Online Resource Allocation with Convex-set Machine-Learned Advice

Negin Golrezaei
Sloan School of Management, Massachusetts Institute of Technology, golrezaei@mit.edu

Patrick Jaillet
Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, jaillet@mit.edu

Zijie Zhou
Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, 02139, zhou98@mit.edu

Decision-makers often have access to a machine-learned prediction about demand, referred to as advice, which can potentially be utilized in online decision-making processes for resource allocation. However, exploiting such advice poses challenges due to its potential inaccuracy. To address this issue, we propose a framework that enhances online resource allocation decisions with potentially unreliable machine-learned (ML) advice. We assume here that this advice is represented by a general convex uncertainty set for the demand vector.

We introduce a parameterized class of Pareto optimal online resource allocation algorithms that strike a balance between consistent and robust ratios. The consistent ratio measures the algorithm's performance (compared to the optimal hindsight solution) when the ML advice is accurate, while the robust ratio captures performance under an adversarial demand process when the advice is inaccurate. Specifically, in a C-Pareto optimal setting, we maximize the robust ratio while ensuring that the consistent ratio is at least C. Our proposed C-Pareto optimal algorithm is an adaptive protection level algorithm, which extends the classical fixed protection level algorithm introduced in Littlewood (2005) and Ball and Queyranne (2009). Solving a complex non-convex continuous optimization problem characterizes the adaptive protection level algorithm. To complement our algorithms, we present a simple method for computing the maximum achievable consistent ratio, which serves as an estimate for the maximum value of the ML advice. Additionally, we present numerical studies to evaluate the performance of our algorithm in comparison to benchmark algorithms. The results demonstrate that by adjusting the parameter C, our algorithms effectively strike a balance between worst-case and average performance, outperforming the benchmark algorithms.

Key words: Online Resource Allocation, Machine-learned Advice, Convex Uncertainty Set, Pareto Optimal Algorithms, Robust Ratio, Consistent Ratio

1. Introduction

The problem of allocating a limited inventory of a single resource to sequentially arriving requests can be examined within the framework of revenue management, a significant discipline in operations research. Originally developed within the airline industry, revenue management has gained widespread recognition and applicability across various sectors, including retail, hospitality, online advertising, and more.

In the context of resource allocation, revenue management aims to optimize a firm's revenue by implementing effective policies to control quantities. This concept finds application in various applications, such as flight seat allocation, online event ticketing, car rental inventory management, and hotel room reservation. To illustrate this concept, let's consider an example involving an airline. Airlines often offer different fare classes to cater to various customer types, including price-sensitive leisure travelers and business travelers. Each fare class comes with distinct prices and additional perks, such as seat selection and flexibility in cancellation. In this scenario, the airline must strategically determine the optimal number of seats to allocate to customers from different fare classes in order to maximize their overall revenue.

However, firms face the challenge of making real-time decisions to allocate their limited resources to incoming demand while lacking precise knowledge of future demand. This challenge arises due to the inherent trade-off between allocating resources, such as seats, to low-reward demand, such as leisure travelers, and reserving resources for potential high-reward demand, such as business travelers.

To address this trade-off, researchers have extensively studied two main models. The first model is the adversarial arrival model (Ball and Queyranne (2009)), which assumes no forecast about demand is available. However, in practice, the demand process is typically not the worst-case scenario and may exhibit some level of predictability. Consequently, the resulting algorithms under this regime tend to be overly conservative as we also show in Section 7. The second model is the stochastic model, which assumes perfect knowledge of the demand process, with the assumption that low-reward demand arrives before high-reward demand (Littlewood 2005). However, demand prediction is often challenging, especially in new and non-stationary settings that arise due to factors like seasonality or natural crises.

In this work, we aim to bridge the gap between the two models by augmenting the adversarial model with a demand forecast in the form of a convex uncertainty set. This uncertainty set that we refer to as *machine-learned advice* is obtained through machine learning or data-driven robust optimization algorithms. Despite the success and ubiquity of these techniques across many domains, leveraging them in online decision-making, such as the aforementioned resource allocation problem, presents significant challenges. The key challenge lies in effectively managing errors and biases in the forecast that inevitably exist, ensuring robust and reliable decision-making in the face of uncertain demand.

To account for this challenge, we go beyond having a point estimate for the demand vector (i.e., the number of customers of different types) as done in Balseiro et al. (2022). (For a detailed comparison between our work and Balseiro et al. (2022), refer to Sections 1.2 and 7.) Instead, we adopt the approach of utilizing an uncertainty set for the demand vector. This modeling choice that also covers a point estimate as its special case offers several advantages:

- First, it allows us to harness biases that exist in a single point estimate, resulting in a robust algorithm that does not overfit to a single point estimate. Please refer to our numerical studies in Section 7 for comparison between resource allocation algorithms with a single point estimate and those we propose with an uncertainty set.
- Second, the framework of uncertainty sets allows for the consideration of inaccuracies within the set itself. In other words, the realized demand may fall outside the bounds defined by the uncertainty set. This level of flexibility greatly enhances the resilience of the allocation process, empowering it to effectively handle unforeseen variations in demand that may deviate from initial expectations.
- Third, the introduction of the uncertainty set allows us to capture the variance in demand process and potential positive or negative correlations between different types of demand. For example, if the uncertainty set establishes bounds on the total number of demands, we can account for the expected correlation between high-reward and low-reward customers. In this case, a large influx of low-reward customers would suggest a corresponding decrease in the number of high-reward customers.

1.1. Our Contributions and Results

A New Resource Allocation Model with a Convex Uncertainty Set. We introduce a novel online resource allocation model with machine-learned (ML) advice (Section 2) that addresses the challenge of allocating m identical units of a resource to arriving requests, categorized as either low-reward or high-reward.

At the onset of the allocation period, the decision-maker receives ML advice in the form of a convex uncertainty set $\mathcal{R}$. This uncertainty set characterizes the total number of high-reward or low-reward requests expected to arrive during the allocation period. For instance, the advice may indicate that the total number of requests falls within a specific interval while ensuring that the total number of high-reward requests remains below a certain threshold. Importantly, the ML advice does not provide any information regarding the order of arrivals.

Our choice of utilizing a convex uncertainty set is motivated by related works in the robust optimization literature, such as Bertsimas and Brown (2009), Bertsimas et al. (2018). These studies leverage offline historical data to construct convex uncertainty sets. Additionally, publications like Cheramin et al. (2021), Bertsimas et al. (2016), Jalilvand-Nejad et al. (2016) have explored the application of convex uncertainty sets in offline robust optimization problems.

In this work, we do not make any assumptions about the accuracy of the ML advice. Instead, we present a class of algorithms that demonstrate robust performance regardless of the accuracy of the advice.

Pareto optimal Algorithms (Section 4). As previously mentioned, our objective is to incorporate ML advice in a robust manner, accounting for potential inaccuracies. To achieve this, we introduce two performance measures: consistent ratio and robust ratio, which are analogous to the traditional competitive ratio used in the analysis of online algorithms.

The consistent ratio of an algorithm represents the worst-case ratio of its expected reward to the optimal hindsight solution under any arrival sequence consistent with the ML advice. On the other hand, the robust ratio is the worst-case ratio of its expected reward to the optimal hindsight solution under any arrival sequence that is not consistent with the advice. The formal definitions of these ratios can be found in Section 2. As one of our main contributions, we present a parameterized class of Pareto optimal algorithms that strike a balance between robust and consistent ratios.

Let $C^\star(\mathcal{R})$ denote the maximum consistent ratio achievable by any algorithm under the ML advice $\mathcal{R}$ (see the formal definition in Section 6). For any $C \leqslant C^\star(\mathcal{R})$, a C-Pareto optimal algorithm maximizes the robust ratio among all deterministic or randomized online algorithms, while ensuring that its consistent ratio is at least C. By adjusting the parameter C, we can emphasize achieving a higher consistent ratio when we have greater confidence in the accuracy of the advice. Conversely, decreasing C reflects concern about potential inaccuracies, leading to a focus on obtaining a higher robust ratio.

Our main result fully characterizes C-Pareto optimal algorithms, demonstrating that they belong to a class of (adaptive) protection level algorithms (PLAs). An adaptive PLA extends the classical fixed protection level algorithms studied in seminal works of Littlewood (2005), Ball and Queyranne (2009). In fixed protection level algorithms, a certain amount of resources is reserved or protected for potential high-reward requests that may arrive later, with the protection level remaining fixed throughout the allocation period. In the adaptive version introduced formally in Section 3, the protection level is represented by a function that maps the total received low-reward demand so far to a protection level. In adaptive PLAs, the protection level can vary or decrease over time, as long as the reduction is not too steep (see the formal definition in Definition 1). Our designed C-Pareto optimal algorithms fall within this class, simplifying their implementation.

THEOREM 1 **(Informal Result: C-Pareto optimal Algorithms)**. *For any convex ML advice $\mathcal{R}$ and $C \leqslant C^{\star}(\mathcal{R})$, there exists a C-Pareto optimal algorithm belonging to the class of (adaptive) PLAs (as defined in Definition 1). Furthermore, the protection level function characterizing the C-Pareto optimal algorithm can be computed in polynomial time.*

To fully characterize the C-Pareto optimal algorithm, we present an optimization problem (referred to as Problem C-Pareto) that optimizes over the protection level function, assuming the algorithm is a PLA. Importantly, our results demonstrate that this assumption is not restrictive, as the designed algorithm is optimal among all deterministic and randomized algorithms, not just the PLAs. However, solving the optimization problem is challenging due to its non-convex and continuous nature.

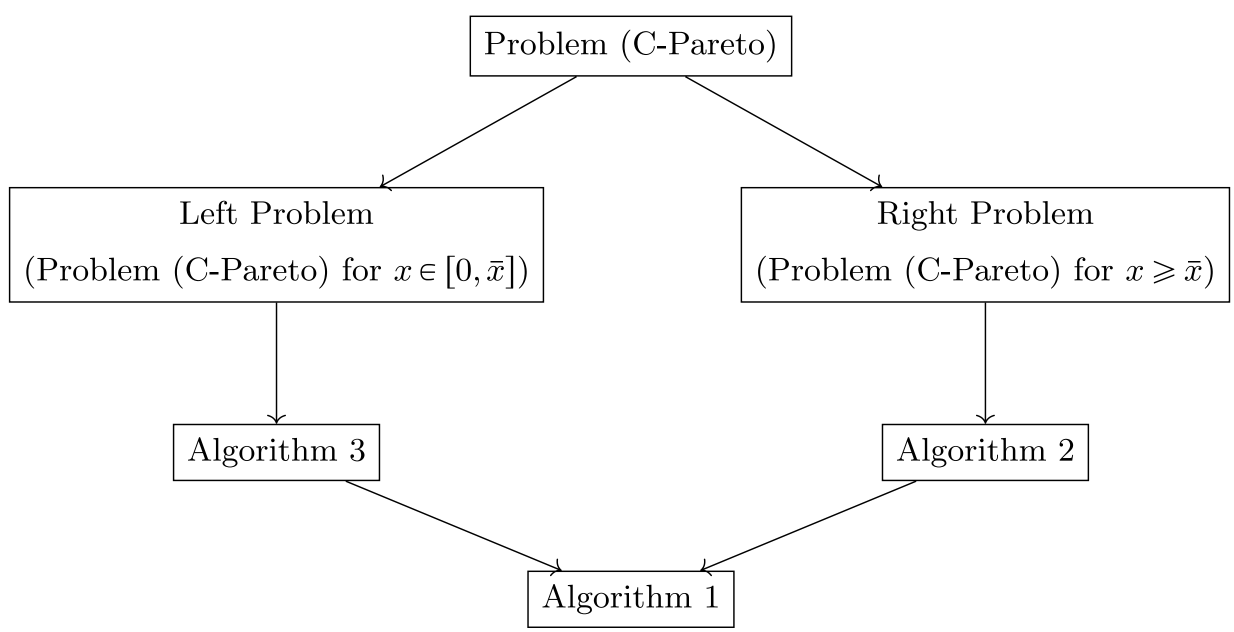


Figure 1 **Decomposition of Problem** (C-Pareto) **into the right and left problems. Here, x is the total low-reward demand and $\bar{x}$ is the maximum total low-reward demand under the ML advice.**

Technical Contributions Regarding C-Pareto optimal Algorithms. One of our main contributions is the development of a polynomial time scheme to solve Problem (C-Pareto) for any convex ML advice $\mathcal{R}$. We achieve this by transforming the consistency constraints into bounds on the protection level (PL) function and applying similar transformations for the robustness constraints. This allows us to decompose the problem into "right" and "left" subproblems. The optimal solution to the right problem coincides with the original problem's solution when the total low-reward demand exceeds a certain threshold, while the left problem considers cases where the demand is below the threshold. The optimal solutions to both subproblems can be fully characterized, resulting in an efficient algorithm for Problem (C-Pareto) with guaranteed performance guarantees. See Figure 1 for an outline of our approach.

Characterizing the Maximum Consistent Ratio $C^\star(\mathcal{R})$ (Section 6). In Section 6, we introduce polynomial-time methods to compute the maximum consistent ratio $C^\star(\mathcal{R})$, which represents the highest value achievable with the ML advice. For general convex ML regions, we propose a bisection method (Algorithm 4) that provides an ϵ-accurate estimate of $C^\star(\mathcal{R})$ in polynomial time, where ϵ can be chosen within the range $(0, 1/2]$. When the ML region is a polyhedron, we present a faster approach (Algorithm 5) based on enumerating the polyhedron vertices. This method computes $C^\star(\mathcal{R})$ exactly by identifying the worst vertices (bottlenecks) of Problem (C-MAX) that determine $C^\star(\mathcal{R})$.

THEOREM 2 **(Informal Result: Maximum Consistent Ratio)**. *Consider any convex ML advice $\mathcal{R}$. For any $\epsilon \in (0, 1/2]$, there exists a bisection method that returns an ϵ-accurate estimate of $C^\star(\mathcal{R})$ in polynomial time. When the ML advice is a polyhedron, there exists a polynomial time enumeration method that computes $C^\star(\mathcal{R})$ exactly.*

Technical Contributions Regarding Maximum Consistent Ratio $C^\star(\mathcal{R})$. For any polyhedron $\mathcal{R}$, we propose a novel enumeration approach to identify the maximum consistent ratio $C^\star(\mathcal{R})$. We assert that the worst over- and under-protected points for any protection level function correspond to the vertices of $\mathcal{R}$. Subsequently, we enumerate each pair of vertices to determine if: (i) they represent the worst over- and under-protected points, and (ii) a feasible protection level function exists if they indeed represent these extreme points. Furthermore, based on the relative positions of each vertex pair, we construct a distinct protection level function. The reasoning behind formulating diverse protection level functions is rooted in our comprehensive understanding of where the protection level function should remain constant and where it should exhibit a decreasing trend.

Numerical Studies (Section 7). We perform numerical studies to assess the performance of our proposed algorithms when the ML advice is derived from a limited number of samples. Our findings demonstrate that, in the presence of ML advice, our algorithms enhance both the average and worst-case performance, surpassing other benchmark algorithms. One such benchmark pertains to resource allocation algorithms based on a single point estimate, which exhibit average and worst-case compatible ratios that are up to 14% and 40% lower than those achieved by our resource allocation algorithms considering convex uncertainty sets.

1.2. Other Related Works

Online Decision-making with ML Advice. Our class of algorithms contributes to a recent literature on using ML advice in the online algorithm design. Examples include Lykouris and Vassilvtiskii (2018) and Rohatgi (2020) for online caching problems, Antoniadis et al. (2020) for online secretary problems, Jin and Ma (2022) for online matching problems, Lattanzi et al. (2020) for online scheduling with job weight advice, and Balseiro et al. (2022) for an online resource allocation problem.[1]

In the context of single-leg revenue management, the work most related to ours is Balseiro et al. (2022), which also explores the impact of ML advice. Specifically, Balseiro et al. (2022) focuses on a single point demand prediction and ML advice and presents a class of Pareto optimal algorithms. However, the LP-based algorithm introduced in Balseiro et al. (2022) suffers from non-monotonicity of the protection levels. This means that it may reject a customer of a certain fare type while accepting another customer of the same fare type at a later time step. Such non-monotonic behavior can lead to strategic actions by customers or third parties aiming to exploit inter-temporal fare arbitrage opportunities. To address this issue, Balseiro et al. (2022) suggests a fixed protection level algorithm, but it is not always optimal.

In our work, there are important distinctions. Firstly, we focus on ML advice in the form of an uncertainty set, rather than a single point estimate. This choice is motivated by the desire to mitigate potential biases inherent in point estimates, such as outliers, asymmetric errors, or incomplete data. Additionally, uncertainty sets provide valuable information about the demand, including variance and correlation between different demand types. Moreover, in some cases, it is impossible to obtain a point estimate for the demand vector (e.g., from a dataset that only contains the total number of arrivals, rather than the number of arrivals of each type). We discuss

[1] See (Mahdian et al. 2012, Esfandiari et al. 2015, Hwang et al. 2021, Golrezaei et al. 2014, 2022) for other works that explore the partially known demand models.

this aspect further in Section 7, supported by numerical studies. Another key difference is the adaptive protection level algorithm we propose, which effectively eliminates the non-monotonicity issues observed in Balseiro et al. (2022). Consequently, our approach offers a more practical and improved solution. Lastly, as stated earlier, our framework can handle the case of a point estimate as ML advice, providing flexibility to accommodate different types of advice.

Online Resource Allocation. The allocation of scarce resources in an online setting has been a subject of research for many years. In the past, researchers have examined this issue under the assumption of stochastic arrival sequences in various works, such as Devanur and Hayes (2009), Feldman et al. (2010), and Agrawal et al. (2014). A popular method for solving this problem is the primal-dual technique, which is known for designing algorithms with sub-linear regret. Additionally, the primal-dual technique can be used not only for stochastic demand arrivals, but also for adversarial demand arrivals, as demonstrated in Mehta et al. (2007), Buchbinder et al. (2007), and Golrezaei et al. (2014). In contrast, our work does not use the primal-dual technique, instead we utilize the protection-level framework, which is a well-established approach in the field of single-leg revenue management.

Single-leg Revenue Management. In this work, we study the single-leg revenue management problem in the presence of ML advice. Single-leg revenue management is a well-established model in the field of revenue management. Littlewood (2005) proposed an optimal policy for the single-leg revenue management problem involving two types of customers under stochastic arrival processes. Brumelle and McGill (1993) extended this problem to include multiple types of customers and designed an optimal policy using dynamic programming. Ball and Queyranne (2009) was the first to address the single-leg revenue management problem under adversarial arrival sequences. They proposed an optimal protection-level policy for the two types of customers case and then introduced the concept of "nesting" to generalize to the multiple types case. See also Jasin (2015), Hwang et al. (2021), Ma et al. (2021), Golrezaei and Yao (2021) for more recent works on single-leg revenue management. Our work contributes to this literature by presenting a new model for single-leg revenue management that bridges the gap between the adversarial and stochastic models.

2. Model

Consider a scenario where a firm has been endowed with m identical units of a divisible resource to allocate over T rounds, but the number of rounds T is unknown to the firm. In each round t, a request with size $s_t > 0$ and type $z_t \in \{\ell, h\}$ arrives, where the size of the request s demands at most s units of the resource. Requests can be categorized into two types based on the normalized reward or revenue they generate upon receiving one unit of the resource, namely low-reward and high-reward requests. (See our discussion in Section 8 about handling more than two types of requests.)

Let the reward for type $z \in \{\ell, h\}$ upon receiving one unit of the resource be denoted as r_z. Without loss of generality, it is assumed that $0 < r_\ell < r_h$.[2] Upon the arrival of a request (s_t, z_t), the firm observes the size and type of the request and must make an irrevocable decision to allocate $a_t \in [0, s_t]$ units of the resource to request (s_t, z_t), and collect a total reward of $a_t \cdot r_{z_t}$. At the time of the decision, the firm has no knowledge of the type and size of future requests.

The goal is to design online allocation algorithms that maximize the cumulative reward of the firm over the course of T rounds. The performance of an algorithm is evaluated by comparing it to the optimal clairvoyant solution, which has complete knowledge of the arrival sequence of the requests $(s_t, z_t)_{t\in[T]}$ in advance. Further details on the evaluation process will be provided later.

2.1. ML Advice

Let $I = (s_t, z_t)_{t\in[T]}$ be the arrival sequence of requests, where the type and size of requests, the order of the requests, and the number of requests T are chosen by an adversary. For an input sequence $I = (s_t, z_t)_{t\in[T]}$, we define the total low-reward and high-reward demand in the input sequence I as $\ell(I) = \sum_{t\in[T], z_t=\ell} s_t$ and $h(I) = \sum_{t\in[T], z_t=h} s_t$, respectively. We assume that at the beginning of round 1, the firm has access to partial knowledge about the arrival sequence I and, more specifically, about $\ell(I)$ and $h(I)$. This partial knowledge, which we refer to as *ML advice*, is represented by a convex region $\mathcal{R} \in \mathbb{R}^2$. The ML advice enforces the demand vector, i.e., $(\ell(I), h(I))$, to fall into region $\mathcal{R}$. Throughout the manuscript, we refer to region $\mathcal{R}$ as the ML region. Then, the set of arrival sequences that is consistent with the ML advice (denoted by $\mathcal{S}(\mathcal{R})$) is given by:

$$\mathcal{S}(\mathcal{R}) = \{I : (\ell(I), h(I)) \in \mathcal{R}\}. \tag{1}$$

We refer to the set $\mathcal{S}(\mathcal{R})$ as the *ML-consistent set*. For example, when $\mathcal{R} = \{(x, y) : x \in [a_1, b_1], y \in [a_2, b_2]\}$ for some $a_1, a_2, b_1, b_2 \geqslant 0$, the ML advice provides lower and upper bounds on the low-reward and high-reward demands. As another example, when $\mathcal{R} = \{(x, y); x + y \in [a, b]\}$ for some $a, b \geqslant 0$, the ML advice gives lower and upper bounds on the total demand. The ML advice does not provide any information on the order of requests. See Section 2.4 for further notation related to ML advice.

[2] Without loss of generality, one can normalize r_h to one.

The ML advice we examine is general and encompasses any convex uncertainty region. As a specific instance, the ML advice can be represented by a single point, as studied in Balseiro et al. (2022). However, our numerical studies reveal that relying solely on a single point estimate as ML advice can result in suboptimal performance due to inherent biases in a single point estimate.

2.2. Performance Measures

In this work, we consider two metrics to measure the performance of any online resource allocation algorithm that has access to the ML advice $\mathcal{R}$. These two metrics, which are called the *consistent ratio* and the *robust ratio*, are inspired by the fact that the ML advice may not be completely accurate.

The consistent ratio of algorithm $\mathcal{A}$, denoted by $\textsc{consis}(\mathcal{A})$, measures how well algorithm $\mathcal{A}$ performs when the ML advice is completely accurate. That is, it measures the performance of algorithm $\mathcal{A}$ on all arrival sequences in the set $\mathcal{S}(\mathcal{R})$ that are consistent with the ML advice $\mathcal{R}$. Similarly, the robust ratio of algorithm $\mathcal{A}$, denoted by $\textsc{robust}(\mathcal{A})$, measures the performance of algorithm $\mathcal{A}$ on all arrival sequences. The robust ratio can evaluate the robustness of the algorithm when the ML advice is misleading.

Mathematically speaking, for any arrival sequence I and an online algorithm $\mathcal{A}$, let $\textsc{R}ew(\mathcal{A}, I)$ be the expected cumulative reward of the algorithm under the arrival sequence I. Further, let $\textsc{opt}(I)$ be the optimal clairvoyant solution under arrival sequence I. Note that the optimal clairvoyant solution, which has knowledge of the arrival sequence in advance, starts by allocating resources to the type h that has the highest reward. If there is any resource remaining after that, it allocates resources to type ℓ requests. Then, we define:

$$\textsc{consis}(\mathcal{A}) = \inf_{I: I\in\mathcal{S}(\mathcal{R})} \frac{\textsc{R}ew(\mathcal{A}, I)}{\textsc{opt}(I)} \quad \text{and} \quad \textsc{robust}(\mathcal{A}) = \inf_{I: I\in\mathcal{S}(\mathcal{R})\cup\mathcal{S}(\mathcal{R})^{\mathsf{C}}} \frac{\textsc{R}ew(\mathcal{A}, I)}{\textsc{opt}(I)}. \tag{2}$$

It is worth noting that when the set $\mathcal{S}(\mathcal{R})$ contains all possible arrival sequences, the consistent ratio is equivalent to the traditional worst-case competitive ratio notion for online algorithm design in the absence of ML advice. In this case, the algorithm's performance is evaluated based on its worst-case performance over all possible arrival sequences. Similarly, when the ML-consistent set $\mathcal{S}(\mathcal{R})$ is empty, the robust ratio is also equivalent to the traditional competitive ratio notion, where the algorithm's performance is measured against the optimal clairvoyant solution that has full knowledge of the arrival sequence.

