Highlights

Bayesian Generalized Network Design ⋆

Yuval Emek, Shay Kutten, Ron Lavi, Yangguang Shi

- Studying the problem of Bayesian Generalized Network Design (BGND), where agents only have partial information regarding the global system configuration, and resources have superlinear cost functions.
- Proposing a new metric, Bayesian competitive ratio (BCR), which evaluates the result by comparing it with the output of an omnipotent algorithm that has a global view and unlimited computational resources.
- Developing a fully combinatorial framework that has a constant BCR only depending on the exponents of the cost functions of resources, and runs in strongly polynomial time.
- Analyzing the proposed framework with the smoothness toolbox for Bayesian games.
- Designing a polynomial-time procedure with the double-sided guarantee for computing the estimation of the expected cost shares over exponentially many possibilities in a Bayesian game with nonlinear costs.

Bayesian Generalized Network Design

Yuval Emek[1,*], Shay Kutten[2,*], Ron Lavi[3,*], Yangguang Shi[4,*]

Faculty of Industrial Engineering and Management, Technion, Haifa, Israel

Abstract

We study network coordination problems, as captured by the setting of *generalized network design* (Emek et al., STOC 2018), in the face of uncertainty resulting from partial information that the network users hold regarding the actions of their peers. This uncertainty is formalized using Alon et al.'s *Bayesian ignorance* framework (TCS 2012). While the approach of Alon et al. is purely combinatorial, the current paper takes into account computational considerations: Our main technical contribution is the development of (strongly) polynomial time algorithms for local decision making in the face of Bayesian uncertainty.

Keywords: Bayesian competitive ratio, Bayesian ignorance, generalized network design, diseconomies of scale, energy consumption, smoothness, best response dynamics

[☆]An extended abstract of this paper is to appear in the 27th Annual European Symposium on Algorithms (ESA 2019).

[*]Corresponding author

Email addresses: `yemek@ie.technion.ac.il` (Yuval Emek), `kutten@ie.technion.ac.il` (Shay Kutten), `ronlavi@ie.technion.ac.il` (Ron Lavi), `shiyangguang@ie.technion.ac.il` (Yangguang Shi)

[1]The work of Yuval Emek was supported in part by an Israeli Science Foundation grant number 1016/17.

[2]The work of Shay Kutten was supported in part by a grant from the ministry of science in the program that is joint with JSPS and in part by the BSF.

[3]The work of Ron Lavi was partially supported by the ISF-NSFC joint research program (grant No. 2560/17).

[4]The work of Yangguang Shi was partially supported at the Technion by a fellowship of the Israel Council for Higher Education.

1. Introduction

In real-life situations, network users are often required to coordinate actions for performance optimization. This challenging coordination task becomes even harder in the face of *uncertainty*, as users often act with partial information regarding their peers. Can users overcome their *local* views and reach a good *global* outcome? How far would this outcome be from optimal?

For a formal treatment of the aforementioned questions, we adopt the *Bayesian ignorance* framework of Alon et al. [1]. Consider N *agents* in a *routing* scenario, where each agent $i \in [N]$ should decide on a (u_i, v_i)-path a_i in the network with the objective of minimizing some global *cost function* that depends on the links' load. The (u_i, v_i) pair, also referred to as the *type* of agent i, is drawn from a distribution p_i. All agents know this distribution, but the actual realization (u_i, v_i) of each agent i is only known to i herself.

Our goal is to construct a *strategy* for each agent i that determines her action a_i based only on her individual type (u_i, v_i). These strategies are computed in a "preprocessing stage" and the actual decision making happens in real-time without further communication. We measure the quality of a tuple of strategies in terms of its *Bayesian competitive ratio (BCR)* defined as the ratio of the expected cost obtained by these strategies to that of an optimal solution computed by an omnipotent algorithm (refer to Section 1.1.1 for the exact definition). To the best of our knowledge, this algorithmic evaluation measure has not been studied so far.

Our main technical contribution is a generic framework that yields strongly polynomial-time algorithms constructing agent strategies with low BCR for *Bayesian generalized network design (BGND)* problems — a setting that includes routing and many other network coordination problems. Our framework assumes cost functions that exhibit *diseconomy of scale (DoS)* [2, 3, 4], capturing the power consumption of network devices that employ the popular *speed scaling* technique.

1.1. Model

For clarity of the exposition, we start with the special case of Bayesian routing in Section 1.1.1 and then present the more general BGND setting in Section 1.1.2. Conceptually, the new algorithmic problem of Bayesian routing that we define here is related to oblivious routing [5, 6, 7], where routing requests should be performed without any knowledge about actual network traffic. This means that the routing path chosen for a routing request may only depend on the network structure and the other parameters of the problem. Oblivious algorithms are attractive as they can be implemented very efficiently in a distributed environment as they base routing decisions only on local knowledge. As will become formally clear below, Bayesian routing has a similar flavor, but with an important additional ingredient. We will assume that the algorithm is equipped with statistical ("Bayesian") knowledge about network traffic. Thus, in a sense, we replace internal randomization techniques, that oblivious routing usually employs, with actual data, while still being oblivious to other actual routing decisions and thus still maintaining the locality principle.[5]

1.1.1. Special Case: Bayesian Routing

In the *full information* variant of the *routing* problem, we are given a (directed or undirected) graph $G = (V, E)$ and a set of N agents, where each agent $i \in [N]$ is associated with a node pair $(u_i, v_i) \in V \times V$, referred to as the (routing) *request* of agent i. This request should be satisfied by choosing some (u_i, v_i)-path in G, referred to as the (feasible) *action* of agent i, and the collection of all such paths is denoted by A_i.

Let $A = A_1 \times \cdots \times A_N$ be the collection of all *action profiles*. The *load* on edge $e \in E$ with respect to action profile $a \in A$, denoted by l_e^a, is defined to be the number of agents whose actions include e, that is, $l_e^a = |\{i \in [N] : e \in a_i\}|$. The cost incurred by load l_e^a on edge e is determined by an (edge specific) *superadditive* cost function $F_e : \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ such that for any $l \geq 0$,

$$F_e(l) \; = \; \xi_e \cdot l^\alpha \, , \tag{1}$$

where $\xi_e > 0$ (a.k.a. the *speed scaling factor*) is a parameter of edge e and $\alpha > 1$ (a.k.a. the *load exponent*) is a global constant parameter. Such a superadditive cost function captures, for example, the power consumption of network devices employing the popular *speed scaling* technique [8, 9, 10, 11, 12, 13] that allows the device to adapt its power level to its actual load. In particular, for those network devices that employ the speed scaling technique, the value of α generally satisfies $1 < \alpha \leq 3$ [14, 15]. Another application of the cost function (1) with $\alpha = 2$ is to model the queuing delay of users in a TCP/IP communication networks [16]. The goal in the (full information)

[5]This is different from stochastic network design as these algorithms are not oblivious. More details are given below.

routing problem is to construct an action profile $a \in A$ with the objective of minimizing the total cost $C(a) = \sum_{e \in E} F_e(l_e^a)$.

Extending to Partial Information. In the current paper, we extend the full information routing problem to the *Bayesian routing* problem, where the request of agent $i \in [N]$ is not fully known to all other agents. In this problem variant, agent $i \in [N]$ is associated with a set T_i of *types* so that each type $t_i \in T_i$ specifies its own routing request $(u_i^{t_i}, v_i^{t_i}) \in V \times V$. Let $A_i^{t_i}$ be the set of all (feasible) actions for (the request of) type t_i, namely, all $(u_i^{t_i}, v_i^{t_i})$-paths in G and let $A_i = \bigcup_{t_i \in T_i} A_i^{t_i}$.

Agent i is also associated with a *prior distribution* p_i over the types in T_i and the crux of the Bayesian routing problem is that agent i should decide on her action while knowing the realization of her own prior distribution p_i (that is, the routing request she should satisfy) but without knowing the realizations of the prior distributions of the other agents $j \neq i$. Formally, let $T = T_1 \times \cdots \times T_N$ be the collection of *type profiles* and $A = A_1 \times \cdots \times A_N$ be the collection of *action profiles.* The set of (feasible) action profiles for a type profile $t \in T$ is denoted by $A^t = A_1^{t_1} \times \cdots \times A_N^{t_N}$ and the prior distribution over the type profiles in T is denoted by p. In this paper, p is assumed to be a product distribution, i.e., the probability of type profile $t \in T$ is $p(t) = \prod_{i=1}^N p_i(t_i)$.

The goal in the Bayesian routing problem is to construct for each agent $i \in [N]$, a *strategy* $s_i : T_i \mapsto A_i$ that maps agent i's realized type $t_i \in T_i$ to an action $a_i \in A_i^{t_i}$. We emphasize that the decision of agent i is taken irrespective of the other agents' realized types which are not (fully) known to agent i. Intuitively, a strategy s_i can be viewed as a *lookup table* constructed in the "preprocessing stage", and queried at real-time to determine a (fixed) path for every (u_i, v_i) pair associated with i (cf. *oblivious routing* [17, 7]).

The set of strategies available for agent i is denoted by S_i and $S = S_1 \times \cdots \times S_N$ denotes the set of *strategy profiles.* For each type profile $t \in T$, the strategy profile $s \in S$ determines an action profile $a = s(t) \in A$ defined so that $a_i = s_i(t_i)$, $i \in [N]$. Using this notation, the objective in the Bayesian routing problem is to construct a strategy profile $s \in S$ that minimizes the total cost

$$C(s) \; = \; \mathbb{E}_{t \sim p}\left[\sum_{e \in E} F_e\left(l_e^{s(t)}\right)\right] .$$

Bayesian Competitive Ratio. Consider an algorithm $\mathcal{A}$ that given a Bayesian routing instance, constructs a strategy profile s. To evaluate the performance of $\mathcal{A}$, we compare the total cost $C(s)$ to $\mathbb{E}_{t\sim p}[\mathtt{OPT}(t)]$, where

$$\mathtt{OPT}(t) \;=\; \min_{a\in A^t}\sum_{e\in E} F_e(l_e^a)$$

is the cost of an optimal action profile for the type profile $t \in T$. This can be regarded as the expectation, over the same prior distribution p, of the total cost incurred by an omnipotent algorithm that has a global view of the whole type profile t and enjoys unlimited computational resources. The *Bayesian competitive ratio (BCR)* of algorithm $\mathcal{A}$ is the smallest $\beta \geq 1$ such that for every Bayesian routing instance, the strategy profile s constructed by $\mathcal{A}$ satisfies $C(s) \leq \beta \cdot \mathbb{E}_{t\sim p}[\mathtt{OPT}(t)]$.

Alon et al. [1] introduced the related criterion of *Bayesian ignorance* defined as $\frac{C(s^*)}{\mathbb{E}_{t\sim p}[\mathtt{OPT}(t)]}$, where $s^* = \mathrm{argmin}_{s\in S}\, C(s)$ is an optimal strategy profile for the given instance. This criterion quantifies the implication of the agents' partial knowledge regarding the global system configuration, irrespective of the computational complexity of constructing this optimal strategy profile. By definition, for any strategy profile $s \in S$,

$$C(s) \;=\; \mathbb{E}_{t\sim p}\Big[\sum_{e\in E} F_e\big(l_e^{s(t)}\big)\Big] \;\geq\; \mathbb{E}_{t\sim p}\Big[\min_{a\in A^t}\sum_{e\in E} F_e(l_e^a)\Big]$$

which implies that the Bayesian ignorance is at least 1. Notice that the BCR is equivalent to the product of the *approximation ratio* $\frac{C(s)}{C(s^*)}$ and the Bayesian ignorance, therefore it evaluates the loss caused by both algorithmic (computational complexity) considerations and the absence of the global information. The first contribution of the current paper is cast in the following theorem.

Theorem 1.1. *For the Bayesian routing problem, there exists an algorithm whose BCR depends only on the load exponent parameter α. This algorithm is fully combinatorial and runs in strongly polynomial time.*

We emphasize that the BCR of the algorithm promised in Theorem 1.1 is independent of the number of agents N, the underlying graph G, the speed scaling factors ξ_e, $e \in E$, and the probability distribution p. Therefore, as α is assumed to be a constant, so is the BCR.

1.1.2. Bayesian Generalized Network Design

Generalized Network Design. The (full information) routing problem has recently been generalized by Emek et al. [18] to the wider family of *generalized network design (GND)* problems. In its full information form (the form considered in [18]), a GND instance is defined over N agents and a set E of *resources*. Each agent $i \in [N]$ is associated with an abstract (not necessarily routing) request characterized by a set $A_i \subseteq 2^E$ of (feasible) actions out of which , some action $a_i \in A_i$ should be selected. As in the routing case, the action profile $a = (a_1, \ldots, a_N)$ induces a load of $l_e^a = |\{i \in [N] : e \in a_i\}|$ on each resource $e \in E$ that subsequently incurs a cost of $F_e(l_e^a)$, where $F_e : \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ is a resource specific cost function. The goal is to construct an action profile $a \in A = A_1 \times \cdots \times A_N$ with the objective of minimizing the total cost $C(a) = \sum_{e \in E} F_e(l_e^a)$.

