}}$, and (20) we can also write

$$\sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}}\hat{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) \geq \sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}}\bar{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) \geq \sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}}\bar{\mathbf{v}}_i(t).\nabla F(\bar{\mathbf{x}}(t)) - {N^{\text{Ex}}}^2\frac{1}{\lambda}d(\mathcal{G})f(\mathcal{R}^\star), \quad (22a)$$

$$\sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}}\hat{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) \geq \sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}}\tilde{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)). \quad (22b)$$

On the other hand, by virtue of [23, Lemma 7.4], $\widetilde{\frac{\partial F}{\partial[\mathbf{x}]_p}}(\mathbf{x}_i(t))$, $p \in \mathcal{P}_i^{\text{Ex}}$ that each agent $i \in \mathcal{A}^{\text{Ex}}$ uses to solve optimization problem (11) satisfies

$$\left|\widetilde{\frac{\partial F}{\partial[\mathbf{x}]_p}}(\mathbf{x}_j(t)) - \frac{\partial F}{\partial[\mathbf{x}]_p}(\mathbf{x}_j(t))\right| \leq \frac{1}{2\lambda}f(\mathcal{R}^\star), \quad (23)$$

with the probability of at least $1 - 2\mathrm{e}^{-\frac{1}{8\lambda^2}K_j}$. Using (22b) and (23), and also that the samples are drawn independently

$$\sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \widetilde{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) \geq \sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \widetilde{\mathbf{v}}_i(t).\widetilde{\nabla F}(\mathbf{x}_i(t)) - N^{\text{Ex}}\frac{1}{2\lambda}f(\mathcal{R}^\star), \quad (24a)$$

$$\sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \widetilde{\mathbf{v}}_i(t).\widetilde{\nabla F}(\mathbf{x}_i(t)) \geq \sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \hat{\mathbf{v}}_i(t).\widetilde{\nabla F}(\mathbf{x}_i(t)) \geq \sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \hat{\mathbf{v}}_i(t).\nabla F(\mathbf{x}_i(t)) - N^{\text{Ex}}\frac{1}{2\lambda}f(\mathcal{R}^\star), \quad (24b)$$

with the probability of at least $\prod_{i\in\mathcal{A}^{\text{Ex}}}(1 - 2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|}$.

From (21), (22a),(24a), and (24b) now we can write

$$\sum\nolimits_{i\in\mathcal{A}^{\text{Ex}}} \widetilde{\mathbf{v}}_i(t).\nabla F(\bar{\mathbf{x}}(t)) \geq$$

$$\sum_{i\in\mathcal{A}^{\text{Ex}}}\bar{\mathbf{v}}_i(t).\nabla F(\bar{\mathbf{x}}(t))-(2N^{\text{Ex}}d(\mathcal{G}))+1)N^{\text{Ex}}\frac{1}{\lambda}f(\mathcal{R}^\star), \quad (25)$$

with the probability of at least $1-2\sum_{i\in\mathcal{A}^{\text{Ex}}}\mathrm{e}^{-\frac{1}{8\lambda^2}K_i}$.

Next, let $\mathbf{v}_i^\star$ be the projection of $\mathbf{1}_{\mathcal{R}^\star}$ into $\mathcal{M}_i$. Knowing that $\mathcal{M}_i$s are disjoint sub-spaces of $\mathcal{M}$ covering the whole space then we can write $\mathbf{1}_{\mathcal{R}^\star}=\sum_{i\in\mathcal{A}^{\text{Ex}}}\mathbf{v}_i^\star$. Then, using (25) and invoking [23, Lemma 7.1] and the fact that $\bar{\mathbf{v}}_i(t).\nabla F(\bar{\mathbf{x}}(t))\geq \mathbf{v}_i^\star(t).\nabla F(\bar{\mathbf{x}}(t))$ we obtain

$$\begin{aligned}&\sum_{i\in\mathcal{A}^{\text{Ex}}}\widetilde{\mathbf{v}}_i(t).\nabla F(\bar{\mathbf{x}}(t))\geq\\&\sum_{i\in\mathcal{A}^{\text{Ex}}}\mathbf{v}_i^\star(t).\nabla F(\bar{\mathbf{x}}(t))-(2N^{\text{Ex}}d(\mathcal{G}))+1)N^{\text{Ex}}\frac{1}{\lambda}f(\mathcal{R}^\star)=\\&\mathbf{1}_{\mathcal{R}^\star}.\nabla F(\bar{\mathbf{x}}(t))-(2N^{\text{Ex}}d(\mathcal{G}))+1)N^{\text{Ex}}\frac{1}{\lambda}f(\mathcal{R}^\star)\geq\\&f(\mathcal{R}^\star)-F(\bar{\mathbf{x}}(t))-(2N^{\text{Ex}}d(\mathcal{G}))+1)\frac{N^{\text{Ex}}}{\lambda}f(\mathcal{R}^\star), \quad (26)\end{aligned}$$