In the absence of ML advice, it has been shown in Ball and Queyranne (2009) that the optimal competitive ratio for online resource allocation algorithms is $1/(2 - r_\ell/r_h)$, where r_ℓ and r_h are the rewards associated with the low- and high-reward types, respectively.

2.3. Objectives

When the ML advice is completely accurate, maximizing the consistent ratio would lead to an optimal algorithm with the highest worst-case competitive ratio on set $\mathcal{S}(\mathcal{R})$. See the details in Section 6. However, such an algorithm may perform poorly when the ML advice is inaccurate (i.e., when the ML-consistent set $\mathcal{S}(\mathcal{R})$ does not occur). To balance this trade-off, our main goal in this work is to present a class of parameterized *Pareto optimal* algorithms.

In a C-Pareto optimal algorithm, we design an algorithm that achieves the highest possible robust ratio while obtaining a consistent ratio of at least C for any $C \leqslant C^\star(\mathcal{R})$. Here, $C^\star(\mathcal{R})$ is the highest possible consistent ratio for a convex ML region $\mathcal{R}$, in the absence of any constraint on the robust ratio. We formally characterize $C^\star(\mathcal{R})$ in Section 6 and suggest an $O(|\mathcal{V}|^3)$ complexity algorithm to find the value of $C^\star(\mathcal{R})$ when $\mathcal{R}$ is a polyhedron, where $\mathcal{V}$ is the set containing all the vertices of $\mathcal{R}$.

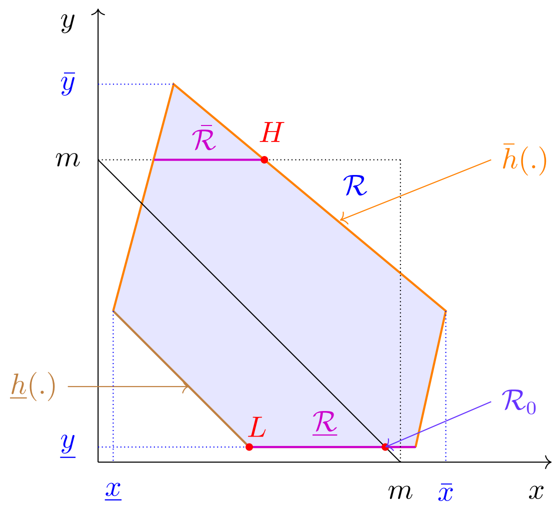
Mathematically, let Π be the set of all online deterministic and randomized algorithms. Then, the C-Pareto optimal algorithm $\mathcal{A}$ solves the following optimization problem:

$$\max_{\mathcal{A}\in\Pi} \ \textsc{robust}(\mathcal{A}) \quad s.t. \quad \textsc{consis}(\mathcal{A}) \geqslant C\,. \tag{3}$$

Observe that by setting C to $C^\star(\mathcal{R})$, we can design an optimal ML-consistent algorithm under which the ML advice is fully trusted. When $\mathcal{R} = \{(x, y) : x, y \geqslant 0\}$—i.e., the ML arrival set $\mathcal{S}(\mathcal{R})$ contains all possible arrival sequences— the protection-level algorithm of Ball and Queyranne (2009) is ML-consistent optimal. In this algorithm, type h requests are always accepted while at most $\frac{m}{2-r_h/r_\ell}$ type ℓ requests are accepted. In other words, we *protect* $m - \frac{m}{2-r_\ell/r_h}$ of the resources for high-reward requests, and by doing so, we obtain $C^\star(\mathcal{R})$ of $\rho := \frac{1}{2-r_\ell/r_h}$. Overall, the design of Pareto optimal algorithms lead to a Pareto curve (e.g., Figure 3) that helps us balance the trade-off between the consistent and robust ratios.

2.4. Notation

In this section, we present a few definitions regarding the ML region $\mathcal{R}$ (refer to the figure below for illustration).



Let

$$\underline{x} = \inf_{(x,y)\in\mathcal{R}} x\,, \qquad \bar{x} = \sup_{(x,y)\in\mathcal{R}} x\,, \qquad \underline{y} = \inf_{(x,y)\in\mathcal{R}} y\,, \qquad \text{and} \qquad \bar{y} = \sup_{(x,y)\in\mathcal{R}} y\,. \tag{4}$$

Define

$$\bar{\mathcal{R}} = \{(x,y)\in\mathcal{R} : y = \sup_{(x',y')\in\mathcal{R}} \min\{y',m\}\}$$

as a subset of region $\mathcal{R}$ under which the total high-reward demand (more precisely $\min\{y',m\}$ for any point $(x',y')\in\mathcal{R}$) is maximized. Similarly, we define

$$\underline{\mathcal{R}} = \{(x,y)\in\mathcal{R} : y = \inf_{(x',y')\in\mathcal{R}} \min\{y',m\}\}$$

to be a subset of region $\mathcal{R}$ under which the total high-reward demand (more precisely $\min\{y',m\}$ for any point $(x',y')\in\mathcal{R}$) is minimized. We then define point $L=(x_L,y_L)\in\underline{\mathcal{R}}$ as the point in set $\underline{\mathcal{R}}$ that has the lowest total low-reward demand. We further define $H=(x_H,y_H)\in\bar{\mathcal{R}}$ as the point in set $\bar{\mathcal{R}}$ that has the highest total low-reward demand. Mathematically speaking, we let

$$L = \inf_x\{(x,y):(x,y)\in\underline{\mathcal{R}}\} \quad \text{and} \quad H = \sup_x\{(x,y):(x,y)\in\overline{\mathcal{R}}\}.$$

Observe that point L has the lowest reward among all the points in set $\underline{\mathcal{R}}$ and point H has the highest reward among all the points in set $\bar{\mathcal{R}}$. Further note that while by definition we have $y_L \leqslant y_H$, we can have x_L less than or greater than x_H.

Then, we define the upper and lower envelop of $\mathcal{R}$ as $\bar{h}(\cdot)$ and $\underline{h}(\cdot)$, where

$$\bar{h}(x) = \sup\{y:(x,y)\in\mathcal{R}\} \quad \text{and} \quad \underline{h}(x) = \inf\{y:(x,y)\in\mathcal{R}\} \tag{5}$$

for $x\in[\underline{x},\bar{x}]$. As $\mathcal{R}$ is a convex set, we have $\underline{h}(x)$ is convex and $\bar{h}(x)$ is concave.

Finally, let

$$\mathcal{R}_0 = \{(x, \underline{h}(x)) : x \in [\underline{x}, \bar{x}]\} \cap \{(x, y) : x + y = m\}$$

be the intersection of the lower envelop $\underline{h}(x)$ of the region $\mathcal{R}$ and line $x + y = m$. Note than for any point above (below respectively) this line, the total demand is greater (less respectively) than the number of resources m. As $\underline{h}(\cdot)$ is a convex function, we have $|\mathcal{R}_0| \in \{0, 1, 2\}$ because a convex function and a line can have at most two intersection points.

3. Adaptive Protection Level Algorithms

The algorithms we have designed have a simple structure, making them easy to implement. We refer to these algorithms as (adaptive) *protection level algorithms*, which can be viewed as an extension of the protection level algorithms (PLA) introduced in Littlewood (2005), Ball and Queyranne (2009). The definition of the (adaptive) protection level algorithm is provided below.

Definition 1 (Adaptive Protection Level Algorithms). *A PLA is defined by a continuous non-increasing Protection Level (PL) function $p : \mathbb{R}_+ \to [0, m]$, with $p'(x) \geqslant -1$ and $p(x) = p(\max\{m, \bar{x}\})$ for any $x \geqslant \max\{m, \bar{x}\}$. Under a PLA with a PL of $p(\cdot)$,*

- *high-reward requests are fully fulfilled unless we do not have enough resources left. That is, for any request $(s, z = h)$, we set the allocation a to the minimum of s and the remaining resources at the time of the decision.*
- *For low-reward requests, let $\bar{s}$ be the sum of the sizes of the low-reward requests received so far (excluding the current one) and define $\bar{a}$ as the total amount of resources we have allocated to low-reward requests so far. Further, let s be the size of the current low-reward request. Under a PLA with a PL of $p(\cdot)$, we allocate $a \in [0, s]$ amount of the resource to the current low-reward request, where*

  $$a = \min\left(\widehat{m},\, Proj_{[0,s]}\big(m - p(\bar{s} + s) - \bar{a}\big)\right).$$

  Here, $Proj_{[0,y]}(x) = \min(\max(x, 0), y)$ and $\widehat{m}$ is the remaining number of resources at the time of the decision. Then, it is guaranteed that $\bar{a} + a \leqslant m - p(\bar{s} + s)$, meaning we protect $p(\bar{s} + s)$ resources for the high-reward requests.

To gain a better understanding of the definition of PLAs, let us consider a scenario where the size of all requests is 1, and $m - p(x)$ is an integer. In this case, a PLA will always accept high-reward agents as long as we have the necessary resources. Moreover, the PLA will reject the x-th low-reward agent if either the number of low-reward agents already accepted is equal to $m-p(x)$, or there are no resources left.

It is worth noting that in a PLA, the number of units that we protect for high-reward requests is solely dependent on the total low-reward demand we have observed so far, which is represented by $\bar{s}+s$ in Definition 1. This property makes PLA a practical and easy-to-implement approach. Another important observation is that the PLA introduced in Ball and Queyranne (2009) can be represented by a fixed protection level function: $p(x) = m - \frac{m}{2-r_\ell/r_h}$ for any $x \geqslant 0$. For a discussion on the necessity of the validity conditions for the PL functions listed in Definition 1, please refer to Appendix A.

3.1. A Property of PLAs

The following lemma shows that under PLAs, the consistent and robust ratios get minimized under ordered arrival sequences under which all low-reward requests arrive before high-reward requests. See Appendix B for the proof.

Lemma 1 **(Ordered Sequences)**. *Suppose that we use a valid PLA $\mathcal{A}$ with a PL function $p(\cdot)$ (per Definition 1). Let $\mathcal{I}(x,y)$ be the set of all arrival instances that contain x low-reward requests and y high-reward requests in any order, and we let $\tilde{I}(x,y)$ be the ordered sequence such that all x low-reward requests arrive first and are followed with all y high-reward requests. For any $I \in \mathcal{I}(x,y)$, we then have* $\frac{\text{Rew}(\mathcal{A},I)}{\text{OPT}(I)} \geqslant \frac{\text{Rew}(\mathcal{A},\tilde{I}(x,y))}{\text{OPT}(\tilde{I}(x,y))}$.

In light of Lemma 1, we finish this section with a few definitions. For any point $A=(x,y)$, let $\text{CP}(p; A=(x,y))$ be the *compatible ratio* of point $A=(x,y)$ under a PL of p when a total of x low-reward requests arrive first, followed by y high-reward requests. That is, $\text{CP}(p;(x,y))$ is the ratio of the obtained reward under *ordered arrivals* associated with point $A=(x,y)$ and a fixed PL of p to the optimal clairvoyant solution. When $p \geqslant \min\{m,y\}$, we over-protect high-reward requests and hence

$$\text{CP}_o(p; A=(x,y)) = \frac{\min\{y,m\}r_h + \min\{x,(m-p)^+\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell}, \qquad \text{if } p \geqslant \min\{m,y\} \ . \tag{6}$$

Here, the subscript "o" in CP_o stands for over-protection. Note that in the definition of CP_o, $\min\{y,m\}r_h + \min\{x, m-p\}r_\ell$ is the reward at the ordered point $A=(x,y)$ with protection level $p \geqslant \min\{m,y\}$. Observe that we accept $\min\{y,m\} \leqslant p$ high-reward request and $\min\{x,m-p\}$ low-reward requests.

On the other hand, when $p < \min\{m, y\}$, we under-protect high-reward requests, and hence

$$\mathrm{CP}_u(p; A = (x,y)) = \frac{\max\{p, \min\{y, (m-x)^+\}\}r_h + \min\{x, m-p\}r_\ell}{\min\{y, m\}r_h + \min\{x, (m-y)^+\}r_\ell}, \qquad \text{if } p < \min\{m, y\} \ . \quad (7)$$

Here, the subscript "u" in CP_u stands for under-protection. We allocate $\min\{x, m-p\}$ to low-reward requests. Then, if the total request is less than m, that is, $y < (m-x)^+$, we will accept all y high-reward type requests. Otherwise, we will accept $\max\{p, (m-x)^+\}$ high-reward type requests. In summary, the total reward is $\max\{p, \min\{y, (m-x)^+\}\}r_h + \min\{x, m-p\}r_\ell$.

When it is not clear whether we are in under- or over-protecting case, we simply use $\mathrm{CP}(p; A = (x,y))$ to denote compatible ratio of point $A = (x,y)$. Note that in the definition of $\mathrm{CP}(p; A)$ we considered ordered arrivals where low-reward agents arrive first. This is because of Lemma 1 where we show for any point A, the compatible ratio is minimized by considering its ordered sequence. Then, by definition, the consistent ratio and robust ratio of a PLA $\mathcal{A}$ with PL function $p(\cdot)$ are given by

$$\textsc{consis}(\mathcal{A}) = \inf_{(x,y)\in\mathcal{R}} \mathrm{CP}(p(x); (x,y)), \qquad \textsc{robust}(\mathcal{A}) = \inf_{(x,y)\geqslant 0} \mathrm{CP}(p(x); (x,y)). \quad (8)$$

4. Optimization Problems to Characterize Pareto Optimal Algorithms

In this section, we outline an optimization problem that characterizes a class of parameterized Pareto optimal algorithms. We proceed by transforming the constraints of this optimization, resulting in a more tractable problem. The solution to this modified optimization problem is then presented in Section 5.

A C-Pareto optimal algorithm achieves the highest possible robust ratio while obtaining a consistent ratio of at least C for any $C \leqslant C^\star(\mathcal{R})$, where $C^\star(\mathcal{R})$ is the highest possible consistent ratio for a convex ML region $\mathcal{R}$ (in the absence of any constraint on the robust ratio); see Section 6 for more details about $C^\star(\mathcal{R})$. Suppose that the C-Pareto optimal algorithm is a PLA. Then, to design a C-Pareto optimal algorithm, by Equation (8), we consider the following optimization problem:

$$\begin{aligned}
\max_{R\in[0,1], p(x): x\in[0,\max\{m,\bar{x}\}]} \quad & R \\
s.t. \quad & \mathrm{CP}(p(x); (x,y)) \geqslant C, && (x,y) \in \mathcal{R}, && (9)\\
& \mathrm{CP}(p(x); (x,y)) \geqslant R, && x, y \geqslant 0, && (10)\\
& p(x) \geqslant 0 \text{ is continuous} && x \in [0, \max\{m, \bar{x}\}], && (11)\\
& -1 \leqslant p'(x) \leqslant 0 \quad \text{a.e.,} && x \in [0, \max\{m, \bar{x}\}] && (12)
\end{aligned}$$

(C-Pareto)

Recall that $\bar{x}$ is defined in Equation (4). The first set of constraints ensures that the compatible ratio of any point $(x,y)\in\mathcal{R}$ at $p(x)$ is at least C. The second set of constraints ensures that the compatible ratio of any point (x,y) in the first quadrant at $p(x)$ is at least R. The third and fourth sets of constraints, which we refer to as validity constraints, ensure that the protection level is valid per Definition 1. Note that the optimal solution to Problem (C-Pareto) only determines the PL function $p(x)$ for $x\in[0,\max m,\bar{x}]$. As per the definition of PLAs in Definition 1, for any $x>\max\{m,\bar{x}\}$, we set $p(x)=p(\max\{m,\bar{x}\})$.

As we will show later in Theorem 3, the optimal solution to Problem (C-Pareto) leads to a PLA that is an optimal C-Pareto algorithm among any deterministic and randomized non-anticipating algorithms (not only the PLAs).

To solve Problem (C-Pareto), we first transform its first and second sets of constraints (i.e., the consistency and robustness constraints). This transformation, which is done in Sections 4.1 and 4.2, plays a key role in our design. We then solve Problem (C-Pareto) using the properties of the transformed constraints.

4.1. Transforming the Consistency Constraints

Here, we transform the consistency constraints in Problem (C-Pareto) that require $\mathrm{CP}(p(x);(x,y))\geqslant C$ for any $(x,y)\in\mathcal{R}$. To do so, let us fix $x\in[\underline{x},\bar{x}]$ and its protection level $p(x)$. Then, the worst points with the minimum consistent ratio occur at the lower and upper boundaries of the ML region. That is, as shown in Lemma 2 stated below, for any $p\in[0,m]$, $\min_{y\in[\underline{h}(x),\bar{h}(x)]}\mathrm{CP}(p;(x,y))$ is either $\mathrm{CP}(p;(x,\underline{h}(x)))$ or $\mathrm{CP}(p;(x,\bar{h}(x)))$, where $\underline{h}(x)$ and $\bar{h}(x)$ are defined in Equation (5). The lemma further shows that in the case of under-protection (i.e., the protection level p less than high-reward requests y), the compatible ratio decreases as y increases while in the case of over-protection (i.e., $p>y$), the compatible ratio decreases as y decreases.

Lemma 2 **(Monotonicity of the Compatible Ratio CP$(p;(x,y))$ w.r.t. y).** *Consider any $x\in[\underline{x},\bar{x}]$ and $\min\{m,y_1\}\leqslant\min\{m,y_2\}$. For any protection level p with $p\leqslant\min\{m,y_1\}$, we have $CP_u(p;(x,y_1))\geqslant CP_u(p;(x,y_2))$, and for any protection level p with $p\geqslant\min\{m,y_2\}$, we have $CP_o(p;(x,y_2))\geqslant CP_o(p;(x,y_1))$. This further implies that for any $x\in[\underline{x},\bar{x}]$ and $p\geqslant 0$, we have*

$$\min_{y\in[\underline{h}(x),\bar{h}(x)]}\{CP(p;(x,y))\}=\min\left\{CP(p;(x,\underline{h}(x))),CP(p;(x,\bar{h}(x)))\right\}.$$

In light of Lemma 2, we define the following two functions $u(x;C)$ and $l(x;C)$. Roughly speaking, under $u(x;C)$, (if possible) the compatible ratio at (the worst over-protected point) $(x,\underline{h}(x))$ is equal to C while under $l(x;C)$, (if possible) the compatible ratio at (the worst under-protected point) $(x,\bar{h}(x))$ is equal to C.

Definition 2 (Upper Bound Function). *For any $C\in[0,1]$, we define*

$$u(x;C)=\sup\left\{p\in[0,m]:CP_o(p;(x,\underline{h}(x)))=C\right\}\qquad x\in[\underline{x}_u,\bar{x}_u] \tag{13}$$

while we set $u(x;C)=m$ for any $x\in[0,\underline{x}_u]$ and $u(x;C)=u(\bar{x}_u;C)$ for any $x\in[\bar{x}_u,\bar{x}]$. Here,

$$\bar{x}_u=\begin{cases} x_L & \text{if } x_L+y_L\geqslant m;\\ \sup\{x\in[x_L,\bar{x}]:(1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-)-C<0\} & \text{Otherwise}, \end{cases} \tag{14}$$

and $\underline{x}_u=\sup\{\underline{x}<x<\bar{x}_u:CP_o(m;(x,\underline{h}(x)))\geqslant C\}$, where $\underline{\mathcal{H}}(x)=\min\{\underline{h}(x),m\}$. We set $\underline{x}_u=\underline{x}$ when its defining set is empty. [3]

Definition 3 (Lower Bound Function). *For any $C\in[0,1]$, let*

$$l(x;C)=\inf\left\{p\in[0,m]:CP_u(p;(x,\bar{h}(x)))=C\right\}\qquad x\in[x_H,\bar{x}_l], \tag{15}$$

while we set $l(x;C)=l(x_H;C)$ for any $x\in[0,x_H]$ and $l(x;C)=0$ for $x\in[\bar{x}_l,\bar{x}]$. Here, $\bar{x}_l=\inf\{x_H<x<\bar{x}:CP_u(0;(x,\bar{h}(x)))\geqslant C\}$. We set $\bar{x}_l$ to $\bar{x}$ when its defining set is empty.

We now discuss the upper and lower bounds. As for the upper bound, at $x\in[\underline{x},\underline{x}_u)$, even if we set the protection level to m—which means rejecting all low-reward type requests—the compatible ratio at point $(x,\underline{h}(x))$ exceeds C. Further, for any $x\in[\underline{x}_u,\bar{x}_u]$, if we set the protection level to $u(x;C)$, the compatible ratio at (the worst over-protected point) $(x,\underline{h}(x))$ is exactly equal to C. As we will show later in Lemma 6, function $u(x;C)$ is convex and decreasing in x.

As for the lower bound, at $x\in(\bar{x}_l,\bar{x}]$, even if we set the protection level to zero (i.e., we accept all low-reward requests when there is resource available), the compatible ratio at point $(x,\bar{h}(x))$ exceeds C. (See Lemma 13 for a formal argument.) Note that for any $x\in[\underline{x}_l,\bar{x}_l]$, if we set the protection level to $l(x;C)$, the compatible ratio at (the worst under-protected point) $(x,\bar{h}(x))$ is exactly equal to C. As we will show later in Lemma 6, function $l(x;C)$ is concave and decreasing in x. Further, $l(x;C)\leqslant u(x;C)$ for any $x\in[0,\bar{x}]$.

Lemma 6 in the appendix presents some key properties of the two functions $u(\cdot;C)$ and $l(\cdot;C)$. It shows that while $l(x;C)$ is concave in x, $u(x;C)$ is convex in x. The lemma also presents the derivative of these functions w.r.t. x and shows that for any $C\leqslant C^\star(\mathcal{R})$, $l(x;C)$ is less than or equal

[3] We note that $\bar{x}_u$ is well defined because L is the lowest point, and hence $\underline{\mathcal{H}}'(x_L^-)\leqslant 0$, and $(1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x_L^-)-C<0$.

to $u(x;C)$ for any $x \in [\underline{x}, \bar{x}]$. In addition, it shows that both $l(x;C)$ and $u(x;C)$ are also continuous monotone function in C for any $x \in [\underline{x}, \bar{x}]$.

We are now ready to present our transformation result.

Lemma 3 **(Transforming the Consistency Constraints-I)**. *For any $C \leqslant C^\star(\mathcal{R})$, Problem* (C-Pareto) *is equivalent to the following optimization problem:*

$$\max_{R\in[0,1],p(x):x\in[0,\max\{m,\bar{x}\}]} R \qquad s.t. \quad l(x;C) \leqslant p(x) \leqslant u(x;C), \quad x \in [0,\bar{x}], \qquad and\ (10),\ (11),\ (12), \tag{16}$$

where $l(x;C)$ and $u(x;C)$ are defined in Equations (15) *and* (13)*, respectively.*

The proof of all lemmas in this section can be found in Appendix C. Lemma 3 demonstrates that enforcing lower and upper bounds on the PL function $p(\cdot)$ satisfies the consistency constraint of Problem (C-Pareto). These bounds, denoted by $l(\cdot;C)$ and $u(\cdot;C)$, respectively, can be easily computed and depend on the ML region. However, these bounds may not be tight since their slope can be less than -1. Recall that the optimal solution to Problem (C-Pareto) should be a valid PL function, i.e., a non-increasing function with a slope greater than or equal to -1. With this in mind, the following lemma presents an alternative (valid) lower bound denoted by $\widetilde{l}(\cdot;C)$, which is a non-increasing function with a slope greater than or equal to -1. This lower bound is defined as follows:

$$\widetilde{l}(x;C) = \begin{cases} l(x;C) & x \in [0, x_{-1}] \\ (-(x - x_{-1}) + l(x_{-1};C))^+ & x \in [x_{-1}, \bar{x}]. \end{cases} \tag{17}$$

Note that $x^+ = \max\{x, 0\}$ and $x_{-1} = \sup\{x \in [x_H, \bar{x}] : \frac{\partial l(x^-;C)}{\partial x} \leqslant -1\}$. When $\frac{\partial l(x^-;C)}{\partial x} \geqslant -1$ for any $x \in [x_H, \bar{x}]$, we set x_{-1} to $\bar{x}$, and in this case, $l(x;C) = \widetilde{l}(x;C)$ for any $x \in [\underline{x}, \bar{x}]$. We recall that $\frac{\partial l(x^-;C)}{\partial x} = C\overline{\mathcal{H}}'(x)$ by Lemma 6, and that $\overline{\mathcal{H}}(\cdot) = \min\{\bar{h}(\cdot), m\}$. We observe that $\widetilde{l}(x;C)$ is a non-increasing continuous function whose slope is greater than or equal to -1, due to the concavity of $l(x;C)$ in x as shown in Lemma 6. Moreover, we have $l(x;C) \leqslant \widetilde{l}(x;C)$ for any $x \in [\underline{x}, \bar{x}]$. Finally, the following lemma shows that $\widetilde{l}(\cdot;C)$ is a tighter lower bound for the PL function $p(\cdot)$.