The request of agent $i \in [N]$ is said to be *succinctly represented* [18] if its corresponding action set A_i can be encoded using poly$(|E|)$ bits. Identifying the resource set E with the edge set of an underlying graph G, the routing requests defined in Section 1.1.1 are clearly succinctly represented since each A_i corresponds to the set of (u_i, v_i)-paths in G, hence A_i can be encoded by specifying u_i and v_i (and G). Other examples for succinctly represented requests, where the resource set E is identified with the edge set of an underlying (directed or undirected) graph $G = (V, E)$, include:

- *multi-routing* requests in directed or undirected graphs, where given a collection $D_i \subseteq V \times V$ of *terminal* pairs, the action set A_i consists of all edge subsets $F \subseteq E$ such that the subgraph (V, F) admits a (u, v)-path for every $(u, v) \in D_i$; and

- *set connectivity* (resp., *set strong connectivity*) in undirected (resp., directed) graphs, where given a set $T_i \subseteq V$ of *terminals*, the action set A_i consists of all edge subsets $F \subseteq E$ that induce on G a connected (resp., strongly connected) subgraph that spans T_i.

All requests mentioned (implicitly or explicitly) hereafter are assumed to be succinctly represented.

Bayesian GND. In the current paper, we extend the (full information) GND setting to *Bayesian GND (BGND)*. This extension is analogous to the extension of full information routing to Bayesian routing as defined in Section 1.1.1. In particular, agent $i \in [N]$ is now associated with a set T_i of types, where each type $t_i \in T_i$ corresponds to a request whose action set is denoted by $A_i^{t_i}$, and a prior distribution p_i over the types in T_i. A strategy s_i of agent i is a function that maps the agent's realized type $t_i \in T_i$ to an action $s_i(t_i) \in A_i^{t_i}$.

Similarly to the notation introduced in Section 1.1.1, let $T = T_1 \times \cdots \times T_N$ be the set of type profiles. Let $A_i = \bigcup_{t_i \in T_i} A_i^{t_i}$ and let $A = A_1 \times \cdots \times A_N$ be the set of action profiles. Let S_i be the set of strategies available for agent i and let $S = S_1 \times \cdots \times S_N$ be the set of strategy profiles. Given a strategy profile $s \in S$ and a type profile $t \in T$, let $a = s(t) \in A$ be the action profile defined so that $a_i = s_i(t_i)$, $i \in [N]$. The goal in the BGND problem is to construct a strategy profile $s \in S$ with the objective of minimizing the total cost

$$C(s) \;=\; \mathbb{E}_{t \sim p}\left[\sum_{e \in E} F_e\left(l_e^{s(t)}\right)\right] . \tag{2}$$

The BCR of Algorithm $\mathcal{A}$ is the smallest $\beta \geq 1$ such that for every BGND instance, the strategy profile $s \in S$ constructed by $\mathcal{A}$ satisfies $C(s) \leq \beta \cdot \mathbb{E}_{t \sim p}[\mathtt{OPT}(t)]$, where

$$\mathtt{OPT}(t) \;=\; \min_{a \in A^t} \sum_{e \in E} F_e(l_e^a) .$$

Generalized Cost Functions. In addition to the generalization of (full information) routing to GND, [18] also generalizes the cost functions defined in Eq. (1) to cost functions of the form

$$F_e(l) \;=\; \sum_{j \in [q]} \xi_{e,j} \cdot l^{\alpha_j} , \tag{3}$$

where q is a positive integer, $\xi_{e,j}$ is a positive real for every $e \in E$ and $j \in [q]$, and α_j is a constant real no smaller than 1 for every $j \in [q]$.[6] We define $\alpha_{\max} = \max_{j \in [q]} \alpha_j$ and assume hereafter that $\alpha_{\max} > 1$. As discussed in [18], this generalization of Eq. (3) is not only interesting from a theoretical perspective, but also makes the model more applicable to practical network energy saving applications. Indeed, in realistic communication networks, a link often consists of several different devices (e.g., transmitter/receiver, amplifier, adapter), all of which are operating when the link is in use. As

[6]The cost functions considered in [18] have a fixed additional term, capturing the resource's *startup cost*, that makes them even more general. Due to technical difficulties, in the current paper we were not able to cope with this additional term.

their energy consumption can vary in terms of the load exponents and speed scaling factors [15], Eq. (3) may often provide a more accurate abstraction of the actual link's power consumption.

Action Oracles. For a BGND problem $\mathcal{P}$, this paper develops a framework which generates an algorithm with BCR $O(\varrho^{\alpha_{\max}})$ when provided with an *action ϱ-oracle* for $\mathcal{P}$. An action ϱ-oracle with parameter $\varrho \geq 1$ for BGND problem $\mathcal{P}$ (cf. the *reply ϱ-oracles* of [18]) is a procedure that given agent $i \in [N]$, type $t_i \in T_i$, and a *weight* vector $w \in \mathbb{R}^E_{\geq 0}$, generates an action $a_i \in A_i^{t_i}$ such that $\sum_{e \in a_i} w(e) \leq \varrho \cdot \sum_{e \in a_i'} w(e)$ for any action $a_i' \in A_i^{t_i}$. An *exact action oracle* is an action ϱ-oracle with parameter $\varrho = 1$.

Notice that the optimization problem behind the action oracle is *not* a BGND problem: It deals with a *single* type of a *single* agent and the role of the resource cost functions is now taken by the weight vector. These differences often make it possible to implement the action oracle with known (approximation) algorithms.

For example, the Bayesian routing problem, which requires paths between the given node pairs, admit an exact action oracle implemented using, e.g., Dijkstra's shortest path algorithm [19, 20]. In contrast, the BGND problem with set connectivity requests in undirected graphs (P1), the BGND problem with set strong connectivity requests in directed graphs (P2), the BGND problem with multi-routing requests in undirected graphs (P3), and the BGND problem with multi-routing requests in directed graphs (P4) do not admit exact action oracles unless P = NP as these would imply exact (efficient) algorithms for the *Steiner tree*, *strongly connected Steiner subgraph*, *Steiner forest*, and *directed Steiner forest* problems, respectively. However, employing known approximation algorithms for the latter (Steiner) problems, one concludes that BGND problem (P1) admits an action ϱ-oracle for $\varrho \leq 1.39$ [21]; BGND problem (P2) admits an action ν^ϵ-oracle, where ν is the number of terminals [22]; BGND problem (P3) admits an action 2-oracle [23]; and BGND problem (P4) admits an action $k^{1/2+\epsilon}$-oracle, where k is the number of terminal pairs [24]. This means, in particular, that BGND problems (P1) and (P3) always admit an action ϱ-oracle with a constant approximation ratio ϱ, whereas BGND problems (P2) and (P4) admit such an oracle when ν and k are fixed [23, 22, 24, 21]. The guarantees of our approximation framework are cast in the following theorem.

Theorem 1.2. *Consider a BGND problem $\mathcal{P}$ with an action ϱ-oracle $\mathcal{O}_{\mathcal{P}}$. When provided access to $\mathcal{O}_{\mathcal{P}}$, the framework proposed in this paper generates* *an algorithm $\mathcal{A}_{\mathcal{P}}$ whose BCR depends only on the load exponent parameters $\alpha_1, \ldots, \alpha_q$ of Eq. (3). This framework is fully combinatorial and runs in strongly polynomial time, hence if $\mathcal{O}_{\mathcal{P}}$ can be implemented to run in strongly polynomial time, then so can $\mathcal{A}_{\mathcal{P}}$.*

Again, we emphasize that the BCR of the algorithm promised in Theorem 1.2 is independent of the number of agents N, the number of resources $|E|$, the speed scaling factors $\xi_{e,j}$, $j \in [q]$, $e \in E$, and the probability distribution p. Therefore, as $\alpha_1, \ldots, \alpha_q$ are assumed to be constants, so is the BCR. Since the Bayesian routing problem admits an exact action oracle, Theorem 1.1 follows trivially from Theorem 1.2. Throughout the remainder of this paper, we focus on the BGND framework promised in Theorem 1.2.

1.2. Related Works

The technical framework that we use is inspired by [18]. Section 3 gives a detailed technical overview including a full comparison.

In the full information case, network design problems with superadditive cost functions as defined in Eq. (1) have been extensively studied with the motivation of improving the energy efficiency of networks [2, 3, 4]. To the best of our knowledge, none of these studies has been extended to the Bayesian case.

In the research works on oblivious routing (e.g., [6, 7, 25, 26]), the absence of global information in routing is modeled in an adversarial (non-Bayesian) manner. In particular, oblivious routing assumes that no knowledge about t_{-i} is available when determining every a_i, and the performance of the algorithm is evaluated by means of its *competitive ratio* $\max_{t \in T} \frac{\sum_{e \in E} F_e(l_e^{s(t)})}{\mathtt{OPT}(t)}$. For the cost function $F_e(l) = l^\alpha$ with $\alpha > 1$, Englert and Räcke [6] propose an $O(\log^\alpha |V|)$-competitive oblivious routing algorithm for the scenario where the traffic requests are allowed to be partitioned into fractional flows. Shi et al. [7] prove that for such a cost function, there exists no oblivious routing algorithm with competitive ratio $O\left(|E|^{\frac{\alpha-1}{\alpha+1}}\right)$ when it is required to choose an integral path for every request.

The Bayesian approach is often used in the game theoretic literature to model the uncertainty a player experiences regarding the actions taken by the other players. Roughgarden [27] studies a *routing game* (among other things) in which the players share (equally) the cost of the edges they use and proposes a theoretical tool called *smoothness* to analyze the *price of anarchy*

(PoA) of this game in a Bayesian setting, defined as $\frac{\max_{s \in S^{\mathrm{BNE}}} C(s)}{\mathbb{E}_{t \sim p}[\mathtt{OPT}(t)]}$, where S^{BNE} denotes the set of *Bayes-Nash equilibria*. In particular, he proves that with the cost function $F_e(l) = \xi_{e,1} \cdot l + \xi_{e,2} \cdot l^2$, the PoA is bounded by $\frac{5}{2}$. We employ the smoothness toolbox in our algorithmic construction, as further described in Section 6 (see also the overview in Section 3).

Alon et al. [1] investigate the Bayesian routing game with a constant cost function $F_e = \xi_e$ and prove that the Bayesian ignorance $\frac{C(s^*)}{\mathbb{E}_{t \sim p}[\mathtt{OPT}(t)]}$ is bounded by $O(N)$ (resp., $O(\log |E|)$) in directed (resp., undirected) graphs $G = (V, E)$. They also introduce game theoretic variants of the Bayesian ignorance notion and analyze them in that game.

To deal with the inherent uncertainty of the demand in realistic networks, many research works have been conducted on *stochastic network design* [28, 29, 30], formulated as a *two-stage stochastic optimization* problem: in the first stage, each link in the network has a fixed cost and the algorithm needs to make decisions to purchase links knowing the probability distribution over the network demands; in the second stage, the network demands are realized (according to the aforementioned probability distribution) and should be satisfied, which may require purchasing additional links, this time with an inflated cost. The objective is to minimize the total cost of the two stages plus a load dependent term, in expectation.

The BGND setting considered in the current paper is different from two-stage stochastic optimization (particularly, stochastic network design) in several aspects, the most significant one is that in BGND, an agent's strategy should dictate her "complete action" (e.g., a path for routing requests) for every possible type, obliviously of the realized types of the other agents. In particular, one cannot "update" the agents' actions and purchase additional resources at a later stage to satisfy the realized demands. Moreover, the current paper evaluates the performance of a BGND algorithm by means of its BCR that takes into consideration computational complexity limitations as well as the lack of global information (see Section 1.1) whereas the literature on two-stage stochastic optimization typically evaluates algorithms using standard approximation guarantees that accounts only for computational complexity limitations.

In [31], Garg et al. investigate online combinatorial optimization problems where the requests arriving online are drawn independently and identically from a known distribution. As an example, Garg et al. [31] study the online Steiner tree problem on an undirected graph $G = (V, E)$. In this problem, at each step the algorithm receives a terminal that is drawn independently from a distribution over V, and needs to maintain a subset of edges connecting all the terminals received so far.

Our work differs from [31] in following four aspects. First, in the stochastic online optimization problem studied in [31], when each request i arrives, the previous requests $\{1, \cdots, i-1\}$ have been realized, and the realization is known. By contrast, in the BGND problem, every agent i needs to be served without knowing the actual realization of the other agents. Second, the cost function studied in [31] maps each resource e to a fixed toll, which is subaddtive in the number of requests using e, while our cost function is superaddtive. Third, in the BGND problem with the set connectivity requests, for each agent i, each type t_i is a set of terminals rather than a single terminal, and each action in $A_i^{t_i}$ is a Steiner tree spanning over the set of terminals corresponding to t_i. Fourth, in the BGND problem, each prior distribution p_i is over the types of agent i, while there is no distribution over the agents.

