Truthful and Welfare-maximizing Resource Scheduling with Application to Electric Vehicles

Ramsundar Anandanarayanan
IIT Bombay
Mumbai, India
ramsundar@cse.iitb.ac.in

Swaprava Nath
IIT Bombay
Mumbai, India
swaprava@cse.iitb.ac.in

Prasant Misra
TCS Research and IISc Bengaluru
Bengaluru, India
prasant.misra@tcs.com

ABSTRACT

We consider the problem of scheduling resources with monetary transfers among agents in a setting where multiple outlets can dispense these resources at different rates within fixed time-slots. This problem is motivated by applications like the electric vehicle (EV) charging that require power and are available for charging within the convenient time window of its owners. The agents' valuations depend on the contiguous collection of time slots at a given outlet that dispense the resource to them. We show that for *monotone* and its special sub-class of *dichotomous* valuations, computing the *social welfare-maximizing allocation* is NP-hard, even if there is only one outlet. For monotone valuations, we provide a randomized 2-approximation mechanism that is *truthful in dominant strategies* and *individually rational* for a single outlet and a randomized $O(\sqrt{|S|})$-approximation algorithm with the same properties for multiple outlets (S is the set of time-slots). However, for *single-minded* valuations, the welfare maximization problem for multiple outlets is in $\mathbb{P}$. This allows us to use standard mechanisms like VCG to ensure truthfulness and individual rationality in such a setting.

KEYWORDS

Resource Scheduling, Electric Vehicle Charging, Social Welfare Maximization, Mechanism Design

ACM Reference Format:
Ramsundar Anandanarayanan, Swaprava Nath, and Prasant Misra. 2025. Truthful and Welfare-maximizing Resource Scheduling with Application to Electric Vehicles. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 12 pages.

1 INTRODUCTION

Allocating resources efficiently among time-constrained consumers is a critical challenge across industries. For instance, power grids distribute electricity to many organizations operating heavy electrical equipments, while food delivery apps allocate incoming food orders to delivery agents. Similarly, a *charging point operator* (CPO) of electric vehicles (EVs) manages multiple charging stations, with EVs coming as consumers to charge the vehicles. These diverse scenarios share some common factors: (a) resources take significant time to serve each consumer, (b) consumers have specific preferences over the schedules (e.g., delivery agents preferring certain geographical areas and times or EVs needing charging at specific times and locations), and (c) payments are allowed, with the option of price discrimination among consumers. This creates a generalized framework of mechanism design for *resource scheduling with monetary transfers*, where the planner (e.g., a grid manager, delivery app, or CPO) must adhere to certain key principles. The first is *truthfulness*, that ensures industries to reveal true electricity demand or delivery agents to disclose real preferences. The second is *social welfare maximization*, that aims to maximize the collective consumer satisfaction.

In this paper, we consider the problem of truthful and welfare-maximizing resource scheduling problem with payments where the allocation and payment decisions are made at certain given epochs, e.g., at certain hours of the day depending on the number of consumer requests that arrive at that time epoch. We keep the electric vehicle allocation as our running example and develop the theory and notation accordingly. However, we want to emphasize that the same framework can also be easily adapted to any resource allocation problem discussed above. The special structure of this setting allows us to show that maximizing welfare is computationally hard and therefore needs to be approximated. However, for such approximated welfare mechanisms, non-trivial allocation and payment rules need to be designed to ensure truthfulness. In this paper, we consider a static setup where the consumers report their values and the mechanism decides the allocation and payments. Making the decision epochs sufficiently fine, a close approximation of the dynamic decision problem can be obtained. Even in such a case, we find the problem to be quite challenging and therefore a general analysis of an online resource scheduling problem is left as a future exercise for these settings.

1.1 Related Work

The literature on truthful resource scheduling is diverse primarily because of the history and application domains of such problems.

The first strand of this literature comes from the classical domain of machine scheduling. In this domain, the primary objective is to minimize *makespan* [1, 9, 10, 14, 16, e.g.]. The question of social welfare has been addressed sporadically, e.g., Koutsoupias [23] defined it as the negation of the sum of executing times of all machines and provided approximation to the optimal.

The second strand comprises of discrete interval scheduling problems, where a set of jobs can be executed over multiple machines and the the goal is to *maximize the weighted sum* of executed jobs [4, 5, 7, 31]. While this literature focuses on providing approximation schemes, it does not consider the objective of truthfulness or capture the rich valuation structure of agents.

The third strand of literature addresses the axiomatic questions of properties such as truthfulness, budget balance, independence

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), A. El Fallah Seghrouchni, Y. Vorobeychik, S. Das, A. Nowe (eds.), May 19 – 23, 2025, Detroit, Michigan, USA.

of irrelevant alternatives [19], risk aversion [24], and provide characterization results. These literature does not consider the computational complexity of the mechanisms that yield these properties. Kress et al. [25] and Kolen et al. [21] provide nice surveys of these three strands.

The fourth strand is from the algorithmic mechanism design viewpoint, where the computational questions in welfare maximization and truthfulness are considered together. If the time is discrete and each agent desires a set of contiguous time-slots of a resource, then this problem reduces to a special combinatorial allocation problem. We consider this setting in our paper and, therefore, this literature is the most relevant one. For mechanisms with payments, VCG [11, 18, 32] is the most widely used one for guaranteeing truthfulness and welfare maximization. However, it requires to compute the *optimal social welfare* (OSW) allocation in order to ensure truthfulness. The OSW problem in combinatorial auctions is known to be NP-Hard [12, 28], even in the case where agents are single-minded. In addition, approximating the social welfare to a factor within $k^{1/2-\epsilon}$ (where k is the number of objects or goods) is also NP-Hard [12, 28]. In the case of multi-unit combinatorial auction under the constraint that no object is allocated more than y times (hence considered as the number of units of every object) and every agent gets at most one bundle, approximating social welfare within a factor of $O(k^{1-\epsilon/y+1})$ is NP-Hard [6]. Thus, the approach taken in the literature is to approximately achieve social welfare ensuring truthfulness and individual rationality (IR).

There are several algorithms that provide $O(k^{1/y+1})$ approximation guarantee [8, 22, 29] to the social welfare maximization problem. For general monotone valuations, only [6, 26] are known to be truthful. Note that the VCG mechanism with approximate social welfare does not generally guarantee truthfulness [27]. Bartal et al. [6] give a deterministic $O(yk^{1/y-2})$ approximation algorithm that ensures truthfulness and IR. However, this approach only works for $y \geqslant 3$. In contrast, Lavi and Swamy [26] provide a randomized mechanism that uses VCG in a computationally tractable manner and achieves $O(k^{1/y+1})$ approximation guarantee for the social welfare $\forall y \geqslant 1$, ensuring truthfulness and IR. Several other works address the single minded buyers [2, 8, 27, e.g.], single-valued buyers [3], and subadditive valuations [15].

Our resource scheduling problem and the results are distinct from that in the literature. In our setup, every outlet-timeslot pair is a good, and there is only one unit of this available. Hence, this naturally falls in the setup of [26]. However, applying their method directly in our setting where the number of goods is $|S||M|$, where S and M are the set of timeslots and outlets respectively, we can achieve an approximation guarantee of $O(\sqrt{|S||M|})$. However, using the structure of the problem we consider, we provide an improved $O(\sqrt{|S|})$-approximation for multiple outlets and a 2-factor approximation for the single outlet (see Section 1.2 and Table 1 for more details).

1.2 Our Contributions

In this paper, we consider the consumers (agents) who are looking for contiguous time-slots to consume resource at a rate that is fixed for that consumer-outlet pair. They have different valuations for different such contiguous slots, e.g., infeasible slots have zero values. The planner wants to allocate the resources to maximize the sum of the valuations of all the agents (i.e., welfare-maximizing) while ensuring that agents are truthful. Monetary transfers can be used to achieve this goal. In this setting, our contributions can be summarized as follows.

- For *monotone* and *dichotomous* valuations, computing the welfare-maximizing allocation is NP-Hard even for a single outlet (Theorems 1 and 4).
- When considering a single outlet for the above valuations, we provide a 2-approximate welfare-maximizing mechanism that satisfies *truthfulness in dominant strategies* and *individual rationality* (Theorem 3).
- For the case with multiple outlets, we provide a $O(\sqrt{|S|})$-approximate welfare-maximizing mechanism (S is the set of time-slots) that is also *truthful in dominant strategies* and *individually rational* (Theorem 2).
- For *single minded* agents (agents who get a fixed positive valuation only when a specific set of contiguous slots at a particular outlet is allocated to them) with multiple outlets, we show that the welfare-maximization problem can be reduced to a linear program and hence is efficiently solvable (Theorem 5). Therefore, truthfulness and IR can be ensured via the classic VCG mechanism.

Our results are summarized in Table 1. The different cases are motivated by the practical limitations of resource scheduling problems. For instance, if the resource outlets are not interconnected and cannot make a simultaneous decision over all the requests coming at all outlets, the planner can run the algorithm individually at every outlet and guarantee a constant factor approximation. For the relatively difficult problem of a single CPO (in the context of EV charging) jointly allocating the consumers over multiple outlets, the approximation guarantee becomes worse because the underlying optimization problem gets harder. For certain special settings, e.g., every EV has a desired (outlet, interval) and does not consider any other (outlet, interval), the problem becomes computationally easy.

2 PRELIMINARIES

In this section, we formally describe the resource scheduling problem using electric vehicle charging as the motivation. We consider monetary transfers with an aim to maximize *social welfare* in a *dominant strategy truthful* manner and design mechanisms that ensure *participation guarantee*.

2.1 Model

Let $N = \{1, 2, 3, \ldots, n\}$ denote the set of electric vehicles (EVs) requesting to charge themselves (e.g., via a mobile application) from a single *charging point operator* (CPO) who owns charging *stations* in a region. Each station has several charging *outlets* and every outlet has a fixed maximum charging rate at which it can charge an EV. We collect together all the outlets in the region (irrespective of whether they are at the same station) and denote $M = \{1, 2, 3, \ldots, m\}$ to be the set of all outlets that the CPO owns in that region. Therefore, each outlet $k \in M$ has a maximum charging rate r_k^{Ch}. EVs have preferences over different outlets based on their location and charging rates. For instance, an EV would prefer to charge at a charging outlet based on its proximity, the rate of charging (fast/slow), pricing,

Table 1: Summary of results.