Lemma 4 **(Transforming the Consistency Constraints-II)**. *For any $C \leqslant C^\star(\mathcal{R})$, Problem* (C-Pareto) *is equivalent to the following optimization problem:*

$$\max_{R\in[0,1],p(x):x\in[0,\max\{m,\bar{x}\}]} R \qquad s.t. \quad \widetilde{l}(x;C) \leqslant p(x) \leqslant u(x;C), \quad x \in [0,\bar{x}], \qquad and\ (10),\ (11),\ (12), \tag{18}$$

where $\widetilde{l}(x;C)$ and $u(x;C)$ are defined in Equations (17) *and* (13)*, respectively.*

We finish this section by presenting an example.

Example 1. *Figure 2 presents these upper and lower bounds for two ML regions*

$$\mathcal{R}_1 = \{4 \leqslant x \leqslant 16\} \cap \{4 \leqslant y \leqslant 16\} \cap \{20 \leqslant x + y \leqslant 25\}$$

and

$$\mathcal{R}_2 = \{4 \leqslant x \leqslant 16\} \cap \{4 \leqslant y \leqslant 16\} \cap \{0 \leqslant y - x \leqslant 5\}.$$

The first region represents a scenario where both the low-reward and high-reward requests are between 4 *and* 16*, and the total number of requests is between* 20 *and* 25*. The second ML region represents a scenario where both the low-reward and high-reward requests are between* 4 *and* 16*, and the difference between the high-reward and low-reward requests is between* 0 *and* 5*. The regions* $\mathcal{R}_1$ *and* $\mathcal{R}_2$ *are illustrated as shaded grey areas in Figure 2. Here,* $m = 20$*,* $r_h = 1$*,* $r_\ell = 1/3$*, and* $C = 0.8$*. Observe that while the lower and upper bounds are constant values for* $\mathcal{R}_1$*, this is not the case for* $\mathcal{R}_2$*.*

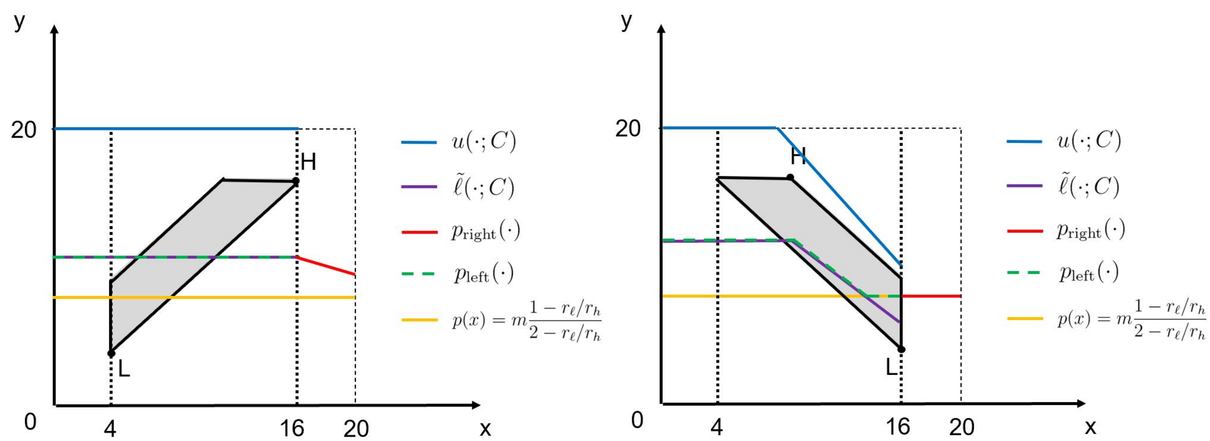


Figure 2 The upper bound $u(\cdot;C)$ (Equation (13)), the lower bound $l(\cdot;C) = \tilde{l}(\cdot;C)$ (Equation (17)), $p_{\text{right}}(x)$ (Equations (23) and (25)), and $p_{\text{left}}(x)$ (Equation (26)) for $\mathcal{R}_1$ and $\mathcal{R}_2$, where $\mathcal{R}_1 = \{4 \leqslant x \leqslant 16\} \cap \{4 \leqslant y \leqslant 16\} \cap \{20 \leqslant x + y \leqslant 25\}$ and $\mathcal{R}_2 = \{4 \leqslant x \leqslant 16\} \cap \{4 \leqslant y \leqslant 16\} \cap \{0 \leqslant y - x \leqslant 5\}$. Here, $m = 20$, $r_h = 1$, $r_\ell = 1/3$, and $C = 0.8$. In both figures, to be C-consistent, the optimal PL $p^\star$, which is a concatenation of p_{left} and p_{right}, protects more resources for high-reward requests than the Ball and Queyranne (2009)'s algorithm with the fixed PL of $m\frac{1-r_\ell/r_h}{2-r_\ell/r_h}$.

4.2. Transforming the Robustness Constraints

In this section, we aim to transform the robustness constraints of Problem (C-Pareto) which mandate that $\mathrm{CP}(p(x);(x,y)) \geqslant R$ for any $(x,y) \geqslant 0$. Similar to the previous section, we replace this constraint with lower and upper bound constraints on the (PL) function $p(\cdot)$. This enables us to effectively address the robustness constraints while simplifying the optimization problem.

Lemma 5 **(Transforming the Robustness Constraints)**. *For any $R \in (0,\rho]$ with $\rho = \frac{1}{2-r_\ell/r_h}$, and for $x \in [0,m]$, define*

$$\underline{g}(x;R) = \frac{m(R - r_\ell/r_h)}{1 - r_\ell/r_h}, \qquad \bar{g}(x;R) = -Rx + m. \tag{19}$$

For $x > m$, we have $\underline{g}(x;R) = \underline{g}(m;R)$ and $\bar{g}(x;R) = \bar{g}(m;R)$. For any $C \leqslant C^\star(\mathcal{R})$, Problem (C-Pareto) *is equivalent to the following optimization problem:*

$$\begin{aligned} &\max_{R\in[0,\rho],p(x):x\in[0,\max\{m,\bar{x}\}]} R \\ &\quad s.t. \quad \widetilde{l}(x;C) \leqslant p(x) \leqslant u(x;C), \qquad x \in [0,\bar{x}]\,, \\ &\qquad\qquad \underline{g}(x;R) \leqslant p(x) \leqslant \bar{g}(x;R), \qquad x \in [0,\max\{m,\bar{x}\}]\,, \\ &\qquad\qquad \textit{Validity Constraints } (11),\ (12)\,. \end{aligned} \tag{C-Pareto-Trans}$$

5. Pareto Optimal Algorithms

In Section we present an optimal solution to the transformed problem (C-Pareto-Trans), denoted by $p^\star(\cdot)$, which achieves the optimal robust ratio, denoted by $R^\star$. To do so, as illustrated in Figure 1, we first focus on solving the transformed problem (C-Pareto-Trans) for any $x \geqslant \bar{x}$; see Section (5.2). We refer to this problem as the *right problem* and denote it by (C-Pareto-right)[4]:

$$\begin{aligned} R_{\text{right}} = &\max_{R\in[0,\rho],p(x):x\in[\bar{x},\max\{m,\bar{x}\}]} R \\ &s.t. \ \ \widetilde{l}(\bar{x};C) \leqslant p(\bar{x}) \leqslant u(\bar{x};C), \\ &\qquad \underline{g}(x;R) \leqslant p(x) \leqslant \bar{g}(x;R), \qquad x \in [\bar{x},\max\{m,\bar{x}\}]\,, \\ &\qquad \text{Validity Constraints (11), (12)}, \qquad x \in [\bar{x},\max\{m,\bar{x}\}] \end{aligned} \tag{C-Pareto-right}$$

In the right problem, it is sufficient to satisfy the consistency lower and upper bound constraints only at $\bar{x}$, as indicated by the first constraint in Problem (C-Pareto-right). Let $p_{\text{right}} : [\bar{x}, \max m, \bar{x}] \mapsto [0,m]$ be the optimal solution to the right problem, and define R_{right} as the optimal objective value of the right problem. Crucially, in Theorem 3, we prove that $p^\star(x)$, which is the optimal solution to Problem (C-Pareto-Trans), is equal to $p_{\text{right}}(x)$ for any $x \in [\bar{x}, \max\{m,\bar{x}\}]$. Note that the PL function $p_{\text{right}}(\cdot)$ depends on C, but for the sake of simplicity, we omit this dependence from our exposition.

After characterizing the optimal solution to $p^\star(x)$ for any $x \geqslant \bar{x}$, we then focus on solving the transformed problem (C-Pareto-Trans) for any $x \in [0,\bar{x}]$. See Section 5.3. We refer to this problem

[4] If $\bar{x} > m$, we only need to optimize a single point $p(\bar{x})$.

as the *left problem* and denote it by (C-Pareto-left). The left problem is the same as the transformed problem (C-Pareto-Trans) for any $x \in [0,\bar{x}]$ with one additional constraint that enforces $p(\bar{x})$ to be $p_{\text{right}}(\bar{x})$, which is equal to $p^\star(\bar{x})$. The left problem can be written as

$$\begin{aligned} R_{\text{left}} = \max_{R\in[0,\rho],p(x):x\in[0,\bar{x}]} \quad & R \\ s.t. \quad & \widetilde{l}(x;C) \leqslant p(x) \leqslant u(x;C), \qquad x \in [\underline{x},\bar{x}]\,, \\ & \underline{g}(x;R) \leqslant p(x) \leqslant \bar{g}(x;R), \qquad x \in [0,\bar{x}]\,, \\ & \text{Validity Constraints (11), (12)} \qquad x \in [0,\bar{x}] \\ & p(\bar{x}) = p_{\text{right}}(\bar{x})\,. \end{aligned} \tag{C-Pareto-left}$$

Let $p_{\text{left}} : [0,\bar{x}] \mapsto [0,m]$ be the optimal solution to the left problem and define R_{left} as the optimal objective value of the left problem. In Theorem 5, we will show that $p^\star(x) = p_{\text{left}}(x)$ for any $x \in [0,\bar{x}]$. That is, the optimal solution to the left problem fully characterizes the optimal solution to Problem (C-Pareto-Trans) for any $x \in [0,\bar{x}]$. Further, we will show in Theorem 3 that the optimal objective value of Problem (C-Pareto-Trans), $R^\star$, is equal to $\min\{R_{\text{right}}, R_{\text{left}}\}$.

5.1. Optimal Solution to Problem C-Pareto-Trans and C-Pareto Optimal Algorithm

Here is the main result of this section where we present the optimal solution to Problem (C-Pareto-Trans) denoted by $p^\star(\cdot)$. More importantly, we show that among any online algorithms Π, the PLA with the PL function of $p^\star(\cdot)$ maximizes the robust ratio while ensuring its consistent ratio is at least C. That is, it is an optimal solution to Problem (3).

THEOREM 3 **(Optimal Solution to** (C-Pareto-Trans) **and C-Pareto Optimal Algorithm)**. *Consider any* $0 \leqslant C \leqslant C^\star(\mathcal{R})$.

1. *The optimal objective value of Problem* (C-Pareto-Trans), *is* $R^\star = \min\{R_{right}, R_{left}\}$, *where* R_{right} *and* R_{left} *are the optimal objective value of Problem* (C-Pareto-right) *and* (C-Pareto-left), *respectively.*
2. *Algorithm 1 presents an optimal solution to Problem* (C-Pareto-Trans). *That is, at the optimal solution to Problem* (C-Pareto-Trans), *denoted by* $p^\star(\cdot)$, *for any* $x \in [0,m]$, *we set*

$$p^\star(x) = \begin{cases} p_{left}(x) & x \in [0,\bar{x}] \\ p_{right}(x) & x \in [\bar{x}, \max\{m,\bar{x}\}]\,, \end{cases} \tag{20}$$

where $p_{right}(\cdot)$ *and* $p_{left}(\cdot)$ *are the optimal solutions to the right and left problems, respectively.*

3. *A PLA with the PL function of* $p^\star(\cdot)$ *is an optimal solution to Problem* (3). *That is, among any online algorithms* Π, *the aforementioned algorithm maximizes the robust ratio while ensuring its consistent ratio is at least* C.

Algorithm 1 Optimal Solution to Problem (C-Pareto-Trans) and C-Pareto Optimal Algorithm.

Input: Convex set $\mathcal{R}$, resource capacity m, and parameter $C \in [0, C^\star(\mathcal{R})]$.

Output: Optimal solution to Problem (C-Pareto-Trans), $p^\star(\cdot) : [0, \max\{m, \bar{x}\}] \mapsto [0, m]$, and its optimal objective $R^\star$.

For any $x \in [0, \max\{m, \bar{x}\}]$, set

$$p^\star(x) = \begin{cases} p_{\text{left}}(x) & x \in [0, \bar{x}] \\ p_{\text{right}}(x) & x \in [\bar{x}, \max\{m, \bar{x}\}] \end{cases}, \tag{21}$$

where $p_{\text{right}}(\cdot)$ and $p_{\text{left}}(\cdot)$ are defined in Algorithms 2 and 3, respectively.

Let $R_{\text{right}} = \min\{\text{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)), \text{CP}_u(p_{\text{right}}(m); (\max\{m, \bar{x}\}, m))$, and set

$$R^\star = \min\{R_{\text{right}}, \inf_{x \in [0, \bar{x}]} \text{CP}_o(p_{\text{left}}(x); (x, 0))\}.$$

The proof of Theorem 3 is stated in Appendix D. To show the first statement, as the main step, we need to argue that $p^\star(\bar{x}) = p_{\text{right}}(\bar{x})$. To do so, we consider two cases, where in the first case $R_{\text{right}} \geqslant R_{\text{left}}$, and in the second case, $R_{\text{right}} < R_{\text{left}}$. In the first case, the optimality of $p^\star(\bar{x})$ can be argued using Theorem 4, where we present an optimal solution to the right problem. Otherwise, for the case where $R_{\text{left}} < R_{\text{right}}$, we show the result by contradiction while using properties of the lower bound $\widetilde{l}(\cdot; C)$. The proof of the second statement follows from Theorems 4 and 5 in which we present an optimal solution to the left and right problems.

To show the third statement, we first note that the optimal robust ratio $R^\star$ is the minimum of three terms: $\text{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0))$, $\text{CP}_u(p_{\text{right}}(m); (\max\{m, \bar{x}\}, m))$, and $\inf_{x \in [0, \bar{x}]} \text{CP}_o(p_{\text{left}}(x); (x, 0))\}$. Depending on which term attains the minimum, we construct worst-case arrival sequences to show that no algorithm can perform better than $R^\star$. For the purpose of this discussion, let us assume that $R^\star$ is equal to the compatible ratio of point (m, m) (i.e., $\text{CP}_u(p_{\text{right}}(m); (\max\{m, \bar{x}\}, m)))$, which is one of the three aforementioned terms.

For this case, we define two (ordered) input sequences. In the first input sequence, I_1, low-reward requests with $\bar{x}_u \leqslant \bar{x}$ arrive first, followed by high-reward requests with $\underline{h}(\bar{x}_u)$. One can think of I_1 as an arrival sequence consistent with the ML advice. In the second input sequence, I_2, m low-reward requests arrive first, followed by m high-reward requests. Here, one can think of I_2 as an arrival sequence outside with the ML advice. Before receiving $\bar{x}_u$ low-reward requests, any deterministic or randomized algorithm cannot differentiate between the two input sequences and must decide how many low-reward requests to accept in expectation. We then show that, on these input sequences, to achieve a consistent ratio of C on I_1, no algorithm can obtain a robust ratio greater than $R^\star$ on

I_2. This is demonstrated by arguing that, on the arrival sequence I_1, to achieve a consistent ratio of at least C, the algorithm must accept at least $m - u(\bar{x}_u; C)$ low-reward agents. This, in turn, prevents any algorithm from performing better than $R^\star$ on I_2. For other cases, we also construct two input sequences, but the contradiction point is different in each case. See Section D for details.

We finish this section by revisiting our running example and present the tradeoff between $R^\star$ and C. The left figure of Figure 3 displays the optimal robust ratio (i.e., the optimal value to Problem (C-Pareto) or the optimal value to the original Problem (3)) versus C for the two ML regions, $\mathcal{R}_1$ and $\mathcal{R}_2$. For each ML region $\mathcal{R}_i$, $i \in [2]$, we consider $C \leqslant C^\star(\mathcal{R}_i)$, as Problem (C-Pareto) is infeasible for any $C > C^\star(\mathcal{R}_i)$.

As expected, for both ML regions, the optimal robust ratio decreases as C increases. The middle and right figures of Figure 3 display $p^\star(\cdot)$ for $C = 0.8$ and 0.89 under the two regions, respectively. We observe that by increasing C, $p^\star(\cdot)$ increases as well, which results in protecting more resources for high-reward requests.

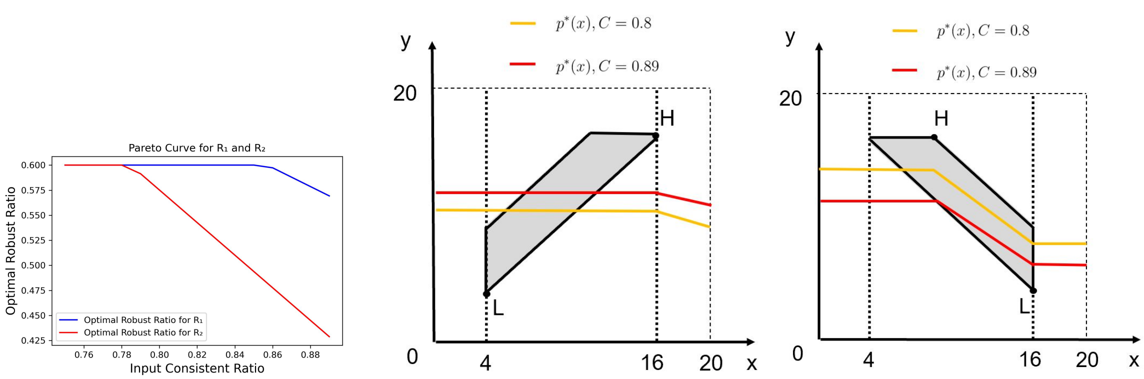


Figure 3 **The figure shows the optimal robust ratio (i.e., the optimal value to Problem** (C-Pareto) **or the optimal value to the original Problem** (3)**) versus** C **for the two ML region** $\mathcal{R}_1$ **and** $\mathcal{R}_2$**. Here,** $\mathcal{R}_1 = \{4 \leqslant x \leqslant 16\} \cap \{4 \leqslant y \leqslant 16\} \cap \{20 \leqslant x + y \leqslant 25\}$ **and** $\mathcal{R}_2 = \{4 \leqslant x \leqslant 16\} \cap \{4 \leqslant y \leqslant 16\} \cap \{0 \leqslant y - x \leqslant 5\}$**.**

In the following sections, we begin by presenting $p_{\text{right}}(\cdot)$, followed by a characterization of $p_{\text{left}}(\cdot)$.

5.2. Optimal Solution to the Right Problem (C-Pareto-right)

Algorithm (2) presents the optimal solution to the right problem (C-Pareto-right), $p_{\text{right}}(\cdot)$, and the optimal objective value of this problem R_{right}. First, the algorithm presents the optimal solution at $\bar{x}$, i.e., $p_{\text{right}}(\bar{x})$, which then determines R_{right}. Second, the algorithm presents the optimal solution at any $x \in (\bar{x}, \max\{m, \bar{x}\}]$, using $p_{\text{right}}(\bar{x})$ and R_{right}. Notice that if $\bar{x} \geqslant m$, $(\bar{x}, \max\{m, \bar{x}\}] = (\bar{x}, \bar{x}]$, is not well-defined. So, we skip the second step, and we only need to solve for $p_{\text{right}}(\bar{x})$.

Algorithm 2 Optimal Solution to Problem (C-Pareto-right).

Input: Convex set $\mathcal{R}$, resource capacity m, and parameter $C \in [0, C^\star(\mathcal{R})]$.

Output: Optimal solution to the right problem (C-Pareto-right), $p_{\text{right}} : [\bar{x}, \max\{m, \bar{x}\}] \mapsto [0, m]$, and optimal objective value of the right problem (C-Pareto-right), denoted by R_{right}.

- **Optimal solution at** $\bar{x}$ Let

$$p_{\text{right}}(\bar{x}) = \underset{p \in [\widetilde{l}(\bar{x};C), u(\bar{x};C)]}{\arg\min} |p - \underline{g}(\bar{x})| \,,$$

where $\underline{g}(x) = \underline{g}(x; R = \frac{1}{2 - r_\ell/r_h}) = \frac{1 - r_\ell/r_h}{2 - r_\ell/r_h} m$, and $\widetilde{l}(\cdot; C)$ and $u(\cdot; C)$ are respectively defined in Equations (17) and (13). Further, define

$$R_{\text{right}} = \min\left\{\text{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)), \text{CP}_u(p_{\text{right}}(\bar{x}); (\max\{m, \bar{x}\}, m))\right\} .$$

- **Optimal Solution at** $x \in (\bar{x}, \max\{m, \bar{x}\}]$**.** For any $x \in (\bar{x}, \max\{m, \bar{x}\}]$, define

$$p_{\text{right}}(x) = \begin{cases} \max\{-x + \bar{x} + p_{\text{right}}(\bar{x}; C), \underline{g}(x)\} & p_{\text{right}}(\bar{x}) \in [\underline{g}(x), \bar{g}(x)] \\ \underline{g}(x; R_{\text{right}}) & p_{\text{right}}(\bar{x}) < \underline{g}(\bar{x}), \\ \bar{g}(x; R_{\text{right}}) & p_{\text{right}}(\bar{x}) > \bar{g}(\bar{x}), \end{cases} \tag{22}$$

where $\underline{g}(\cdot; R)$, $\bar{g}(\cdot; R)$ are respectively defined in Equation (19).

Return. $p_{\text{right}}(x)$, $x \in [\bar{x}, \max\{m, \bar{x}\}]$, and R_{right}.

Recall that $\rho = \frac{1}{2 - r_\ell/r_h}$ is the optimal consistent ratio (obtained by Ball and Queyranne (2009)) when the ML region is $\{(x, y) : x, y \geqslant 0\}$. With a little abuse of notation, for $x \in [\bar{x}, \max\{m, \bar{x}\}]$, we let

$$\bar{g}(x) = \bar{g}(x; R = \rho), \quad \text{and} \quad \underline{g}(x) = \underline{g}(x; R = \rho) \,,$$

where $\bar{g}(x; R)$ and $\underline{g}(x; R)$ are defined in Equation (19). Then, in Algorithm 2, we have

$$p_{\text{right}}(\bar{x}) = \underset{p \in [\widetilde{l}(\bar{x};C), u(\bar{x};C)]}{\arg\min} |p - \underline{g}(\bar{x})| \,. \tag{23}$$

Clearly, $p_{\text{right}}(\bar{x})$ satisfies the first constraint in the right problem; that is, $p_{\text{right}}(\bar{x}) \in [\widetilde{l}(\bar{x}; C), u(\bar{x}; C)]$, as desired. To set $p_{\text{right}}(\bar{x})$, the algorithm compares the feasible interval $[\widetilde{l}(\bar{x}; C), u(\bar{x}; C)]$ with $\underline{g}(x) = \underline{g}(x; R = \rho)$. Here, $\underline{g}(x; \rho) = \frac{1 - r_\ell/r_h}{2 - r_\ell/r_h} m$ is the optimal PL in the setting studied in Ball and Queyranne (2009) when the ML region is $\{(x, y) : x, y \geqslant 0\}$. At a high level, if we can set $p_{\text{right}}(\bar{x})$ to $\underline{g}(\bar{x})$, the optimal (right) robust ratio R_{right} will be equal to ρ (which is the maximum robust ratio for any ML region). However, setting $p_{\text{right}}(\bar{x})$ to $\underline{g}(\bar{x})$ is not always possible. In such a case, we either set $p_{\text{right}}(\bar{x})$ to $u(\bar{x}; C)$ or $\widetilde{l}(\bar{x}; C)$. The PL $p_{\text{right}}(\bar{x})$ then determines R_{right}:

$$R_{\text{right}} = \min\left\{\text{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)), \text{CP}_u(p_{\text{right}}(\bar{x}); (\max\{m, \bar{x}\}, m))\right\} . \tag{24}$$

This shows that in the right problem, points $(\bar{x}, 0)$ or $(\max\{m, \bar{x}\}, m)$ play a crucial role, determining the optimal objective value.