1.3. Paper Organization

The rest of this paper is organized as follows. Section 2 introduces some of the concepts employed in our approximation framework together with some notation and terminology. The main challenges that we had to overcome when developing this framework and some of the techniques used for that purpose are discussed in Section 3. Section 4 is dedicated to a detailed exposition of our approximation framework. Its performance is then analyzed in Section 5 using certain game theoretic properties which are investigated in Sections 6–8.

2. Preliminaries

We follow the common convention that for an N-tuple $x = (x_1, \ldots, x_N)$ and for $i \in [N]$, the notation x_{-i} denotes the $(N-1)$-tuple $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_N)$. Likewise, for a Cartesian product $X = X_1 \times \cdots \times X_N$ and for $i \in [N]$, the notation X_{-i} denotes the Cartesian product $X_1 \times \cdots \times X_{i-1} \times X_{i+1} \times \cdots \times X_N$.

2.1. The BGND Game

Given an instance $\mathcal{I} = \left\langle N, E, \{T_i, p_i\}_{i\in[N]}, \{\xi_{e,j}\}_{e\in E, j\in[q]}, \{\alpha_j\}_{j\in[q]} \right\rangle$ of a BGND problem $\mathcal{P}$, we define a *BGND game* by associating every agent $i \in [N]$ with a strategic player who decides on the strategy s_i with the objective of minimizing her own individual cost defined as follows. Given an action profile $a \in A$ and a resource $e \in E$, the corresponding cost $F_e(l_e^a)$ is equally divided among the players $i \in [N]$ satisfying $e \in a_i$; in other words, the *cost share* of player i in resource e under action profile a, denoted by $f_{i,e}(a)$, is defined to be

$$f_{i,e}(a) \;=\; \begin{cases} 0\,, & e \notin a_i \\ \frac{F_e(l_e^a)}{|i:e\in a_i|} = \sum_j \xi_{e,j}\,(l_e^a)^{\alpha_j - 1}\;, & \text{otherwise} \end{cases}.$$

Informally, the individual cost of player i is the sum of her cost shares over all resources.

For a more formal treatment of the BGND game, we occasionally need to explicitly specify the type t_i of player i in the expressions involving her cost share in which case we use the notation $f_{i,e}(t_i; a)$, following the convention that $f_{i,e}(t_i; a) = f_{i,e}(a)$ if $a_i \in A_i^{t_i}$; and $f_{i,e}(t_i; a) = \infty$ otherwise. The individual cost of a player i with respect to the type t_i and a fixed action profile a is defined as $C_i(t_i; a) = \sum_{e \in E} f_{i,e}(t_i; a)$. Correspondingly, for each player $i \in [N]$ and each type $t_i \in T_i$, we define the *type-specified* expected individual cost

$$C_i(t_i; s) \;=\; \mathbb{E}_{t_{-i} \sim p_{-i}}\left[C_i(t_i; s(t_i, t_{-i}))\right].$$

The objective function that player i wishes to minimize is her *type-averaged* expected individual cost

$$C_i(s) \;=\; \mathbb{E}_{t_i \sim p_i}\left[C_i(t_i; s)\right],$$

irrespective of the total cost $C(s)$, often referred to as the *social cost*.

Observation 2.1. *The social cost satisfies $C(s) = \sum_{i \in [N]} C_i(s)$ for every strategy profile $s \in S$.*

Let $\mathfrak{f}_{i,e}(a_i; s_{-i}) = \mathbb{E}_{t_{-i} \sim p_{-i}}[f_{i,e}(a_i, s_{-i}(t_{-i}))]$ be the expected cost share of player $i \in [N]$ on resource $e \in E$ with respect to action $a_i \in A_i$ and strategy profile $s_{-i} \in S_{-i}$. Fixing $a_{-i} \in A_{-i}$ (resp., $s_{-i} \in S_{-i}$), the cost share $f_{i,e}(a_i, a_{-i})$ (resp., expected cost share $\mathfrak{f}_{i,e}(a_i; s_{-i})$) of player i on resource e is the same for every action $a_i \in A_i$ such that $e \in a_i$. Therefore, it is often convenient to ignore the specifics of action a_i and use the notations

$f_{i,e}(+, a_{-i})$ and $\mathfrak{f}_{i,e}(+; s_{-i})$ instead of $f_{i,e}(a_i, a_{-i})$ and $\mathfrak{f}_{i,e}(a_i; s_{-i})$, respectively, given that $e \in a_i$.[7]

2.2. Definitions for the Algorithm Design and Analysis

The following definitions play key roles in the design and analysis of our approximation framework.

Definition (Choice Function [27])**.** A *choice function* $\sigma : T \mapsto A$ maps every type profile $t \in T$ to an action profile $a \in A^t$. The action specified by σ for player $i \in [N]$ with respect to type profile t is denoted by $\sigma_i(t)$. In particular, the choice function that maps each type profile t to an action profile that realizes $\texttt{OPT}(t)$ is denoted by σ^*.

Definition (Smoothness [27])**.** Given parameters $\lambda > 0$ and $0 < \mu < 1$, a BGND game is said to be (λ, μ)-*smooth* if

$$\sum_{i \in [N]} C_i(t_i; (\sigma_i^*(t), a_{-i})) \;\leq\; \lambda \cdot \texttt{OPT}(t) + \mu \cdot \sum_{i \in [N]} C_i(t_i', a)$$

for every type profiles $t, t' \in T$ and action profile $a \in A^{t'}$.

Definition (Potential Function)**.** A function $\Phi : S \mapsto \mathbb{R}_{\geq 0}$ is said to be a *potential function* of the BGND game if

$$\Phi(s) - \Phi(s_i', s_{-i}) \;=\; C_i(s) - C_i(s_i', s_{-i})$$

for every strategy profile $s \in S$, player $i \in [N]$, and strategy $s_i' \in S_i$. The potential function $\Phi(\cdot)$ is said to be K-*bounded* for a parameter $K \geq 1$ if $\Phi(s) \leq C(s) \leq K \cdot \Phi(s)$ for every strategy profile $s \in S$.

Definition ($(\underline{\eta}, \overline{\eta})$-Estimation)**.** Given real parameters $\underline{\eta}, \overline{\eta} \geq 1$, a value x is said to be an $(\underline{\eta}, \overline{\eta})$-*estimation* of the expected cost share $\mathfrak{f}_{i,e}(a_i; s_{-i})$ (resp., $\mathfrak{f}_{i,e}(+; s_{-i})$) if it satisfies $x/\underline{\eta} \leq \mathfrak{f}_{i,e}(a_i; s_{-i}) \leq x \cdot \overline{\eta}$ (resp., $x/\underline{\eta} \leq \mathfrak{f}_{i,e}(+; s_{-i}) \leq x \cdot \overline{\eta}$). We typically denote this estimation x by

[7] To avoid ambiguity concerning the definition of $f_{i,e}(+, a_{-i})$ and $\mathfrak{f}_{i,e}(+; s_{-i})$ for resources $e \notin A_i$, we assume (in the scope of using these notations) that $A_i = E$ for all $i \in [N]$. This is without loss of generality as one can augment T_i with a virtual type $\tilde{t}_i$ such that $A_i^{\tilde{t}_i} = \{E\}$ and $p_i(\tilde{t}_i)$ is arbitrarily small.

$\widehat{\mathfrak{f}}_{i,e}(a_i; s_{-i})$ (resp., $\widehat{\mathfrak{f}}_{i,e}(+; s_{-i})$). The BGND game is said to be *poly-time* $(\underline{\eta}, \overline{\eta})$-*estimable* if for every player $i \in [N]$ and strategy profile $s_{-i} \in S_{-i}$, there exists an algorithm which runs in time $\text{poly}(N, q, |T_1|, \cdots, |T_N|)$ and outputs an $(\underline{\eta}, \overline{\eta})$-estimation of the expected cost share $\mathfrak{f}_{i,e}(+; s_{-i})$. The BGND game is said to be *tractable* if it is poly-time $(\underline{\eta}, \overline{\eta})$-estimable with $\overline{\eta} = \underline{\eta} = 1$.

Fix some player $i \in [N]$, type $t_i \in T_i$, and $(\underline{\eta}, \overline{\eta})$-estimations $\widehat{\mathfrak{f}}_{i,e}(s_i(t_i); s_{-i})$, $e \in E$. With respect to these variables, let $\widehat{C}_i(t_i; s) = \sum_{e \in E} \widehat{\mathfrak{f}}_{i,e}(s_i(t_i); s_{-i})$ and $\widehat{C}_i(s) = \mathbb{E}_{t_i \sim p_i}[\widehat{C}_i(t_i; s)]$. By the linearity of expectation, we know that

$$\widehat{C}_i(t_i; s)/\underline{\eta} \leq C_i(t_i; s) \leq \widehat{C}_i(t_i; s) \cdot \overline{\eta} \quad \text{and} \quad \widehat{C}_i(s)/\underline{\eta} \leq C_i(s) \leq \widehat{C}_i(s) \cdot \overline{\eta} .$$

Consequently, we refer to $\widehat{C}_i(t_i; s)$ and $\widehat{C}_i(s)$ as $(\underline{\eta}, \overline{\eta})$-*estimations* of $C_i(t_i; s)$ and $C_i(s)$, respectively.

Definition (Approximate Best Response)**.** For strategy profile $s \in S$ and player $i \in [N]$, strategy $s_i \in S_i$ is said to be an *approximate best response (ABR)* of i with approximation parameter $\chi \geq 1$ if $C_i(s_i, s_{-i}) \leq \chi \cdot C_i(s'_i, s_{-i})$ holds for any $s'_i \in S_i$. We may omit the explicit mention of the approximation parameter χ when it is clear from the context. A *best response (BR)* is an ABR with approximation parameter $\chi = 1$.

Definition (Approximate Best Response Dynamics)**.** An *approximate best response dynamic (ABRD)* is a procedure that starts from a predetermined strategy profile $s^0 \in S$ and generates a series of strategy profiles $s^1, \cdots, s^R$ such that for every $1 \leq r \leq R$, there exists some player $i \in [N]$ satisfying (1) $s^r_{-i} = s^{r-1}_{-i}$; and (2) s^r_i is an ABR of i to s^{r-1}_{-i}.

3. Overview of the Main Challenges and Techniques

The approximation framework presented in Section 4 for BGND problems is inspired by the framework designed in [18] for full information GND problems only in the conceptual sense that both algorithms employ approximate best response dynamics. In a high-level, for a certain number R of rounds that will be carefully chosen in order to achieve the approximation promise, and starting from some properly chosen initial strategy profile s^0, for each round $1 \leq r \leq R$ the strategy profile s^r is generated from s^{r-1} in the following manner:

1. For every player $i \in [N]$ and resource $e \in E$, compute an $(\underline{\eta}, \overline{\eta})$-estimation $\widehat{\mathfrak{f}}_{i,e}(+; s_{-i}^{r-1})$ of the expected cost share $\mathfrak{f}_{i,e}(+; s_{-i}^{r-1})$.
2. For every player $i \in [N]$, construct the strategy s'_i by mapping each type $t_i \in T_i$ to the action $a_i \in A_i^{t_i}$ computed by invoking the action ϱ-oracle with weight vector w defined by setting $w(e) = \widehat{\mathfrak{f}}_{i,e}(+; s_{-i}^{r-1})$.
3. Choose player $i \in [N]$ according to the game theoretic criterion presented in Section 4 regarding the estimations $\widehat{C}_i(s^{r-1})$ and $\widehat{C}_i(s'_i, s_{-i}^{r-1})$ of the type-averaged expected individual costs. Construct s^r by updating the strategy of the chosen player i to s'_i.

However, beyond the similar high-level structure, the technical construction in this paper is entirely different from [18] since the incomplete information assumption of the BGND setting exhibits new algorithmic challenges that require novel techniques. Specifically, the main challenges that our technical analysis in this paper handles are as follows.

A first obstacle here is the difficulty in computing the estimation $\widehat{\mathfrak{f}}_{i,e}(+; s_{-i}^{r-1}) = \mathbb{E}_{t_{-i} \sim p_{-i}}[f_{i,e}(+, s_{-i}(t_{-i}))]$ in step 1 since there are exponentially (in N) many possibilities for t_{-i}. Another source of difficulty in this regard is that the function $f_{i,e}(+, s_{-i}(t_{-i}))$ is nonlinear in $l_e^{s_{-i}(t_{-i})}$. One may hope that Jensen's inequality [32] can resolve this issue, however, as we explain in the technical sections, it is not enough for obtaining proper bounds on both $\underline{\eta}$ and $\overline{\eta}$. This obstacle is addressed in Section 8 where we employ probabilistic tools from [33] and using Cantelli's inequality [34] to obtain the required estimation of the expression $\mathbb{E}_{t_{-i} \sim p_{-i}}[f_{i,e}(+, s_{-i}(t_{-i}))]$.