with the probability of at least $\prod_{i\in\mathcal{A}^{\text{Ex}}}(1-2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|}$. Hence, using (19) and (26), we conclude that

$$\begin{aligned}&F(\bar{\mathbf{x}}(t+1))-F(\bar{\mathbf{x}}(t))\geq\\&\frac{1}{\lambda}(f(\mathcal{R}^\star)-F(\bar{\mathbf{x}}(t))-(2N^{\text{Ex}}d(\mathcal{G}))+\frac{1}{2}N^{\text{Ex}}+1)\frac{N^{\text{Ex}}}{\lambda^2}f(\mathcal{R}^\star), \quad (27)\end{aligned}$$

with the probability of at least $\prod_{i\in\mathcal{A}^{\text{Ex}}}(1-2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|}$.

Next, let $g(t)=f(\mathcal{R}^\star)-F(\bar{\mathbf{x}}(t))$ and $\beta=(2N^{\text{Ex}}d(\mathcal{G}))+\frac{1}{2}N^{\text{Ex}}+1)\frac{N^{\text{Ex}}}{\lambda^2}f(\mathcal{R}^\star)$, to rewrite (27) as

$$\begin{aligned}&(f(\mathcal{R}^\star)-F(\bar{\mathbf{x}}(t)))-(f(\mathcal{R}^\star)-F(\bar{\mathbf{x}}(t+1)))=\\&g(t)-g(t+1)\geq\frac{1}{\lambda}(f(\mathcal{R}^\star)-F(\bar{\mathbf{x}}(t)))-\beta=\frac{1}{\lambda}g(t)-\beta. \quad (28)\end{aligned}$$

Then from inequality (28) we get

$$g(t+1)\leq(1-\frac{1}{\lambda})g(t)+\beta, \quad (29)$$

with the probability of at least $\prod_{i\in\mathcal{A}^{\text{Ex}}}(1-2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|}$. Solving for inequality (29) at time T yields

$$\begin{aligned}&g(T)\leq(1-\frac{1}{\lambda})^Tg(0)+\beta\sum_{k=0}^{T-1}(1-\frac{1}{\lambda})^k=\\&(1-\frac{1}{\lambda})^Tg(0)+\lambda\beta(1-(1-\frac{1}{\lambda})^T), \quad (30)\end{aligned}$$

with the probability of at least $\left(\prod_{i\in\mathcal{A}^{\text{Ex}}}(1-2\mathrm{e}^{-\frac{1}{8\lambda^2}K_j})^{|\mathcal{P}_i^{\text{Ex}}|}\right)^T$. Substituting back $g(T)=f(\mathcal{R}^\star)-F(\bar{\mathbf{x}}(T))$ and $g(0)=f(\mathcal{R}^\star)-F(\mathbf{x}(0))=f(\mathcal{R}^\star)$, in (30) we then obtain

$$\begin{aligned}&(1-(1-\frac{1}{\lambda})^T)(f(\mathcal{R}^\star)-\lambda\beta)=\\&(1-(1-\frac{1}{\lambda})^T)(1-(2N^{\text{Ex}}d(\mathcal{G}))+\frac{1}{2}N^{\text{Ex}}+1)\frac{N^{\text{Ex}}}{\lambda})f(\mathcal{R}^\star)\\&\leq F(\bar{\mathbf{x}}(T)), \quad (31)\end{aligned}$$

with the probability of at least $\left(\prod_{i\in\mathcal{A}^{\text{Ex}}}(1-2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|}\right)^T$. Knowing that $\frac{1}{\mathrm{e}}\geq(1-\frac{1}{\lambda})^\lambda$, we can write $(\frac{1}{\mathrm{e}})\geq(1-\frac{1}{\lambda})^\lambda$ and $\lambda=\frac{T}{1-\sqrt[\kappa_{\max}]{1-\gamma}}$ from equation (9), we can write $(\frac{1}{\mathrm{e}})^{1-\sqrt[\kappa_{\max}]{1-\gamma}}\geq(1-\frac{1}{\lambda})^T$ and consequently $1-(\frac{1}{\mathrm{e}})^{1-\sqrt[\kappa_{\max}]{1-\gamma}}\leq1-(1-\frac{1}{\lambda})^T$. Hence, we conclude

$$\alpha(1-\frac{1}{\mathrm{e}^{1-\sqrt[\kappa_{\max}]{1-\gamma}}})f(\mathcal{R}^\star)\leq F(\bar{\mathbf{x}}(T)),$$

with the probability of at least $\left(\prod_{i\in\mathcal{A}^{\text{Ex}}}(1-2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|}\right)^T$ where

$$\alpha=1-(2N^{\text{Ex}}d(\mathcal{G})+\frac{1}{2}N^{\text{Ex}}+1)\frac{N^{\text{Ex}}}{\lambda}.$$