Given the optimal objective value of R_{right}, if $\bar{x} < m$, for any $x \in (\bar{x}, m]$, we only need to make sure that $\underline{g}(x; R_{\text{right}}) \leqslant p(x) \leqslant \bar{g}(x; R_{\text{right}})$. This is achieved by setting

$$p_{\text{right}}(x) = \begin{cases} \max\{-x + \bar{x} + p_{\text{right}}(\bar{x}; C), \underline{g}(x)\} & p_{\text{right}}(\bar{x}) \in [\underline{g}(x), \bar{g}(x)] \\ \underline{g}(x; R_{\text{right}}) & p_{\text{right}}(\bar{x}) < \underline{g}(\bar{x}), \\ \bar{g}(x; R_{\text{right}}) & p_{\text{right}}(\bar{x}) > \bar{g}(\bar{x}). \end{cases} \tag{25}$$

As it becomes more clear in the proof of Theorem 4, when $p_{\text{right}}(\bar{x}) \in [\underline{g}(\bar{x}), \bar{g}(\bar{x})]$, the optimal objective value of the right problem R_{right} is indeed $\rho = 1/(2 - r_\ell / r_h)$. For the other cases where either $p_{\text{right}}(\bar{x}) < \underline{g}(\bar{x})$ or $p_{\text{right}}(\bar{x}) > \bar{g}(\bar{x})$, we have $R_{\text{right}} < \rho$. There, we set $p_{\text{right}}(x)$ such that the compatible ratio at any points $(x, 0)$ and (x, m) (with $x \in (\bar{x}, m]$) is greater than or equal to R_{right}, defined in Equation (24).

Theorem 4 **(Optimal Solution to the Right Problem)**. *Algorithm 2 presents an optimal solution to Problem* (C-Pareto-right)*. That is, at the optimal solution to Problem* (C-Pareto-right)*, denoted by $p_{right}(\cdot)$, we set $p_{right}(x)$ based on Equations* (23) *and* (25)*. Furthermore, the optimal objective value of Problem* (C-Pareto-right)*, R_{right}, is given in Equation* (24).

We finish this section by revisiting the examples in Figure 2 (Example 1). The figure displays the optimal $p_{\text{right}}(\cdot) : [16, 20] \mapsto [0, 20]$ for $\mathcal{R}_1$ and $\mathcal{R}_2$ respectively. Both regions have $\underline{g}(\bar{x}) = 8$ and $\bar{g}(\bar{x}) = 10.4$. In $\mathcal{R}_1$, since $p_{\text{right}}(\bar{x}) \geqslant \tilde{l}(\bar{x}; C) = 10.8 > \bar{g}(\bar{x})$, we cannot achieve $\rho = 1/(2 - r_\ell/r_h) = 0.6$ if the adversary chooses $(\bar{x}, 0)$. Thus, we choose $p_{\text{right}}(\bar{x}) = \tilde{l}(\bar{x}; C) = 10.8$ by Equation (23), and for $x \in (\bar{x}, m]$, we set $p_{\text{right}}(x)$ such that $\text{CP}_o(p_{\text{right}}(x); (x, 0)) \geqslant \text{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)) = R_{\text{right}}$. In $\mathcal{R}_2$, since $\underline{g}(\bar{x}) = 8 \in [\tilde{l}(\bar{x}; C), u(\bar{x}; C)]$, we can achieve ρ, and we set $p_{\text{right}}(x) = \underline{g}(x)$ for $x \in [\bar{x}, m]$.

5.3. Optimal Solution to Problem (C-Pareto-left)

In this section, we present an optimal solution to the left problem (C-Pareto-left), denoted by $p_{\text{left}}(\cdot) : [0, \bar{x}] \mapsto [0, m]$. See Algorithm 3. The algorithm shows that at the optimal solution, we set

$$p_{\text{left}}(x) = \max\{\tilde{l}(x; C), p_{\text{right}}(\bar{x})\}, \quad x \in [0, \bar{x}], \tag{26}$$

where $p_{\text{right}}(\bar{x})$ is the optimal solution to Problem (C-Pareto-right) at $\bar{x}$, and $\tilde{l}(x; C)$ is defined in Equation (17).

The optimal solution to the left problem is obtained by one observation. We show in the proof of Theorem 5 that in this problem, one can ignore the lower bound constraint $p(x) \geqslant \underline{g}(x; R)$, $x \in [0, \bar{x}]$.

Algorithm 3 Optimal Solution to Problem (C-Pareto-left).

Input: Convex set $\mathcal{R}$, resource capacity m, and parameter $C \in [0, C^{\star}(\mathcal{R})]$.

Output: Optimal solution to the right problem (C-Pareto-left), $p_{\text{left}} : [0, \bar{x}] \mapsto [0, m]$.

For any $x \in [0, \bar{x}]$, define

$$p_{\text{left}}(x) = \max\{\widetilde{l}(x; C), p_{\text{right}}(\bar{x})\}\,, \quad x \in [0, \bar{x}]\,, \tag{27}$$

where $p_{\text{right}}(\bar{x})$ is the optimal solution to Problem (C-Pareto-right) at $\bar{x}$, and $\widetilde{l}(x; C)$ is defined in Equation (17).

Return. $p_{\text{left}}(x)$, $x \in [0, \bar{x}]$.

Given this simplification, to present an optimal solution to Problem (C-Pareto-left), we need to find the largest value of R that satisfies the conditions $\max\{\widetilde{l}(x; C), p_{\text{right}}(\bar{x})\} \leqslant p(x) \leqslant \bar{g}(x; R)$, where $\bar{g}(x; R)$ is a decreasing function of R and $p(x)$ is a non-increasing function. Then, considering the fact that $\bar{g}(x; R)$ is decreasing in R, to maximize R while ensuring $\max\{\widetilde{l}(x; C), p_{\text{right}}(\bar{x})\} \leqslant p(x) \leqslant \bar{g}(x; R)$, we set $p_{\text{left}}(x) = \max\{\widetilde{l}(x; C), p_{\text{right}}(\bar{x})\}$. See Figure 2 for $p_{\text{left}}(\cdot)$ in our running example.

Theorem 5 **(Optimal Solution to the Left Problem)**. *Algorithm 3 presents an optimal solution to Problem* (C-Pareto-left)*. That is, at the optimal solution to Problem* (C-Pareto-left)*, denoted by $p_{left}(\cdot)$, we set $p_{left}(x)$ based on Equation* (26)*. The optimal objective value of Problem* (C-Pareto-left) *is:*

$$R_{left} = \min\{\, CP_u(p_{left}(\bar{x}); (\bar{x}, m))), \inf_{x \in [0, \bar{x}]} CP_o(p_{left}(x); (x, 0))\}\,.$$

6. Optimal Consistent Ratio

In this section, using an optimization problem, we first characterize the optimal/maximum consistent ratio ($C^{\star}(\mathcal{R})$) that any algorithm can achieve under ML region $\mathcal{R}$. (See Section 6.1.) For any convex ML region $\mathcal{R}$, in Section 6.2, we then present a simple bisection method that allows us to obtain a good approximation for $C^{\star}(\mathcal{R})$. Section 6.3 presents a simpler algorithm to obtain the exact value of $C^{\star}(\mathcal{R})$ when the ML region $\mathcal{R}$ is a polyhedron.

6.1. Characterizing the Optimal Consistent Ratio

We begin by the following theorem:

THEOREM 6 **(Characterizing Optimal Consistent Ratio)**. *Consider any convex ML region $\mathcal{R}$. Then, the consistent ratio of any online algorithm under ML region $\mathcal{R}$ is at most $C^\star(\mathcal{R})$ where*

$$\begin{aligned} C^\star(\mathcal{R}) = \max_{C\geqslant 0, p(x): x\in[0,\bar{x}]} \quad & C \\ s.t. \quad & \widetilde{l}(x;C) \leqslant p(x) \leqslant u(x;C), \qquad x\in[\underline{x},\bar{x}] \\ & \textit{Validity Constraints } (11),\ (12) \quad x\in[0,\bar{x}]\,. \end{aligned} \tag{C-MAX}$$

Here, $\widetilde{l}(x;C)$ and $u(x;C)$ are defined in Equations (17) and (13), respectively. Furthermore, $C^\star(\mathcal{R}) \geqslant \rho$, where $\rho = 1/(2 - r_\ell/r_h)$.

Theorem 6 characterizes $C^\star(\mathcal{R})$ using an optimization problem (C-MAX) that bears resemblance with Problem (C-Pareto-Trans). In Problem (C-MAX), we aim to characterize a valid PL function under which the consistent ratio is maximized. Note that by the transformation Lemma 4, the first constraint (i.e., $\widetilde{l}(x;C) \leqslant p(x) \leqslant u(x;C)$) is equivalent to $\mathrm{CP}(p(x);(x,y)) \geqslant C$ for any $(x,y)\in\mathcal{R}$. The resulting PL function then leads an optimal PLA that obtains the consistent ratio of $C^\star(\mathcal{R})$, which is the maximum consistent ratio that any online algorithms can achieve.

6.2. A Bisection Method to Compute $C^\star(\mathcal{R})$ for a General Convex Region

Having characterized $C^\star(\mathcal{R})$, here we present a simple bisection method that allows us to compute an ϵ-accurate estimate of $C^\star(\mathcal{R})$. The bisection method (Algorithm 4) crucially uses the first set of constraints in Problem (C-MAX):

Necessary and Sufficient Conditions for $C \leqslant C^\star(\mathcal{R})$: By Lemma 4, we have for any $C \in [\rho, 1]$, $C^\star(\mathcal{R}) \geqslant C$ if and only if for any $x \in [\underline{x}, \bar{x}]$, we have

$$\widetilde{l}(x;C) \leqslant u(x;C)\,,$$

where we recall that $\widetilde{l}(x;C)$ and $u(x;C)$ are defined in Equations (17) and (13), respectively.

Algorithm 4 A Bisection Method to Compute $C^\star(\mathcal{R})$.

Input: Convex set $\mathcal{R}$, resource capacity m, and accuracy parameter $\epsilon \in [0, 1/2]$.

Output: An ϵ-accurate estimate of $C^\star(\mathcal{R})$.

Initialize $C_0 = \rho$ and $C_1 = 1$, where $\rho = 1/(2 - r_\ell/r_h)$.

While $|C_1 - C_0| \geqslant \epsilon$:

- Compute the mid point $C_m = (C_0 + C_1)/2$.
- If $\widetilde{l}(x;C_m) \leqslant u(x;C_m)$ for any $x \in [\underline{x}, \bar{x}]$, set C_0 to C_m.
- If $\widetilde{l}(x;C_m) > u(x;C_m)$ for some $x \in [\underline{x}, \bar{x}]$, set C_1 to C_m.

Return: C_0.

In Algorithm 4, we use a bisection procedure that repeatedly checks if $\widetilde{l}(x;C) \leqslant u(x;C)$, $x \in [\underline{x}, \bar{x}]$, for some given C. Note that this condition can be easily checked considering the fact that we have a closed form solution for $\widetilde{l}(x;C)$ and $u(x;C)$. Furthermore, checking this condition is equivalent to check if $\min_{x\in[\underline{x},\bar{x}]} u(x;C) - \widetilde{l}(x;C) \geqslant 0$, where we highlight this optimization can be easily solved. This is because by Lemma 6, we know that $u(x;C)$ is convex in x, and $\widetilde{l}(x;C)$ is concave in x. This implies that $u(x;C) - \widetilde{l}(x;C)$ is convex, and the aforementioned problem is a convex optimization problem. The following proposition sheds light on the performance of Algorithm 4.

Proposition 1 **(Bisection Method to Compute $C^\star(\mathcal{R})$).** *Consider Algorithm 4 with an accuracy parameter $\epsilon \in [0, 1/2]$. Given a general convex set $\mathcal{R}$, Algorithm 4 returns a $C_0 \in [C^\star(\mathcal{R}) - \epsilon, C^\star(\mathcal{R}) + \epsilon]$, where $C^\star(\mathcal{R})$ is the optimal solution to Problem* (C-max). *In addition, the computational complexity of Algorithm 4 is $O(\log(1/\epsilon))$.*

6.3. A Faster Method to Compute $C^\star(\mathcal{R})$ for Polyhedron Convex Regions

In the previous section, we present a bisection method to estimate $C^\star(\mathcal{R})$ for a general convex ML region. Here, we present a faster method to compute $C^\star(\mathcal{R})$ when the ML region is a convex polyhedron. This method relies on the following theorem.

Theorem 7 **(Properties of $C^\star(\mathcal{R})$ under Polyhedron ML Regions).** *Let $\mathcal{V}$ be the set containing the x value of all vertices of $\mathcal{R}$ and of the set $\mathcal{R}_0$, where $\mathcal{R}_0 = \{(x, \underline{h}(x) : x \in [\underline{x}, \bar{x}]\} \cap \{(x,y) : x + y = m\}$. Then, $C^\star(\mathcal{R}) = C$ for some $C \in [\rho, 1]$ if and only if the following two conditions holds.*

1. *for any $x \in \mathcal{V}$, $\widetilde{l}(x;C) \leqslant u(x;C)$.*
2. *there exists $\widehat{x} \in \mathcal{V}$, such that $\widetilde{l}(\widehat{x};C) = u(\widehat{x};C)$.*

Here, we recall that $\widetilde{l}(x;C)$ and $u(x;C)$ are defined in Equations (17) *and* (13)*, respectively.*

Theorem 7 shows that when the ML region $\mathcal{R}$ is a polyhedron, for any $x \in \mathcal{V}$, we have $\widetilde{l}(x;C^\star(\mathcal{R})) \leqslant u(x;C^\star(\mathcal{R}))$ while there exists $\widehat{x} \in \mathcal{V}$ under which the lower bound $\widetilde{l}(\widehat{x};C^\star(\mathcal{R}))$ is equal to the upper bound $u(\widehat{x};C^\star(\mathcal{R}))$. Here, $\mathcal{V}$ is the sets contain the x value of all vertices of $\mathcal{R}$, and the x value of $\mathcal{R}_0$. We refer to $\mathcal{V}$ as the set of x-vertices. This theorem shows that when the ML region is a polyhedron, we can simplify the feasibility check in Algorithm 4 by checking the condition $\widetilde{l}(x;C) \leqslant u(x;C)$ only for any x-vertices $x \in \mathcal{V}$. While this is an improvement, we present a faster algorithm that returns the exact value of $C^\star(\mathcal{R})$ by taking advantage of properties of the lower and upper bounds $\widetilde{l}(\cdot;C)$ and $u(\cdot;C)$, presented in Lemma 6.

Algorithm 5 An Algorithm to Compute $C^\star(\mathcal{R})$ for Polyhedron ML region $\mathcal{R}$

Input: Polyhedron ML region $\mathcal{R}$, resource capacity m.

Output: Optimal consistent ratio $C^\star(\mathcal{R})$.

Initialization: Set $\mathcal{S} = \varnothing$.

- For any pair of x-vertices $x_1, x_2 \in \mathcal{V}$ with $\bar{h}(x_1) \geqslant \underline{h}(x_2)$ and $x_2 \leqslant x_1$, find the following balancing PL $p \in [\underline{h}(x_2), \bar{h}(x_1))]$ such that

$$\mathrm{CP}_u(p; (x_1, \bar{h}(x_1))) = \mathrm{CP}_o(p; (x_2, \underline{h}(x_2)) . \tag{28}$$

(Note that x_1 can be equal to x_2.) Add $\mathrm{CP}_u(p; (x_1, \bar{h}(x_1)))$ to $\mathcal{S}$.

- For any pair of x-vertices $x_1, x_2 \in \mathcal{V}$ with $\bar{h}(x_1) - \underline{h}(x_2) \geqslant x_2 - x_1$ and $x_2 > x_1$, find the following balancing PL $p \in [\underline{h}(x_2) + (x_2 - x_1), \bar{h}(x_1)]$ such that

$$\mathrm{CP}_u(p; (x_1, \bar{h}(x_1))) = \mathrm{CP}_o(p - (x_2 - x_1); (x_2, \underline{h}(x_2))) . \tag{29}$$

Add $\mathrm{CP}_u(p; (x_1, \bar{h}(x_1)))$ to $\mathcal{S}$. That is, update $\mathcal{S}$ to $\mathcal{S} \cup \{\mathrm{CP}_u(p; (x_1, \bar{h}(x_1)))\}$.

Return: Return the largest $C \in \mathcal{S}$ under which $\widetilde{l}(x; C) \leqslant u(x; C)$ for any $x \in \mathcal{V}$ as $C^\star(\mathcal{R})$:

$$C^\star(\mathcal{R}) = \max\{C \in \mathcal{S} : \widetilde{l}(x; C) \leqslant u(x; C) \quad \text{for any } x \in \mathcal{V}\} .$$

In the faster algorithm, at a high level, we aim to find the x-vertex $\widehat{x}$ under which $\widetilde{l}(\widehat{x}; C^\star(\mathcal{R})) = u(\widehat{x}; C^\star(\mathcal{R}))$. To do so, we follow an enumeration technique that uses the property of $\widehat{x}$ along with the first condition in Theorem 7 that allows us to only focus on x-vertices to determine $C^\star(\mathcal{R})$. To explain the idea behind the algorithm, let us recall that in defining the upper bound $u(x; C)$, when possible, we choose the protection level $u(x; C)$ such that the compatible ratio at point $(x, \underline{h}(x))$ is equal to C. That is, for any $x \in [\underline{x}_u, \bar{x}_u]$, we have $u(x; C) = \sup\{p \in [0, m] : \mathrm{CP}_o(p; (x, \underline{h}(x))) = C\}$. Similarly, in defining the upper bound $l(x; C)$ (which we later use to define $\widetilde{l}$), when possible, we choose the protection level $l(x; C)$ such that the compatible ratio at point $(x, \bar{h}(x))$ is equal to C. That is, for any $x \in [x_H, \bar{x}_l]$, we have $l(x; C) = \inf\{p \in [0, m] : \mathrm{CP}_u(p; (x, \bar{h}(x))) = C\}$.

Now suppose that at $\widehat{x}$, we have $l(\widehat{x}; C^\star(\mathcal{R})) = \widetilde{l}(\widehat{x}; C^\star(\mathcal{R}))$. Then, if $\widehat{x} \in [\underline{x}_u, \bar{x}_u] \cap [x_H, \bar{x}_l]$, the condition $\widetilde{l}(\widehat{x}; C^\star(\mathcal{R})) = u(\widehat{x}; C^\star(\mathcal{R}))$ leads to *balancing* two compatible ratios. That is, we need to find a protection level $p \in [\underline{h}(\widehat{x}), \bar{h}(\widehat{x}))]$ such that

$$\mathrm{CP}_o(p; (\widehat{x}, \underline{h}(\widehat{x}))) = \mathrm{CP}_u(p; (\widehat{x}, \bar{h}(\widehat{x}))) .$$

This balancing idea explains Equation (28) in Algorithm 5. Now, suppose that at $\widehat{x}$, we have $l(\widehat{x}; C^\star(\mathcal{R})) \neq \widetilde{l}(\widehat{x}; C^\star(\mathcal{R}))$, which only happens when $\widehat{x} \geqslant x_{-1}$, where $x_{-1} = \sup\{x \in [x_H, \bar{x}] : \frac{\partial l(x^-; C^\star(\mathcal{R}))}{\partial x} \leqslant -1\}$. By definition of $\widetilde{l}$ in Equation (17), we then know that $\widetilde{l}(\widehat{x}; C^\star(\mathcal{R})) = l(x_{-1}; C^\star(\mathcal{R})) - (\widehat{x} - x_{-1})$. Then, the condition $\widetilde{l}(\widehat{x}; C^\star(\mathcal{R})) = u(\widehat{x}; C^\star(\mathcal{R}))$ leads to a slightly different balancing procedure in which we need to find a protection level $p \in [\underline{h}(\widehat{x}), \bar{h}(\widehat{x}))]$ (i.e., $l(\widehat{x}; C^\star(\mathcal{R}))$) such that

$$\mathrm{CP}_u(p; (x_{-1}, \bar{h}(x_{-1}))) = \mathrm{CP}_o(p - (\widehat{x} - x_{-1}); (\widehat{x}, \underline{h}(\widehat{x}))) \,.$$

This justifies Equation (29) in Algorithm 5. (Note that as we show in the proof of Theorem 8, for any C, x_{-1} is a x-vertex.)

Theorem 8 **(Optimal Consistent Ratio for Convex Polyhedron ML Regions)**. *Suppose that the ML region $\mathcal{R}$ is a convex polyhedron. Algorithm 5 returns the optimal consistent ratio $C^\star(\mathcal{R})$ for any given polyhedron $\mathcal{R}$ in run time $O(|\mathcal{V}|^3)$.*

7. Numerical Studies

In this section, we present the results of our numerical studies, which highlight the efficiency of resource allocation achieved through the integration of our algorithms with ML advice. Specifically, we investigate two distinct categories of uncertainty sets provided by ML advice: box advice and ellipsoid advice. We will show that thanks to our algorithm, ML advice can significantly improve the average and worst case performance, outperforming other benchmarks.

7.1. Setup

Demand/Arrival models. We conduct an analysis of two demand models, one with a uniform distribution, and the other with a normal distribution. In both models, we introduce random noise to the demand process using a uniform distribution. Specifically, denoting x and y as the number of high reward arrivals and low reward arrivals, respectively, in the first demand model, we have:

$$x, y \sim \begin{cases} \text{Uniform}(10, 20) & \text{with probability } 0.9, \\ \text{Uniform}(0, 30) & \text{with probability } 0.1. \end{cases} \tag{30}$$

Similarly, in the second model, we have:

$$x, y \sim \begin{cases} \text{Normal}(15, 3) & \text{with probability } 0.9, \\ \text{Uniform}(0, 30) & \text{with probability } 0.1\,. \end{cases} \tag{31}$$

We set $m = 20$, $r_h = 1$, and $r_\ell = 1/3$. We analyze both worst-case and uniform arriving orders in each demand model.

Construction of ML advice. For the uniform demand distribution in Equation (30), as shown in Figure 4, constructing a box advice is a natural choice. However, for the normal demand distribution in Equation (31), an ellipsoid advice is more appropriate, as shown in Figure 4. We use $n = \{10, 25\}$ as the number of samples used to construct the box and ellipsoid advice. To construct the box (ellipsoid) advice, we identify the smallest rectangle (ellipsoid) that encompasses at least $z\%$ of the sample points, where $z\% \in \{80\%, 90\%\}$. For the ellipsoid advice, we apply the method in Gärtner and Schönherr (1997) to construct such an ellipsoid. The left plot of Figure 4 illustrates an example of how we construct the box advice, while the right plot of Figure 4 displays an example of the ellipsoid advice.

To construct the ML advice, we sample $n \in \{10, 25\}$ data points $K = 1000$ times from the demand models in Equations (30) and (31). For each sample set, with size n, we then construct ML region in the shape of a box or an ellipsoid, depending on the demand model. Let S_k, $k \in [K]$, be the sample dataset and $\mathcal{R}_k$ be the ML advice associated with it.

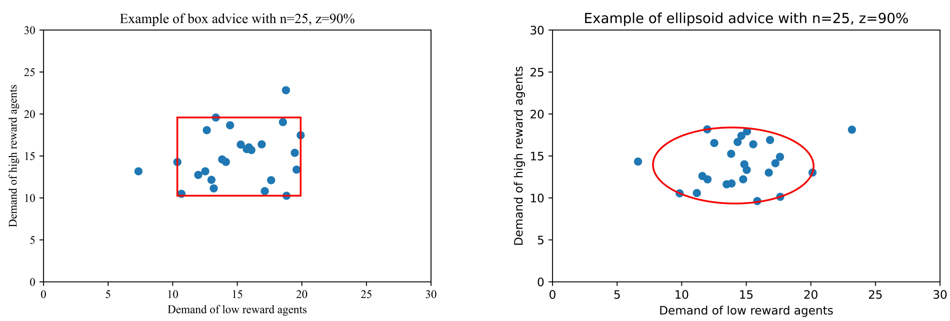


Figure 4 **Constructing the box and ellipsoid advice with $n = 25$ samples for the uniform and normal demand model in Equations (30) and (31) , respectively. Here, $z = 90\%$ fraction of the samples fall into the constructed box.**

Performance evaluation. To evaluate the performance of our algorithm and the benchmark algorithms that we will define shortly, we generate a test set which contains 100 instances. Each test instance corresponds to a point (x, y) drawn from the demand models described in Equation (30) (for box advice) and Equation (31) (for ellipsoid advice). We assess the algorithm's performance using these test instances under two different scenarios: worst-case (ordered) arrival sequences and stochastic (uniform order) arrival sequences.