A second obstacle is that the ABRD-based approximation framework expresses its approximation guarantees in terms of smoothness parameters and bounded potential functions. However, neither the smoothness parameters nor the existence of a bounded potential function are known for the BGND game that we have defined here. We provide a new analysis for these two issues in Sections 6 and 7, respectively.

A third obstacle involves the stopping condition of the best response dynamics. A stopping condition for the full information case, via the smoothness framework, was developed by [35] (showing that if the current outcome in a best response dynamics is far from optimal there must exist a player whose best response significantly improves his own utility). For the Bayesian case, to the best of our knowledge, no such general stopping condition was known prior to the current paper. In fact, the smoothness framework for the Bayesian case which was developed in [27] did not include any results on best response dynamics. One specific technical difficulty is that Bayesian smoothness is defined in [27] w.r.t. a deviation to the optimal choice function rather than to a best response. This obstacle is resolved in Section 5 where we provide such a stopping condition by proving that if the outcome of the current step of the ABRD in the Bayesian case is far from optimal, there must exist a player whose approximate best response must significantly improve her utility.

A fourth obstacle regards the output of the algorithm, once the ABRD terminates. Although we prove that there exists at least one strategy profile s^r, $1 \leq r \leq R$, with a sufficiently small social cost $C(s^r)$, we do not know how to find it. In particular, we wish to emphasize that we cannot simply evaluate the social cost function $C(\cdot)$ (see Eq. (2)) due to the exponential number of type profiles. This obstacle does not exist in [18] where they can explicitly go over all steps of the full information ABRD and find the exact step whose outcome has minimal cost. To resolve this issue, we output the last strategy profile s^R generated in the ABRD and bound its loss. This is described in Section 5.

Our technical constructions and our analysis employ various techniques from algorithmic game theory, demonstrating once again (as in [18]) the usefulness of this literature as a toolbox for algorithmic constructions that, on the face of it, have nothing to do with selfish agents. In particular, in this paper (and as assumed in the literature on oblivious routing [5, 6, 7]), we construct an algorithm that receives a correct input and outputs routing tables that the agents are going to follow without issues of selfish deviations.

4. The Algorithm

In this part, we present an algorithm, which is referred to as `Bayes-ABRD`, for a given BGND problem $\mathcal{P}$. The algorithm is assumed to have free access to an action ϱ-oracle for $\mathcal{P}$, which is denoted by $\mathcal{O}_{\mathcal{P}}$.

With an input instance $\mathcal{I} = \left\langle N, E, \{T_i, p_i\}_{i\in[N]}, \{\xi_{e,j}\}_{e\in E, j\in[q]}, \{\alpha_j\}_{j\in[q]} \right\rangle$, the first step of the algorithm is to (conceptually) construct a BGND game, and choose a tuple of parameters $(\lambda, \mu, K, \underline{\eta}, \overline{\eta})$ such that the BGND game

1. is (λ, μ)-smooth with $\varrho(\underline{\eta}\overline{\eta})^2\mu < 1$,
2. has a potential function Φ that is K-bounded,
3. is poly-time $(\underline{\eta}, \overline{\eta})$-estimable.

The existence and exact values of the parameters in this tuple are presented in the following sections. In particular, the smoothness parameters (λ, μ) are analyzed in Section 6, the potential function is established in Section 7, and the estimation parameters $(\underline{\eta}, \overline{\eta})$ are specified in Section 8.

Lemma 4.1. *For any $i \in [N]$ and any $s_{-i} \in S_{-i}$, there exists a $\mathrm{poly}(|E|, N, q, \{|T_i|\}_{i\in[N]})$-time procedure which generates a strategy $s_i \in S_i$ and the corresponding $(\underline{\eta}, \overline{\eta})$-estimation $\widehat{C}_i(s_i, s_{-i})$ of the individual costs such that $\widehat{C}_i(s_i, s_{-i}) \le \varrho \cdot \underline{\eta} \cdot C_i(s_i', s_{-i})$ for any $s_i' \in S_i$. This means in particular that s_i is an ABR of i to s_{-i} with approximation parameter $\varrho \cdot \underline{\eta}\overline{\eta}$.*[8]

Proof. For each player $i \in [N]$, construct the weight vector $w_{i,s_{-i}} : E \to \mathbb{R}_{\ge 0}$ by setting $w_{i,s_{-i}}(e)$ to be the $(\underline{\eta}, \overline{\eta})$-estimation $\widehat{\mathfrak{f}}_{i,e}(+; s_{-i})$ of the expected share. This weight vector can be obtained in time $\mathrm{poly}(|E|, N, q, \{|T_i|\}_{i\in[N]})$ since the BGND game is poly-time $(\underline{\eta}, \overline{\eta})$-estimable. By definition, for any action $a_i' \in A_i$ satisfying $e \in a_i'$, it holds that $\mathfrak{f}_{i,e}(a_i'; s_{-i}) = \mathfrak{f}_{i,e}(+; s_{-i})$. It implies that $w_{i,s_{-i}}(e)$ can be taken as an $(\underline{\eta}, \overline{\eta})$-estimation $\widehat{\mathfrak{f}}_{i,e}(a_i'; s_{-i})$ of the expected share $\mathfrak{f}_{i,e}(a_i'; s_{-i})$.

Then, through accessing the action ϱ-oracle $\mathcal{O}_{\mathcal{P}}$ for each type $t_i \in T_i$, a strategy s_i can be found such that for any strategy $s_i' \in S_i$,

$$\begin{aligned}
\sum_{e\in E} \widehat{\mathfrak{f}}_{i,e}(s_i(t_i); s_{-i}) &= \sum_{e\in s_i(t_i)} w_{i,s_{-i}}(e) \\
&\le \varrho \cdot \sum_{e\in s_i'(t_i)} w_{i,s_{-i}}(e) \\
&\le \varrho \cdot \underline{\eta} \sum_{e\in E} \mathfrak{f}_{i,e}(s_i'(t_i); s_{-i}) \,,
\end{aligned}$$

which means that $\widehat{C}_i(t_i; (s_i, s_{-i})) \le \varrho \cdot \underline{\eta} \cdot C_i(t_i; (s_i', s_{-i}))$.

By the linearity of the expectation, $\sum_{t_i\in T_i} p_i(t_i) \sum_{e\in s_i(t_i)} w_{i,s_{-i}}(e)$ gives

[8] All subsequent occurrences of the term ABR (and ABRD) share the same approximation parameter $\varrho\underline{\eta}\overline{\eta}$, hence we may refrain from mentioning this parameter explicitly.

the desired $(\underline{\eta}, \overline{\eta})$-estimation $\widehat{C}_i(s_i, s_{-i})$, and for any $s'_i \in S_i$, it holds that

$$\begin{aligned}\widehat{C}_i(s_i, s_{-i}) &= \sum_{t_i \in T_i} p_i(t_i) \cdot \widehat{C}_i(t_i; (s_i, s_{-i})) \\ &\leq \varrho\underline{\eta} \sum_{t_i \in T_i} p_i(t_i) \cdot C_i(t_i; (s'_i, s_{-i})) \\ &\leq \varrho\underline{\eta} C_i(s'_i, s_{-i})\,.\end{aligned}$$

□

Employing the procedure promised by Lemma 4.1, `Bayes-ABRD` simulates an ABRD of at most R rounds $s^0, s^1, \ldots$ for the BGND game induced by $\mathcal{I}$. Here R is a positive integer depending on the tuple $(\lambda, \mu, K, \underline{\eta}, \overline{\eta})$, and its exact value is also deferred to the following parts (Section 5). The ABRD simulated in our algorithm is done as follows.

Each player i chooses her initial strategy s_i^0 by taking each $s_i^0(t_i)$ to be the action generated by $\mathcal{O}_{\mathcal{P}}$ for type t_i with respect to the weight vector w^0 defined by setting $w^0(e) = \sum_{j \in [q]} \xi_{e,j}$, that is, as if i is playing alone. The obtained strategy s_i^0 is broadcast by player i to all the other players such that the full strategy profile s^0 is known by every player. Assuming that s^{r-1}, $1 \leq r \leq R$, was already constructed and known by all the players, s^r is obtained as follows. Every player $i \in [N]$ employs the procedure promised by Lemma 4.1 to generate an ABR $\widehat{s}_i^{r-1}$ to s_{-i}^{r-1}, and computes $\Delta_i^r = \widehat{C}_i(s^{r-1}) - (\underline{\eta}\overline{\eta}) \cdot \widehat{C}_i(\widehat{s}_i^{r-1}, s_{-i}^{r-1})$. Both the strategy $\widehat{s}_i^{r-1}$ and the value Δ_i^r are broadcast to all the other players. If $\Delta_i^r \leq 0$ for all $i \in [N]$, then the ABRD stops, and every player i sets $s_i^r = s_i^{r-1}$; in this case, we say that the ABRD *converges*. Otherwise, fix $\Delta^r = \sum_{i \in [N]} \Delta_i^r$ and choose some player $i' \in [N]$ so that

$$\Delta_{i'}^r > 0 \quad \text{and} \quad \Delta_{i'}^r \geq \frac{1}{N}\Delta^r \tag{4}$$

to update her strategy, setting $s^r = (\widehat{s}_{i'}^{r-1}, s_{-i'}^{r-1})$ (the existence of such a player is guaranteed by the pigeonhole principle, and ties are always broken by choosing the player with the smallest index). Such an update can be performed by each player in a distributed manner, as every player has the knowledge of the full vectors $\{s_i^r\}_{i \in [N]}$ and $\{\Delta_i^r\}_{i \in [N]}$.

When the ABRD terminates (either because it has reached round $r = R$ or because it converges), `Bayes-ABRD` outputs the strategy generated in the last round.

Remark 4.2. Note that `Bayes-ABRD` is designed for computing the strategy profile, not for invoking the strategies to decide the actions in real-time. All the operations of `Bayes-ABRD`, including broadcasting the strategy $\widehat{s}_i^{r-1}$ and the value Δ_i^r for every player i in every round $r \in [R]$, are carried out in a "precomputing stage" without seeing the realized type profile. The decision making that happens in real-time does not involve any further communication.

5. Bounding the BCR with Game Theoretic Parameters

Lemma 5.1. *For every player i and every strategy profile s, if s'_i is the BR of i to s, then*

$$C_i\big(t_i; (s'_i, s_{-i})\big) \ \le\ \mathbb{E}_{t_{-i}\sim p_{-i}}\Big[C_i\big(t_i; (a_i, s_{-i}(t_{-i}))\big)\Big]$$

holds for every type t_i and every action $a_i \in A_i^{t_i}$.

Proof. Suppose that there exists a type t'_i and an action $a'_i \in A_i^{t'_i}$ such that $C_i\big(t_i; (s'_i, s_{-i})\big) > \mathbb{E}_{t_{-i}\sim p_{-i}}\Big[C_i\big(t_i; (a_i, s_{-i}(t_{-i}))\big)\Big]$. Now construct a new strategy s''_i of player i which maps every type $t_i \neq t'_i$ to the same action as s'_i, and maps t'_i to a'_i. Then

$$\begin{aligned}
C_i(s''_i, s_{-i}) &= \mathbb{E}_{t_i\sim p_i}\Big[C_i\big(t_i; (s''_i, s_{-i})\big)\Big] \\
&= \sum_{t_i\neq t'_i} p_i(t_i) C_i\big(t_i; (s''_i, s_{-i})\big) + p_i(t'_i)\mathbb{E}_{t_{-i}\sim p_{-i}}\Big[C_i\big(t_i; (a_i, s_{-i}(t_{-i}))\big)\Big] \\
&< \sum_{t_i\neq t'_i} p_i(t_i) C_i\big(t_i; (s''_i, s_{-i})\big) + p_i(t'_i) C_i(t'_i, (s'_i, s_{-i})) \ =\ C_i(s'_i, s_{-i})\,,
\end{aligned}$$

which conflicts with the assumption that s'_i is the BR of i to s. □

Lemma 5.2. *For a BGND game that is (λ, μ)-smooth with $\lambda > 0$ and $0 < \mu < \frac{1}{\varrho(\underline{\eta}\overline{\eta})^2}$ and every strategy profile s, let s'_i be the BR of each player i to s, then*

$$\sum_{i\in[N]} C_i(s'_i, s_{-i}) \ \le\ \lambda \cdot \mathbb{E}_{t\in T}\left[\mathtt{OPT}(t)\right] + \mu \cdot C(s)\,.$$