Valuation	*# outlets*	*Complexity*	*Mechanism*	*Guarantee*	*DSIC*	*IR*
Monotone	multiple	NP-Hard	RAE (Algorithm 1) + Algorithm 2 in separation oracle	$O(\sqrt{\lvert S\rvert})$-approx	✓	✓
Monotone	single	NP-Hard	RAE (Algorithm 1) + Algorithm 3 in separation oracle	2-approx	✓	✓
Dichotomous	single	NP-Hard	RAE (Algorithm 1) + Algorithm 3 in separation oracle	2-approx	✓	✓
Single-minded	multiple	Polynomial time	DAE (Algorithm 4)	Optimal	✓	✓

and various similar factors. We consider CPO as the planner whose goal is to allocate EVs (agents) to outlets and decide an appropriate pricing scheme for the allocation. Since charging an EV requires time, the planner also needs to factor in the time allocated while assigning agents to the outlets. Consider a time horizon (e.g. the working hours of a day) which is discretized into s slots of equal duration denoted by $S = \{1, 2, 3, \ldots, s\}$. Each slot $j \in S$ is an indivisible unit representing the minimum amount of time an agent must charge once plugged in at an outlet $k \in M$. The planner solves the problem of allocating agents to slots at the outlets given a set of charging requests by EVs. Hence, the resource that each EV can be allocated is a pair of 'time slot and outlet'.

Each EV $i \in N$ is allocated a collection of (slot, outlet) pairs which we will be calling a *bundle*. Since no EV can charge at two different outlets at the same time slot, we denote a bundle by $b \in (M \cup \{0\})^S$, which implies that a bundle is a vector of length $|S|$ where the coordinates correspond to the time slots and the value at each coordinate represents the assigned outlets $\{1, 2, \ldots, m\} \cup \{0\}$ at the corresponding time-slot. The special outlet 0 denotes 'unassigned' at that slot. We assume that each EV wants to be assigned contiguous time-slots exactly at one outlet. This assumption captures the practical problem of repeatedly switching between outlets or stop-starting charging, which are infeasible in practice. This implies that we are only considering the types of bundles that satisfy the following: (i) $\forall i, j \in S$, if $b_i, b_j \neq 0$, then $b_i = b_j$ and (ii) there exists i^*, j^*, s.t. $b_i = 0, \forall i < i^*, i > j^*$ and $b_i \neq 0, \forall i^* \leqslant i \leqslant j^*$. The first condition ensures that the bundle consists of time slots at exactly one outlet, while the second condition imposes the contiguity requirement. Denote the set of all such *feasible* bundles by B.

Each agent $i \in N$ comes with a type $\theta_i : B \to \mathbb{R}$, where $\theta_i(b)$ represents the satisfaction of agent i for bundle $b \in B$. We assume that the types satisfy monotonicity unless stated otherwise, i.e., for all $b, b' \in B$ where b' is a sub-bundle of b (b' is a *sub-bundle* of b if b contains all the allocated time-slots of b' at the same outlet, and is represented as $b' \sqsubseteq b$)

$$\theta_i(b') \leqslant \theta_i(b), \forall i \in N, \text{ and } \theta_i(\{0\}^{|S|}) = 0, \forall i \in N. \quad (1)$$

Note that, in this definition, θ_is account for the agent i's preference over outlets (fast/slow chargers), time slots (e.g., their preferred arrival and departure), and their charge demand, in a consolidated manner. Since θ_i is agent i's private information, we need mechanisms to truthfully elicit this information to take an *efficient* decision. We use θ_{-i} to represent the types of agents other than i and $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ to denote the type profile. Set Θ_i denotes the *monotone* type set of agent i and $\Theta = \prod_{i \in N} \Theta_i$ denotes the set of type profiles. When requesting for charging services, each agent reports $\hat{\theta}_i$ which could be different from their true type θ_i. The planner needs to design a mechanism (the allocation and payment schemes) using the reported type vector $\hat{\theta}$. Note, when time slots are categorized into different time-periods of the day, e.g., morning, afternoon, evening, night, such mechanisms can allocate slots to all the EVs that placed a request before the time-period started.

An allocation is represented as $x = [x(i, b), \forall i \in N, b \in B]$, where $x(i, b) = 1$ when agent i is allocated $b \in B$, and $x(i, b) = 0$ otherwise. We call an allocation feasible if it satisfies the following: (a) every agent is allocated at most one bundle, i.e., $\sum_{b \in B} x(i, b) \leqslant 1, \forall i \in N$, and, (b) no more than a single unit of any "time slot, outlet" pair is allocated. Let $B_{jk} = \{b \in B : b_j = k\}$ denote the set of bundles in which the pair $(j, k), j \in S, k \in M$, exists. Then $\sum_{b \in B_{jk}} \sum_{i \in N} x(i, b) \leqslant 1, \quad \forall j \in S, k \in M$. All allocations that satisfy conditions (a) and (b) are called *feasible allocations* and the set of all feasible allocations is denoted by X.

An allocation function $f : \Theta \to X$ is a mapping that yields a feasible allocation $f(\theta) \in X$ for every type profile $\theta \in \Theta$. The *valuation* of agent $i \in N$ is described by $v_i : X \times \Theta_i \to \mathbb{R}$, which for a given $\theta_i \in \theta$ and a feasible allocation $x \in X$ gives a value $v_i(x, \theta_i) = \sum_{b \in B} \theta_i(b)\ x(i, b)$. Note, if θ_i satisfies monotonicity then so does the valuation v_i. Every EV is also asked a payment for a given allocation. A payment function for agent i is given by $\pi_i : \Theta \to \mathbb{R}$ which maps the reported type profile $\theta \in \Theta$ to a real number.

Given the above formulation, the utilities of the agents take a quasi-linear form. Formally, given the reported type profile of agents $\hat{\theta}$, an allocation function f and payment functions $\pi_i, \forall i \in N$, the utility of agent i when its true type is θ_i is given by: $u_i((f(\hat{\theta}), \pi(\hat{\theta})), \theta_i) = v_i(f(\hat{\theta}), \theta_i) - \pi_i(\hat{\theta})$.

Note that in the definitions above, we defined the allocation and the payments to be deterministic. But more generally, the planner can also output randomized allocation and payments. A randomized allocation can be seen as a probability distribution over all deterministic allocations in X. Denote the set of all randomized allocations by $\Delta_X = \{\lambda \in [0, 1]^{|X|} : \sum_{x \in X} \lambda_x = 1 \text{ and } \lambda_x \geqslant 0, \forall x \in X\}$, where λ represents a randomized allocation and λ_x denotes the probability of choosing the deterministic allocation $x \in X$. Note, Δ_X is the convex hull of the set X. Given this, we extend the allocation function $f : \Theta \to \Delta_X$ to be a mapping which yields a randomized allocation $f(\theta) \in \Delta_X$ for a given type profile $\theta \in \Theta$. We also extend the the valuation function of agent i, $v_i : \Delta_X \times \Theta_i \to \mathbb{R}$ to have all randomized allocations Δ_x in the domain. Thus, with a slight abuse of notation we denote $v_i(\lambda, \theta_i) = \sum_{x \in X} \lambda_x\ v_i(x, \theta_i) = \sum_{x \in X} \lambda_x\ \sum_{b \in B} \theta_i(b)\ x(i, b)$ to be the expected valuation of agent i for the randomized allocation $\lambda \in \Delta_X$ when its type is θ_i. Likewise, the payment $\pi_i(\theta)$ denotes the expected payment to be made by agent i. For a given the reported type profile $\hat{\theta}$, this gives us the expected utility of an agent i when its true type is θ_i as follows: $u_i((f(\hat{\theta}), \pi(\hat{\theta})), \theta_i) = \mathbb{E}[v_i(f(\hat{\theta}), \theta_i) - \pi_i(\hat{\theta})]$, where the expectation is taken w.r.t. the randomized allocation $f(\hat{\theta})$ and randomized payment $\pi_i(\hat{\theta})$.

In summary, the planner needs to design a social choice function or a mechanism (f, π) such that several desirable properties are satisfied. We define the desirable properties in the following section.

2.2 Design Desiderata

In this paper, our objective is to maximize social welfare through a mechanism that is dominant strategy incentive compatible and individually rational. These properties are defined as follows.

Definition 1 (Efficiency). A deterministic mechanism (f, π) maximizes social welfare and therefore is *efficient* if for every $\theta \in \Theta$, $f(\theta) = \arg\max_{x \in X} \sum_{i \in N} \sum_{b \in B} \theta_i(b)\ x(i, b)$. Correspondingly, a randomized mechanism (f, π) is efficient if for every $\theta \in \Theta$, $f(\theta) = \arg\max_{\lambda \in \Delta_X} \sum_{x \in X} \lambda_x\ \sum_{i \in N} \sum_{b \in B} \theta_i(b)\ x(i, b)$.

The next property incentivizes each agent to participate in the game ensuring that their utility is non-negative for every type profile.

Definition 2 (Individual Rationality (IR)). A deterministic mechanism (f, π) is *individually rational (IR)* if for every $\theta \in \Theta$ and for every $i \in N$, $v_i(f(\theta), \theta_i) - \pi_i(\theta) \geqslant 0$. Likewise, a randomized mechanism (f, π) is *ex-post* individually rational if for every $\theta \in \Theta$ and for every $i \in N$, $v_i(x, \theta_i) - p_i \geqslant 0$ for every sample x and p_i drawn from $f(\theta)$ and $\pi_i(\theta)$ respectively.

Finally, since the planner's decision is dependent on the agents' reported types $\hat{\theta}$, we need to incentivize them to report it truthfully.