Let $\mathcal{T}$ denote the set of test instances. Then, the worst CP and Avg. CP of an algorithm $\mathcal{A}$ are respectively defined as:

$$\text{WORST CP} = \frac{1}{K}\sum_{k=1}^{K}\min_{(x,y)\in\mathcal{T}}\text{CP}_{\mathcal{A}}(x,y;S_k) \quad \text{AVG. CP} = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\mathcal{T}|}\sum_{(x,y)\in\mathcal{T}}\text{CP}_{\mathcal{A}}(x,y;S_k)\,. \tag{32}$$

Here, $\text{CP}_{\mathcal{A}}(x,y;S_k)$ is the compatible ratio of algorithm $\mathcal{A}$ under the sample set S_k, which is used to construct the ML advice when algorithm $\mathcal{A}$ is, for example, Algorithm 1.

For the stochastic arrival sequences, for each instance $(x,y)\in\mathcal{T}$, we generate 100 random permutations. Similarly, we define:

$$\text{WORST CP} = \frac{1}{K}\sum_{k=1}^{K}\min_{(x,y)\in\mathcal{T}}\mathbb{E}[\text{CP}_{\mathcal{A}}(x,y;S_k)] \quad \text{AVG. CP} = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\mathcal{T}|}\sum_{(x,y)\in\mathcal{T}}\mathbb{E}[\text{CP}_{\mathcal{A}}(x,y;S_k)]\,,$$

where the expectation is taken with respect to the randomness in arrival permutations.

Benchmarks. To further assess the effectiveness of our algorithms, we propose several benchmarks for comparison with Algorithm 5.

1. *Point estimate benchmark.* Under this benchmark, we consider a sample set S_k with a size of $n\in\{10,25\}$. To estimate the central location, we use the point estimate $(\widehat{x}_k,\widehat{y}_k) = \frac{1}{n}(\sum_{i\in S_k}x_i,\sum_{i\in S_k}y_i)$. We then use this point estimate to construct a convex set $\mathcal{R}_k$, which serves as an input for Algorithm 1. Upon observation, we note that when $\widehat{y}_k\leqslant 8$, Algorithm 1 produces a fixed PL. In our specific case, the value 8 corresponds to the fixed PL proposed in Ball and Queyranne (2009). On the other hand, when $\widehat{y}_k > 8$, although the protection level function remains constant for $x\leqslant\hat{x}$, Algorithm 1 produces a decreasing PL for $x>\hat{x}_k$.
2. *The BQ benchmark (Ball and Queyranne 2009).* This benchmark algorithm is the PLA with a fixed PL function as proposed by Ball and Queyranne (2009). In this benchmark that we refer to as the *BQ*, the PL is set to $\frac{1-r_\ell/r_h}{2-r_\ell/r_h}m = 8$.
3. *PLA with an ML-augmented fixed PL function (Perakis and Roels 2010).* Under this benchmark, we again have a PLA with a fixed PL function $p(x)=p$. This benchmark that we refer to as *PR* is introduced by Perakis and Roels (2010) and is specifically designed for box ML advice, assuming that the ML advice is completely accurate. To determine the value of p in this benchmark, Perakis and Roels (2010) solve a mixed integer programming (MIP) problem to obtain the optimal consistent ratio. However, their algorithm does not account for inaccurate ML advice or accommodate ellipsoid advice. Thus, we will only consider this benchmark for the uniform demand model with box ML advice.

Table 1 **Results under box ML advice (demand model in Equation** (30)**) and adversarial order. The standard error of all the numbers is less than** 0.003**.**

	# of samples n	n=10			n=25		
	Consistent ratio C	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$
Alg. 1 with input C ($z = 90\%$)	Avg. CP	**0.902**	0.880	0.824	0.912	0.891	0.834
	Worst CP	0.631	**0.686**	0.644	0.646	0.703	0.664
Alg. 1 with input C ($z = 80\%$)	Avg. CP	0.899	0.877	0.823	**0.914**	0.899	0.845
	Worst CP	0.632	0.685	0.649	0.629	**0.705**	0.673
Point ML Advice	Avg. CP	0.901	0.823	0.745	0.905	0.819	0.745
	Worst CP	0.454	0.520	0.604	0.459	0.523	0.606
BQ Benchmark	Avg. CP	0.762	-	-	0.762	-	-
	Worst CP	0.600	-	-	0.600	-	-
PR Benchmark ($z = 90\%$)	Avg. CP	0.900	-	-	0.907	-	-
	Worst CP	0.626	-	-	0.641	-	-
PR Benchmark ($z = 80\%$)	Avg. CP	0.896	-	-	0.909	-	-
	Worst CP	0.627	-	-	0.628	-	-

7.2. Results

Results under box advice. Tables 1 and 2 present the average CP (Avg. CP) and worst CP (Worst CP) metrics, as defined in Equation (32), with the box ML advice associated with demand model in Equation (30) under adversarial and stochastic arrival sequences, respectively. In these tables, we show the performance of Algorithm 1 with different settings (specifically, $z \in \{80\%, 90\%\}$ and $C \in \{C^\star(\mathcal{R}), 0.9 \cdot C^\star(\mathcal{R}), 0.8 \cdot C^\star(\mathcal{R})\}$), along with our benchmarks: point ML advice (with $C \in \{C^\star(\mathcal{R}), 0.9 \cdot C^\star(\mathcal{R}), 0.8 \cdot C^\star(\mathcal{R})\}$), BQ, and PR (with $z \in \{80\%, 90\%\}$). Here, we recall that the parameter z is used to determine the ML advice for both Algorithm 1 and the PR benchmark.

As a general observation, all algorithms demonstrate a better performance under stochastic arrivals, compared with adversarial arrivals. For both adversarial and stochastic arrivals, we observe that Algorithm 1, with a fixed number of samples n used in constructing the ML advice, effectively achieves a balance between the average and worst case CP. To illustrate this, let's consider some examples.

Under adversarial setting, when $n = 10$ and $z = 90\%$, Algorithm 1 with $C = 0.9 \cdot C^\star(\mathcal{R})$ surpasses the PR benchmark in terms of worst case CP by 10%, with values of 0.686 and 0.626 respectively. However, there is a slight sacrifice in the average CP, with Algorithm 1 achieving 0.880 compared to the PR benchmark's 0.900.

Another scenario to consider is when $n = 25$ and $z = 80\%$ and we have stochastic arrival model. In this case, Algorithm 1 with $C = 0.9 \cdot C^\star(\mathcal{R})$ outperforms the point ML advice benchmark significantly. It surpasses the point ML advice benchmark in terms of worst case CP by a remarkable 24% (0.736 versus 0.592), while also excelling in terms of average CP by 9% (0.934 versus 0.857). This demonstrates the limitations of the point ML advice approach in effectively extracting and

Table 2 Results under box ML advice (demand model in Equation (30)) and stochastic order. The standard error of all the numbers is less than 0.003.

	# of samples n	n=10			n=25		
	Consistent ratio C	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$
Alg. 1 with input C	Avg. CP	**0.943**	0.929	0.901	**0.951**	0.937	0.904
($z = 90\%$)	Worst CP	0.676	0.733	**0.741**	0.677	0.741	**0.744**
Alg. 1 with input C	Avg. CP	0.936	0.922	0.893	0.944	0.934	0.899
($z = 80\%$)	Worst CP	0.667	0.726	0.732	0.678	0.736	0.734
Point ML	Avg. CP	0.910	0.853	0.814	0.914	0.857	0.817
Advice	Worst CP	0.555	0.584	0.673	0.554	0.592	0.674
BQ	Avg. CP	0.823	-	-	0.823	-	-
Benchmark	Worst CP	0.674	-	-	0.674	-	-
PR Benchmark	Avg. CP	0.933	-	-	0.941	-	-
($z = 90\%$)	Worst CP	0.666	-	-	0.688	-	-
PR Benchmark	Avg. CP	0.926	-	-	0.935	-	-
($z = 80\%$)	Worst CP	0.658	-	-	0.668	-	-

exploiting valuable information in the past (training) dataset. The result is slightly worse for the BQ benchmark as it does not consider any ML advice and aims to hedge against the worst case scenario.

Table 3 Results under ellipsoid ML advice (demand model in Equation (31)) and adversarial order. The standard error of all the numbers is less than 0.003.

	# of samples n	n=10			n=25		
	Input consistent ratio C	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$
Alg. 1 with input C	Avg. CP	**0.921**	0.894	0.830	**0.933**	0.914	0.850
($z = 90\%$)	Worst CP	0.646	**0.661**	0.613	0.660	**0.705**	0.647
Alg. 1 with input C	Avg. CP	0.914	0.877	0.810	0.923	0.889	0.832
($z = 80\%$)	Worst CP	0.633	0.653	0.605	0.650	0.677	0.639
Point ML	Avg. CP	0.906	0.822	0.733	0.912	0.823	0.734
Advice	Worst CP	0.539	0.566	0.599	0.523	0.575	0.600
BQ	Avg. CP	0.725	-	-	0.725	-	-
Benchmark	Worst CP	0.600	-	-	0.600	-	-

Table 4 Results under ellipsoid ML advice (demand model in Equation (31)) and stochastic order. The standard error of all the numbers is less than 0.005.

	# of samples n	n=10			n=25		
	Consistent ratio C	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$	$C^\star(\mathcal{R})$	$0.9 \cdot C^\star(\mathcal{R})$	$0.8 \cdot C^\star(\mathcal{R})$
Alg. 1 with input C	Avg. CP	**0.947**	0.922	0.891	**0.952**	0.935	0.909
($z = 90\%$)	Worst CP	0.679	0.698	**0.703**	0.689	0.713	**0.722**
Alg. 1 with input C	Avg. CP	0.934	0.918	0.880	0.949	0.929	0.892
($z = 80\%$)	Worst CP	0.664	0.683	0.692	0.669	0.691	0.701
Point ML	Avg. CP	0.909	0.852	0.819	0.911	0.858	0.818
Advice	Worst CP	0.551	0.583	0.639	0.535	0.592	0.649
BQ	Avg. CP	0.834	-	-	0.834	-	-
Benchmark	Worst CP	0.670	-	-	0.670	-	-

Results under ellipsoid advice. Tables 3 and 4 present the results obtained using ellipsoid advice, specifically with the demand model described by Equation (31), under both adversarial and stochastic arrival models. Note that the PR benchmark is not included in these tables, as it is specifically designed for box advice.

Once again, we observe that Algorithm 1 achieves a balance between the average and worst case CP, outperforming the existing benchmarks for both adversarial and stochastic arrivals. For instance, when considering both values of n and selecting $C = 0.9 \cdot C^\star(\mathcal{R})$, Algorithm 1 achieves the best worst CP under adversarial arrivals while there is a small sacrifice of approximately 2-3% in the worst CP when compared to the maximum average CP obtained under the same algorithm with input $C = C^\star(\mathcal{R})$.

8. Concluding Remarks and Future Directions

In this work, we have proposed a novel online resource allocation model that addresses the challenge of integrating machine learned predictions into resource allocation decisions in a robust and efficient manner. Efficient online resource allocation is essential for entities such as hospitals, governments, and various industries that often face a trade-off between meeting low-reward and high-reward demands without precise knowledge of future demand. However, factors such as environmental fluctuations, data biases, and insufficient data points can impede consistent accuracy, making it challenging to allocate resources efficiently.

The proposed model is based on the concept of convex uncertainty sets, which use historical data to construct sets of plausible demand scenarios, allowing for flexibility and robustness in decision-making. We examine the benefits of utilizing ML advice in online resource allocation problems by proposing C-Pareto PLAs that balance the robust and the consistent ratios. Compared to traditional fixed protection level algorithms, we find that adaptive PLAs often manage to obtain high consistent and robust ratios, highlighting the significance and advantages of adjusting the protection level.

This work highlights the substantial advantages of employing uncertainty set ML advice, as opposed to point estimate advice, in sequential decision-making under uncertainty. It also paves the way for further exploration in other sequential decision-making problems. One such problem is single-leg revenue management with $K \geqslant 3$ types.

When dealing with more than two types of requests, relying on a single protection level function may not suffice for making decisions. One approach to address this challenge is to consider an adaptive nested PLA that incorporates $n-1$ protection level functions $p_i(\cdot)_{i\in[K-1]}$. In this case, given rewards $r_1 \geqslant r_2 \geqslant \ldots \geqslant r_K$, the protection level $p_i(s_j)$ is applied to type i once we observe s_j type j requests for all $i+1 \leqslant j \leqslant K$. As a potential future research direction, it would be valuable to explore the effectiveness of the nested PLA in this context. However, further investigation is required to explore this direction, which we leave for future research.

References

Shipra Agrawal, Zizhuo Wang, and Yinyu Ye. 2014. A dynamic near-optimal algorithm for online linear programming. *Operations Research* 62, 4 (2014), 876–890.

Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. 2020. Secretary and online matching problems with machine learned advice. *arXiv preprint arXiv:2006.01026* (2020).

Michael O Ball and Maurice Queyranne. 2009. Toward robust revenue management: Competitive analysis of online booking. *Operations Research* 57, 4 (2009), 950–963.

Santiago Balseiro, Christian Kroer, and Rachitesh Kumar. 2022. Single-Leg Revenue Management with Advice. *arXiv preprint arXiv:2202.10939* (2022).

Dimitris Bertsimas and David B Brown. 2009. Constructing uncertainty sets for robust linear optimization. *Operations research* 57, 6 (2009), 1483–1495.

Dimitris Bertsimas, Iain Dunning, and Miles Lubin. 2016. Reformulation versus cutting-planes for robust optimization. *Computational Management Science* 13, 2 (2016), 195–217.

Dimitris Bertsimas, Vishal Gupta, and Nathan Kallus. 2018. Data-driven robust optimization. *Mathematical Programming* 167, 2 (2018), 235–292.

Shelby L Brumelle and Jeffrey I McGill. 1993. Airline seat allocation with multiple nested fare classes. *Operations research* 41, 1 (1993), 127–137.

Niv Buchbinder, Kamal Jain, and Joseph Seffi Naor. 2007. Online primal-dual algorithms for maximizing ad-auctions revenue. In *European Symposium on Algorithms*. Springer, 253–264.

Meysam Cheramin, Richard Li-Yang Chen, Jianqiang Cheng, and Ali Pinar. 2021. Data-Driven Robust Optimization Using Scenario-Induced Uncertainty Sets. *arXiv preprint arXiv:2107.04977* (2021).

Nikhil R Devanur and Thomas P Hayes. 2009. The adwords problem: online keyword matching with budgeted bidders under random permutations. In *Proceedings of the 10th ACM conference on Electronic commerce*. 71–78.

Hossein Esfandiari, Nitish Korula, and Vahab Mirrokni. 2015. Online allocation with traffic spikes: Mixing adversarial and stochastic models. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*. 169–186.

Jon Feldman, Monika Henzinger, Nitish Korula, Vahab S Mirrokni, and Cliff Stein. 2010. Online stochastic packing applied to display ad allocation. In *European Symposium on Algorithms*. Springer, 182–194.

Bernd Gärtner and Sven Schönherr. 1997. Smallest enclosing ellipses: fast and exact. (1997).

Negin Golrezaei, Patrick Jaillet, and Zijie Zhou. 2022. Online Resource Allocation with Samples. *Available at SSRN* (2022).

Negin Golrezaei, Hamid Nazerzadeh, and Paat Rusmevichientong. 2014. Real-time optimization of personalized assortments. *Management Science* 60, 6 (2014), 1532–1551.

Negin Golrezaei and Evan Yao. 2021. Upfront Commitment in Online Resource Allocation with Patient Customers. *arXiv preprint arXiv:2108.03517* (2021).

Dawsen Hwang, Patrick Jaillet, and Vahideh Manshadi. 2021. Online resource allocation under partially predictable demand. *Operations Research* 69, 3 (2021), 895–915.

Amir Jalilvand-Nejad, Rasoul Shafaei, and Hamid Shahriari. 2016. Robust optimization under correlated polyhedral uncertainty set. *Computers & Industrial Engineering* 92 (2016), 82–94.

Stefanus Jasin. 2015. Performance of an lp-based control for revenue management with unknown demand parameters. *Operations Research* 63, 4 (2015), 909–915.

Billy Jin and Will Ma. 2022. Online Bipartite Matching with Advice: Tight Robustness-Consistency Tradeoffs for the Two-Stage Model. *arXiv preprint arXiv:2206.11397* (2022).

Silvio Lattanzi, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. 2020. Online scheduling via learned weights. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 1859–1877.

Ken Littlewood. 2005. Special issue papers: Forecasting and control of passenger bookings. *Journal of Revenue and Pricing Management* 4, 2 (2005), 111–123.

Thodoris Lykouris and Sergei Vassilvtiskii. 2018. Competitive caching with machine learned advice. In *International Conference on Machine Learning*. PMLR, 3296–3305.

Will Ma, David Simchi-Levi, and Chung-Piaw Teo. 2021. On policies for single-leg revenue management with limited demand information. *Operations Research* 69, 1 (2021), 207–226.

Mohammad Mahdian, Hamid Nazerzadeh, and Amin Saberi. 2012. Online optimization with uncertain information. *ACM Transactions on Algorithms (TALG)* 8, 1 (2012), 1–29.

Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. 2007. Adwords and generalized online matching. *Journal of the ACM (JACM)* 54, 5 (2007), 22–es.

Georgia Perakis and Guillaume Roels. 2010. Robust controls for network revenue management. *Manufacturing & Service Operations Management* 12, 1 (2010), 56–76.

Dhruv Rohatgi. 2020. Near-optimal bounds for online caching with machine learned advice. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 1834–1845.

Appendix A: More on Validity Conditions of Protection Level Algorithms

Here, we justify the necessity of the validity conditions of PLAs in Definition 1.

The PL function belongs to PLAs should be non-increasing with $p'(x) \geqslant -1$. We refer to these conditions as *validity conditions* and we say a PLA is valid if its PL function is non-increasing with $p'(x) \geqslant -1$. First, it is clear that the PL function cannot be increasing. To see that, suppose when deciding about a low-reward request with size s, we can allocate $m - p(\bar{s}+s) - \bar{a}$ resources to the request, and protect $p(\bar{s}+s)$ units for high-reward requests. Now, suppose that we have already allocated $p(\bar{s}+s)$ units to high-reward requests and hence we fully utilize the m units of resources. Then, if the PL function $p(\cdot)$ increases at $\bar{s}+s$, upon the arrival of the next low-reward request, we would like to protect more than $p(\bar{s}+s)$ units for high-reward requests which is not possible as all resources are allocated.

Second, the derivative of the PL function $p(\cdot)$ should be always greater than equal to -1. This condition ensures that PLAs can be implemented. Recall that under a PLA, upon the arrival of a low-reward request with size s, we set $a = \min\left\{\widehat{m}, \mathrm{Proj}_{[0,s]}\left(m - p(\bar{s}+s) - \bar{a}\right)\right\}$. Suppose that we can set $a = m - p(\bar{s}+s) - \bar{a}$ and contrary to our assumption, suppose that $p'(\bar{s}+s) < -1$. Then, assume that we receive a low-reward request with size $s_1 > 0$, where s_1 is an arbitrary small number. Suppose that we can set $a_1 = m - p(\bar{s}+s+s_1) - \bar{a_1} \leqslant s_1$, where $\bar{a_1}$, which is the total accepted low-reward request, is equal to $\bar{a}+a$. Then, we have

$$a_1 - a = p(\bar{s}+s) - p(\bar{s}+s+s_1) - a > s_1 - a,$$

where the inequality is because $p'(\bar{s}+s) < -1$ and s_1 is an arbitrary small positive number. This equation implies that $a_1 > s_1$, which cannot happen because we can accept at most s_1 low-reward requests. Therefore, the derivative of the PL function p cannot be less than -1.

Appendix B: Proof of Lemma 1

Let (x, y) be fixed, and let I be the ordered sequence of arrivals such that x low-reward requests arrive first, followed by y high-reward requests. For any adaptive protection level algorithm $\mathcal{A}$, let $\tilde{x}(\mathcal{A})$ be the total number of low-reward requests that are accepted under the ordered sequence I. Then, the algorithm accepts $\min\{y, m - \tilde{x}(\mathcal{A})\}$ high-reward requests. Observe that algorithm $\mathcal{A}$ rejects the x-th low-reward request if and only if

- condition (1) the number of low-reward requests accepted so far is not less than $m - p(x)$, where $p(\cdot)$ is the protection level function, or
- condition (2) all the resources have been used.

We split the proof into two cases.

Case 1: $y < m - \tilde{x}(\mathcal{A})$. In this case, we claim that under any other ordering of the arrivals, the number of high-reward requests that are accepted cannot be smaller than y. We also claim that under any other ordering of the arrivals, the number of low-reward requests that are accepted cannot be smaller than $\tilde{x}(\mathcal{A})$. Showing these claims, complete the proof of this case.

We begin with the first part. First note that $\tilde{x}(\mathcal{A})$ is an upper bound on the total number of low-reward requests that algorithm $\mathcal{A}$ accepts under any ordering of the arrivals. This is because condition (2) never fails as $y \leqslant m - \tilde{x}(\mathcal{A})$, and hence with even accepting y high-reward requests, there are resources left for $\tilde{x}(\mathcal{A})$ low-reward requests, and condition (1) is independent of the order of high-reward requests. Moreover, since $y \leqslant m - \tilde{x}(\mathcal{A})$, any arriving high-reward request is accepted. Thus, for any ordering of the arrivals, algorithm $\mathcal{A}$ can accept y high-reward requests.

We now show the second part. Here, we want to show that under any other ordering of the arrivals, the number of low-reward requests that are accepted is greater than or equal to $\tilde{x}(\mathcal{A})$. Contrary to our claim, suppose that the total number of low-reward requests that algorithm $\mathcal{A}$ accepts under any unordered sequence is strictly less than $\tilde{x}(\mathcal{A})$. This means that at some point, condition (2) is not satisfied. Therefore, there exists a time t such that $x(t) + y(t) = m$, where $x(t)$ and $y(t)$ denote the number of low- and high-reward requests accepted up to time t, respectively. However, we know that $x(t) < \tilde{x}(\mathcal{A})$, $y(t) \leqslant y$, and $\tilde{x}(\mathcal{A}) + y \leqslant m$. Therefore, $x(t) + y(t) = m$ cannot hold, which contradicts the assumption that algorithm $\mathcal{A}$ has accepted fewer than $\tilde{x}(\mathcal{A})$ low-reward requests. This completes the proof of the first case.

Case 2: $y \geqslant m - \tilde{x}(\mathcal{A})$. In this case, we claim that under any other ordering of the requests, the number of high-reward requests accepted cannot be smaller than $m - \tilde{x}(\mathcal{A})$. Additionally, we claim that if the number of low-reward requests accepted under any other ordering is less than $\tilde{x}(\mathcal{A})$, then the total reward generated by the algorithm is larger than the case where the number of low-reward requests accepted is $\tilde{x}(\mathcal{A})$.

We begin with the first part. We will show that under any other ordering of the requests, the number of high-reward requests accepted cannot be smaller than $m - \tilde{x}(\mathcal{A})$. As we know, high-reward requests are rejected only when all resources are used up. Therefore, the later a high-reward request arrives, the less chance we accept the request. Compare any unordered arrival sequence with the ordered arrival sequence, each high-reward request arrives earlier, which implies that the number of high-reward requests accepted cannot be smaller than $m - \tilde{x}(\mathcal{A})$.

Next, we show the second part, which is if the number of low-reward requests accepted under any other ordering is less than $\tilde{x}(\mathcal{A})$, then the total reward generated by the algorithm is larger than the case where the number of low-reward requests accepted is $\tilde{x}(\mathcal{A})$. If the total number of low-reward requests the algorithm $\mathcal{A}$ accepts is strictly less than $\tilde{x}(\mathcal{A})$, then at some point, condition (2) is not satisfied. This means that for some time t, $x(t) + y(t) = m$, where $x(t)$ and $y(t)$ denote the number of low-reward and high-reward requests accepted before time t, respectively. Since $x(t) < \tilde{x}(\mathcal{A})$, we have $y(t) > m - \tilde{x}(\mathcal{A})$. Therefore, the reward generated by the algorithm is

$$y(t)r_h + x(t)r_\ell = y(t)r_h + (m - y(t))r_\ell > (m - \tilde{x}(\mathcal{A}))r_h + \tilde{x}(\mathcal{A})r_\ell,$$

which implies that the algorithm generates a larger reward.