Proof. For every fixed $t'_{-i} \in T_{-i}$, Lemma 5.1 indicates that for every i, every t_i, and every $t'_{-i} \in T_{-i}$,

$$C_i(t_i; (s'_i, s_{-i})) \;\le\; \mathbb{E}_{t_{-i}\sim p_{-i}}\Big[C_i\big(t_i; (\sigma^*_i(t_i, t'_{-i}), s_{-i}(t_{-i}))\big)\Big]\,,$$

because the action $\sigma^*_i(t_i, t'_{-i})$ does not depend on t_{-i}. Taking the expectation over t_i, we get

$$\begin{aligned}\sum_{i\in[N]} C_i(s'_i, s_{-i}) \;&=\; \sum_{i\in[N]} \mathbb{E}_{t_i\sim p_i}\Big[C_i(t_i; (s'_i, s_{-i}))\Big] \\ &\le\; \sum_{i\in[N]} \mathbb{E}_{t_i\sim p_i}\Big[\mathbb{E}_{t_{-i}\sim p_{-i}}\Big[C_i\big(t_i; (\sigma^*_i(t_i, t'_{-i}), s_{-i}(t_{-i}))\big)\Big]\Big] \\ &=\; \sum_{i\in[N]} \mathbb{E}_{t\sim p}\Big[C_i\big(t_i; (\sigma^*_i(t_i, t'_{-i}), s_{-i}(t_{-i}))\big)\Big]\,.\end{aligned}$$

The last transition holds because the prior distribution p is assumed to be a product distribution. Since the formula above holds for every $t'_{-i} \in T_{-i}$, it can be derived from the definition of expectation that

$$\begin{aligned}\sum_{i\in[N]} C_i(s'_i, s_{-i}) \;&\le \mathbb{E}_{t'_{-i}\sim p_{-i}}\left[\sum_{i\in[N]} \mathbb{E}_{t\sim p}\Big[C_i\big(t_i; (\sigma^*_i(t_i, t'_{-i}), s_{-i}(t_{-i}))\big)\Big]\right] \\ &= \sum_{i\in[N]} \mathbb{E}_{t\sim p}\left[\mathbb{E}_{t'_{-i}\sim p_{-i}}\left[C_i\Big(t_i; \big(\sigma^*_i(t_i, t'_{-i}), s_{-i}(t_{-i})\big)\Big)\right]\right]\,.\end{aligned}$$

The last transition holds because t'_{-i} is independent of t. In [27], it is proved that in a BGND game that is (λ, μ)-smooth, it holds for any strategy profile s that

$$\sum_{i\in[N]} \mathbb{E}_{t\sim p}\left[\mathbb{E}_{t'_{-i}\sim p_{-i}}\left[C_i\Big(t_i; \big(\sigma^*_i(t_i, t'_{-i}), s_{-i}(t_{-i})\big)\Big)\right]\right] \le \lambda\cdot\mathbb{E}_{t\sim p}\left[\mathtt{OPT}(t)\right] + \mu C(s)\,.$$

Since $\mu < \frac{1}{\varrho(\underline{\eta}\overline{\eta})^2} \le 1$, this proposition follows. □

Lemma 5.3. *If the ABRD simulated in* ***`Bayes-ABRD`*** *converges at round r for any $r \in [R]$, then the last strategy profile s^r satisfies*

$$C(s^r) \le \frac{\varrho(\underline{\eta}\overline{\eta})^2\lambda}{1 - \varrho(\underline{\eta}\overline{\eta})^2\mu} \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big]\,.$$

Proof. Recalling that we use s'_i and $\widehat{s}_i^r$ to respectively represent the BR and ABR of player i to s^r, we observe that

$$\begin{aligned} C(s^r) \;=\; \sum_i C_i(s^r) \;&\le \overline{\eta} \sum_i \widehat{C}_i(s^r) \\ &\le \overline{\eta}(\underline{\eta}\overline{\eta}) \sum_i \widehat{C}_i(\widehat{s}_i^r, s_{-i}^r) \\ &\le \varrho(\underline{\eta}\overline{\eta})^2 \cdot \sum_i C_i(s'_i, s_{-i}^r) \\ &\le \varrho(\underline{\eta}\overline{\eta})^2 (\lambda \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big] + \mu \cdot C(s^r))\,, \end{aligned}$$

where the second transitions follow from the definition of the $(\underline{\eta}, \overline{\eta})$-estimation of the individual cost, the third transition holds since the ABRD converges at round r, the fourth transition holds following Lemma 4.1, and the fifth transition follows from Lemma 5.2. $\square$

Lemma 5.4. *The initial strategy profile s^0 of* ***Bayes-ABRD*** *satisfies $C(s^0) \le \varrho \cdot N^{\alpha_{\max}-1} \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big]$.*

Proof. The construction of s^0 guarantees that

$$\sum_{e\in s_i^0(t_i)} \sum_{j\in[q]} \xi_{e,j} \;\le\; \varrho \cdot \sum_{e\in\sigma_i^*(t_i,t_{-i})} \sum_{j\in[q]} \xi_{e,j}$$

holds for any i, any t_i, and any t_{-i}. It implies that,

$$\begin{aligned} \sum_{i\in[N]} \sum_{e\in s_i^0(t_i)} \sum_{j\in[q]} \xi_{e,j} \;&\le \varrho \cdot \sum_{i\in[N]} \sum_{e\in\sigma_i^*(t_i,t_{-i})} \sum_{j\in[q]} \xi_{e,j} \\ &\le \varrho \cdot \sum_{i\in[N]} \sum_{e\in\sigma_i^*(t_i,t_{-i})} \sum_{j\in[q]} \xi_{e,j} \left(l_e^{\sigma^*(t_i,t_{-i})}\right)^{\alpha_j-1} \\ &= \varrho \cdot \mathtt{OPT}(t_i, t_{-i})\,, \end{aligned}$$

where the second transition holds because $l_e^{\sigma^*(t)} \in \mathbb{Z}_{\ge 1}$ for any $e \in \sigma_i^*(t)$, and

$\alpha_j - 1 \geq 0$. Then,

$$\begin{aligned}
C(s^0) &= \mathbb{E}_{t\sim T}\left[\sum_{e\in E}\sum_{j\in[q]}\xi_{e,j}\left(l_e^{s^0(t)}\right)^{\alpha_j}\right] \\
&= \mathbb{E}_{t\sim T}\left[\sum_{e\in E}\sum_{i:e\in s_i^0(t_i)}\sum_{j\in[q]}\xi_{e,j}\left(l_e^{s^0(t)}\right)^{\alpha_j-1}\right] \\
&\leq \mathbb{E}_{t\sim T}\left[\sum_{e\in E}\sum_{i:e\in s_i^0(t_i)}\sum_{j\in[q]}\xi_{e,j}\cdot N^{\alpha_j-1}\right] \\
&\leq N^{\alpha_{\max}-1}\mathbb{E}_{t\sim T}\left[\sum_{e\in E}\sum_{i:e\in s_i^0(t_i)}\sum_{j\in[q]}\xi_{e,j}\right] \\
&= N^{\alpha_{\max}-1}\mathbb{E}_{t\sim T}\left[\sum_{i\in[N]}\sum_{e\in s_i^0(t_i)}\sum_{j\in[q]}\xi_{e,j}\right] \\
&\leq \varrho N^{\alpha_{\max}-1}\cdot\mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big]\,.
\end{aligned}$$

The assertion follows. □

Lemma 5.5. *For any round $r < R$ such that the ABRD does not converge at round $r+1$, as long as the player selected for strategy update satisfies Eq. (4), we have $\Phi(s^r) - \Phi(s^{r+1}) > 0$.*

Proof. Since the ABRD does not converge at round r, there exists a player i^r who is selected to update her strategy. By the definition of the potential function,

$$\begin{aligned}
\Phi(s^r) - \Phi(s^{r+1}) &= C_{i^r}(s^r) - C_{i^r}(\widehat{s}^r_{i^r}, s^r_{-i^r}) \\
&\geq \frac{1}{\underline{\eta}}\widehat{C}_{i^r}(s^r) - \overline{\eta}\widehat{C}_{i^r}(\widehat{s}^r_{i^r}, s^r_{-i^r}) \\
&> \frac{1}{\underline{\eta}}(\underline{\eta}\overline{\eta})\widehat{C}_{i^r}(\widehat{s}^r_{i^r}, s^r_{-i^r}) - \overline{\eta}\widehat{C}_{i^r}(\widehat{s}^r_{i^r}, s^r_{-i^r}) \\
&= 0\,.
\end{aligned}$$

The second formula follows from the definition of the ϵ-individual cost. The third one follows from Eq. (4). □

Theorem 5.6. *Let $Q = \frac{2(\underline{\eta}\overline{\eta})N}{1-\varrho(\underline{\eta}\overline{\eta})^2\mu}$. If $R = \lceil Q \cdot \ln(KN^{\alpha_{\max}-1})\rceil$, then the output s^{out} of `Bayes-ABRD` satisfies*

$$C(s^{out}) \le \frac{2K\varrho(\underline{\eta}\overline{\eta})^2\lambda}{1-\varrho(\underline{\eta}\overline{\eta})^2\mu} \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big] .$$

Proof. Lemma 5.3 ensures that the assertion holds if the ABRD simulated in `Bayes-ABRD` converges in any round $r \le R$, so it is left to analyze the case where the ABRD does not converge. We say a profile s^r involved in the ABRD is *bad* if

$$C(s^r) > \frac{2\varrho(\underline{\eta}\overline{\eta})^2\lambda}{1-\varrho(\underline{\eta}\overline{\eta})^2\mu} \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big] .$$

Claim 5.7. *For any $r < R$, if s^r is bad, then $\Phi(s^{r+1}) < (1 - 1/Q) \cdot \Phi(s^r)$.*

Proof. Fix

$$d^r = \overline{\eta}\Big[\sum_{i\in[N]} \widehat{C}_i(s^r) - (\underline{\eta}\overline{\eta}) \sum_{i\in[N]} \widehat{C}_i(\widehat{s}_i^r, s_{-i}^r)\Big] . \tag{5}$$

This means that

$$\begin{aligned} C(s^r) = \sum_{i\in[N]} C_i(s^r) &\le \overline{\eta} \sum_{i\in[N]} \widehat{C}_i(s^r) \\ &= \overline{\eta}(\underline{\eta}\overline{\eta}) \sum_{i\in[N]} \widehat{C}_i(\widehat{s}_i^r, s_{-i}^r) + d^r \\ &\le \varrho(\underline{\eta}\overline{\eta})^2 \sum_{i\in[N]} C_i(s'_i, s_{-i}^r) + d^r \\ &\le \varrho(\underline{\eta}\overline{\eta})^2 \Big(\lambda \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big] + \mu C(s^r)\Big) + d^r . \end{aligned}$$

Therefore, $d^r \ge \big[1 - \varrho(\underline{\eta}\overline{\eta})^2\mu\big]\, C(s^r) - \varrho(\underline{\eta}\overline{\eta})^2\lambda \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big]$, hence, if s^r is bad, then d^r satisfies

$$d^r > \big[1 - \varrho(\underline{\eta}\overline{\eta})^2\mu\big]\, C(s^r) - \frac{1-\varrho(\underline{\eta}\overline{\eta})^2\mu}{2} C(s^r) = \frac{1-\varrho(\underline{\eta}\overline{\eta})^2\mu}{2} C(s^r) . \tag{6}$$

Since the ABRD does not converge at round r, there exists a player i^r being selected to update her strategy. Recalling that the ABR of player i to

s^r is denoted by $\widehat{s}_i^r$, we observe that

$$\begin{aligned}
\Phi(s^r) - \Phi(s^{r+1}) &= C_{i^r}(s^r) - C_{i^r}(\widehat{s}_{i^r}^r, s_{-i^r}^r) \\
&\geq \frac{1}{\underline{\eta}} \widehat{C}_{i^r}(s^r) - \overline{\eta} \cdot \widehat{C}_{i^r}(\widehat{s}_{i^r}^r, s_{-i^r}^r) \\
&= \frac{1}{\underline{\eta}} \left[\widehat{C}_{i^r}(s^r) - (\underline{\eta}\overline{\eta}) \widehat{C}_{i^r}(\widehat{s}_{i^r}^r, s_{-i^r}^r) \right] \\
&\geq \frac{1}{\underline{\eta}} \cdot \frac{1}{N} \sum_{i \in [N]} \left[\widehat{C}_i(s^r) - (\underline{\eta}\overline{\eta}) \widehat{C}_i(\widehat{s}_i^r, s_{-i}^r) \right] \\
&= \frac{1}{\underline{\eta}\overline{\eta}} \cdot \frac{d^r}{N} \\
&> \frac{1}{\underline{\eta}\overline{\eta}} \cdot \frac{1}{2N} \left[1 - \varrho (\underline{\eta}\overline{\eta})^2 \mu \right] C(s^r) \\
&\geq \frac{1}{\underline{\eta}\overline{\eta}} \cdot \frac{1}{2N} \left[1 - \varrho (\underline{\eta}\overline{\eta})^2 \mu \right] \Phi(s^r) ,
\end{aligned}$$

where the fourth transition follows from Eq. (4), the fifth and sixth transitions follow from Eq. (5) and Eq. (6), respectively, and the last transition holds because the potential function is assumed to be K-bounded. Therefore,

$$\Phi(s^{r+1}) \;<\; \Phi(s^r) \left(1 - \frac{1 - \varrho(\underline{\eta}\overline{\eta})^2 \mu}{2(\underline{\eta}\overline{\eta}) N} \right) \;=\; (1 - 1/Q) \cdot \Phi(s^r)$$

as promised. ■ (Claim 5.7)