Definition 3 (Dominant Strategy Incentive Compatible (DSIC)). A deterministic mechanism (f, π) is *dominant strategy incentive compatible (DSIC)* if for every agent $i \in N, \forall\ \theta_i, \tilde{\theta}_i \in \Theta_i$, and $\forall\ \theta_{-i} \in \Theta_{-i}, v_i(f(\theta_i, \theta_{-i}), \theta_i) - \pi_i(\theta_i, \theta_{-i}) \geqslant v_i(f(\tilde{\theta}_i, \theta_{-i}), \theta_i) - \pi_i(\tilde{\theta}_i, \theta_{-i})$. Correspondingly, a randomized mechanism (f, π) is DSIC if $\forall i \in N, \forall \theta_i, \tilde{\theta}_i \in \Theta_i$, and $\forall \theta_{-i} \in \Theta_{-i}$,

$$\mathbb{E}[v_i(f(\theta_i, \theta_{-i}), \theta_i) - \pi_i(\theta_i, \theta_{-i})] \geqslant \mathbb{E}[v_i(f(\tilde{\theta}_i, \theta_{-i}), \theta_i) - \pi_i(\tilde{\theta}_i, \theta_{-i})].$$

In the sections that follow, we focus on mechanisms that achieve the above set of properties in a computationally efficient manner.

3 PROBLEM SETUP

In this section, we present the central problem of the paper. The social welfare maximization problem for the general monotone valuations is given as follows.

$$\begin{aligned} \max \quad & \sum_{i \in N} \sum_{b \in B} \theta_i(b)\ x(i, b) \\ \text{s.t.} \quad & \sum_{b \in B_{jk}} \sum_{i \in N} x(i, b) \leqslant 1, \forall j \in S, k \in M, \\ & \sum_{b \in B} x(i, b) \leqslant 1, \forall i \in N, \\ & x(i, b) \in \{0, 1\}, \forall i \in N, b \in B. \end{aligned} \tag{2}$$

We show that it is NP-Hard in various reasonable settings and provide approximate solutions to welfare while maintaining the DSIC and IR properties.

4 MONOTONE VALUATIONS

We first show that the social welfare maximization problem for monotone valuations given by ILP (2) is NP-Hard. We prove this via a polynomial reduction from the *Job Interval Selection Problem* [12, 31] which is known to be NP-Complete.

Theorem 1. *For monotone valuations and for a given K, the decision problem of whether the optimal allocation to the EV charging problem has a social welfare of at least K is NP-complete even when the number of outlets $|M| = 1$.*

Due to paucity of space, we move the proof of this result and the proofs some other results to the supplementary material.

4.1 Mechanism for multi-outlet scenario

Given the above result, the VCG mechanism is intractable for our setup. Thus, we focus on maximizing social welfare approximately. To obtain this, we use the classic VCG mechanism in the *fractional* space to obtain an optimal *fractional* allocation that is *efficient* and payments that ensure DSIC and IR. A randomized mechanism is then constructed such that the randomized allocation is a convex decomposition of the fractional allocation scaled by a factor α and the expected payment of every agent is set to be the α-scaled VCG payment calculated in the fractional space. Note that to get the convex decomposition of the α-scaled fractional allocation, an α-approximation algorithm that gives guarantees w.r.t. the fractional optimal solution for every monotone valuation is required. We provide a greedy algorithm with $O(\sqrt{|S|})$-approximation factor to do this. The above method approximates the social welfare to within a factor of $O(\sqrt{|S|})$ and retains DSIC and IR via VCG in the fractional space. We call this method Randomized Allocatively Efficient (RAE) mechanism, which is detailed out in Algorithm 1. Note that Algorithm 1 takes an α-approximation algorithm $\mathcal{A}$ as input. It internally employs the ellipsoid method with a *separation oracle* [33] that uses the approximation algorithm $\mathcal{A}$. Thus, Algorithm 1 acts as a template, where the variable $\mathcal{A}$ can be set appropriately. We show in the following result how we can achieve all desirable properties.

Theorem 2. *For monotone valuations and mutiple outlets, the* RAE *mechanism (Algorithm 1) that uses Algorithm 2 as $\mathcal{A}$ in the separation oracle approximates the social welfare within a factor of $O(\sqrt{|S|})$ and ensures DSIC and IR.*

The general technique of constructing a randomized mechanism using VCG in a tractable manner was originally proposed by Lavi and Swamy [26] in the context of combinatorial auctions. We adapt their general technique to ensure DSIC and IR for monotone valuations, but improve on the approximation guarantees for this setup. In particular, the $O(\sqrt{\rho})$ approximation algorithm (ρ is the number of goods) proposed in [26] translates to an approximation factor of $O(\sqrt{|S||M|})$ in our setting since every (j, k) pair, where $j \in S, k \in M$ can be seen as a good. However, Theorem 2 provides an improved $O(\sqrt{|S|})$-approximation for the multi-outlet case. Later, we improve it to a constant factor for a single outlet.

For given reported types $\hat{\theta}$, Algorithm 1 first solves the LP relaxation of ILP (2) given by LP (3) to obtain an optimal fractional allocation. This can be computed in polynomial time since the number of variables and constraints in LP (3) are polynomial in

Algorithm 1: RAE Mechanism

Input: Agent reports $\hat{\theta}$ and an α-approximation algorithm $\mathcal{A}$ ($\alpha > 1$) that provides an integer solution with a value of at least $1/\alpha$ times the value of the fractional optimal for any monotone $\hat{\theta}$.

Output: A randomized allocation $f(\hat{\theta}) \in \Delta_X$ and randomized payments $\pi_i(\hat{\theta}), \forall i \in N$.

Solve LP (3) to get an optimal fractional allocation $x^{\text{fr}}(\hat{\theta})$.

Set payments $p_i^{\text{fr}}(\hat{\theta})$ for every agent i using VCG in the fractional space X^{fr} as given by Equation (4).

Scale $x^{\text{fr}}(\hat{\theta})$ and $p_i^{\text{fr}}(\hat{\theta}), \forall i \in N$ by α.

Using `GetConvexDecomposition`$(x^{\text{fr}}(\hat{\theta}), \mathcal{A})$, construct a convex decomposition $x^{\text{fr}}(\hat{\theta})/\alpha = \sum_{x^I \in X} \lambda^*_{x^I} x^I$ with polynomially many $\lambda^*_{x^I} > 0$.

Set the randomized allocation $f(\hat{\theta})$ and payments $\pi_i(\hat{\theta}), \forall i \in N$ according to Equation (8).

return $f(\hat{\theta}), \pi(\hat{\theta})$

Procedure `GetConvexDecomposition`$(x^{\text{fr}}(\hat{\theta}), \mathcal{A})$:

  Solve the dual LP (6) using ellipsoid method with `SeparationOracle()` that uses $\mathcal{A}$ and $x^{\text{fr}}(\hat{\theta})$. This identifies an LP that is equivalent to LP (6) but with only polynomial no. of constraints from $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\ w(i,b) + z \leqslant 1, \forall x^I \in X$.

  Solve the primal LP (5) by considering polynomially many variables corresponding to the above identified constraints to get the optimal solution λ^*.

  return λ^*

Procedure `SeparationOracle()`:

  Input: $x^{\text{fr}}(\hat{\theta})$, an α-approximation algorithm $\mathcal{A}$, and any point (w, z), where $w = [w(i,b), \forall i \in N, b \in B]$ is unconstrained.

  Output: A separating hyperplane which is used to cut the ellipsoid in a given iteration.

  if $z + \frac{1}{\alpha} \sum_{i \in N} \sum_{b \in B} x^{\text{fr}}(i,b)\ w(i,b) > 1$ **then**

    Using $\mathcal{A}$, get an $x^I \in X$ s.t. $\sum_{i \in N} \sum_{b \in B} x^I(i,b) w(i,b) \geqslant \frac{1}{\alpha} \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i,b) w(i,b)$.

    Using the above inequality and the condition in the if statement, we get a violated constraint $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\ w(i,b) + z > 1$ of the LP (6) for the point (w, z).

    return $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\ w(i,b) + z = 1$

  else

    return $z + \frac{1}{\alpha} \sum_{i \in N} \sum_{b \in B} x^{\text{fr}}(i,b)\ w(i,b) = 1$

  end

$|N|$, $|M|$, and $|S|$. Particularly, note that number of bundles in $B = O(|M||S|^2)$ since EVs are assigned contiguous time slots at exactly any single outlet. Denote the optimal solution of LP (3) by $x^{\text{fr}}(\hat{\theta})$. Wherever clear from context, we will use x^{fr} instead of $x^{\text{fr}}(\hat{\theta})$ for brevity. Note that, x^{fr} can be seen as a fractional allocation, where $x^{\text{fr}}(i,b) \in [0,1], \forall i \in N, b \in B$ denotes the fraction of bundle b allocated to agent i. Denote X^{fr} to be the set of all feasible fractional allocations[1].

$$\begin{aligned} \max \quad & \sum_{i \in N} \sum_{b \in B} \hat{\theta}_i(b)\ x(i,b) \\ \text{s.t.} \quad & \sum_{b \in B_{jk}} \sum_{i \in N} x(i,b) \leqslant 1, \forall j \in S, k \in M \\ & \sum_{b \in B} x(i,b) \leqslant 1, \forall i \in N \\ & x(i,b) \geqslant 0, \forall i \in N, b \in B \end{aligned} \tag{3}$$

The payment of every agent i is then given by the VCG payment in the fractional space X^{fr} as follows.

$$p_i^{\text{fr}}(\hat{\theta}) = \max_{x \in X^{\text{fr}}} \sum_{i' \in N \setminus \{i\}} \sum_{b \in B} \hat{\theta}_{i'}(b) x(i', b) - \sum_{i' \in N \setminus \{i\}} \sum_{b \in B} \hat{\theta}_{i'}(b) x^{\text{fr}}(i', b) \tag{4}$$

The *fractional mechanism* $(x^{\text{fr}}, p^{\text{fr}})$ guarantees DSIC and IR since it is the VCG mechanism in the fractional space X^{fr}. Note that, even if the allocation and the payments are scaled by some $\alpha > 1$, i.e., $x^{\text{fr}}(i,b)/\alpha, \forall i \in N, b \in B$ and $p_i^{\text{fr}}(\hat{\theta})/\alpha, \forall i \in N$, DSIC and IR still hold. This is due to v_i's linearity in x^{fr} i.e., $v_i(x^{\text{fr}}(\hat{\theta}), \theta_i) = \sum_{b \in B} \theta_i(b)\ x^{\text{fr}}(i,b), \forall x^{\text{fr}}(\hat{\theta}) \in X^{\text{fr}}$. Note that we also overload v_i for a fractional allocation. From the above discussion, we get the following lemma.