Therefore, we conclude that the adversary should choose the first instance where all x low reward requests arrive first and follow with all y high reward requests.

Appendix C: Proof of Statements in Section 4

Throughout the proofs, we make use of some preliminary lemmas that are presented in Section F.

C.1. Proof of Lemma 2

We first show that for any $p \leqslant \min\{m, y_1\}$, we have $\mathrm{CP}_u(p;(x,y_1)) \geqslant \mathrm{CP}_u(p;(x,y_2))$, where we recall that $y_1 \leqslant y_2$. Observe that when $p \leqslant y_1$ and $x + y_1 < m$, by Lemma 8, we have $\mathrm{CP}_u(p;(x,y_1)) = 1$, and hence $1 = \mathrm{CP}_u(p;(x,y_1)) \geqslant \mathrm{CP}_u(p;(x,y_2))$ trivially holds. Now suppose that $x + y_1 \geqslant m$. By definition, we have

$$\begin{aligned}
\mathrm{CP}_u(p;(x,y_1)) &= \frac{\max\{p,(m-x)^+\}r_h + \min\{x, m-p\}r_\ell}{\min\{y_1, m\}r_h + \min\{x,(m-y_1)^+\}r_\ell} \\
&\geqslant \frac{\max\{p,(m-x)^+\}r_h + \min\{x, m-p\}r_\ell}{\min\{y_2, m\}r_h + \min\{x,(m-y_2)^+\}r_\ell} = \mathrm{CP}_u(p;(x,y_2))\,,
\end{aligned}$$

where the inequality holds because $y_2 \geqslant y_1$ and $y \mapsto yr_h + (m-y)r_\ell$ is increasing in y as $r_h > r_\ell$.

Second, we show that any protection level p with $p \geqslant \min\{m, y_2\}$, we have $\mathrm{CP}_o(p;(x,y_2)) \geqslant \mathrm{CP}_o(p;(x,y_1))$. To show this, first consider the case where $x + y_1 < m$ and $x + y_2 < m$. Then, by the definition of CP_o in Equation (6), we have

$$\mathrm{CP}_o(p;(x,y_2)) = \frac{\min\{y_2, m\}r_h + \min\{x, m-p\}r_\ell}{\min\{y_2, m\}r_h + \min\{x,(m-y_2)^+\}r_\ell} = \frac{y_2 r_h + \min\{x, m-p\}r_\ell}{y_2 r_h + x r_\ell} \tag{33}$$

$$\geqslant \frac{y_1 r_h + \min\{x, m-p\}r_\ell}{y_1 r_h + x r_\ell} = \mathrm{CP}_o(p;(x,y_1))\,, \tag{34}$$

where the inequality holds because $y \mapsto \frac{yr_h + \min\{x,m-p\}r_\ell}{yr_h + xr_\ell}$ is increasing in y.

Now consider the case where $x + y_2 \geqslant m$. Next, we show that by fixing an x, for any y such that $x + y > m$, $\mathrm{CP}_o(p;(x,y))$ is increasing in y. Observe that for any y with $x + y \geqslant m$ and $p \geqslant y$, we have

$$\mathrm{CP}_o(p;(x,y)) = \frac{\min\{y, m\}r_h + \min\{x, m-p\}r_\ell}{\min\{y, m\}r_h + \min\{x,(m-y)^+\}r_\ell},$$

If $y \geqslant m$, the statement is trivial. Otherwise,

$$\begin{aligned}
\frac{\partial \mathrm{CP}_o(p;(x,y))}{\partial y} &= \frac{r_h(yr_h + (m-y)r_\ell) - (r_h - r_\ell)(yr_h + \min\{m-p, x\}r_\ell)}{(yr_h + (m-y)r_\ell)^2} \\
&\geqslant \frac{r_h(yr_h + (m-y)r_\ell) - (r_h - r_\ell)(yr_h + (m-p)r_\ell)}{(yr_h + (m-y)r_\ell)^2} \\
&\geqslant \frac{(r_h - r_\ell)(yr_h + (m-y)r_\ell) - (r_h - r_\ell)(yr_h + (m-p)r_\ell)}{(yr_h + (m-y)r_\ell)^2} \\
&= \frac{(r_h - r_\ell)(yr_h + (p-y)r_\ell)}{(yr_h + (m-y)r_\ell)^2} \geqslant 0,
\end{aligned}$$

where the last inequality is because $p \geqslant y$. The chain of inequalities shows that $\mathrm{CP}_o(p;(x,y))$ is increasing in y when $x + y \geqslant m$ and $p \geqslant y$. This implies that we have $\mathrm{CP}_o(p;(x,y_2)) \geqslant \mathrm{CP}_o(p;(x,y_1))$ when $x + y_i \geqslant m$, $i \in \{1, 2\}$, as desired.

For the case where $x + y_2 \geqslant m$ and $x + y_1 < m$, we have

$$\mathrm{CP}_o(p;(x,y_2)) \geqslant \mathrm{CP}_o(p;(x,m-x)) \geqslant \mathrm{CP}_o(p;(x,y_1))\,,$$

where the first inequality holds because $\frac{\partial \mathrm{CP}_o(p;(x,y))}{\partial y} \geqslant 0$ when $x + y \geqslant m$, and the second inequality holds because of Equation (33).

Finally, we show that any $x \in [\underline{x}, \bar{x}]$ and $p \geqslant 0$, we have $\min_{y \in [\underline{h}(x), \bar{h}(x)]}\{\mathrm{CP}(p;(x,y))\} = \min\left\{\mathrm{CP}(p;(x,\underline{h}(x))), \mathrm{CP}(p;(x,\bar{h}(x)))\right\}$. Suppose that $p \leqslant \underline{h}(x)$. Then, $\mathrm{CP}(p;(x,y)) = \mathrm{CP}_u(p;(x,y))$ for any $y \in [\underline{h}(x), \bar{h}(x)]$, and hence by the first result of this lemma, we have

$$\min_{y \in [\underline{h}(x), \bar{h}(x)]}\{\mathrm{CP}(p;(x,y))\} = \mathrm{CP}_u(p;(x,\bar{h}(x)))\,,$$

as desired. Now, suppose that $p \geqslant \bar{h}(x)$. Then, $\mathrm{CP}(p;(x,y)) = \mathrm{CP}_o(p;(x,y))$ for any $y \in [\underline{h}(x), \bar{h}(x)]$, and hence by the second result of this lemma, we have

$$\min_{y\in[\underline{h}(x),\bar{h}(x)]}\{\mathrm{CP}(p;(x,y))\} = \mathrm{CP}_o(p;(x,\underline{h}(x)))\,,$$

as desired. Now, suppose the final case where $p \in (\underline{h}(x), \bar{h}(x))$. Then,

$$\begin{aligned}\min_{y\in[\underline{h}(x),\bar{h}(x)]}\{\mathrm{CP}(p;(x,y))\} &= \min\left\{\min_{y\in[\underline{h}(x),p]}\{\mathrm{CP}_o(p;(x,y))\},\ \min_{y\in[p,\bar{h}(x)]}\{\mathrm{CP}_u(p;(x,y))\}\right\}\\ &= \min\left\{\mathrm{CP}_o(p;(x,\underline{h}(x))), \mathrm{CP}_u(p;(x,\bar{h}(x)))\right\}\,,\end{aligned}$$

where the last inequality follows from the first and second results of this lemma.

□

C.2. Properties of functions $u(\cdot;C)$ and $l(\cdot;C)$: Lemma 6 and its Proof

Lemma 6 **(Properties of functions $u(\cdot;C)$ and $l(\cdot;C)$).** *The functions $u(\cdot;C)$ and $l(\cdot;C)$, which are respectively defined in Equations (13) and (15), have the following properties.*

1. *For any $x \in (\underline{x}_u, \bar{x}_u)$ and $C \in [0,1]$, let $\underline{\mathcal{H}}(x) = \min\{\underline{h}(x), m\}$. When $\underline{\mathcal{H}}'(x)$ exists, we have*
$$\frac{\partial u(x;C)}{\partial x} = \begin{cases} ((1-C)\frac{r_h}{r_\ell} + C)\underline{\mathcal{H}}'(x) & \text{if } x + \underline{\mathcal{H}}(x) \geqslant m;\\ (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x) - C & \text{if } x + \underline{\mathcal{H}}(x) < m.\end{cases} \tag{35}$$

2. *For any $x \in (x_H, \bar{x}_l)$ and $C \in [0,1]$, let $\overline{\mathcal{H}}(x) = \min\{\bar{h}(x), m\}$. When $\overline{\mathcal{H}}'(x)$ exists, we have*
$$\frac{\partial l(x;C)}{\partial x} = C\overline{\mathcal{H}}'(x). \tag{36}$$

3. *For any $C \in [0,1]$, $u(x;C)$ is non-increasing in $x \in [0,\bar{x}]$ and is convex for $x \in [\underline{x}_u, \bar{x}]$.*

4. *For any $C \in [0,1]$, $l(x;C)$ is non-increasing in $x \in [0,\bar{x}]$ and is concave for $x \in [\underline{x}, \bar{x}_\ell]$.*

5. *For any $C \leqslant C^\star(\mathcal{R})$ and any $x \in [0,\bar{x}]$, we have $l(x;C) \leqslant u(x;C)$.*

6. *For any $x \in [0,\bar{x}]$, $l(x;C)$ is continuously increasing in C and $u(x;C)$ is continuously decreasing in C.*

Proof of Lemma 6 Here, we will show the following six properties.

C.2.1. Property 1 We first show Equation (50). We split the analysis into two cases: *Case 1: $x + \underline{\mathcal{H}}(x) \geqslant m$* and *Case 2: $x + \underline{\mathcal{H}}(x) < m$.*

Case 1: ($x + \underline{\mathcal{H}}(x) \geqslant m$). By Lemma 12, we have for any $x \in (\underline{x}_u, \bar{x}_u)$, $\underline{\mathcal{H}}(x) = \underline{h}(x)$, which implies that $\underline{h}(x) \leqslant m$. Then, we take an arbitrary point $(x_1, \underline{h}(x_1))$ with $x_1 \in (\underline{x_u}, \bar{x}_u)$. By definition of $u(\cdot;C)$, we should have $\mathrm{CP}_o(u(x_1;C);(x_1,\underline{h}(x_1))) = C$, and by Lemma 14, we have such $u(x_1;C)$ always exists and $u(x_1;C) \geqslant \underline{h}(x_1)$. By Equation (6),

$$\mathrm{CP}_o(u(x_1;C);(x_1,\underline{h}(x_1))) = \frac{\underline{h}(x_1)r_h + \min\{x_1, m - u(x_1;C)\}r_\ell}{\underline{h}(x_1)r_h + \min\{x_1, m - \underline{h}(x_1)\}r_\ell} = C. \tag{37}$$

As in this case, if $x_1 + \underline{h}(x_1) \geqslant m$, we have $\min\{x_1, m - \underline{h}(x_1)\} = m - \underline{h}(x_1)$. We then argue that $\min\{x_1, m - u(x_1;C)\} = m - u(x_1;C)$. Suppose that contrary to our claim, $\min\{x_1, m - u(x_1;C)\} = x_1$. We then have

$$\mathrm{CP}_o(u(x_1;C);(x_1,\underline{h}(x_1))) = \frac{\underline{h}(x_1)r_h + x_1 r_\ell}{\underline{h}(x_1)r_h + (m - \underline{h}(x_1))r_\ell} = \mathrm{CP}_o(m;(x_1,\underline{h}(x_1))),$$

which implies that $x_1 \leqslant \underline{x}_u$. However, we define $x_1 \in (\underline{x}_u, \bar{x}_u)$, therefore, this is a contradiction, and we can only have $\min\{x_1, m-u(x_1;C)\} = m-u(x_1;C)$. Then, by Equation (37), we have

$$\frac{\underline{h}(x_1)r_h + (m-u(x_1;C))r_\ell}{\underline{h}(x_1)r_h + (m-\underline{h}(x_1))r_\ell} = C,$$

which is equivalent to

$$u(x_1;C) = ((1-C)\frac{r_h}{r_\ell} + C)\underline{h}(x_1) + m(1-C).$$

This implies that $\frac{\partial u(x;C)}{\partial x} = ((1-C)\frac{r_h}{r_\ell} + C)\underline{h}'(x)$ when $x + \underline{h}(x) \geqslant m$. As we have $\underline{\mathcal{H}}(x) = \underline{h}(x)$ for $x \in [\underline{x_u}, \bar{x}_u]$, this implies the desired result.

Case 2 ($x + \underline{\mathcal{H}}(x) < m$). In this case, as $x + \underline{\mathcal{H}}(x) < m$, we have $\underline{\mathcal{H}}(x) < m$, which implies that $\underline{\mathcal{H}}(x) = \underline{h}(x)$. Let us take a point $(x_1, \underline{h}(x_1))$ with $x_1 \in (\underline{x_u}, \bar{x}_u)$. By definition of $u(\cdot;C)$, we have $\mathrm{CP}_o(u(x_1;C);(x_1,\underline{h}(x_1))) = C$ and by Lemma 14, we have such $u(x_1;C)$ always exists and $u(x_1;C) \geqslant \underline{h}(x_1)$. By Equation (6),

$$\mathrm{CP}_o(u(x_1;C);(x_1,\underline{h}(x_1))) = \frac{\underline{h}(x_1)r_h + \min\{x_1, m-u(x_1;C)\}r_\ell}{\underline{h}(x_1)r_h + \min\{x_1, m-\underline{h}(x_1)\}r_\ell} = C.$$

As in this case $x_1 + \underline{h}(x_1) < m$, we have $\min\{x_1, m-\underline{h}(x_1)\} = x_1$. Here, we argue that $\min\{x_1, m-u(x_1;C)\} = m-u(x_1;C)$. Contrary to our claim, suppose that $\min\{x_1, m-u(x_1;C)\} = x_1$. We then have

$$\mathrm{CP}_o(u(x_1;C);(x_1,\underline{h}(x_1))) = \frac{\underline{h}(x_1)r_h + x_1 r_\ell}{\underline{h}(x_1)r_h + x_1 r_\ell} = 1 > C,$$

which is a contradiction. Therefore, we can only have $\min\{x_1, m-u(x_1;C)\} = m-u(x_1;C)$. Then, we have

$$\frac{\underline{h}(x_1)r_h + (m-u(x_1;C))r_\ell}{\underline{h}(x_1)r_h + x_1 r_\ell} = C,$$

which is equivalent as

$$u(x_1;C) = (1-C)\frac{r_h}{r_\ell}\underline{h}(x_1) - Cx_1 + m.$$

This implies that $\frac{\partial u(x;C)}{\partial x} = (1-C)\frac{r_h}{r_\ell}\underline{h}'(x) - C$ when $x + \underline{h}(x) < m$. As we have $\underline{\mathcal{H}}(x) = \underline{h}(x)$ for $x \in [\underline{x_u}, \bar{x}_u]$, this implies the desired result.

C.2.2. Property 2 We take a point $(x_1, \overline{\mathcal{H}}(x_1))$ with $x_1 \in (x_H, \bar{x}_\ell)$. By definition of $l(\cdot;C)$, we have $\mathrm{CP}_u(l(x_1;C);(x_1,\bar{h}(x_1))) = C$. As $\overline{\mathcal{H}}(x_1) = \min\{m, \bar{h}(x_1)\}$, by Lemma 10, we have $\mathrm{CP}_u(l(x_1;C);(x_1,\overline{\mathcal{H}}(x_1))) = \mathrm{CP}_u(l(x_1;C);(x_1,\bar{h}(x_1))) = C$. By Lemma 14, such $l(x_1;C)$ always exists and $l(x_1;C) \leqslant \overline{\mathcal{H}}(x_1)$. By Equation (7),

$$\mathrm{CP}_u(l(x_1;C);(x_1,\overline{\mathcal{H}}(x_1))) = \frac{\max\{l(x_1;C), \min\{\overline{\mathcal{H}}(x_1), m-x_1\}\}r_h + \min\{x_1, m-l(x_1;C)\}r_\ell}{\overline{\mathcal{H}}(x_1)r_h + \min\{x_1, m-\overline{\mathcal{H}}(x_1)\}r_\ell} = C.$$

If $\min\{x_1, m-\overline{\mathcal{H}}(x_1)\} = x_1$, by Lemma 8 and the fact that $l(x_1;C) \leqslant \overline{\mathcal{H}}(x_1)$, $\mathrm{CP}_u(l(x_1;C);(x_1,\overline{\mathcal{H}}(x_1))) = 1 \neq C$, which cannot happen, and hence $\min\{x_1, m-\overline{\mathcal{H}}(x_1)\} = m-\overline{\mathcal{H}}(x_1)$. If $\min\{x_1, m-l(x_1;C)\} = x_1$, then

$$\mathrm{CP}_u(l(x_1;C);(x_1,\overline{\mathcal{H}}(x_1))) = \frac{(m-x_1)r_h + x_1 r_\ell}{\overline{\mathcal{H}}(x_1)r_h + (m-\overline{\mathcal{H}}(x_1))r_\ell} = \mathrm{CP}_u(0;(x_1,\overline{\mathcal{H}}(x_1))),$$

which implies that $x_1 \geqslant \bar{x}_\ell$. However, we define $x_1 \in (x_H, \bar{x}_\ell)$, therefore, this is a contradiction, and we can only have $\min\{x_1, m-\overline{\mathcal{H}}(x_1)\} = m-\overline{\mathcal{H}}(x_1)$ and $\min\{x_1, m-l(x_1;C)\} = m-l(x_1;C)$. Then, we have

$$\frac{l(x_1;C)r_h + (m-l(x_1;C))r_\ell}{\overline{\mathcal{H}}(x_1)r_h + (m-\overline{\mathcal{H}}(x_1))r_\ell} = C,$$

which is equivalent to

$$l(x_1;C) = C\overline{\mathcal{H}}(x_1) - \frac{(1-C)mr_\ell}{r_h - r_\ell},$$

and verifies Equation (36).

C.2.3. Property 3 We first show that $u(x;C)$ is non-increasing for $x \in [\underline{x}, \bar{x}]$. First, by the definition of $\underline{x}_u$, we have $u(x;C) = m$ for $x \in [\underline{x}, \underline{x}_u]$, which is non-increasing.

For $x \in (\underline{x}_u, \bar{x}_u)$, first recall that

$$\bar{x}_u = \begin{cases} x_L & \text{if } x_L + y_L \geqslant m; \\ \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\} & \text{Otherwise}, \end{cases}$$

Case 1 $-$ $\mathbf{x_L + y_L \geqslant m}$. If $x_L + y_L \geqslant m$, we have $\bar{x}_u = x_L$. Since point $L = (x_L, y_L)$ is the lowest point, $\underline{h}(x)$ decreases for $x < x_L$, increases for $x > x_L$, which implies that $\underline{h}'(x) \leqslant 0$ for $x < x_L$ and $\underline{h}'(x) \geqslant 0$ for $x > x_L$. Now, recall Property 1 that we just showed:

$$\frac{\partial u(x;C)}{\partial x} = \begin{cases} ((1-C)\frac{r_h}{r_\ell} + C)\underline{\mathcal{H}}'(x) & \text{if } x + \underline{\mathcal{H}}(x) \geqslant m; \\ (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x) - C & \text{if } x + \underline{\mathcal{H}}(x) < m. \end{cases}$$

where by Lemma 12, we have $\underline{\mathcal{H}}(x) = \underline{h}(x)$ for $x \in (\underline{x}_u, \bar{x}_u)$. This property and the fact that $\underline{h}'(x) \leqslant 0$ for $x < x_L$ and $\underline{h}'(x) \geqslant 0$ for $x > x_L$ imply that $u(x;C)$ decreases for $x < x_L$ and increases for $x > x_L$. As we force $u(x;C) = u(x_L;C)$ for $x > \bar{x}_u = x_L$, we have $u(x;C)$ is always non-increasing.

Case 2 $-$ $\mathbf{x_L + y_L < m}$. If $x_L + y_L < m$, we have $\bar{x}_u = \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{h}'(x) - C < 0\}$. For $x \in (\underline{x}_u, \bar{x}_u)$, by Lemma 12, we have $\underline{\mathcal{H}}(x) = \underline{h}(x)$. As $\underline{h}(x)$ is convex, we have its subderivative is increasing. Then, for $x < \bar{x}_u$, we have $(1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x) - C < 0$. Therefore, Property (1) that we just showed (i.e., Equation (50)) implies that $u(x;C)$ decreases for $x \in [\underline{x}_u, \bar{x}_u]$. As we force $u(x;C)$ to be a constant for $x \in [\bar{x}_u, \bar{x}]$, we have $u(x;C)$ is non-increasing for $x \in [\underline{x}, \bar{x}]$.

Finally, we show that $u(x;C)$ is convex for $x \in [\underline{x}_u, \bar{x}]$ by proving that its subderivative is increasing. As $\underline{h}(x)$ is convex, we have the subderivative of $\underline{h}(x)$ is increasing for $x \in [\underline{x}_u, \bar{x}_u]$. By Lemma 12, we have $\underline{\mathcal{H}}(x) = \underline{h}(x)$ for $x \in [\underline{x}_u, \bar{x}_u]$. Then, we have the subderivative of $\underline{\mathcal{H}}(x)$ is increasing for $x \in [\underline{x}_u, \bar{x}_u]$. Therefore, Equation (50) implies that $u(x;C)$ has increasing subderivative, and $u(x;C)$ is convex for $x \in [\underline{x}_u, \bar{x}_u]$. As $u(x;C)$ is non-increasing and convex for $x \in [\underline{x}_u, \bar{x}_u]$ and constant for $x \in [\bar{x}_u, \bar{x}]$, we have $u(x;C)$ is convex for $x \in [\underline{x}_u, \bar{x}]$.

C.2.4. Property 4 Because H is the highest point and $\mathcal{R}$ is convex, $\bar{h}(x)$ increases for $x < x_H$ and decreases for $x > x_H$, which implies that $\bar{h}'(x) \geqslant 0$ a.e. for $x < x_H$ and $\bar{h}'(x) \leqslant 0$ a.e. for $x > x_H$. (Recall that $\bar{h}(\cdot)$ is concave and point $H = (x_H, y_H) \in \bar{\mathcal{R}}$, which lies on the upper envelope $\bar{h}(\cdot)$, is the point in set $\bar{\mathcal{R}}$ that has the highest low-reward demand, where $\bar{\mathcal{R}} = \{(x,y) \in \mathcal{R} : y = \sup_{(x',y') \in \mathcal{R}} \min\{y', m\}\}$ is a subset of region $\mathcal{R}$ under which the high-reward demand (more precisely $\min\{y', m\}$ for any point $(x', y') \in \mathcal{R}$) is maximized.) Therefore, Property (2) that we just showed, (i.e., $\frac{\partial l(x;C)}{\partial x} = C\overline{\mathcal{H}}'(x)$) implies that $l(x;C)$ increases for $x < x_H$. As we force $l(x;C) = l(x_H;C)$ for $x \geqslant x_H$, we have $l(x;C)$ is always non-increasing.

Next, we show that $l(\cdot;C)$ is concave. As stated earlier, because $\mathcal{R}$ is a convex set, we have $\bar{h}(x)$ is concave. Since both $\bar{h}(x)$ and $y = m$ are concave, we have $\overline{\mathcal{H}}(x) = \min\{\bar{h}(x), m\}$ is concave. Therefore, the subderivative of $\overline{\mathcal{H}}(x)$ is decreasing. By Equation (36), we have the subderivative of $l(x;C)$ decreases for $x \in [\underline{x}_\ell, \bar{x}_\ell]$, which implies that $l(x;C)$ is concave for $x \in [\underline{x}_\ell, \bar{x}_\ell]$. As $l(x;C)$ is a constant for $x \in [\underline{x}, \underline{x}_\ell]$ and $l(x;C)$ is concave and non-increasing for $x \in [\underline{x}_\ell, \bar{x}_\ell]$, we have $l(x;C)$ is concave for $x \in [\underline{x}, \bar{x}_\ell]$.