Claim 5.8. *Assuming that all the $R+1$ strategy profiles in the ABRD are bad, we have $C(s^R) < \varrho \cdot \mathbb{E}_{t \sim T}\Big[\mathtt{OPT}(t)\Big]$.*

Proof. Claim 5.7 implies that if all the $R+1$ profiles involved in the ABRD are bad, then

$$\begin{aligned}
\Phi(s^R) \;&<\; \left(1 - \frac{1}{Q} \right)^R \Phi(s^0) \\
&= \left(1 - \frac{1}{Q} \right)^{\left\lceil Q \cdot \ln \left(K N^{\alpha_{\max} - 1} \right) \right\rceil} \Phi(s^0) \\
&\leq \frac{1}{K N^{\alpha_{\max} - 1}} \Phi(s^0) .
\end{aligned}$$

By the definition of the bounded potential function and by Lemma 5.4, we have

$$C(s^R) \leq K \cdot \Phi(s^R) < \frac{K\Phi(s^0)}{KN^{\alpha_{\max}-1}} \leq \frac{C(s^0)}{N^{\alpha_{\max}-1}} \leq \frac{\varrho N^{\alpha_{\max}-1}\mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big]}{N^{\alpha_{\max}-1}},$$

which completes the proof. ■ (Claim 5.8)

Claim 5.9. $\varrho < \frac{2\varrho(\underline{\eta}\overline{\eta})^2\lambda}{1-\varrho(\underline{\eta}\overline{\eta})^2\mu}$.

Proof. It can be inferred from [27] that the parameters (λ, μ) should satisfy $\frac{\lambda}{1-\mu} \geq 1$ if the game is (λ, μ)-smooth. Therefore, $\frac{2\varrho(\underline{\eta}\overline{\eta})^2\lambda}{1-\varrho(\underline{\eta}\overline{\eta})^2\mu} > \frac{2\varrho\lambda}{1-\mu} > \varrho$. ■ (Claim 5.9)

By Claim 5.9 and the definition of bad strategy profiles, $C(s^R) < \varrho \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big]$ implies that s^R is not bad, which conflicts with the assumption of Claim 5.8 that all the $R+1$ strategy profiles are bad. This means that there exists at least one round r^* whose corresponding strategy profile s^{r^*} is not bad. Therefore,

$$\begin{aligned} C(s^R) \leq K \cdot \Phi(s^R) &\leq K \cdot \Phi(s^{r^*}) \\ &\leq K \cdot C(s^{r^*}) \\ &\leq K \cdot \frac{2\varrho(\underline{\eta}\overline{\eta})^2\lambda}{1-\varrho(\underline{\eta}\overline{\eta})^2\mu} \cdot \mathbb{E}_{t\sim T}\Big[\mathtt{OPT}(t)\Big]. \end{aligned}$$

The first transition and the third one holds because the potential function is K-bounded. The second transition follows from Lemma 5.5. The last transition holds because s^{r^*} is not bad. This completes the proof. □

6. Smoothness Parameters

In this section, we consider the case where the parameters ϱ, $\underline{\eta}$ and $\overline{\eta}$ are fixed, and focus on finding proper parameters (λ, μ) such that the BGND game is (λ, μ)-smooth, and $\mu < 1/[\varrho(\underline{\eta}\overline{\eta})^2]$.

Lemma 6.1. *For any pair of parameters $\lambda' > 0$ and $0 < \mu' < 1\Big/\Big[\varrho\big(\underline{\eta}\overline{\eta}\big)^2\Big]$, if*

$$y \cdot (x+y)^{\alpha_j - 1} \leq \lambda' \cdot y^{\alpha_j} + \mu' \cdot x^{\alpha_j} \tag{7}$$

holds for any $x, y \in \mathbb{R}_{\geq 0}$ and every $j \in [q]$, then the BGND game is (λ', μ')-smooth.

Proof. This proposition can be proved in a similar way as [27]. For any resource e and any type t, we say $e \in \sigma^*(t)$ if there exists some player i such that $e \in \sigma_i^*(t)$. For any type profiles t, t', every action profile $a \in A^{t'}$, and every resource $e \in \sigma^*(t)$,

$$\begin{aligned}
\sum_{i \in [N]} f_{i,e}(t_i; (\sigma_i^*(t), a_{-i})) &= \sum_{i \in [N]: e \in \sigma_i^*(t)} \sum_{j \in [q]} \xi_{e,j} \left(l_e^{(\sigma_i^*(t), a_{-i})} \right)^{\alpha_j - 1} \\
&\leq |l_e^{\sigma^*(t)}| \sum_{j \in [q]} \xi_{e,j} \left(l_e^{\sigma^*(t)} + l_e^a \right)^{\alpha_j - 1} \\
&\leq \sum_{j \in [q]} \xi_{e,j} \left[\lambda' \cdot \left(l_e^{\sigma^*(t)} \right)^{\alpha_j} + \mu' \cdot \left(l_e^a \right)^{\alpha_j} \right],
\end{aligned}$$

where the third transition follows from Eq. (7). Then

$$\begin{aligned}
\sum_{i \in [N]} C_i(t_i; (\sigma_i^*(t), a_{-i})) &= \sum_{i \in [N]} \sum_{e \in E} f_{i,e}(t_i; (\sigma_i^*(t), a_{-i})) \\
&= \sum_{e \in \sigma^*(t)} \sum_{i \in [N]} f_{i,e}(t_i; (\sigma_i^*(t), a_{-i})) \\
&\leq \lambda' \cdot \sum_{e \in \sigma^*(t)} \sum_{j \in [q]} \xi_{e,j} \left(l_e^{\sigma^*(t)} \right)^{\alpha_j} + \mu' \cdot \sum_{e \in \sigma^*(t)} \sum_{j \in [q]} \xi_{e,j} \left(l_e^a \right)^{\alpha_j} \\
&\leq \lambda' \cdot \texttt{OPT}(t) + \mu' \cdot \sum_{i \in [N]} C_i(t_i', a).
\end{aligned}$$

The second transition above holds because for any $e \notin \sigma^*(t)$, $f_{i,e}(t_i; (\sigma_i^*(t), a_{-i})) = 0$ for every player i. The last transition holds because $l_e^a = 0$ for any $e \notin a$, which implies that $\sum_{e \in \sigma^*(t)} \sum_{j \in [q]} \xi_{e,j} \left(l_e^a \right)^{\alpha_j} = \sum_{e \in \sigma^*(t) \cap a} \sum_{j \in [q]} \xi_{e,j} \left(l_e^a \right)^{\alpha_j}$. ◻

Lemma 6.2 ([36])**.** *For any $\mu' \in (0, \frac{1}{\varrho(\underline{\eta}\overline{\eta})^2})$, setting $\lambda' = \max_{x \in \mathbb{R}_{>0}} (x + 1)^{\alpha_{\max} - 1} - \mu' \cdot x^{\alpha_{\max}}$ satisfies Eq. (7).*

For any $\mu' \in (0, \frac{1}{\varrho(\underline{\eta}\overline{\eta})^2})$, define $g_{\mu'}(x) = (x+1)^{\alpha_{\max} - 1} - \mu' \cdot x^{\alpha_{\max}}$ and $h(x) = \left[(\alpha_{\max} - 1)(x+1)^{\alpha_{\max} - 2} \right] \Big/ \left[\alpha_{\max} \cdot x^{\alpha_{\max} - 1} \right]$. Then:

Lemma 6.3 ([36]). *For any $\mu' \in (0, \frac{1}{\varrho(\underline{\eta}\overline{\eta})^2})$, there exists a unique real positive number $x_{\mu'}$ that maximizes $g_{\mu'}(x)$ over $x \in (0, +\infty)$, and this number satisfies $h_{\alpha_{\max}}(x_{\mu'}) = \mu'$.*

Remark 6.4. Note that in [36], the propositions corresponding to Lemma 6.2 and Lemma 6.3 are proved for the case $\alpha_{\max} \in \mathbb{Z}_{\geq 2}$, but their proofs directly hold for the case where $\alpha_{\max} \in \mathbb{R}_{\geq 1}$.

It can be inferred from the derivative that $(x+1)^{\alpha_{\max}-1} = x^{\alpha_{\max}}$ has a unique positive root, which is denoted by $\gamma_{\alpha_{\max}}$. In [36], it is proved that if $\alpha_{\max} \geq 3$, $\gamma_{\alpha_{\max}}$ is bounded by $O\Big(\frac{\alpha_{\max}-1}{\ln(\alpha_{\max}-1)}\Big)$. Furthermore, it can be verified that $\gamma_{\alpha_{\max}} < 2$ when $\alpha_{\max} \in (1,2)$, and $\gamma_{\alpha_{\max}} < 3$ when $\alpha_{\max} \in [2,3)$.

Let $\mu_\alpha = h\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big)$, and $\lambda_\alpha = g_{\mu_\alpha}\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big)$. Then we have:

Theorem 6.5. *The BGND game is $(\lambda_\alpha, \mu_\alpha)$-smooth, and $\varrho(\underline{\eta}\overline{\eta})^2 \mu_\alpha < 1 - 1/\alpha_{\max}$.*

Proof.

$$
\begin{aligned}
h\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big) &= \frac{(\alpha_{\max}-1)\cdot\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}+1\Big)^{\alpha_{\max}-2}}{\alpha_{\max}\cdot\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big)^{\alpha_{\max}-1}} \\
&= \frac{(\alpha_{\max}-1)\cdot\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}+1\Big)^{\alpha_{\max}-1}\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big)}{\alpha_{\max}\cdot\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big)^{\alpha_{\max}}\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}+1\Big)} \\
&= \frac{(\alpha_{\max}-1)\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}+1\Big)^{\alpha_{\max}-1}\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big)}{\alpha_{\max}\varrho(\underline{\eta}\overline{\eta})^2\Big[\varrho(\underline{\eta}\overline{\eta})^2 \cdot \Big(\gamma_{\alpha_{\max}}+1\Big)\Big]^{\alpha_{\max}-1}\Big(\varrho(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}+1\Big)} \\
&< \frac{(\alpha_{\max}-1)}{\alpha_{\max}\varrho(\underline{\eta}\overline{\eta})^2},
\end{aligned}
$$

which implies that $\varrho(\underline{\eta}\overline{\eta})^2 \cdot \mu_\alpha < (\alpha_{\max}-1)/\alpha_{\max}$. The third transition holds because $\gamma_{\alpha_{\max}}$ is the positive root of $(x+1)^{\alpha_{\max}-1} - x^{\alpha_{\max}} = 0$. The fourth transition holds because $\varrho(\underline{\eta}\overline{\eta})^2 \geq 1$, $\gamma_{\alpha_{\max}} > 0$ and $\alpha_{\max}-1 > 0$.

Since $\mu_\alpha < \frac{\alpha_{\max}-1}{\alpha_{\max}} \cdot \frac{1}{\varrho(\underline{\eta}\overline{\eta})^2} < \frac{1}{\varrho(\underline{\eta}\overline{\eta})^2}$, it can be inferred from Lemma 6.3 that $g_{\mu_\alpha}\Big(\varrho\cdot(\underline{\eta}\overline{\eta})^2 \cdot \gamma_{\alpha_{\max}}\Big) = \max_{x>0}(x+1)^{\alpha_{\max}-1} - \mu_\alpha \cdot x^{\alpha_{\max}}$. It implies that λ_α satisfies the condition given in Lemma 6.2. By Lemma 6.1, this theorem holds. $\square$

7. Bounded Potential Function

In this section, it is proved that the BGND game admits a potential function that is K-bounded with $K = \lceil \alpha_{\max} \rceil$.

Lemma 7.1 ([1]). *If for every type profile t, there exists a function $\Phi_t : A \mapsto \mathbb{R}_{\geq 0}$ such that for every action profile $a \in A^t$, every $i \in [N]$ and every $a_i' \in A_i^{t_i}$,*

$$\Phi_t(a) - \Phi_t(a_i', a_{-i}) = C_i(t_i; a) - C_i(t_i; (a_i', a_{-i})) \,, \tag{8}$$

then $\Phi(s) = \sum_{t \in T} p(t) \Phi_t(s(t))$ is a potential function of the BGND game.