Lemma 1. *For every $\theta \in \Theta$, a mechanism that outputs the fractional allocation $x^{\text{fr}}(\theta)/\alpha$ and the VCG payments $p^{\text{fr}}(\theta)/\alpha$, for every $\alpha > 1$ is DSIC and IR in X^{fr}.*

However, note that the mechanism $(x^{\text{fr}}(\hat{\theta})/\alpha, p^{\text{fr}}(\hat{\theta})/\alpha)$ cannot be implemented since it gives a fractional allocation. For this reason, we construct a convex decomposition of $x^{\text{fr}}(\hat{\theta})/\alpha = \sum_{x^I \in X} \lambda_{x^I} x^I$ to obtain a randomized allocation $\lambda \in \Delta_X$ that has only polynomially many $\lambda_{x^I} > 0$. The problem of finding such a decomposition can be formulated as the following linear program.

$$\begin{aligned} \min \quad & \sum_{x^I \in X} \lambda_{x^I} \\ \text{s.t.} \quad & \sum_{x^I \in X} \lambda_{x^I} x^I(i,b) = x^{\text{fr}}(\hat{\theta})(i,b)/\alpha, \forall i \in N, b \in B \\ & \sum_{x^I \in X} \lambda_{x^I} \geqslant 1 \\ & \lambda_x \geqslant 0, \forall x^I \in X. \end{aligned} \tag{5}$$

If we can show that the optimal value of LP (5) is 1 (for some fixed $\alpha > 1$), then we get the required convex decomposition of the fractional allocation. This gives a randomized allocation that approximates the social welfare to within a factor of α since we have $\sum_{x^I \in X} \lambda_{x^I} \sum_{i \in N} \sum_{b \in B} \theta_i(b)\ x^I(i,b) = \sum_{i \in N} \sum_{b \in B} \theta_i(b) \sum_{x^I \in X} \lambda_{x^I}\ x^I(i,b) = \sum_{i \in N} \sum_{b \in B} \theta_i(b)\ x^{\text{fr}}(i,b)/\alpha$. In addition, using the properties of *fractional mechanism* (Lemma 1) we can also ensure DSIC and IR. We show that for a particular

[1] Note that, X^{fr} is the feasible region of LP (3). This may be different from the convex hull of X, since the corner points of the feasible region X^{fr} may not be deterministic allocations. If that happens to be the case, then ILP (2) is solvable in polynomial time.

choice of α we can guarantee an optimal value of 1 for LP (5) for every monotone $\hat{\theta}$. This also gives a $\lambda \in \Delta_X$ having only polynomially many $\lambda_{x^I} > 0$, (for $x^I \in X$) in polynomial time.

Observe that LP (5) can have exponentially many variables, since the number of deterministic allocations for a given instance of our EV charging problem can be exponential in the number of agents, outlets and time slots. For this reason, we consider its dual LP (6).

$$\begin{aligned} \max \quad & z + \frac{1}{\alpha} \sum_{i \in N} \sum_{b \in B} x^{\text{fr}}(\hat{\theta})(i,b)\ w(i,b) \\ \text{s.t.} \quad & z \geqslant 0 \\ & \sum_{i \in N} \sum_{b \in B} x^I(i,b)\ w(i,b)\ + z \leqslant 1, \forall x^I \in X \end{aligned} \tag{6}$$

The dual program has a polynomial number of variables and an exponential number of constraints. But an LP with exponentially many constraints can be solved in polynomial time using the ellipsoid method if one can construct an efficient *separation oracle* [17, 33]. This is because the ellipsoid method solves an LP without the explicit description of the program itself. For our dual LP (6), an α-approximation algorithm that provides an integer solution with a value of at least $1/\alpha$ times the value of the optimal fractional solution of LP(3) for every *monotone* $\theta \in \Theta$ can be used to construct such an efficient separation oracle (see Algorithm 1). This has two implications for the choice of α: (1) We require α to be at least the *integrality gap (IG)*; (2) To obtain the convex decomposition we need an accompanying α-approximation algorithm which provides an integer solution having guarantees w.r.t. to the fractional optimal for every *monotone* $\theta \in \Theta$. The *integrality gap* is the maximal ratio between the optimal fractional solution and optimal integer solution of the social welfare maximization problem across all *monotone* valuations as defined below.

$$\text{IG} := \sup_{\theta \in \Theta^{\text{MONO}}} \frac{\max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} \theta_i(b)\ x(i,b)}{\max_{x^I \in X} \sum_{i \in N} \sum_{b \in B} \theta_i(b)\ x^I(i,b)} \tag{7}$$

If α was less than the integrality gap, then by the above definition there exists a $\theta \in \Theta$ for which no integer solution $x^I \in X$ gives $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\ \theta_i(b) \geqslant \frac{1}{\alpha} \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i,b)\ \theta_i(b)$.

An important point to highlight is that in the separation oracle the α-approximation algorithm is used to provide an integer solution with a value of at least $1/\alpha$ times the value of the fractional optimal for any unconstrained w (not monotone w). However, for packing[2] problems an α-approximation algorithm that works for *monotone* w can also be used to provide the required integer solution for any unconstrained w. This is stated as the following lemma. We note that for packing problems an α-approximation algorithm that provides an integer solution with a value of at least $1/\alpha$ times the value of the fractional optimal for every *positive* $\theta \in \Theta$ can also be used for the separation oracle. For more details, we refer the reader to the supplementary material.

Lemma 2. *For any unconstrained $w = [w(i,b), \forall i \in N, b \in B]$, an α-approximation algorithm that provides an integer solution with a value of at least $1/\alpha$ times the value of the fractional optimal of Equation* (3) *for every monotone $\theta \in \Theta$ can be used to construct an $x^I \in X$ in polynomial time such that $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\ w(i,b) \geqslant 1/\alpha \cdot \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i,b)\ w(i,b)$.*

The ellipsoid method with this efficient separation oracle identifies an LP that is equivalent[3] to the dual LP (6), but with only polynomially many constraints from $\sum_{i \in N} \sum_{b \in B} x(i,b)\ w(i,b) + z \leqslant 1, \forall x \in X$. These constitute the set of violated constraints returned by the separation oracle which is used to cut the ellipsoid at each iteration. The primal LP (5) is then solved by considering only polynomially many variables corresponding to these violated constraints to get the optimal solution λ^*. Since the ellipsoid method runs in polynomially many iterations, we get the decomposition in polynomial time. It can also be shown that the optimal value of the dual, and hence, the primal is 1 which yields the desired convex combination. This gives us the following result.

Lemma 3. *A decomposition of $x^{\text{fr}}(\hat{\theta})/\alpha = \sum_{x^I \in X} \lambda^*_{x^I} x^I$ with only polynomially many $\lambda^*_{x^I} > 0$ and $\sum_{x^I \in X} \lambda^*_{x^I} = 1$ can be obtained in polynomial time.*

Finally, the allocation and payments of the randomized mechanism are given by Equation (8). The allocation $f(\hat{\theta})$ is set to the randomized allocation λ^* and the expected payment $\pi_i(\hat{\theta})$ of agent i is set to $p_i^{\text{fr}}(\hat{\theta})/\alpha$. This ensures DSIC, IR, and gives an α-approximation to the social welfare. In summary, the above discussion highlights that for packing problems any α-approximation that gives guarantees w.r.t to the fractional optimal for *monotone* inputs can be used to give a α-approximation mechanism that is DSIC and IR.

$$\begin{aligned} & f(\hat{\theta}) = \{\lambda^*_{x^I}, \forall x^I \in X\} \\ & \pi_i(\hat{\theta}) = \begin{cases} \frac{p_i^{\text{fr}}(\hat{\theta})/\alpha}{v_i(f(\hat{\theta}),\hat{\theta}_i)}\ v_i(x^I, \hat{\theta}_i), \text{ if } v_i(f(\hat{\theta}), \hat{\theta}_i) > 0\ \&\ x^I \in X \text{ is sampled.} \\ 0, \text{ Otherwise.} \end{cases} \end{aligned} \tag{8}$$

Lemma 4. *The randomized mechanism (f, π) given by Equation* (8) *is DSIC, IR, and approximates the social welfare to within a factor of α.*

For the α-approximation algorithm, we leverage [27] to provide a greedy strategy that gives an improved $O(\sqrt{|S|})$-approximation guarantee w.r.t. the optimal fractional solution in our setup. Note that the approximation guarantee depends only on the number of time slots $|S|$ and not on the outlet count $|M|$.

Lemma 5. *For monotone valuations and multiple outlets, the greedy Algorithm 2 approximates the social welfare of the EV charging problem within a factor of $O(\sqrt{|S|})$.*

From Lemmas 1 to 5, we conclude Theorem 2.

4.2 Mechanism for single outlet scenario

As shown in Theorem 1, the social welfare maximization problem under monotone valuations is NP-Hard even for the case of single outlet. However, we show that we can achieve an improved 2-factor approximation for this scenario. This is particularly useful when each agent prefers to be charged at a single outlet but has monotonic

[2]Note that the EV allocation problem (can also be represented as a combinatorial auction problem) is an instance of the set packing problem [13].

[3]It has the same the same optimal value as LP (6), but has only polynomially many constraints from $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\, w(i,b)\ + z \leqslant 1, \forall x^I \in X$.