C.2.5. Property 5 Here, we want to show that for any $C \leqslant C^\star(\mathcal{R})$ and any $x \in [\underline{x}, \bar{x}]$, we have $l(x;C) \leqslant u(x;C)$. For $x \in [\underline{x}, \bar{x}]$, let $p_b(x)$ be a function that

$$\mathrm{CP}_o(p_b(x);(x,\underline{h}(x))) = \mathrm{CP}_u(p_b(x);(x,\bar{h}(x))).$$

Lemma 17 shows that such $p_b(x)$ always exists, and $\mathrm{CP}_o(p_b(x);(x,\underline{h}(x))) = \mathrm{CP}_u(p_b(x);(x,\bar{h}(x))) \geqslant C^\star(\mathcal{R})$. Then, by Lemma 9, for any $C \leqslant C^\star(\mathcal{R})$, we have $\mathrm{CP}_o(l(x;C);(x,\underline{h}(x))) = C$ implies that $l(x;C) \leqslant p_b(x)$, and $\mathrm{CP}_u(u(x;C);(x,\bar{h}(x))) = C \leqslant C^\star(\mathcal{R})$ implies that $u(x;C) \geqslant p_b(x)$. Therefore, we have $l(x;C) \leqslant p_b(x) \leqslant u(x;C)$.

C.2.6. Property 6 Here, we would like to show for any $x \in [\underline{x}, \bar{x}]$, $l(x;C)$ is continuously increasing in C and $u(x;C)$ is continuously decreasing in C. This is because we have showed

$$l(x_1;C) = C\overline{\mathcal{H}}(x_1) - \frac{(1-C)mr_\ell}{r_h - r_\ell},$$

and

$$u(x_1;C) = (1-C)\frac{r_h}{r_\ell}\underline{h}(x_1) - Cx_1 + m.$$

From these two equations, we can simply find that for any $x \in [\underline{x}, \bar{x}]$, $l(x;C)$ is continuously increasing in C and $u(x;C)$ is continuously decreasing in C.

□

C.3. Proof of Lemma 3

First Direction. We first show that if $l(x;C) \leqslant p(x) \leqslant u(x;C)$, we have $\underline{\mathrm{CP}}(p(x);x) \geqslant C$ for any $x \in [\underline{x}, \bar{x}]$, where we define

$$\underline{\mathrm{CP}}(p(x);x) = \min\{\mathrm{CP}_o(p(x);(x,\underline{h}(x))), \mathrm{CP}_u(p(x);(x,\bar{h}(x)))\}. \tag{38}$$

This gives us the desired result because by Lemma 2, for any $x \in [\underline{x}, \bar{x}]$ and $p \geqslant 0$, we have

$$\min_{y \in [\underline{h}(x),\bar{h}(x)]} \{\mathrm{CP}(p;(x,y))\} = \min\left\{\mathrm{CP}(p;(x,\underline{h}(x))), \mathrm{CP}(p;(x,\bar{h}(x)))\right\}.$$

Part 1: $\mathrm{CP}_o(p(x);(x,\underline{h}(x))) \geqslant C$ if $p(x) \leqslant u(x;C)$. Here, we show that for any $x \in [\underline{x}, \bar{x}]$, $\mathrm{CP}_o(p(x);(x,\underline{h}(x)))$ is greater than or equal to C as long as $p(x) \leqslant u(x;C)$. Let us first focus on $x \in [\underline{x}, \underline{x}_u]$ and $x \in [\bar{x}_u, \bar{x}]$. By Lemma 13, for any $\underline{x} < x \leqslant \underline{x}_u$, we have $\mathrm{CP}_o(m;(x,\underline{h}(x))) \geqslant C$. By Lemma 15, for any $x \in [\bar{x}_u, \bar{x}]$, we have $\mathrm{CP}_o(m;(x,\underline{h}(x))) \geqslant C$. By Lemma 9, for any $p(x) \leqslant m = u(x;C)$ for $x \in [\underline{x}, \underline{x}_u]$ and $[\bar{x}_u, \bar{x}]$, we have

$$\mathrm{CP}_o(p(x);(x,\underline{h}(x))) \geqslant \mathrm{CP}_o(m;(x,\underline{h}(x))) \geqslant C.$$

Next, we consider $x \in [\underline{x}_u, \bar{x}_u]$. By Lemma 14, we have

$$\mathrm{CP}_o(u(x;C);(x,\underline{h}(x))) = C,$$

and by Lemma 9, we have for any $p(x) \leqslant u(x;C)$,

$$\mathrm{CP}_o(p(x);(x,\underline{h}(x))) \geqslant \mathrm{CP}_o(u(x;C);(x,\underline{h}(x))) = C,$$

which is the desired result.

Part 2: $\mathbf{CP}_u(p(x);(x,\bar{h}(x))) \geqslant C$ if $p(x) \geqslant l(x;C)$. Here, we show that for any $x \in [\underline{x}, \bar{x}]$, $\mathrm{CP}_u(p(x);(x,\bar{h}(x)))$ is greater than or equal to C as long as $p(x) \geqslant l(x;C)$. Let us first consider any $x \in [\underline{x}, x_H]$ and $x \in [\bar{x}_l, \bar{x}]$. By the definition of x_H and Lemma 13, for $x \in [\underline{x}, x_H]$ and $[\bar{x}_\ell, \bar{x}]$, we have $\mathrm{CP}_u(0;(x,\bar{h}(x))) \geqslant C$. By Lemma 9, for any $p(x) \geqslant 0 = l(x;C)$ for $x \in [\underline{x}, x_H]$ and $[\bar{x}_\ell, \bar{x}]$, we then have

$$\mathrm{CP}_u(p(x);(x,\bar{h}(x))) \geqslant \mathrm{CP}_u(0;(x,\bar{h}(x))) \geqslant C\,,$$

which is the desired result. Now, let us consider any $x \in [x_H, \bar{x}_\ell]$, By Lemma 14, we have $\mathrm{CP}_u(l(x;C);(x,\bar{h}(x))) = C$, and by Lemma 9, we have for any $p(x) \geqslant l(x;C)$,

$$\mathrm{CP}_u(p(x);(x,\bar{h}(x))) \geqslant \mathrm{CP}_u(l(x;C);(x,\bar{h}(x))) = C\,,$$

which is the desired result.

Second Direction. So far we have established that if $p(x) \in [l(x;C), u(x;C)]$, we have $\underline{\mathrm{CP}}(p(x);x) \geqslant C$ for any $(x,y) \in \mathcal{R}$. Next, we show that if $p(x) > u(x;C)$ or $p(x) < l(x;C)$, we have $\underline{\mathrm{CP}}(p(x);x) < C$.

Part 1: $\mathbf{CP}_o(p(x);(x,\underline{h}(x))) < C$ if $p(x) > u(x;C)$. First, as $u(x;C) = m$ for $x \in [\underline{x}, \underline{x}_u]$ and $[\bar{x}_u, \bar{x}]$, and $p(x) \leqslant m$, we cannot have $p(x) > u(x;C)$. Thus, we need to only consider $x \in [\underline{x}_u, \bar{x}_u]$. For any $x \in [\underline{x}_u, \bar{x}_u]$, by definition, $u(x;C)$ is the largest PL value such that

$$\mathrm{CP}_o(u(x;C);(x,\underline{h}(x))) = C,$$

and by Lemma 9, if $p(x) > u(x;C)$, we have

$$\mathrm{CP}_o(p(x);(x,\underline{h}(x))) < \mathrm{CP}_o(u(x;C);(x,\underline{h}(x))) = C\,,$$

which is the desired result.

Part 2: $\mathbf{CP}_u(p(x);(x,\bar{h}(x))) < C$ if $p(x) < l(x;C)$. As $l(x;C) = 0$ for $x \in [\underline{x}, x_H]$ and $[\bar{x}_\ell, \bar{x}]$, and $p(x) \geqslant 0$, we cannot have $p(x) < l(x;C)$. Thus, we consider $x \in [x_H, \bar{x}_\ell]$. For any $x \in [x_H, \bar{x}_\ell]$, by definition, $l(x;C)$ is the smallest PL value such that

$$\mathrm{CP}_u(l(x;C);(x,\bar{h}(x))) = C,$$

and by Lemma 9, if $p(x) < l(x;C)$, we have

$$\mathrm{CP}_u(p(x);(x,\bar{h}(x))) < \mathrm{CP}_u(l(x;C);(x,\bar{h}(x))) = C\,,$$

which is the desired result.

C.4. Proof of Lemma 4

To show the result, we show the optimization problem in Equation (18) is equivalent to that in Equation (16). Since the only difference between these two problems is their first set of constraints, we only need to show that the feasible regions of these two problems are identical. To do so, we show that any feasible solution to Problem (18) is a feasible solution to Problem (16) and vice versa.

Considering a feasible solution to Problem (18) with $p(x) \in [\widetilde{l}(x;C), u(x;C)]$ for any $x \in [\underline{x}, \bar{x}]$. By Equation (17), we know that for any $C \in [0,1]$ and $x \in [\underline{x}, \bar{x}]$, $\widetilde{l}(x;C) \geqslant l(x;C)$. Therefore, $\widetilde{l}(x;C) \leqslant p(x) \leqslant u(x;C)$

implies that $l(x;C) \leqslant p(x) \leqslant u(x;C)$, as desired. Recall that $l(x;C) \leqslant p(x) \leqslant u(x;C)$ is the first constraint in Problem (16), and hence the above argument shows that any feasible solution to Problem (18) is a feasible solution to Problem (16).

Next, we show the opposite direction. Contrary to our claim, suppose that there exists a feasible solution $p(x)$ to Problem (16) with $l(x_1;C) \leqslant p(x_1) < \widetilde{l}(x_1;C)$ for some $x_1 \in [\underline{x}, \bar{x}]$. (This shows that there exists a feasible solution to Problem (16), which is not a feasible solution to Problem (18).) By Equation (17), we must have $x_1 > x_{-1}$, where x_{-1} is defined in Equation (17). As $l(x_{-1};C) = \widetilde{l}(x_{-1};C)$, we have $p(x_{-1}) \geqslant l(x_{-1};C) = \widetilde{l}(x_{-1};C)$. Then, we have

$$\frac{p(x_1) - p(x_{-1})}{x_1 - x_{-1}} < \frac{\widetilde{l}(x_1;C) - \widetilde{l}(x_{-1};C)}{x_1 - x_{-1}} = -1,$$

where the equation holds because by definition of $\widetilde{l}(x;C)$, the slope of $\widetilde{l}(x;C)$ w.r.t. x is -1 for any $x \in [x_{-1}, x_1]$. That $\frac{p(x_1)-p(x_{-1})}{x_1 - x_{-1}} < -1$ implies that $p'(x) < -1$ on a positive measure set, and hence, $p(x)$ is not a valid PL function, which is a contradiction.

C.5. Proof of Lemma 5

First Direction. We first show the 'if' statement. That is, if $\underline{g}(x;R) \leqslant p(x) \leqslant \bar{g}(x;R)$, we have $\underline{\text{CP}}(p(x);x) \geqslant R$ for any $x \in [0, \max\{m, \bar{x}\}]$, where with a slight abuse of notation, we define

$$\underline{\text{CP}}(p(x);x) = \min\{\text{CP}_o(p(x);(x,0)), \text{CP}_u(p(x);(x,m))\}. \tag{39}$$

Notice that by Lemma 10, we have $\text{CP}_u(p(x);(x,m)) = \text{CP}_u(p(x);(x,y))$ for any $y \geqslant m$. Then, by Lemma 2, it suffices to show $\underline{\text{CP}}(p(x);x) \geqslant R$ when $\underline{g}(x;R) \leqslant p(x) \leqslant \bar{g}(x;R)$.

Part 1: $\mathbf{CP}_o(p(x);(x,0)) \geqslant R$ if $p(x) \leqslant \bar{g}(x;R)$. First observe that, by Definition of CP_o in Equation (6), if we set $p(x) = \bar{g}(x;R)$, we have

$$\text{CP}_o(\bar{g}(x;R);(x,0)) = \frac{0 \cdot r_h + \min\{x, m - \bar{g}(x;R)\}r_\ell}{0 \cdot r_h + \min\{x, m-0\}r_\ell} = \frac{\min\{x, m - \bar{g}(x;R)\}r_\ell}{\min\{x,m\}r_\ell}.$$

If $x \leqslant m$, we have $\bar{g}(x;R) = -Rx + m$, and we can obtain

$$\frac{\min\{x, m - \bar{g}(x;R)\}r_\ell}{\min\{x,m\}r_\ell} = \frac{\min\{x, m - (-Rx+m)\}r_\ell}{xr_\ell} = \frac{Rxr_\ell}{xr_\ell} = R.$$

Otherwise, if $x > m$, we have $\bar{g}(x;R) = \bar{g}(m;R) = -Rm + m$, and we can obtain

$$\frac{\min\{x, m - \bar{g}(x;R)\}r_\ell}{mr_\ell} = \frac{\min\{x, m - (-Rm+m)\}r_\ell}{mr_\ell} = \frac{Rmr_\ell}{mr_\ell} = R.$$

Then, by Lemma 9, we have for any $p(x) \leqslant \bar{g}(x;R)$, we have

$$\text{CP}_o(p(x);(x,0)) \geqslant \text{CP}_o(\bar{g}(x;R);(x,0)) = R,$$

which is the desired result.

Part 2: $\mathbf{CP}_u(p(x);(x,m)) \geqslant R$ if $p(x) \geqslant \underline{g}(x;R)$. By Definition of CP_u in Equation (7), we have

$$\begin{aligned}\text{CP}_u(p(x);(x,m)) &= \frac{\max\{p(x), \min\{m, m-x\}\}r_h + \min\{x, m - p(x)\}r_\ell}{mr_h + \min\{x, m-m\}r_\ell} \\ &= \frac{\max\{p(x), m-x\}r_h + \min\{x, m-p(x)\}r_\ell}{mr_h}.\end{aligned}$$

We would like to show that if $p(x) \geqslant \underline{g}(x;R)$, we have $\mathrm{CP}_u(p(x);(x,m)) \geqslant R$, where $\underline{g}(x;R) = \frac{m(R - r_\ell/r_h)}{1 - r_\ell/r_h}$ for $x \in [0, \max\{m, \bar{x}\}]$. If $p(x) \geqslant m - x$, we have

$$\mathrm{CP}_u(p(x);(x,m)) = \frac{p(x) r_h + (m - p(x)) r_\ell}{m r_h} \geqslant \frac{\frac{m(R - r_\ell/r_h)}{1 - r_\ell/r_h} r_h + (m - \frac{m(R - r_\ell/r_h)}{1 - r_\ell/r_h}) r_\ell}{m r_h} = R\,,$$

where the inequality holds because $p(x) \geqslant \underline{g}(x;R) = \frac{m(R - r_\ell/r_h)}{1 - r_\ell/r_h}$. Otherwise, if $p(x) < m - x$, as $p(x) \geqslant \underline{g}(x;R)$, we have $\underline{g}(x;R) < m - x$. Then,

$$\mathrm{CP}_u(p(x);(x,m)) = \frac{(m - x) r_h + x r_\ell}{m r_h} \geqslant \frac{p(x) r_h + (m - p(x)) r_\ell}{m r_h} \geqslant R,$$

where the first inequality is because $\frac{(m-x) r_h + x r_\ell}{m r_h}$ is decreasing in x and $x < m - p(x)$. The last inequality, which is the desired result, is because

$$p(x) r_h + (m - p(x)) r_\ell = p(x)(r_h - r_\ell) + m r_\ell \geqslant \underline{g}(x;R)(r_h - r_\ell) + m r_\ell\,,$$

and by some calculations, we have $\frac{\underline{g}(x;R)(r_h - r_\ell) + m r_\ell}{m r_h} = R$.

Second Direction. So far, we have established that if $p(x) \in [\underline{g}(x;R), \bar{g}(x;R)]$, we have $\underline{\mathrm{CP}}(p(x);x) \geqslant R$ for any $(x,y) \in \mathcal{R}$. Next, we show that if $p(x) > \bar{g}(x;R)$ or $p(x) < \underline{g}(x;C)$, we have $\underline{\mathrm{CP}}(p(x);x) < R$.

If $p(x) > \bar{g}(x;R)$ for some $x \in [0, \max\{m, \bar{x}\}]$, by Equation (19), we have if $x \in [0, m]$, $\bar{g}(x;R) = -Rx + m \geqslant -x + m$ and hence $p(x) + x > m$. If $x > m$, we have $\bar{g}(x;R) = \bar{g}(m;R)$. Then, we have

$$\mathrm{CP}_o(p(x);(x,0)) = \frac{0 \cdot r_h + \min\{x, m - p(x)\} r_\ell}{0 \cdot r_h + \min\{x, m - 0\} r_\ell} = \frac{\min\{x, m - p(x)\} r_\ell}{\min\{x, m\} r_\ell}.$$

If $x \leqslant m$, we have

$$\frac{\min\{x, m - p(x)\} r_\ell}{\min\{x, m\} r_\ell} = \frac{\min\{x, m - p(x)\} r_\ell}{x r_\ell} < \frac{(m - \bar{g}(x;R))\, r_\ell}{x r_\ell} = R,$$

where the inequality is because $p(x) > \bar{g}(x;R)$. Otherwise, if $x > m$, we have

$$\frac{\min\{x, m - p(x)\} r_\ell}{\min\{x, m\} r_\ell} = \frac{(m - p(x)) r_\ell}{m r_\ell} = \frac{m - p(x)}{m} < \frac{m - \bar{g}(m;R)}{m} = R.$$

If $p(x) < \underline{g}(x;R)$ for some $x \in [0, \max\{m, \bar{x}\}]$, as a valid PL function $p(x)$ is non-increasing, we have $p(\max\{m, \bar{x}\}) < \underline{g}(x;R)$. Therefore,

$$\begin{aligned}\mathrm{CP}_u(p(\max\{m, \bar{x}\}); (\max\{m, \bar{x}\}, m)) &= \frac{p(\max\{m, \bar{x}\}) r_h + (m - p(\max\{m, \bar{x}\})) r_\ell}{m r_h} \\ &< \frac{\frac{m(R - r_\ell/r_h)}{1 - r_\ell/r_h} r_h + (m - \frac{m(R - r_\ell/r_h)}{1 - r_\ell/r_h}) r_\ell}{m r_h} = R.\end{aligned}$$

C.6. Proof of Theorem 4

Algorithm 2 presents an optimal solution to Problem (C-Pareto-right). That is, at the optimal solution to Problem (C-Pareto-right), denoted by $p_{\text{right}}(\cdot)$, we set $p_{\text{right}}(x)$ based on Equations (23) and (25). Furthermore, the optimal objective value of Problem (C-Pareto-right), R_{right}, is given in Equation (24).

We split the proof into three cases, in each case, we first figure out the robust ratio under the PL function $p_{\text{right}}(\cdot)$ and check the feasibility and optimality of $p_{\text{right}}(\cdot)$:

- *Case 1:* $[\widetilde{l}(\bar{x};C),u(\bar{x};C)] \cap [\underline{g}(\bar{x}),\bar{g}(\bar{x})] \neq \varnothing$. In this case, if $\widetilde{l}(\bar{x};C) < \underline{g}(\bar{x})$,

$$p_{\text{right}}(\bar{x}) = \arg \min_{p \in [\widetilde{l}(\bar{x};C),u(\bar{x};C)]} |p - \underline{g}(\bar{x})| = \underline{g}(\bar{x}).$$

If $\widetilde{l}(\bar{x};C) \geqslant \underline{g}(\bar{x})$, we have $p_{\text{right}}(\bar{x}) = \widetilde{l}(\bar{x};C)$. If $\bar{x} \geqslant m$, then as the right problem is only defined on $\bar{x}$, by definition, we have $\underline{g}(\bar{x}) = \bar{g}(\bar{x}) = \frac{1-r_\ell/r_h}{2-r_\ell/r_h}m$. Then, $p_{\text{right}}(\bar{x}) = \frac{1-r_\ell/r_h}{2-r_\ell/r_h}m$, which is feasible. By Lemma 10, we have

$$\begin{aligned} R_{\text{right}} &= \min\{\text{CP}_o(p_{\text{right}}(\bar{x});(\bar{x},0)),\text{CP}_u(p_{\text{right}}(\bar{x});(\bar{x},m))\} \\ &= \min\left\{\text{CP}_o(\frac{1-r_\ell/r_h}{2-r_\ell/r_h}m;(m,0)),\text{CP}_u(\frac{1-r_\ell/r_h}{2-r_\ell/r_h}m;(m,m))\right\} = \frac{1}{2-r_\ell/r_h}, \end{aligned}$$

which matches the upper bound of the robust ratio in the absence of ML advice. Therefore, $p_{\text{right}}(\bar{x})$ is optimal in this case.

Otherwise, if $\bar{x} < m$, by Equation (22), $p_{\text{right}}(x) = \max\{-x+\bar{x}+p_{\text{right}}(\bar{x};C),\underline{g}(x)\}$ for $x \in [\bar{x},m]$. By definition, we have $\underline{g}(x) \leqslant p_{\text{right}}(x)$. For the part where $p_{\text{right}}(x) = \underline{g}(x)$, we have $p_{\text{right}}(x) \leqslant \bar{g}(x)$ because by Equation (19), $\underline{g}(x) \leqslant \bar{g}(x)$ for any $x \in [0,m]$. Then, we check that $-x+\bar{x}+p_{\text{right}}(\bar{x};C) \leqslant \bar{g}(x)$ for $x \in [\bar{x},m]$. Notice that $-x+\bar{x}+p_{\text{right}}(\bar{x};C)$ is a line with slope -1 and by Equation (19), $\bar{g}(x)$ is a line with slope $-R \geqslant -1$. Moreover, $-\bar{x}+\bar{x}+p_{\text{right}}(\bar{x};C) = p_{\text{right}}(\bar{x};C) = \max\{\underline{g}(\bar{x}),\widetilde{l}(\bar{x};C)\} \leqslant \bar{g}(x)$ since $[\widetilde{l}(\bar{x};C),u(\bar{x};C)] \cap [\underline{g}(\bar{x}),\bar{g}(\bar{x})] \neq \varnothing$. We have $p_{\text{right}}(x) \leqslant \bar{g}(x)$. By taking $R=\rho$ in Problem (C-Pareto-Trans), we can find $p_{\text{right}}(\cdot)$ is a feasible solution, and therefore, it achieves a robust ratio of at least ρ. By Ball and Queyranne (2009), we know ρ is the upper bound among all algorithms, and therefore, $p_{\text{right}}(\cdot)$ is optimal. In addition, notice that $p_{\text{right}}(m) = \underline{g}(m)$ and we can check $\text{CP}_u(p_{\text{right}}(\bar{x});(m,m)) = \rho$, and hence $R_{\text{right}} = \min\{\text{CP}_o(p_{\text{right}}(\bar{x});(\bar{x},0)),\text{CP}_u(p_{\text{right}}(\bar{x});(m,m))\}$.

- *Case 2:* $u(\bar{x};C) < \underline{g}(\bar{x})$. In this case, we first show $p_{\text{right}}(\cdot)$ achieves a robust ratio of $R_{\text{right}} = \text{CP}_u(p_{\text{right}}(\bar{x});(\max\{m,\bar{x}\},m))$ and $\text{CP}_o(p_{\text{right}}(\bar{x});(\bar{x},0)) \geqslant \text{CP}_u(p_{\text{right}}(\bar{x});(\max\{m,\bar{x}\},m))$. Then, we show $p_{\text{right}}(\cdot)$ is feasible, and finally, we show it is optimal among all PL functions.

In this case, $p_{\text{right}}(x) = \arg\min_{p \in [\widetilde{l}(\bar{x};C),u(\bar{x};C)]} |p - \underline{g}(\bar{x})| = u(\bar{x};C)$, and by definition, $p_{\text{right}}(x) = p_{\text{right}}(\bar{x})$ for $x \in [\bar{x},\max\{m,\bar{x}\}]$. By Lemmas 2 and 10, we know the worst case is achieved on $(x,0)$ or (x,m) for some $x \in [\bar{x},\max\{m,\bar{x}\}]$; that is, $R_{\text{right}} = \inf_{x \in [\bar{x},\max\{m,\bar{x}\}]} \min\{\text{CP}_u(p_{\text{right}}(x);(x,m)),\text{CP}_o(p_{\text{right}}(x);(x,0))\}$. As we have $p_{\text{right}}(x) \leqslant m$, by Lemma 7, we have

$$\text{CP}_u(p_{\text{right}}(x);(x,m)) \geqslant \text{CP}_u(p_{\text{right}}(\max\{m,\bar{x}\});(\max\{m,\bar{x}\},m)).$$

As $p_{\text{right}}(\max\{m,\bar{x}\}) = p_{\text{right}}(\bar{x}) < \underline{g}(\bar{x}) = \underline{g}(\max\{m,\bar{x}\}) = m(1-r_\ell/r_h)/(2-r_\ell/r_h)$, where the first inequality is because $p_{\text{right}}(\bar{x}) \leqslant u(\bar{x};C) < \underline{g}(\bar{x})$, by Lemma 9, we have

$$\text{CP}_u(p_{\text{right}}(\max\{m,\bar{x}\});(\max\{m,\bar{x}\},m)) < \text{CP}_u(\underline{g}(\max\{m,\bar{x}\});(\max\{m,\bar{x}\},m)) = \rho.$$

As $p_{\text{right}}(\bar{x}) < \underline{g}(\bar{x})$, we also have $p_{\text{right}}(x) < \bar{g}(\bar{x})$ for any $x \in [\bar{x},m]$. This is because by definition, for any $x \in [\bar{x},m]$, $\underline{g}(x) \leqslant \bar{g}(x)$. Then, by Lemma 9, for $x \in [\bar{x},m]$, we have

$$\text{CP}_o(p_{\text{right}}(x);(x,0)) \geqslant \text{CP}_o(\bar{g}(x);(x,0)) = \rho > \text{CP}_u(p_{\text{right}}(m);(m,m)),$$

where the equality is because $\bar{g}(x) = -\rho x + m$ and one can easily check $\mathrm{CP}_o(-\rho x + m; (x, 0)) = \rho$ for any $x \in [0, m]$.