Theorem 7.2. *For the BGND game, there exists a potential function $\Phi(s)$ that is $\lceil \alpha_{\max} \rceil$-bounded.*

Proof. For every type profile t and every action profile $a \in A^t$, define the function

$$\Phi_t(a) \;=\; \sum_{e \in E} \sum_{l=1}^{l_e^a} \sum_{j \in [q]} \xi_{e,j} \cdot l^{\alpha_j - 1} \,.$$

In [37], it is proved that such a function satisfies Eq. (8), which implies that the BGND game admits a potential function $\Phi(s) = \sum_{t \in T} p(t) \Phi_t(s(t))$. Furthermore, for every $e \in E$ and every $j \in [q]$, $\sum_{l=1}^{l_e^a} l^{\alpha_j - 1} \leq (l_e^a)^{\alpha_j}$ trivially holds, and

$$\sum_{l=1}^{l_e^a} l^{\alpha_j - 1} \;\geq\; \frac{1}{(l_e^a)^{\lceil \alpha_j \rceil - \alpha_j}} \sum_{l=1}^{l_e^a} l^{\lceil \alpha_j \rceil - 1} \;\geq\; \frac{1}{(l_e^a)^{\lceil \alpha_j \rceil - \alpha_j}} \cdot \frac{1}{\lceil \alpha_j \rceil} (l_e^a)^{\lceil \alpha_j \rceil} \;=\; \frac{1}{\lceil \alpha_j \rceil} (l_e^a)^{\alpha_j} \,,$$

where the second transition follows from [38]. Therefore, it can be obtained that $\Phi_t(a) \leq \sum_{e \in E} F_e(l_e^a) \leq \lceil \alpha_{\max} \rceil \Phi_t(a)$. By the linearity of expectation, $\Phi(s)$ is $\lceil \alpha_{\max} \rceil$-bounded. $\square$

8. Efficient Estimation of the Cost Share

This section focuses on the $(\underline{\eta}, \overline{\eta})$-estimation of the expected cost shares. For any $z \in (0, 1)$ and $z' \geq 1$, define $b_z = \Big((\beta^\circ)^2 + 1\Big)\Big(1 - \frac{1}{\beta^\circ}\Big)^{-z}$ with β° being the unique root of $2\beta^3 - (z+2)\beta^2 - 2 = 0$ in the interval $(1, +\infty)$, $B_{z'}$ to be the fractional Bell number with the parameter z' [3, 4], and $\gamma_{z'}$ to be the unique positive root of $(x+1)^{z'-1} = x^{z'}$ [36].

For any $i \in [N]$, $e \in E$ and any s_{-i}, it is shown that there exists a $(\max\{1, \max_{\alpha_j \in (1,2)} b_{\alpha_j - 1}\}, \max\{\max_{\alpha_j \geq 2} B_{\alpha_j - 1}, 1\})$-estimation $\widehat{\mathfrak{f}}_{i,e}(+; s_{-i})$ of $\mathfrak{f}_{i,e}(+; s_{-i})$ that can be obtained in poly$(q, N, \{|T_i|\}_{i \in [N]})$-time. Particularly, consider the special case of the BGND game where $q = 1$ and $\alpha_1 = 2$, which means that the cost function associated with each resource $e \in E$ can be written as

$$F_e(l) = \xi_e \cdot l^2 \,. \tag{9}$$

The BGND game is proved to be tractable with such a quadratic function.

Lemma 8.1 ([33]). *Let $\{X_1, X_2, \cdots, X_k, \cdots\}$ be a finite set of mutually independent random variables following the Bernoulli distribution supported on $\{0, 1\}$. Then for any $z \geq 1$,*

$$\mathbb{E}\Big[\big(\sum_k X_k\big)^z\Big] \;\leq\; B_z \cdot \max\Big\{\mathbb{E}\big[\sum_k X_k\big],\, \Big(\mathbb{E}\big[\sum_k X_k\big]\Big)^z\Big\}\,.$$

Lemma 8.2. *Let $\{X_1, X_2, \cdots, X_k, \cdots\}$ be a finite set of Bernoulli random variables that are mutually independent. For any $z' \in (0, 1)$ and $\beta > 1$:*

$$\frac{1}{\beta^2+1}\Big(1-\frac{1}{\beta}\Big)^{z'}\Big(\mathbb{E}\Big[1+\sum_k X_k\Big]\Big)^{z'} \leq \mathbb{E}\Big[\Big(1+\sum_k X_k\Big)^{z'}\Big] \leq \Big(\mathbb{E}\Big[1+\sum_k X_k\Big]\Big)^{z'}.$$

Proof. The expression $\mathbb{E}\Big[\Big(1 + \sum_k X_k\Big)^{z'}\Big] \leq \Big(\mathbb{E}\Big[1 + \sum_k X_k\Big]\Big)^{z'}$ follows from Jensen's inequality [32], since the function $\varphi(x) = x^{z'}$ is concave when $z' \in (0, 1)$. Now consider the lower bound on $\mathbb{E}\Big[\Big(1 + \sum_k X_k\Big)^{z'}\Big]$. Let $\mathrm{Var}\Big[1 + \sum_k X_k\Big]$ be the variance of the random variable $1 + \sum_k X_k$. Then we have

$$\begin{aligned}
&\mathrm{Var}\Big[1 + \sum_k X_k\Big] \\
&= \mathbb{E}\Big[\Big(1 + \sum_k X_k\Big)^2\Big] - \Big(\mathbb{E}\Big[1 + \sum_k X_k\Big]\Big)^2 \\
&\leq B_2 \cdot \max\Big\{\mathbb{E}\Big[1 + \sum_k X_k\Big],\, \Big(\mathbb{E}\Big[1 + \sum_k X_k\Big]\Big)^2\Big\} - \Big(\mathbb{E}\Big[1 + \sum_k X_k\Big]\Big)^2 \\
&= 2 \cdot \Big(\mathbb{E}\Big[1 + \sum_k X_k\Big]\Big)^2 - \Big(\mathbb{E}\Big[1 + \sum_k X_k\Big]\Big)^2 \\
&= \Big(\mathbb{E}\Big[1 + \sum_k X_k\Big]\Big)^2.
\end{aligned}$$

The second transition above follows from Lemma 8.1, because $\{X_k\}$ are mutually independent Bernoulli random variables, and the constant 1 can also be viewed as a Bernoulli random variable which equals to 1 with probability 1. The third transition holds because $1 + \sum_k X_k \geq 1$. By using Cantelli's inequality [34], it can be obtained that for any $\beta > 1$,

$$\begin{aligned}\Pr\left[\Big(1+\sum_k X_k\Big) < \Big(1-\frac{1}{\beta}\Big)\mathbb{E}\Big[1+\sum_k X_k\Big]\right] &\leq \frac{1}{1+\frac{\left(\mathbb{E}\left[1+\sum_k X_k\right]\right)^2}{\beta^2\cdot\mathrm{Var}\left[1+\sum_k X_k\right]}} \\ &= \frac{\beta^2}{\beta^2+1}\,,\end{aligned}$$

where $\Pr[\cdot]$ denotes the probability of random events. Therefore,

$$\begin{aligned}&\mathbb{E}\Big[\Big(1+\sum_k X_k\Big)^{z'}\Big] \\ \geq\ &\Pr\left[\Big(1+\sum_k X_k\Big) \geq \Big(1-\frac{1}{\beta}\Big)\mathbb{E}\Big[1+\sum_k X_k\Big]\right]\cdot\left[\Big(1-\frac{1}{\beta}\Big)\mathbb{E}\Big[1+\sum_k X_k\Big]\right]^{z'} \\ \geq\ &\Big(1-\frac{\beta^2}{\beta^2+1}\Big)\Big(1-\frac{1}{\beta}\Big)^{z'}\Big(\mathbb{E}\Big[1+\sum_k X_k\Big]\Big)^{z'}.\end{aligned}$$

□

Recalling that $b_{z'} = \Big((\beta^\circ)^2+1\Big)\Big(1-\frac{1}{\beta^\circ}\Big)^{-z'}$ with β° being the unique root of $2\beta^3-(z+2)\beta^2-2=0$ in the interval $(1,+\infty)$, we have the following lemma.

Lemma 8.3. *For any $z' \in (0,1)$, $b_{z'} = \min_{\beta>1}(\beta^2+1)\Big(1-\frac{1}{\beta}\Big)^{-z'}$.*

Proof. Let $\varphi(\beta) = (\beta^2+1)\Big(1-\frac{1}{\beta}\Big)^{-z'}$. Fix z', the derivative of φ with respect to β is

$$\frac{\mathrm{d}\varphi}{\mathrm{d}\beta} = \Big(1-\frac{1}{\beta}\Big)^{-z'-1}\frac{1}{\beta^2}\Big(2\beta^3-(2+z')\beta^2-2\Big).$$

It can be further derived from the derivative that $2\beta^3-(2+z')\beta^2-2$ is monotonically increasing in the interval $(\frac{2+z'}{3},\infty)$. Since $\frac{2+z'}{3} < 1$, $2\beta^3-(2+$

$z')\beta^2 - 2 < 0$ for $\beta = 1$, and $2\beta^3 - (2+z')\beta^2 - 2 > 0$ for $\beta = 2$, there exists a unique $\beta^\circ \in (1, +\infty)$ such that $2\beta^3 - (2+z')\beta^2 - 2 = 0$, and β° minimizes $(\beta^2+1)\left(1-\frac{1}{\beta}\right)^{-z'}$ because $\left(1-\frac{1}{\beta}\right)^{-z'-1}\frac{1}{\beta^2} > 0$ for any $\beta > 1$. $\square$

For each action a_i of player i and each resource e, denote the indicator of whether e is contained in a_i by $\delta(a_i, e)$. Formally,

$$\delta(a_i, e) \;=\; \begin{cases} 0 & \text{if } e \notin a_i \\ 1 & \text{otherwise} \end{cases}$$

Theorem 8.4. *For any player i, any edge e, any action a_i, and any strategies s_{-i}, let*

$$\widehat{\mathfrak{f}}_{i,e}(+; s_{-i}) \;=\; \sum_{j\in[q]} \xi_{e,j}\Big[1 + \sum_{i'\neq i}\sum_{t_{i'}\in T_{i'}} p_{i'}(t_{i'})\delta\big(s_{i'}(t_{i'}), e\big)\Big]^{\alpha_j - 1}, \tag{10}$$

then

$$\frac{\widehat{\mathfrak{f}}_{i,e}(+; s_{-i})}{\max\Big\{1,\, \max_{j:\alpha_j\in(1,2)} b_{\alpha_j-1}\Big\}} \;\le\; \mathfrak{f}_{i,e}(+; s_{-i}) \;\le\; \widehat{\mathfrak{f}}_{i,e}(+; s_{-i})\cdot\Big\{1,\, \max_{j:\alpha_j\ge 2} B_{\alpha_j-1}\Big\}. \tag{11}$$

In particular, if for every resource e, $F_e(l)$ is a quadratic function given in Eq. (9), then $\widehat{\mathfrak{f}}_{i,e}(+; s_{-i}) = \mathfrak{f}_{i,e}(+; s_{-i})$.

Proof. Let a_i be an action in A_i satisfying $e \in a_i$. By definition, we have

$$\begin{aligned}
\mathfrak{f}_{i,e}(+; s_{-i}) &= \mathbb{E}_{t_{-i}\sim p_{-i}}\left[f_{i,e}(a_i, s_{-i}(t_{-i}))\right] \\
&= \sum_{j\in[N]} \xi_{e,j}\cdot \mathbb{E}_{t_{-i}\sim p_{-i}}\left[\left(l_e^{a_i, s_{-i}(t_{-i})}\right)^{\alpha_j-1}\right] \\
&= \sum_{j\in[N]} \xi_{e,j}\cdot \mathbb{E}_{t_{-i}\sim p_{-i}}\left[\left(1 + \sum_{i'\in[N]:i'\neq i} \delta(s_{i'}(t_{-i}(i')), e)\right)^{\alpha_j-1}\right] \\
&= \sum_{j\in[N]} \xi_{e,j}\mathbb{E}_{\{t_{i'}\sim p_{i'}\}_{i'\neq i}}\left[\left(1 + \sum_{i'\neq i} \delta(s_{i'}(t_{i'}), e)\right)^{\alpha_j-1}\right].
\end{aligned}$$

The last transition holds because the prior distribution p is assumed to be a product distribution.