Algorithm 2: Greedy $O(\sqrt{|S|})$-approximation algorithm

Input: Montone $\hat{\theta} = [\hat{\theta}_i(b), \forall i \in N, b \in B]$.
Output: $x^I \in X$ such that $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\ \hat{\theta}_i(b) \geqslant \frac{1}{2\sqrt{|S|}} \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i,b)\ \hat{\theta}_i(b)$.

Initialize allocation $x^I = [0,\ \forall i \in N, b \in B]$.
Initialize set $Y = \{(i,b), \forall i \in N, b \in B\}$.
while $Y \neq \emptyset$ **do**
  Determine $(i', b') = \arg\max_{i \in N, b \in B} \hat{\theta}_i(b) / \sqrt{\sum_{j \in S} \mathbb{I}\{b_j \neq 0\}}$
  Set $x^I(i', b') = 1$.
  For every (i, b) such that $i = i'$ or $b'_j = b_j \neq 0$ (for some $j \in S$), $Y = Y \setminus \{(i,b)\}$.
end
return x^I

valuations for contiguous time slots at that outlet. In such cases, we can solve for each outlet independently, and achieve a 2-factor approximation across multiple outlets.

We leverage [5] to give a 2-approximation w.r.t. the optimal fractional solution. In particular, their rounding and graph coloring ideas can be extended to our single outlet setup while retaining the same approximation guarantees. Since the algorithm works for all monotone inputs, this gives DSIC and IR via the RAE mechanism.

Firstly, LP(9) is solved to obtain the optimal fractional solution (x^*, OPT) for any monotone w[4]. Note that since $|M| = 1$, a bundle $b \in B$ is a vector of length $|S|$ i.e., $b \in \{0,1\}^{|S|}$, such that : (i) $\forall i, j \in S$, if $b_i, b_j \neq 0$, then $b_i = b_j = 1$ and (ii) there exists i^*, j^*, s.t. $b_i = 0, \forall i < i^*, i > j^*$ and $b_i = 1, \forall i^* \leqslant i \leqslant j^*$.

$$\begin{aligned} \max \quad & \sum_{i \in N} \sum_{b \in B} w(i,b) x(i,b) \\ \text{s.t.} \quad & \sum_{b \in B} x(i,b) \leqslant 1, \forall i \in N \\ & \sum_{b \in B_j} \sum_{i \in N} x(i,b) \leqslant 1, \forall j \in S \\ & x(i,b) \geqslant 0, \forall i \in N, b \in B \end{aligned} \tag{9}$$

Each $x^*(i,b)$ is then rounded down to the nearest fraction of the form p/Q for some $p \in \{1, 2, \dots, Q\}$, where $Q = |N|^2 (|S|(|S|+1)/2)^2$. Denote the rounded solution by x^{rou}. Observe that every $x^{\text{rou}}(i,b)$ is at most $1/Q$ smaller than $x^*(i,b)$. This implies that the value of the objective function for x^{rou} decreases by at most $\max_{i,b} w(i,b)/\sqrt{Q}$. This is because the summation is taken over all agents ($|N|$) and bundles ($|S|(|S|+1)/2$). Moreover, since $\texttt{OPT} \geqslant \max_{i,b} w(i,b)$, we have

$$\sum_{i \in N} \sum_{b \in B} w(i,b)\ x^{\text{rou}}(i,b) \geqslant (1 - 1/\sqrt{Q})\ \texttt{OPT}. \tag{10}$$

Denote $x^\ell, \forall \ell \in L$ be a set of feasible integral solutions to LP(9), where $L = \{1, 2, \dots, l\}$. Let $\texttt{val}(x^\ell) = \sum_{i \in N} \sum_{b \in B} w(i,b)\ x^\ell(i,b)$. It is easy to see that if $\sum_{\ell \in L} \texttt{val}(x^\ell) \beta_\ell \geqslant (1 - 1/\sqrt{Q})$ OPT and $\sum_{\ell \in L} \beta_\ell \leqslant 2$, then there exists an $\ell' \in L$ such that $\texttt{val}(x^{\ell'}) \geqslant (1 - 1/\sqrt{Q})\texttt{OPT}/2$. Thus, if one can find such a set of integral solutions $x^\ell, \forall \ell \in L$ with polynomial size of L, then we get a 2-factor approximation (with a negligible rounding loss) for the single outlet case. Using the rounded solution x^{rou}, we next construct a graph and color it appropriately to get the desired set of integral solutions.

[4]We omit the outlets M from the linear program since $|M| = 1$.

Construct a graph G with $x^{\text{rou}}(i,b) \cdot Q$ vertices corresponding to each $i \in N, b \in B$. Any two vertices y, z corresponding to (i^y, b^y) and (i^z, b^z) respectively have an edge between them if $i^y = i^z$ or $b^y_j = b^z_j = 1$ for some $j \in S$. This implies that two vertices have an edge if either they correspond to the same agent or if their corresponding bundles overlap (i.e., have a common slot allotted). The vertices of G are colored such that no two vertices y, z having an edge between them get the same color. Observe that the set of vertices that get the same color is an independent set in G and form a feasible integral solution for LP(9). Hence, we will call such a coloring of vertices of G as a *feasible coloring*. It can be shown that a *feasible coloring* can be achieved with at most $(2Q - 1)$ colors using a greedy strategy. See supplementary material for more details.

Lemma 6. *For graph G, there exists a feasible coloring for vertices that requires at most $(2Q - 1)$ colors.*

Let L be the set of colors and x^ℓ for $\ell \in L$ denote an integral solution where $x^\ell(i,b) = 1$ if a vertex corresponding to (i,b) has color ℓ, and $x^\ell(i,b) = 0$ otherwise. As before, denote $\texttt{val}(x^\ell) = \sum_{i \in N} \sum_{b \in B} w(i,b)\ x^\ell(i,b)$. Let $\beta_\ell = 1/Q, \forall \ell \in L$. This gives $\sum_{\ell \in L} \beta_\ell \leqslant (2Q-1)/Q \leqslant 2$ since the size of L is at most $2Q - 1$. Moreover, we have

$$\sum_{\ell \in L} \texttt{val}(x^\ell) = \sum_{i \in N} \sum_{b \in B} w(i,b)(Q\ x^{\text{rou}}(i,b)) \geqslant Q((1 - 1/\sqrt{Q})\texttt{OPT}).$$

The equality holds because graph G contains $x^{\text{rou}}(i,b) \cdot Q$ vertices for each $i \in N, b \in B$. The inequality holds due to Equation (10). This implies $\sum_{\ell \in L} \texttt{val}(x^\ell)\beta_\ell = \sum_{\ell \in L} \texttt{val}(x^\ell) 1/Q \geqslant (1 - 1/\sqrt{Q})\texttt{OPT}$. Since $\sum_{\ell \in L} \beta_\ell \leqslant 2$, there exists a color $\ell' \in L$ for which $\texttt{val}(x^{\ell'}) \geqslant (1 - 1/\sqrt{Q}) \frac{\texttt{OPT}}{2}$. We can obtain $x^{\ell'}$ by choosing the color having the maximum value. From the above discussion, we conclude the following.

Lemma 7. *For the single outlet case, Algorithm 3 that rounds the optimal fractional solution approximates the social welfare to within a factor of 2 in polynomial time.*

From Lemmas 1 to 4 and 7, we conclude the following result.

Theorem 3. *For monotone valuations and a single outlet, the* RAE *mechanism (Algorithm 1) that uses approximation Algorithm 3 as $\mathcal{A}$ in the separation oracle, approximates the social welfare within a factor of* 2 *and ensures DSIC and IR.*

5 DICHOTOMOUS VALUATIONS

Although our assumption that agents have monotone valuations is fairly general, it becomes quite demanding in the EV charging setup. A more restricted, yet practical scenario arises when agents have dichotomous valuations. Consider a scenario where each agent requires c_i units of charge and is available between the time slots a_i and d_i for charging, where $a_i, d_i \in S$ denote the arrival and departure time slots with $a_i \leqslant d_i$. Furthermore, each agent i derives a value of v^*_i if assigned c_i units of charge and 0 if they receive anything less. This implies that each agent i must be allocated some

Algorithm 3: 2-approximation algorithm

Input: Montone $w = [w(i,b), \forall i \in N, b \in B]$.

Output: A feasible integer allocation $x^I = [x^I(i,b) : \forall i \in N, b \in B]$ s.t. $\sum_{i \in N} \sum_{b \in B} x^I(i,b)\ w(i,b) \geqslant \frac{1}{2} \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i,b)\ w(i,b)$.

1. Solve LP(9) to get the optimal fractional solution x^*.
2. Set $Q = |N|^2 (|S|(|S|+1)/2)^2$.
3. Round down every $x^*(i,b)$ to the nearest fraction of the form p/Q for some $p \in \{1, 2, \ldots, Q\}$ to get x^{rou}.
4. Construct a graph G with $x^{\text{rou}}(i,b) \cdot Q$ vertices corresponding to the each $i \in N, b \in B$. Add an edge between any two vertices y, z if either $i^y = i^z$ or $b^y_j = b^z_j = 1$ for some $j \in S$.
5. For every vertex y denote $b^y_{\min} = \min_{j \in S : b^y_j = 1} j$.
6. Sort the vertices in ascending order of $b_{\min}$ and color them using at most $2Q - 1$ colors (Lemma 6) from left to right s.t. no two vertices y, z with an edge between them get the same color.
7. Let L be the set of colors and let $x^\ell, \forall \ell \in L$ be an integer solution where $x^\ell(i,b) = 1$ if a vertex corresponding to (i,b) has color ℓ, and $x^\ell(i,b) = 0$ otherwise.
8. $x^I = \arg\max_{x^\ell : \ell \in L} \sum_{i \in N} \sum_{b \in B} w(i,b)\ x^\ell(i,b)$.
9. **return** x^I

ℓ_{ik} time slots at the outlet k to obtain c_i amount of charge. This scenario induces the following dichotomous type θ_i for each agent. For every $b \in B$, let $b^{\text{arr}} = \min_{j \in S : b_j \neq 0} j$ and $b^{\text{dep}} = \max_{j \in S : b_j \neq 0} j$ denote the earliest and the latest time slot allotted as part of b respectively. In addition, denote by $b^{\text{len}} = b^{\text{dep}} - b^{\text{arr}}$ the number of contiguous time slots allotted within b and $k^b \in M$ as the outlet such that $b_j = k^b, \forall j \in S$ having $b_j \neq 0$. Then θ_i is said to be dichotomous if $\theta_i(b) = v^*_i, \forall b \in B^*_i$, where $B^*_i = \{b \in B : b^{\text{arr}} \geqslant a_i, b^{\text{dep}} \leqslant d_i, b^{\text{len}} = \ell_{ik^b}\}$, else $\theta_i(b) = 0$. It can be shown that the social welfare maximization problem for dichotomous valuations is also NP-Hard. This is because we can show a reduction from the *Job Interval Selection Problem (JISPk)* [31] where all intervals have equal length, which is known to be NP-Complete.