Moreover, by defining $\bar{\mathbf{z}}=\max_{i\in\mathcal{A}^{\text{Ex}}}\mathbf{z}_i$ and because of equations (14), (15), we can conclude $[\bar{\mathbf{z}}]_p\geq[\bar{\mathbf{x}}(T)]_p$ for $p\in\mathcal{P}^{\text{Ex}}$. Hence, by the stochastic definition of extended function F given by equation (5), we have $F(\bar{\mathbf{x}})\leq F(\bar{\mathbf{z}})$ and consequently

$$\alpha(1-\frac{1}{\mathrm{e}^{1-\sqrt[\kappa_{\max}]{1-\gamma}}})f(\mathcal{R}^\star)\leq F(\bar{\mathbf{z}}), \quad (32)$$

By the stochastic interpretation of the extended function (5), we can write

$$F(\bar{\mathbf{z}})=\mathbb{E}[f(\mathcal{R}_{\bar{\mathbf{z}}})]. \quad (33)$$

By decomposing $\bar{\mathbf{z}}$ to sub-agent level components, we can write

$$\mathbb{E}[\mathcal{R}_{\bar{\mathbf{z}}}]=\mathbb{E}[f(\bigcup_{a\in\mathcal{A}}\bigcup_{i\in\mathcal{A}_a}\mathcal{R}_{\bar{\mathbf{z}}_i})]. \quad (34)$$

Moreover, for a random set $\mathcal{S}\in\mathcal{P}^{\text{Ex}}$ and the random set $\mathcal{R}_{\mathbf{z}_i}=\{p_1,\cdots,p_m\}\subset\mathcal{P}_i^{\text{Ex}}$, $i\in\mathcal{A}^{\text{Ex}}$, we have

$$\begin{aligned}&\mathbb{E}[f(\mathcal{S}\cup\mathcal{R}_{\mathbf{z}_i})]=\mathbb{E}[f(\mathcal{S})+\sum_{k=1}^m\Delta_f(p_k|\mathcal{S}\cup\{p_1,\cdots,p_{l-1}\})]\\&\leq\mathbb{E}[f(\mathcal{S})+\sum_{p\in\mathcal{R}_{\mathbf{z}_i}}\Delta_f(p|\mathcal{S})]=\mathbb{E}[f(\mathcal{S})+\sum_{p\in\mathcal{P}_i^{\text{Ex}}}\Delta_f(p|\mathcal{S})]\\&=\sum_{p\in\mathcal{P}_i^{\text{Ex}}}[\mathbf{z}_i]_pf(\mathcal{S}\cup\{p\}). \quad (35)\end{aligned}$$

Since $\mathbf{1}.\mathbf{z}_i=1$ and $[\mathbf{z}_i]_p\geq0, p\in\mathcal{P}_i^{\text{Ex}}$, then the expression $\sum_{p\in\mathcal{P}_i^{\text{Ex}}}[\mathbf{z}_i]_pf(\mathcal{S}\cup\{p\})$ is equivalent to $\mathbb{E}[f(\mathcal{S}\cup\{p\})]$ when the policy p is chosen randomly according to belief vector $\mathbf{z}_i$. Hence, by referring to equations (34) and (35), we have

$$\mathbb{E}[\mathcal{R}_{\bar{\mathbf{z}}}]\leq\mathbb{E}[f(\bigcup_{a\in\mathcal{A}}\bigcup_{i\in\mathcal{A}_a}p_i]=\mathbb{E}[f(\bigcup_{a\in\mathcal{A}}\mathcal{T}_a],$$

where p_i is the randomly selected policy according to belief vector $\mathbf{z}_i$ and $\mathcal{T}_a=\bigcup_{i\in\mathcal{A}_a}p_i$ is the set of randomly selected policies of sub-agents of agent a. Having that $f(\mathcal{T}_a)=f(\bar{\mathcal{R}}_a)$ where $\bar{\mathcal{R}}_a=\{\mathsf{PolicyMap}(p)|p\in\mathcal{T}_a\}$, and using the equations (32) and (33), we can write

$$\alpha(1-\frac{1}{\mathrm{e}^{1-\sqrt[\kappa_{\max}]{1-\gamma}}})f(\mathcal{R}^\star)\leq\mathbb{E}[f(\bigcup_{a\in\mathcal{A}}\mathcal{T}_a]=f(\bar{\mathcal{R}}), \quad (36)$$

with the probability of at least $\left(\prod_{i\in\mathcal{A}^{\text{Ex}}}(1-2\mathrm{e}^{-\frac{1}{8\lambda^2}K_i})^{|\mathcal{P}_i^{\text{Ex}}|}\right)^T$, which concludes the proof. ■