Now define a finite set of mutually independent Bernoulli random variables $\{X_{i',e}(s)\}_{i' \neq i}$ such that each $X_{i',e}(s)$ takes the value 1 with probability $\sum_{t_{i'}: e \in s_{i'}(t_{i'})} p_{i'}(t_{i'})$. Fixing a player $i'' \neq i$, we have

$$
\begin{aligned}
&\mathbb{E}_{\{t_{i'} \sim p_{i'}\}_{i' \neq i}} \left[\left(1 + \sum_{i' \neq i} \delta(s_{i'}(t_{i'}), e) \right)^{\alpha_j - 1} \right] \\
&= \sum_{t_{i''} \in T_{i''}} p_{i''}(t_{i''}) \mathbb{E}_{\{t_{i'} \sim p_{i'}\}_{i' \neq i \wedge i' \neq i''}} \left[\left(1 + \delta(s_{i''}(t_{i''}), e) + \sum_{i' \neq i \wedge i' \neq i''} \delta(s_{i'}(t_{i'}), e) \right)^{\alpha_j - 1} \right] \\
&= \sum_{t_{i''}: e \in s_{i''}(t_{i''})} p_{i''}(t_{i''}) \mathbb{E}_{\{t_{i'} \sim p_{i'}\}_{i' \neq i \wedge i' \neq i''}} \left[\left(1 + 1 + \sum_{i' \neq i \wedge i' \neq i''} \delta(s_{i'}(t_{i'}), e) \right)^{\alpha_j - 1} \right] \\
&\quad + \sum_{t_{i''}: e \notin s_{i''}(t_{i''})} p_{i''}(t_{i''}) \mathbb{E}_{\{t_{i'} \sim p_{i'}\}_{i' \neq i \wedge i' \neq i''}} \left[\left(1 + \sum_{i' \neq i \wedge i' \neq i''} \delta(s_{i'}(t_{i'}), e) \right)^{\alpha_j - 1} \right] \\
&= \mathbb{E}_{X_{i'',e}(s)} \left[\mathbb{E}_{\{t_{i'} \sim p_{i'}\}_{i' \neq i \wedge i' \neq i''}} \left[\left(1 + X_{i'',e}(s) + \sum_{i' \neq i \wedge i' \neq i''} \delta(s_{i'}(t_{i'}), e) \right)^{\alpha_j - 1} \right] \right] .
\end{aligned}
$$

Therefore, it can be inductively proved that

$$
\mathbb{E}_{\{t_{i'} \sim p_{i'}\}_{i' \neq i}} \left[\left(1 + \sum_{i' \neq i} \delta(s_{i'}(t_{i'}), e) \right)^{\alpha_j - 1} \right] = \mathbb{E} \left[\left(1 + \sum_{i' \neq i} X_{i',e}(s) \right)^{\alpha_j - 1} \right] .
$$

Recall that the constant 1 in the last expression above can also be viewed as a Bernoulli random variable which equals to 1 with probability 1. For every $\alpha_j \geq 2$, Lemma 8.1 can be applied to obtain the following expression.

$$
\begin{aligned}
&\mathbb{E} \left[\left(1 + \sum_{i' \neq i} X_{i',e}(s) \right)^{\alpha_j - 1} \right] \\
&\leq B_{\alpha_j - 1} \cdot \max \left\{ \mathbb{E} \left[1 + \sum_{i' \neq i} X_{i',e}(s) \right], \left(\mathbb{E} \left[1 + \sum_{i' \neq i} X_{i',e}(s) \right] \right)^{\alpha_{j-1}} \right\} \\
&= B_{\alpha_j - 1} \cdot \left(\mathbb{E} \left[1 + \sum_{i' \neq i} X_{i',e}(s) \right] \right)^{\alpha_j - 1} .
\end{aligned}
$$

The second line holds because $\mathbb{E}\Big[1 + \sum_{i' \neq i} X_{i',e}(s)\Big] > 1$. Similarly, it can be derived from Lemma 8.2 that for every $\alpha_j \in (1, 2)$,

$$\mathbb{E}\left[\left(1 + \sum_{i' \neq i} X_{i',e}(s)\right)^{\alpha_j - 1}\right] \leq \left(\mathbb{E}\left[1 + \sum_{i' \neq i} X_{i',e}(s)\right]\right)^{\alpha_j - 1},$$

which also trivially holds for $\alpha_j = 1$. So, $\mathbb{E}\left[\left(1 + \sum_{i' \neq i} X_{i',e}(s)\right)^{\alpha_j - 1}\right] \leq \max\Big\{1, \max_{j:\alpha_j \geq 2} B_{\alpha_j - 1}\Big\}\left(\mathbb{E}\Big[1 + \sum_{i' \neq i} X_{i',e}(s)\Big]\right)^{\alpha_j - 1}$, and in a similar way, it also be inferred from Lemma 8.1 and Lemma 8.2 that $\mathbb{E}\left[\left(1 + \sum_{i' \neq i} X_{i',e}(s)\right)^{\alpha_j - 1}\right] \geq \left(\mathbb{E}\Big[1 + \sum_{i' \neq i} X_{i',e}(s)\Big]\right)^{\alpha_j - 1} \Big/ \max\Big\{1, \max_{j:\alpha_j < 2} b_{\alpha_j - 1}\Big\}$. Since $\mathbb{E}\Big[1 + \sum_{i' \neq i} X_{i',e}(s)\Big] = 1 + \sum_{i' \neq i} \sum_{t_{i'}} p_{i'}(t_{i'}) \delta\Big(s_{i'}(t_{i'}), e\Big)$, Eq. (11) holds. For the special case where every F_e is a quadratic function, by the linearity of the expectation, we have

$$\mathbb{E}_{\{t_{i'} \sim p_{i'}\}_{i' \neq i}}\left[\left(1 + \sum_{i' \neq i} \delta(s_{i'}(t_{i'}), e)\right)^{2-1}\right] = 1 + \sum_{i' \neq i} \mathbb{E}_{t_{i'} \sim p_{i'}}\Big[\delta(s_{i'}(t_{i'}), e)\Big],$$

which completes the proof. □

Corollary 8.5. *By computing Eq.* (10)*, the desired estimation of each expected cost share is obtained in* $O(q \cdot \sum_{i \in [N]} |T_i|)$*-time.*

Plugging Theorem 6.5, Theorem 7.2, Theorem 8.4, and Corollary 8.5 into Theorem 5.6 proves our main result, Theorem 1.2.

References

[1] N. Alon, Y. Emek, M. Feldman, M. Tennenholtz, Bayesian ignorance, Theoretical Computer Science 452 (2012) 1–11. Preliminary version appears in Proceedings of the 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, pp. 384-391. ACM, 2010.

[2] M. Andrews, A. F. Anta, L. Zhang, W. Zhao, Routing for power minimization in the speed scaling model, IEEE/ACM Transactions on Networking 20 (2012) 285 –294.

[3] E. Bampis, A. Kononov, D. Letsios, G. Lucarelli, M. Sviridenko, Energy efficient scheduling and routing via randomized rounding, in: 33nd International Conference on Foundations of Software Technology and Theoretical Computer Science, 2013, p. 449.

[4] K. Makarychev, M. Sviridenko, Solving optimization problems with diseconomies of scale via decoupling, in: 55th IEEE Annual Symposium on Foundations of Computer Science, FOCS, 2014.

[5] A. Gupta, M. T. Hajiaghayi, H. Räcke, Oblivious network design, in: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm, Society for Industrial and Applied Mathematics, 2006, pp. 970–979.

[6] M. Englert, H. Räcke, Oblivious routing for the lp-norm, in: Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science, FOCS '09, IEEE Computer Society, Washington, DC, USA, 2009, pp. 32–40.

[7] Y. Shi, F. Zhang, J. Wu, Z. Liu, Randomized oblivious integral routing for minimizing power cost, Theoretical Computer Science 607 (2015) 221–246.

[8] F. Yao, A. Demers, S. Shenker, A scheduling model for reduced cpu energy, in: Proceedings of IEEE 36th Annual Foundations of Computer Science, 1995, pp. 374–382.

[9] S. Irani, K. R. Pruhs, Algorithmic problems in power management, SIGACT News 36 (2005) 63–76.

[10] N. Bansal, T. Kimbrel, K. Pruhs, Speed scaling to manage energy and temperature, J. ACM 54 (2007) 3:1–3:39.

[11] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, D. Wetherall, Reducing network energy consumption via sleeping and rate-adaptation, in: Proceedings of the 5th USENIX Symposium on Networked Systems

Design and Implementation, NSDI'08, USENIX Association, 2008, pp. 323–336.

[12] K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, J. A. Maestro, Ieee 802.3 az: the road to energy efficient ethernet, IEEE Communications Magazine 48 (2010).

[13] S. Albers, Energy-efficient algorithms, Commun. ACM 53 (2010) 86–96.

[14] Intel, Enhanced intel speedstep technology for the intel pentium m processor, in: Intel White Paper 301170-001, 2004.

[15] A. Wierman, L. L. Andrew, A. Tang, Power-aware speed scaling in processor sharing systems, in: 28th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, INFOCOM, 2009, pp. 2007 –2015.

[16] M. Englert, H. Räcke, Oblivious routing for the lp-norm, in: Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on, IEEE, 2009, pp. 32–40.

[17] H. Räcke, Survey on oblivious routing strategies, in: Mathematical Theory and Computational Practice, volume 5635 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2009, pp. 419–429.

[18] Y. Emek, S. Kutten, R. Lavi, Y. Shi, Approximating generalized network design under (dis)economies of scale with applications to energy efficiency, in: Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, ACM, New York, NY, USA, 2018, pp. 598–606.

[19] E. W. Dijkstra, A note on two problems in connexion with graphs, Numer. Math. 1 (1959) 269–271.

[20] M. L. Fredman, R. E. Tarjan, Fibonacci heaps and their uses in improved network optimization algorithms, J. ACM 34 (1987) 596–615.

[21] J. Byrka, F. Grandoni, T. Rothvoss, L. Sanità, Steiner tree approximation via iterative randomized rounding, Journal of the ACM (JACM) 60 (2013) 6. Preliminary version in STOC'10.

[22] M. Charikar, C. Chekuri, T.-y. Cheung, Z. Dai, A. Goel, S. Guha, M. Li, Approximation algorithms for directed steiner problems, in: Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, 1998, pp. 192–200.

[23] A. Agrawal, P. Klein, R. Ravi, When trees collide: An approximation algorithm for the generalized steiner problem on networks, SIAM Journal on Computing 24 (1995) 440–456.

[24] C. Chekuri, G. Even, A. Gupta, D. Segev, Set connectivity problems in undirected graphs and the directed steiner network problem, ACM Trans. Algorithms 7 (2011) 18:1–18:17.

[25] G. Lawler, H. Narayanan, Mixing times and ℓp bounds for oblivious routing, in: Proceedings of the Meeting on Analytic Algorithmics and Combinatorics, ANALCO '09, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2009, pp. 66–74.

[26] P. Harsha, T. P. Hayes, H. Narayanan, H. Räcke, J. Radhakrishnan, Minimizing average latency in oblivious routing, in: Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '08, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2008, pp. 200–207.

[27] T. Roughgarden, The price of anarchy in games of incomplete information, in: Proceedings of the 13th ACM Conference on Electronic Commerce, ACM, 2012, pp. 862–879.

[28] A. Gupta, R. Ravi, A. Sinha, An edge in time saves nine: Lp rounding approximation algorithms for stochastic network design, in: FOCS, 2004, pp. 218–227.

[29] T. G. Crainic, X. Fu, M. Gendreau, W. Rei, S. W. Wallace, Progressive hedging-based metaheuristics for stochastic network design, Networks 58 (2011) 114–124.

[30] R. Rahmaniani, T. G. Crainic, M. Gendreau, W. Rei, Accelerating the benders decomposition method: Application to stochastic network design problems, SIAM Journal on Optimization 28 (2018) 875–903.

[31] N. Garg, A. Gupta, S. Leonardi, P. Sankowski, Stochastic analyses for online combinatorial optimization problems, in: Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '08, Society for Industrial and Applied Mathematics, 2008, pp. 942–951.

[32] J. Jensen, Sur les fonctions convexes et les inégalités entre les valeurs moyennes, Acta Mathematica 30 (1906) 175–193.

[33] D. Berend, T. Tassa, Improved bounds on bell numbers and on moments of sums of random variables, Probability and Mathematical Statistics 30 (2010) 185–205.

[34] I. R. Savage, Probability inequalities of the tchebycheff type, Journal of Research of the National Bureau of Standards-B. Mathematics and Mathematical Physics B 65 (1961) 211–222.

[35] T. Roughgarden, Intrinsic robustness of the price of anarchy, Journal of the ACM (JACM) 62 (2015) 32. Preliminary version in STOC'09.

[36] S. Aland, D. Dumrauf, M. Gairing, B. Monien, F. Schoppmann, Exact price of anarchy for polynomial congestion games, in: Annual Symposium on Theoretical Aspects of Computer Science, Springer, 2006, pp. 218–229.

[37] D. Monderer, L. S. Shapley, Potential games, Games and economic behavior 14 (1996) 124–143.

[38] E. Anshelevich, A. Dasgupta, J. Kleinberg, E. Tardos, T. Wexler, T. Roughgarden, The price of stability for network design with fair cost allocation, SIAM Journal on Computing 38 (2008) 1602–1623. Preliminary version appears in Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, 2004.