Theorem 4. *For dichotomous valuations, the social welfare maximization problem is NP-Hard even if the number of outlets $|M| = 1$.*

Given the above result, all the approximation mechanisms given for monotone valuations also extend to this case, since dichotomous valuations are a strict subset of monotone valuations.

6 SINGLE-MINDED VALUATIONS

The type θ_i for every agent $i \in N$ is said to be single-minded if there exists a bundle $b^i \in B$ and $q \in \mathbb{R}$ such that $\theta_i(b) = q, \forall b \supseteq b^i$ and $\theta_i(b) = 0$ otherwise. In other words, each agent prefers to charge at a single outlet for a specific set of contiguous time slots or not charge at all. For single-minded reports $\hat{\theta}$, we can drop the constraint $\sum_{b \in B} x(i,b) \leqslant 1, \forall i \in N$ from ILP (2) since each agent is interested in exactly one bundle. Hence, the LP-relaxation reduces

Algorithm 4: DAE mechanism

Input: Agents report type profile $\hat{\theta}$.

Output: A allocation $f(\hat{\theta}) \in X$ and payments $\pi_i(\hat{\theta}) \in \mathbb{R}, \forall i \in N$.

1. Solve LP (11) with parameters given by $\hat{\theta}$ to get an optimal deterministic allocation $f(\hat{\theta}) = x^*$.
2. For every agent i, set payment using VCG $\pi_i(\hat{\theta}) = \max_{x \in X} \sum_{i' \in N \setminus \{i\}} \sum_{b \in B} \hat{\theta}_{i'}(b) x(i', b) - \sum_{i' \in N \setminus \{i\}} \sum_{b \in B} \hat{\theta}_{i'}(b) x^*(i', b)$.
3. **return** $f(\hat{\theta}), \pi(\hat{\theta})$

to LP (11). It can be shown that this LP always has an optimal integer solution (since the constraints are totally unimodular [20]). This implies it can be solved to obtain an optimal deterministic allocation.

$$\begin{aligned} \max \quad & \sum_{i \in N} \sum_{b \in B} \hat{\theta}_i(b)\ x(i,b) \\ \text{s.t.} \quad & \sum_{b \in B_{jk}} \sum_{i \in N} x(i,b) \leqslant 1, \forall j \in S, k \in M, \\ & x(i,b) \geqslant 0, \forall i \in N, b \in B. \end{aligned} \tag{11}$$

A polytope ($Ax \leqslant b,\ x \geqslant 0$) is said to be integral if and only if all its corners have integer coordinates. It is well known that a linear program with an integral polytope always has an optimal integer solution [30]. A sufficient condition to identify integral polytopes is by total unimodularity. A polytope $Ax \leqslant b, x \geqslant 0$ is integral if A is totally unimodular (TU)[5] and b is integral [30]. Observe from the constraints ($Ax \leqslant b,\ x \geqslant 0$) of LP (11) that b is integral. Additionally, the constraint matrix A is totally unimodular because it is a 0-1 matrix with consecutive ones in each column. This implies the LP (11) is integral and can be solved to obtain an optimal deterministic allocation in polynomial time. Since computing the *efficient* allocation is tractable, the VCG mechanism can be used to ensure DSIC and IR which concludes the following result.

Theorem 5. *For single-minded valuations, the* Deterministic Allocatively Efficient *(DAE) mechanism (Algorithm 4) ensures DSIC and IR, and gives an efficient allocation in polynomial time.*

7 CONCLUSION

We investigated the problem of scheduling resources with monetary transfers among agents across multiple outlets with varying dispensing rates in this paper. We established NP-hardness of maximizing social welfare for both monotone and dichotomous valuation functions, for a single outlet. For monotone valuations, we presented a randomized 2-approximation mechanism for a single outlet and an $O(\sqrt{S})$-approximation mechanism for multiple outlets, ensuring DSIC and IR. For single-minded agents, the allocation problem is in $\mathbb{P}$ and hence VCG mechanism can be used.

In future, we would like to explore improving the approximation ratios and finding matching lower bounds. Extending the model to consider dynamic arrivals and departures of agents is another important future direction.

[5] A matrix A is TU if every square sub-matrix of A has a determinant of 0, 1, or -1.

REFERENCES

[1] Eric Angel, Evripidis Bampis, and Nicolas Thibault. 2012. Randomized truthful algorithms for scheduling selfish tasks on parallel machines. *Theoretical computer science* 414, 1 (2012), 1–8.

[2] Aaron Archer, Christos Papadimitriou, Kunal Talwar, and Éva Tardos. 2003. An approximate truthful mechanism for combinatorial auctions with single parameter agents. In *Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*. 205–214.

[3] Moshe Babaioff, Ron Lavi, and Elan Pavlov. 2009. Single-value combinatorial auctions and algorithmic implementation in undominated strategies. *Journal of the ACM (JACM)* 56, 1 (2009), 1–32.

[4] Amotz Bar-Noy, Reuven Bar-Yehuda, Ari Freund, Joseph Naor, and Baruch Schieber. 2001. A unified approach to approximating resource allocation and scheduling. *Journal of the ACM (JACM)* 48, 5 (2001), 1069–1090.

[5] Amotz Bar-Noy, Sudipto Guha, Joseph Naor, and Baruch Schieber. 1999. Approximating the throughput of multiple machines under real-time scheduling. In *Proceedings of the thirty-first annual ACM symposium on Theory of Computing*. 622–631.

[6] Yair Bartal, Rica Gonen, and Noam Nisan. 2003. Incentive compatible multi unit combinatorial auctions. In *Proceedings of the 9th conference on Theoretical aspects of rationality and knowledge*. 72–87.

[7] Piotr Berman and Bhaskar DasGupta. 2000. Multi-phase algorithms for throughput maximization for real-time scheduling. *Journal of Combinatorial Optimization* 4 (2000), 307–323.

[8] Patrick Briest, Piotr Krysta, and Berthold Vöcking. 2005. Approximation techniques for utilitarian mechanism design. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. 39–48.

[9] George Christodoulou, Laurent Gourves, and Fanny Pascual. 2007. Scheduling selfish tasks: about the performance of truthful algorithms. In *Computing and Combinatorics: 13th Annual International Conference, COCOON 2007, Banff, Canada, July 16-19, 2007. Proceedings 13*. Springer, 187–197.

[10] George Christodoulou and Annamária Kovács. 2013. A deterministic truthful PTAS for scheduling related machines. *SIAM J. Comput.* 42, 4 (2013), 1572–1595.

[11] Edward H Clarke. 1971. Multipart pricing of public goods. *Public choice* (1971), 17–33.

[12] Peter C Cramton, Yoav Shoham, Richard Steinberg, and Vernon L Smith. 2006. *Combinatorial auctions*. Vol. 1. MIT press Cambridge.

[13] Sven De Vries and Rakesh V Vohra. 2003. Combinatorial auctions: A survey. *INFORMS Journal on computing* 15, 3 (2003), 284–309.

[14] Peerapong Dhangwatnotai, Shahar Dobzinski, Shaddin Dughmi, and Tim Roughgarden. 2011. Truthful approximation schemes for single-parameter agents. *SIAM J. Comput.* 40, 3 (2011), 915–933.

[15] Shahar Dobzinski, Noam Nisan, and Michael Schapira. 2005. Approximation algorithms for combinatorial auctions with complement-free bidders. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. 610–618.

[16] Leah Epstein, Asaf Levin, and Rob Van Stee. 2016. A unified approach to truthful scheduling on related machines. *Mathematics of Operations Research* 41, 1 (2016), 332–351.

[17] Martin Grötschel, László Lovász, and Alexander Schrijver. 1981. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica* 1 (1981), 169–197.

[18] Theodore Groves. 1973. Incentives in teams. *Econometrica: Journal of the Econometric Society* (1973), 617–631.

[19] Birgit Heydenreich, Debasis Mishra, Rudolf Müller, and Marc Uetz. 2008. Optimal mechanisms for single machine scheduling. In *Internet and Network Economics: 4th International Workshop, WINE 2008, Shanghai, China, December 17-20, 2008. Proceedings 4*. Springer, 414–425.

[20] Alan J Hoffman and Joseph B Kruskal. 2010. Integral boundary points of convex polyhedra. *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art* (2010), 49–76.

[21] Antoon WJ Kolen, Jan Karel Lenstra, Christos H Papadimitriou, and Frits CR Spieksma. 2007. Interval scheduling: A survey. *Naval Research Logistics (NRL)* 54, 5 (2007), 530–543.

[22] Stavros G Kolliopoulos and Clifford Stein. 1998. Approximating disjoint-path problems using greedy algorithms and packing integer programs. In *International Conference on Integer Programming and Combinatorial Optimization*. Springer, 153–168.

[23] Elias Koutsoupias. 2014. Scheduling without payments. *Theory of Computing Systems* 54, 3 (2014), 375–387.

[24] Mikhail Y Kovalyov, Dominik Kress, Sebastian Meiswinkel, and Erwin Pesch. 2019. A parallel machine schedule updating game with compensations and clients averse to uncertain loss. *Computers & Operations Research* 103 (2019), 148–157.

[25] Dominik Kress, Sebastian Meiswinkel, and Erwin Pesch. 2018. Mechanism design for machine scheduling problems: Classification and literature overview. *OR Spectrum* 40 (2018), 583–611.

[26] Ron Lavi and Chaitanya Swamy. 2011. Truthful and near-optimal mechanism design via linear programming. *Journal of the ACM (JACM)* 58, 6 (2011), 1–24.

[27] Daniel Lehmann, Liadan Ita Ocallaghan, and Yoav Shoham. 2002. Truth revelation in approximately efficient combinatorial auctions. *Journal of the ACM (JACM)* 49, 5 (2002), 577–602.

[28] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. 2007. *Algorithmic Game Theory*. Cambridge University Press.

[29] Prabhakar Raghavan. 1988. Probabilistic construction of deterministic algorithms: approximating packing integer programs. *J. Comput. System Sci.* 37, 2 (1988), 130–143.

[30] Alexander Schrijver et al. 2003. *Combinatorial optimization: polyhedra and efficiency*. Vol. 24. Springer. 74–76 pages.

[31] Frits CR Spieksma. 1999. On the approximability of an interval scheduling problem. *Journal of Scheduling* 2, 5 (1999), 215–227.

[32] William Vickrey. 1961. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance* 16, 1 (1961), 8–37.

[33] Nisheeth K. Vishnoi. 2021. *Algorithms for Convex Optimization*. Cambridge University Press. 250–274 pages.

A SUPPLEMENTARY MATERIAL

A.1 Proof of Theorem 1

Job Interval Selection Problem: Consider a time horizon which is slotted into time slots $T = \{1, 2, \ldots, t\}$. An interval $\ell \in I$ can be represented using $[s_\ell, f_\ell]$ where $s_\ell \leqslant f_\ell$ and $s_\ell, f_\ell \in T$, which is a collection of contiguous time slots starting from s_ℓ until f_ℓ. Two intervals $\ell, \ell' \in I$ overlap if there exists $t \in T$ such that $t \in [s_\ell, f_\ell] \cap [s_{\ell'}, f_{\ell'}]$. The job interval selection problem (JISPk) considers a set of jobs J which can be executed over any set of k intervals (can be different set for each job) in I. For a given integer K, the goal is to decide if we can select at least K intervals from I such that: (1) no two selected intervals overlap, and (2) at most one interval is selected for each job. We denote $a(j, \ell) = 1$ if the interval $\ell \in I$ is selected for the job $j \in J$, and $a(j, \ell) = 0$ otherwise. The problem JISPk ($k \geqslant 2$) is known to be NP-complete [31]. Additionally, the maximization version of JISPk has no PTAS unless $\mathcal{P} = \mathcal{NP}$, i.e., there exists no known polynomial time algorithm which can approximate the solution to within a factor of $(1 - \epsilon)$ to the optimal, $\forall \epsilon > 0$ (see [31] for details).

Proof. We can construct an instance of the EV charging problem from JISPk as follows. The set of agents N and the set of bundles B in ILP (2) denote the set of jobs J and the set of intervals I in JISPk respectively. For every agent $i \in N$ and $\forall b, b' \in B$ such that $b \sqsubseteq b'$, set $\theta_i(b') = 1$ if the job $j \in J$ corresponding to agent i can be executed in the interval $\ell \in I$ corresponding to $b \in B$. For the remaining bundles $b' \in B$ set $\theta_i(b') = 0$.

A solution of JISPk can be constructed from the solution of EV charging problem as follows. For every $i \in N$ and $b' \in B$, if $x^*(i, b') = 1$, then for the job j corresponding to agent i select any one of its valid interval $\ell \in I$ such that bundle $b \in B$ corresponding to interval ℓ is a subset of b' and set $a^*(j, \ell) = 1$. For the remaining set $a^*(j, \ell) = 0$. It is easy to see that the selected intervals are a valid set of intervals for JISPk. Additionally, the social welfare of the optimal allocation is the same as the number of such selected intervals. This implies, we can select at least K intervals from I for JISPk if and only if the optimal allocation to the EV charging problem has a social welfare of at least K. Since the reduction requires only polynomial number of steps, the decision version of the EV charging problem is NP-Hard. The problem also belongs to NP because given a solution we can verify in polynomial time if its social welfare exceeds K. □

A.2 Proof of Lemma 2

Proof. Let $\mathsf{OPT} = \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i, b)\ w(i, b)$. For every $i \in N, b \in B$, denote $w^+(i, b) = \max(w(i, b), 0)$ and $w^{\text{mn}}(i, b) = \max_{b' \sqsubseteq b} w^+(i, b')$. Since w^{mn} is monotone, the α-approximation algorithm can be used to get a feasible integral $x^{\text{mn}} \in X$ such that

$$\sum_{i \in N} \sum_{b \in B} x^{\text{mn}}(i, b) w^{\text{mn}}(i, b) \geqslant \frac{1}{\alpha} \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i, b)\ w^{\text{mn}}(i, b) \geqslant \mathsf{OPT}/\alpha.$$

Note that the second inequality holds since $w^{\text{mn}}(i, b) \geqslant w(i, b), \forall i \in N, b \in B$. Now, construct x^+ from x^{mn} as follows. For each $i \in N, b \in B$, if $x^{\text{mn}}(i, b) = 1$, set $x^+(i, b') = 1$ for exactly one b' such that $b' = \arg\max_{b'' \sqsubseteq b} w^+(i, b'')$. For the other variables, set 0. Clearly, x^+ is a feasible integer solution and $\sum_{i \in N} \sum_{b \in B} w^{\text{mn}}(i, b) x^{\text{mn}}(i, b) = \sum_{i \in N} \sum_{b \in B} w^+(i, b) x^+(i, b)$ by construction. Finally, we construct $x^I \in X$ satisfying $\sum_{i \in N} \sum_{b \in B} w(i, b) x^I(i, b) = \sum_{i \in N} \sum_{b \in B} w^+(i, b) x^+(i, b) \geqslant \mathsf{OPT}/\alpha$ which concludes the proof. If $w^+(i, b) \geqslant 0$, then set $x^I(i, b) = x^+(i, b)$, else $x^I(i, b) = 0$. Since $x^+ \in X$ and x^I is an integral solution such that $x^I \leqslant x^+$, from the packing property of the EV charging problem $x^I \in X$. The packing property states that if any $x \in X$ is an integral solution and some integral $x' \leqslant x$, then $x' \in X$.

We note that if the α-approximation algorithm provides an integer solution with a value of at least $1/\alpha$ times the value of the fractional optimal for every *positive* $\theta \in \Theta$ then we can directly feed $w^+(i, b) = \max(w(i, b), 0)$ to the algorithm to get $x^+(i, b)$ such that $\sum_{i \in N} \sum_{b \in B} w^+(i, b) x^+(i, b) \geqslant \mathsf{OPT}/\alpha$. Then, $x^I \in X$ can be constructed as mentioned above using the packing property. □

A.3 Proof of Lemma 3

Proof. Note, the dual LP (6) is a maximization problem and $w = 0, z = 1$ is a feasible solution with a value of 1. This implies $\max\ z + \frac{1}{\alpha} \sum_{i \in N} \sum_{b \in B} x^{\text{fr}}(\hat{\theta})(i, b)\ w(i, b) \geqslant 1$. Assume that for any feasible solution w, z of LP (6)

$$z + \frac{1}{\alpha} \sum_{i \in N} \sum_{b \in B} x^{\text{fr}}(\hat{\theta})(i, b)\ w(i, b) > 1. \tag{12}$$

Using the α-approximation algorithm, we can get an $x^I \in X$ that satisfies the following (from Lemma 2).

$$\begin{aligned} \sum_{i \in N} \sum_{b \in B} x^I(i, b) w(i, b) &\geqslant \frac{1}{\alpha} \cdot \max_{x \in X^{\text{fr}}} \sum_{i \in N} \sum_{b \in B} x(i, b) w(i, b) \\ &\geqslant \frac{1}{\alpha} \sum_{i \in N} \sum_{b \in B} x^{\text{fr}}(\hat{\theta})(i, b)\ w(i, b) \end{aligned} \tag{13}$$

From (12) and (13) we get $z + \sum_{i \in N} \sum_{b \in B} x^I(i, b) w(i, b) > 1$ which violates a constraint of LP (6). This gives us a contradiction since w, z is a feasible solution of LP (6). Hence, for any feasible w, z we have $z + \frac{1}{\alpha} \sum_{i \in N} \sum_{b \in B} x^{\text{fr}}(\hat{\theta})(i, b)\ w(i, b) \leqslant 1$. This implies that the optimal value of the dual LP (6), and hence, the primal LP (5) is exactly 1.

The ellipsoid method is used with the efficient separation oracle described in Algorithm 1. This identifies an LP that is equivalent to the dual LP (6), but with only polynomially many constraints from $\sum_{i \in N} \sum_{b \in B} x^I(i, b)\ w(i, b) + z \leqslant 1, \forall x^I \in X$. These constitute the set of violated constraints returned by the separation oracle which is used to cut the ellipsoid at each iteration of the method. The primal LP (5) can then solved by considering only polynomially many variables corresponding to these violated constraints to get the optimal solution λ^*. For more details on the ellipsoid method, we refer the reader to [17, 33]. In particular, the above discussion of the ellipsoid method is a direct consequence of Theorem (3.10) Grötschel et al. [17]. Since the ellipsoid method runs in polynomially many iterations, we get the decomposition in polynomial time.

□

A.4 Proof of Lemma 4

Proof. For a given $\hat{\theta}$, the expected value derived by an agent i under $f(\hat{\theta})$ is $v_i(f(\hat{\theta}),\theta_i) = \sum_{x^I \in X} \lambda^*_{x^I}\ v_i(x^I,\theta_i)$.

$$\begin{aligned}\sum_{x^I\in X}\lambda^*_{x^I}\ v_i(x^I,\theta_i) &= \sum_{x^I\in X}\lambda^*_{x^I}\ \sum_{b\in B}\theta_i(b)x^I(i,b)\\ &= \sum_{b\in B}\theta_i(b)\sum_{x^I\in X}\lambda^*_{x^I}x^I(i,b)\\ &= \sum_{b\in B}\theta_i(b)\ x^{\text{fr}}(\hat{\theta})(i,b)/\alpha\\ &= v_i(x^{\text{fr}}(\hat{\theta})/\alpha,\theta_i) = v_i(x^{\text{fr}}(\hat{\theta}),\theta_i)/\alpha.\end{aligned}$$

The expected payment $\pi_i(\hat{\theta})$ is as follows.

$$\begin{aligned}\pi_i(\hat{\theta}) &= \sum_{x^I\in X}\lambda_{x^I}\left(\frac{p_i^{\text{fr}}(\hat{\theta})/\alpha}{v_i(f(\hat{\theta}),\hat{\theta}_i)}\ v_i(x^I,\hat{\theta}_i)\right)\\ &= \frac{p_i^{\text{fr}}(\hat{\theta})/\alpha}{v_i(f(\hat{\theta}),\hat{\theta}_i)}\sum_{x^I\in X}\lambda_{x^I}\ v_i(x^I,\hat{\theta}_i) = \frac{p_i^{\text{fr}}(\hat{\theta})}{\alpha}\end{aligned}$$

Thus, for a given $\hat{\theta}$ the expected utility for every agent under f,π is equal to its utility under fractional VCG mechanism $x^{\text{fr}}(\hat{\theta}), p^{\text{fr}}(\hat{\theta})$ scaled by α. This from Lemma 1 implies that the randomized mechanism (f,π) is DSIC . Moreover, IR is guaranteed since for every sample $x^I \in X$ the payment $\pi_i(\theta) \leqslant v_i(x^I,\theta_i)$. This is because $v_i(f(\theta),\theta_i) = v_i(x^{\text{fr}}(\theta),\theta_i)/\alpha \geqslant p_i^{\text{fr}}(\theta)/\alpha$ (since Lemma 1 guarantees IR for the fractional mechanism) which gives $\frac{p_i^{\text{fr}}(\theta)/\alpha}{v_i(f(\theta),\theta_i)} \leqslant 1$.

Finally, for the social welfare we have,

$$\begin{aligned}\sum_{x^I\in X}\lambda^*_{x^I}\sum_{i\in N}\sum_{b\in B}\hat{\theta}_i(b)\ x^I(i,b) &= \sum_{i\in N}\sum_{b\in B}\hat{\theta}_i(b)\sum_{x^I\in X}\lambda^*_{x^I}\ x^I(i,b)\\ &= \frac{1}{\alpha}\sum_{i\in N}\sum_{b\in B}\hat{\theta}_i(b)\ x^{\text{fr}}(\hat{\theta})(i,b).\end{aligned}$$

Since $\sum_{i\in N}\sum_{b\in B}\hat{\theta}_i(b)\ x^{\text{fr}}(\hat{\theta})(i,b)$ is an upper bound on the social welfare, the randomized mechanism approximates the social welfare to within a factor of α. ◻

A.5 Proof of Lemma 5

Proof. Consider x^I to be the integer solution returned by the greedy algorithm and $x^{\text{fr}} \in X$ be the optimal fractional solution of LP (3). Define for every agent $i' \in N$ with $x^I(i',b') = 1$, a term $P_{i'}$ which is the set of pairs (i,b) s.t. $x^{\text{fr}}(i,b) > 0$ and (i,b) was removed by the greedy strategy when $x^I(i',b')$ was set to 1. Note that $(i',b') \in P_{i'}$. It can be shown that $\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\hat{\theta}_i(b) \leqslant 2\sqrt{|S|}\ \hat{\theta}_{i'}(b')$ which proves the lemma. To begin with, we have $\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\hat{\theta}_i(b)$

$$\begin{aligned}&= \sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\frac{\hat{\theta}_i(b)}{\sqrt{\sum_{j\in S}\mathbb{I}\{b_j\neq 0\}}}\sqrt{\sum_{j\in S}\mathbb{I}\{b_j\neq 0\}}\\ &\leqslant \frac{\hat{\theta}_{i'}(b')}{\sqrt{\sum_{j\in S}\mathbb{I}\{b'_j\neq 0\}}}\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\sqrt{\sum_{j\in S}\mathbb{I}\{b_j\neq 0\}} \quad \text{(from greedy strategy)}\\ &\leqslant \frac{\hat{\theta}_{i'}(b')}{\sqrt{\sum_{j\in S}\mathbb{I}\{b'_j\neq 0\}}}\sqrt{\left(\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\right)\left(\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\cdot\left(\sum_{j\in S}\mathbb{I}\{b_j\neq 0\}\right)\right)}\end{aligned} \tag{14}$$

Note, b' is a set of contiguous slots at a particular outlet. Let that outlet be $k' \in M$ i.e., $b'_j = k', \forall j \in S$ such that $b'_j \neq 0$. In addition, by definition of $P_{i'}$, for each $(i,b) \in P_{i'}$ we have either $i = i'$ or $b_j = b'_j = k'$ for some $j \in S$. This implies

$$\begin{aligned}\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b) &\leqslant \sum_{(i,b)\in P_{i'}:i=i'} x^{\text{fr}}(i,b) + \sum_{\substack{j\in S:\\ b'_j=k'}}\ \sum_{\substack{(i,b)\in P_{i'}:\\ b_j=k'}} x^{\text{fr}}(i,b)\\ &\leqslant 1 + \sum_{j\in S:b'_j\neq 0} 1 \quad \text{(from constraints of LP(3))}\\ &= 1 + \sum_{j\in S}\mathbb{I}\{b'_j\neq 0\}.\end{aligned} \tag{15}$$

Moreover we have the following:

$$\begin{aligned}&\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\cdot\left(\sum_{j\in S}\mathbb{I}\{b_j\neq 0\}\right)\\ &\leqslant \sum_{j\in S}\sum_{\substack{(i,b)\in P_{i'}:\\ b_j\neq 0}} x^{\text{fr}}(i,b) \leqslant \sum_{j\in S}\sum_{k\in M}\sum_{\substack{(i,b)\in P_{i'}:\\ b_j=k}} x^{\text{fr}}(i,b)\\ &\leqslant \sum_{j\in S}\sum_{\substack{(i,b)\in P_{i'}:\\ b_j=k'}} x^{\text{fr}}(i,b) + \sum_{j\in S}\sum_{k\in M\setminus\{k'\}}\sum_{\substack{(i,b)\in P_{i'}:\\ b_j=k}} x^{\text{fr}}(i,b)\end{aligned}$$

From constraints of LP(3),

$$\begin{aligned}&\leqslant \sum_{j\in S}1 + \sum_{j\in S}\sum_{k\in M\setminus\{k'\}}\sum_{\substack{(i,b)\in P_{i'}:\\ b_j=k}} x^{\text{fr}}(i,b)\\ &= \sum_{j\in S}1 + \sum_{j\in S}\sum_{k\in M\setminus\{k'\}}\sum_{\substack{(i,b)\in P_{i'}:\\ b_j=k,i=i'}} x^{\text{fr}}(i,b)\\ &\leqslant \sum_{j\in S}1 + \sum_{j\in S}\sum_{\substack{(i,b)\in P_{i'}:\\ i=i'}} x^{\text{fr}}(i,b) \leqslant \sum_{j\in S}1 + \sum_{j\in S}1 \leqslant 2|S|\end{aligned} \tag{16}$$

Substituting Equation (15) and Equation (16) in Equation (14) we get the following which concludes the proof.

$$\begin{aligned}&\sum_{(i,b)\in P_{i'}} x^{\text{fr}}(i,b)\hat{\theta}_i(b)\\ &\leqslant \frac{\hat{\theta}_{i'}(b')}{\sqrt{\sum_{j\in S}\mathbb{I}\{b'_j\neq 0\}}}\sqrt{\left(1+\sum_{j\in S}\mathbb{I}\{b'_j\neq 0\}\right)2|S|}\\ &\leqslant \hat{\theta}_{i'}(b')\sqrt{2}\sqrt{2|S|} \leqslant \hat{\theta}_{i'}(b')2\sqrt{|S|}\end{aligned}$$

◻

A.6 Proof of Lemma 6

Proof. For every vertex y in G let $b^y_{\min} = \min_{j\in S:b^y_j=1} j$. We sort the vertices in increasing order of $b^y_{\min}$ and color them one at a time in that order so that no two vertices y,z having an edge get the same color. Using induction, we can show that there always exists a free color for a vertex when using $2Q-1$ colors.

In the beginning, since no vertices are colored, this holds trivially. For any vertex y, the color corresponding to any colored vertex $z \neq y$ where both y and z correspond to the same agent (i.e., $i^y = i^z$) cannot be used. By construction, we have at most $Q - 1$ such vertices because from the constraints of Equation (9) we get $\sum_{b \in B} x(i,b) \leqslant 1$ for every $i \in N$ which implies that $\sum_{b \in B} Q\, x^{\text{rou}}(i,b) \leqslant Q$. Thus, the number of vertices corresponding to the same agent is upper-bounded by Q. Furthermore, the color corresponding to any colored vertex $z \neq y$ where y and z correspond to bundles that overlap (i.e., $b^y_j = b^z_j = 1$ for some $j \in S$) cannot also be used. Observe that the vertices are colored in ascending order of the value $b^y_{\min}$ and all the colored vertices whose corresponding bundles overlap with the bundle b^y also have overlapping bundles among themselves. Since $\sum_{b \in B_j} \sum_{i \in N} x(i,b) \leqslant 1, \forall j \in S$ from the constraints of Equation (9), we get $\sum_{b \in B_j} \sum_{i \in N} Q\, x^{\text{rou}}(i,b) \leqslant Q$ which results in excluding at most another $Q - 1$ colors. Thus, at any given time for any vertex y at most $2Q - 2$ colors cannot be used, but one free color remains. This concludes the proof.

□