Therefore, the robust ratio of $p_{\text{right}}(x)$ for $x \in [\bar{x}, m]$ is $R_{\text{right}} = \mathrm{CP}_u(p_{\text{right}}(m); (m, m))$, and by Lemma 5, we have $\underline{g}(x; R_{\text{right}}) \leqslant p_{\text{right}}(x) \leqslant \bar{g}(x; R_{\text{right}})$. Also, $p_{\text{right}}(\cdot)$ is a constant function and is valid. We obtain $p_{\text{right}}(\cdot)$ is feasible.

Finally, we show that $p_{\text{right}}(\cdot)$ is optimal among all PL algorithms. We prove by contradiction. Suppose that a valid $p_1(x)$ can achieve a robust ratio greater than R_{right}. Then, we have

$$\inf_{x \in [\bar{x}, m]} \min\{\mathrm{CP}_o(p_1(x); (x, 0)), \mathrm{CP}_u(p_1(x); (x, m))\} > \mathrm{CP}_u(p_{\text{right}}(m); (m, m)),$$

which implies that $\mathrm{CP}_u(p_1(m); (m, m)) > \mathrm{CP}_u(p_{\text{right}}(m); (m, m))$. By Lemma 9, we have $p_1(m) > p_{\text{right}}(m)$. As $p_1(\cdot)$ is valid, it is non-increasing, we have $p_1(\bar{x}) \geqslant p_1(m) > p_{\text{right}}(m) = u(\bar{x}; C)$, which is a contradiction because $p_1(\bar{x}) > u(\bar{x}; C)$ means it is infeasible.

- *Case 3:* $\widetilde{l}(\bar{x}; C) > \bar{g}(\bar{x})$. In this case, $p_{\text{right}}(\bar{x}) = \arg\min_{p \in [\widetilde{l}(\bar{x}; C), u(\bar{x}; C)]} |p - \underline{g}(\bar{x})| = \widetilde{l}(\bar{x}; C)$, and we have $p_{\text{right}}(x) = \bar{g}(x; R_{\text{right}})$ for $x \in [\bar{x}, \max\{m, \bar{x}\}]$ where $R_{\text{right}} = \mathrm{CP}_o(\widetilde{l}(\bar{x}; C); (\bar{x}, 0))$. Observe that $p_{\text{right}}(\cdot)$ is continuous at $\bar{x}$ because $p_{\text{right}}(\bar{x}) = \bar{g}(\bar{x}; R_{\text{right}}) = \bar{g}(\bar{x}; \mathrm{CP}_o(\widetilde{l}(\bar{x}; C); (\bar{x}, 0)))$, and if $\bar{x} \leqslant m$, we have $\bar{g}(\bar{x}; \mathrm{CP}_o(\widetilde{l}(\bar{x}; C); (\bar{x}, 0))) = -\frac{m - \widetilde{l}(\bar{x}; C)}{\bar{x}} \bar{x} + m = \widetilde{l}(\bar{x}; C)$. Similarly, if $\bar{x} > m$, we have $\bar{g}(\bar{x}; \mathrm{CP}_o(\widetilde{l}(\bar{x}; C); (\bar{x}, 0))) = -\frac{m - \widetilde{l}(\bar{x}; C)}{m} m + m = \widetilde{l}(\bar{x}; C)$. By Lemmas 2 and 10, for $x \in [\bar{x}, \max\{m, \bar{x}\}]$, the worst over- and under-protected points are $(x, 0)$ and (x, m), respectively; that is, $R_{\text{right}} = \inf_{x \in [\bar{x}, \max\{m, \bar{x}\}]} \min\{\mathrm{CP}_o(p_{\text{right}}(x); (x, 0)), \mathrm{CP}_u(p_{\text{right}}(x); (x, m))\}$. So, we first claim that $p_{\text{right}}(\cdot)$ achieves a robust ratio of $R_{\text{right}} = \mathrm{CP}_o(\widetilde{l}(\bar{x}; C); (\bar{x}, 0))$ by showing that for any $x \in [\bar{x}, \max\{m, \bar{x}\}]$, $\mathrm{CP}_o(p_{\text{right}}(x); (x, 0)) \geqslant R_{\text{right}}$ and $\mathrm{CP}_u(p_{\text{right}}(x); (x, m)) \geqslant R_{\text{right}}$. Then, we show its feasibility and optimality.

By Equation (6), we have for any $x \in [\bar{x}, \max\{m, \bar{x}\}]$,

$$\mathrm{CP}_o(p_{\text{right}}(x); (x, 0)) = \frac{0 \cdot r_h + \min\{x, m - p_{\text{right}}(x)\} r_\ell}{0 \cdot r_h + \min\{x, m - 0\} r_\ell} = \frac{\min\{x, m - p_{\text{right}}(x)\} r_\ell}{\min\{x, m\} r_\ell}.$$

If $x \leqslant m$, we have

$$\begin{aligned} \frac{\min\{x, m - p_{\text{right}}(x)\} r_\ell}{\min\{x, m\} r_\ell} &= \frac{(m - p_{\text{right}}(x)) r_\ell}{x r_\ell} \\ &= \frac{(m - \bar{g}(x; R_{\text{right}})) r_\ell}{x r_\ell} \\ &= \frac{m - \bar{g}(x; R_{\text{right}})}{x} = R_{\text{right}}, \end{aligned}$$

where the second equality is because $p_{\text{right}}(x) = \bar{g}(x; R_{\text{right}}) = -R_{\text{right}} x + m$ and $m - (-R_{\text{right}} x + m) = R_{\text{right}} x \leqslant x$. If $x > m$, we have

$$\begin{aligned} \frac{\min\{x, m - p_{\text{right}}(x)\} r_\ell}{\min\{x, m\} r_\ell} &= \frac{(m - p_{\text{right}}(x)) r_\ell}{m r_\ell} \\ &= \frac{(m - \bar{g}(x; R_{\text{right}})) r_\ell}{m r_\ell} \\ &= \frac{m - \bar{g}(m; R_{\text{right}})}{m} = R_{\text{right}}. \end{aligned}$$

For any $x \in [\bar{x}, \max\{m, \bar{x}\}]$, by Lemma 7, we have

$$\mathrm{CP}_u(p_{\text{right}}(x); (x, m)) \geqslant \mathrm{CP}_u(p_{\text{right}}(\max\{m, \bar{x}\}); (\max\{m, \bar{x}\}, m)).$$

As we have $\mathrm{CP}_u(\underline{g}(\max\{m, \bar{x}\}); (\max\{m, \bar{x}\}, m)) = \rho$, and $p_{\text{right}}(\max\{m, \bar{x}\}) = \bar{g}(\max\{m, \bar{x}\}) \geqslant \underline{g}(\max\{m, \bar{x}\})$, by Lemma 9, we have

$$\mathrm{CP}_u(p_{\text{right}}(\max\{m, \bar{x}\}); (\max\{m, \bar{x}\}, m)) \geqslant \rho \geqslant R_{\text{right}}.$$

Therefore, the robust ratio of $p_{\text{right}}(x)$ is R_{right}. As $p_{\text{right}}(x) = \bar{g}(x; R_{\text{right}})$, we have $\underline{g}(x; R_{\text{right}}) \leqslant p_{\text{right}}(x) \leqslant \bar{g}(x; R_{\text{right}})$. Also, since $p_{\text{right}}(\bar{x}) = \tilde{l}(\bar{x}; C)$, we have $\tilde{l}(\bar{x}; C) \leqslant p_{\text{right}}(\bar{x}) \leqslant u(\bar{x}; C)$. In addition, $p_{\text{right}}(x)$ has slope $-R_{\text{right}} \leqslant -1$, which means it is valid. We have $p_{\text{right}}(\cdot)$ is a feasible solution.

Finally, we show that $p_{\text{right}}(\cdot)$ is optimal among all PL algorithms. We prove by contradiction. Suppose that a valid $p_1(x)$ can achieve a robust ratio greater than R_{right}. Then, we have

$$\inf_{x \in [\bar{x}, \max\{m, \bar{x}\}]} \min\{\mathrm{CP}_o(p_1(x); (x, 0)), \mathrm{CP}_u(p_1(x); (x, m))\} > \mathrm{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)),$$

which implies that $\mathrm{CP}_o(p_1(\bar{x}); (\bar{x}, 0)) > \mathrm{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0))$. By Lemma 9, we have $p_1(\bar{x}) < p_{\text{right}}(\bar{x})$. However, as we have $p_{\text{right}}(\bar{x}) = \tilde{l}(\bar{x}; C)$, we obtain $p_1(\bar{x}) < \tilde{l}(\bar{x}; C)$, which means $p_1(\cdot)$ is infeasible and forms a contradiction.

□

C.7. Proof of Theorem 5

We first show that $(p_{\text{left}}(\cdot), R_{\text{left}})$ is a feasible solution to Problem (C-Pareto-left), where $p_{\text{left}}(x) = \max\{\tilde{l}(x; C), p_{\text{right}}(\bar{x})\}, x \in [0, \bar{x}]$ and $R_{\text{left}} = \min\{\mathrm{CP}_u(p_{\text{left}}(\bar{x}); (\bar{x}, m))), \inf_{x \in [0, \bar{x}]} \mathrm{CP}_o(p_{\text{left}}(x); (x, 0))\}$. Under the PL $p_{\text{left}}(\cdot)$, we first note that by Lemma 2, we only need to consider the points $(x, 0)$ and (x, m) for any $x \in [0, \bar{x}]$. That is,

$$R_{\text{left}} = \min\left\{\inf_{x \in [0, \bar{x}]} \mathrm{CP}_o(p_{\text{left}}(x); (x, 0)), \inf_{x \in [0, \bar{x}]} \mathrm{CP}_u(p_{\text{left}}(x); (x, m))\right\},$$

By Lemma 7, we then have

$$\mathrm{CP}_u(p_{\text{left}}(x); (x, m)) \geqslant \mathrm{CP}_u(p_{\text{left}}(\bar{x}); (\bar{x}, m)) \quad \Rightarrow \quad \inf_{x \in [0, \bar{x}]} \mathrm{CP}_u(p_{\text{left}}(x); (x, m)) = \mathrm{CP}_u(p_{\text{left}}(\bar{x}); (\bar{x}, m)).$$

This implies that $R_{\text{left}} = \min\{\mathrm{CP}_u(p_{\text{left}}(\bar{x}); (\bar{x}, m)), \inf_{x \in [0, \bar{x}]} \mathrm{CP}_o(p_{\text{left}}(x); (x, 0))\}$, as desired.

R_{left} is feasible because the range of compatible ratio is $[0, 1]$. To show that $p_{\text{left}}(x)$ is feasible, first, as we have shown it achieves a robust ratio of R_{left}, by Lemma 5, we have $\underline{g}(x; R_{\text{left}}) \leqslant p_{\text{left}}(x) \leqslant \bar{g}(x; R_{\text{left}})$. Second, we show that $p_{\text{left}}(\cdot)$ is a valid PL function and $p_{\text{left}}(x) \in [\tilde{l}(x; C), u(x; C)]$. Observe that $\tilde{l}(x; C)$ is a valid PL function by definition, and hence we have $p_{\text{left}}(x)$ is also valid. Third, we show that $\tilde{l}x; C \leqslant p_{\text{left}}(x) \leqslant u(x; C)$ for $x \in [0, \bar{x}]$. As $p_{\text{left}}(x) = \max\{\tilde{l}(x; C), p_{\text{right}}(\bar{x})\}$, we have $\tilde{l}(x; C) \leqslant p_{\text{left}}(x)$ for any $x \in [0, \bar{x}]$. In addition, we have $p_{\text{left}}(\bar{x}) = p_{\text{right}}(\bar{x}) \leqslant u(\bar{x}; C)$, and by Lemma 6, we have $u(x; C)$ is a non-increasing function in x. This implies that $u(x; C) \geqslant p_{\text{left}}(\bar{x})$ for any $x \in [0, \bar{x}]$. Also, as C is assumed to be less than $C^\star(\mathcal{R})$, by Lemma 4, we have $\tilde{l}(x; C) \leqslant u(x; C)$ for $x \in [0, \bar{x}]$. Therefore, we have $p_{\text{left}}(x) = \max\{p_{\text{left}}(\bar{x}), \tilde{l}(x; C)\} \leqslant u(x; C)$, which is the desired result.

Second, we show that $p_{\mathrm{left}}(\cdot)$ is optimal. To do so, we argue that (i) by Lemma 5, we have

$$\underline{g}(x; R_{\mathrm{left}}) \leqslant p_{\mathrm{left}}(x) \leqslant \bar{g}(x; R_{\mathrm{left}}),$$

and (ii) there does not exist any other valid PL function that achieves a higher robust ratio than R_{left} while satisfying the consistency lower and upper bounds.

Recall that $R_{\mathrm{left}} = \min\{\mathrm{CP}_u(p_{\mathrm{left}}(\bar{x}); (\bar{x}, m)), \inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p_{\mathrm{left}}(x); (x, 0))\}$. As Problem (C-Pareto-left) restricts the value of $p(\bar{x})$, we have any PL algorithm has the same worst under-protected ratio, i.e. $\mathrm{CP}_u(p_{\mathrm{left}}(\bar{x}); (\bar{x}, m))$. Then, we show that any PL algorithm cannot get a larger worst over-protected ratio. For over-protected case, we let

$$\widetilde{R_{\mathrm{left}}} = \inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p_{\mathrm{left}}(x); (x, 0)),$$

and let the infimum is achieved on x_1, i.e. $\mathrm{CP}_o(p_{\mathrm{left}}(x_1); (x_1, 0)) = \widetilde{R_{\mathrm{left}}}$. If there exists a PL algorithm $p(x)$ such that $\inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p(x); (x, 0)) > R_{\mathrm{left}}$, then $\mathrm{CP}_o(p(x_1); (x_1, 0)) > \mathrm{CP}_o(p_{\mathrm{left}}(x_1); (x_1, 0))$. By Lemma 9, we have $p(x_1) < p_{\mathrm{left}}(x_1)$. This implies that either $p(x_1) < p_{\mathrm{right}}(\bar{x})$ or $p(x_1) < \tilde{\ell}(x; C)$. However, if $p(x_1) < p_{\mathrm{right}}(\bar{x})$, as $p(\cdot)$ is non-increasing, we have $p(\bar{x}) < p_{\mathrm{right}}(\bar{x})$, which implies $p(\cdot)$ is infeasible. If $p(x_1) < \tilde{\ell}(x; C)$, this immediately contradicts to $\tilde{\ell}(x; C) \leqslant p(x) \leqslant u(x; C)$. Therefore, such $p(x)$ does not exist.

□

Appendix D: Proof of Theorem 3

The proof is naturally divided into three parts.

D.1. Result 1: $R^\star = \min\{R_{\mathbf{right}}, R_{\mathbf{left}}\}$

First, we show that $R^\star = \min\{R_{\mathrm{right}}, R_{\mathrm{left}}\}$, where Theorems 4 and 5 show that

$$R_{\mathrm{left}} = \min\{\mathrm{CP}_u(p_{\mathrm{left}}(\bar{x}); (\bar{x}, m)), \inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p_{\mathrm{left}}(x); (x, 0))\}.$$

and

$$R_{\mathrm{right}} = \min\{\mathrm{CP}_o(p_{\mathrm{right}}(\bar{x}); (\bar{x}, 0)), \mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\max\{m, \bar{x}\}, m))\}.$$

Let us denote $\widetilde{R_{\mathrm{left}}} = \inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p_{\mathrm{left}}(x); (x, 0))$, and note that by Lemma 7, we have $\mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\bar{x}, m)) \geqslant \mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\max\{m, \bar{x}\}, m))$, where by construction, we have $\mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\bar{x}, m)) = \mathrm{CP}_u(p_{\mathrm{left}}(\bar{x}); (\bar{x}, m))$. Therefore, we have

$$\begin{aligned}
&\min\{R_{\mathrm{right}}, R_{\mathrm{left}}\} \\
&= \min\left\{\mathrm{CP}_o(p_{\mathrm{right}}(\bar{x}); (\bar{x}, 0)), \mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\max\{m, \bar{x}\}, m)), \mathrm{CP}_u(p_{\mathrm{left}}(\bar{x}); (\bar{x}, m)), \inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p_{\mathrm{left}}(x); (x, 0))\right\} \\
&= \min\left\{\mathrm{CP}_o(p_{\mathrm{right}}(\bar{x}); (\bar{x}, 0)), \mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\max\{m, \bar{x}\}, m)), \mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\bar{x}, m)), \inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p_{\mathrm{left}}(x); (x, 0))\right\} \\
&= \min\left\{\mathrm{CP}_o(p_{\mathrm{right}}(\bar{x}); (\bar{x}, 0)), \mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\max\{m, \bar{x}\}, m)), \inf_{x\in[0,\bar{x}]} \mathrm{CP}_o(p_{\mathrm{left}}(x); (x, 0))\right\} = \min\{R_{\mathrm{right}}, \widetilde{R_{\mathrm{left}}}\},
\end{aligned}$$

where the second equality is because $p_{\mathrm{right}}(\bar{x}) = p_{\mathrm{left}}(\bar{x})$, and the third equality is because $\mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\bar{x}, m)) \geqslant \mathrm{CP}_u(p_{\mathrm{right}}(\bar{x}); (\max\{m, \bar{x}\}, m))$.

Therefore, showing $R^\star = \min\{R_{\mathrm{right}}, R_{\mathrm{left}}\}$ is equivalent to show that $R^\star = \min\{R_{\mathrm{right}}, \widetilde{R_{\mathrm{left}}}\}$. As Theorems 4 and 5 show, if we set $p(\bar{x})$ optimally, our $p^\star(\cdot)$, presented in Algorithm 1, achieves an optimal robust ratio.

Therefore, it suffices to show that for any valid $\widehat{p}(x)$ such that $\widehat{p}(\bar{x}) \neq p_{\text{right}}(\bar{x})$, we have $\textsc{robust}(p(\cdot)) \leqslant \textsc{robust}(p^\star(\cdot))$. Here, with a slight abuse of notation, $\textsc{robust}(p(\cdot))$ is the robust ratio of a PLA with PL of $p(\cdot)$. We split the analysis into two cases based on the value of $\widetilde{R_{\text{left}}}$ and R_{right}, where in case 1, we have $R_{\text{right}} \leqslant \widetilde{R_{\text{left}}}$, and in case 2, we have $R_{\text{right}} > \widetilde{R_{\text{left}}}$.

- $R_{\text{right}} \leqslant \widetilde{R_{\text{left}}}$. In this case, $\textsc{robust}(p^\star(\cdot)) = \min\{\widetilde{R_{\text{left}}}, R_{\text{right}}\} = R_{\text{right}}$. By Theorem 4, we know that no PL can achieve a robust ratio greater than R_{right} for $x \in [\bar{x}, \max\{m, \bar{x}\}]$. Therefore, we have $\textsc{robust}(\widehat{p}(\cdot)) \leqslant \textsc{robust}(p^\star(\cdot)) = R_{\text{right}}$, which is the desired result.

- $\widetilde{R_{\text{left}}} < R_{\text{right}}$. As is shown in Theorem 5, by fixing $p^\star(\bar{x}) = p_{\text{right}}(\bar{x})$, no PL can achieve a robust ratio greater than $\widetilde{R_{\text{left}}}$. Then, in this part, we show that if a valid and feasible PL function $\hat{p}(x)$ for $x \in [0, \bar{x}]$ does not have restriction on $\bar{x}$, it can still not achieve a robust ratio greater than $\widetilde{R_{\text{left}}}$. To show this, we define $\widehat{x} \in [0, \bar{x}]$ as such that $\widetilde{l}(\widehat{x}; C) = p_{\text{right}}(\bar{x})$. We start with showing that such $\widehat{x}$ always exists, and then we show that $p_{\text{left}}(\cdot)$ achieves $\widetilde{R_{\text{left}}}$ in $[0, \widehat{x}]$ and no PL function $\hat{p}(x)$ can outperform $\widetilde{R_{\text{left}}}$ in $[0, \widehat{x}]$.

  To show the existence, we use a contradiction argument. Contrary to our claim, suppose that there does not exist any $\widehat{x} \in [0, \bar{x}]$ such that $\widetilde{l}(\widehat{x}; C) = p_{\text{right}}(\bar{x})$. Observe that $p_{\text{right}}(\bar{x})$ is a constant and is greater than $\widetilde{l}(\bar{x}; C)$ by feasibility of $p_{\text{right}}(\cdot)$. Further, note that by Lemma 16, $\widetilde{l}(x; C)$ is non-increasing in x. Then, when $\widehat{x}$ does not exist, we must have $p_{\text{right}}(\bar{x}) > \widetilde{l}(x; C)$ for any $x \in [0, \bar{x}]$. By Theorem 5, in this case, $p_{\text{left}}(x) = p_{\text{right}}(\bar{x})$ is a constant function. (Recall that $p_{\text{left}}(x) = \max\{\widetilde{l}(x; C), p_{\text{right}}(\bar{x})\}$ for any $x \in [0, \bar{x}]$.) By Lemma 18, we have

$$\widetilde{R_{\text{left}}} = \inf_{x \in [0, \bar{x}]} \mathrm{CP}_o(p_{\text{right}}(\bar{x}); (x, 0)) = \mathrm{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)).$$

  However, as $\mathrm{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)) \geqslant \min\{\mathrm{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)), \mathrm{CP}_u(p_{\text{right}}(\bar{x}); (\max\{m, \bar{x}\}, m))\}$, we have $\widetilde{R_{\text{left}}} \geqslant R_{\text{right}}$, which is a contradiction. Therefore, such $\widehat{x}$ exists.

  Then, $p_{\text{left}}(x) = p_{\text{right}}(\bar{x})$ for $x \in [\widehat{x}, \bar{x}]$ and $p_{\text{left}}(x) = \widetilde{l}(x; C)$ for $x \in [0, \widehat{x}]$. By definition of $\widetilde{R_{\text{left}}}$, we have

$$\widetilde{R_{\text{left}}} = \min\{\inf_{x \in [0, \widehat{x}]} \mathrm{CP}_o(\widetilde{l}(x; C); (x, 0)), \inf_{x \in [\widehat{x}, \bar{x}]} \mathrm{CP}_o(p_{\text{right}}(\bar{x}); (x, 0))\}.$$

  By Lemma 18, we have

$$\inf_{x \in [\widehat{x}, \bar{x}]} \mathrm{CP}_o(p_{\text{right}}(\bar{x}); (x, 0)) = \mathrm{CP}_o(p_{\text{right}}(\bar{x}); (\bar{x}, 0)) \geqslant R_{\text{right}}.$$

  Given that $\widetilde{R_{\text{left}}} < R_{\text{right}}$, we have

$$\widetilde{R_{\text{left}}} = \inf_{x \in [0, \widehat{x}]} \mathrm{CP}_o(\widetilde{l}(x; C); (x, 0)).$$

  Finally, we show that no valid and feasible PL $\hat{p}(x)$ can outperform $\widetilde{R_{\text{left}}}$ in $[0, \widehat{x}]$. Suppose that $x_1 = \arg\min_{x \in [0, \widehat{x}]} \mathrm{CP}_o(\widetilde{l}(x; C); (x, 0))$, which means that $\mathrm{CP}_o(\widetilde{l}(x_1; C); (x_1, 0)) = \widetilde{R_{\text{left}}}$. If $\hat{p}(x)$ outperforms $\widetilde{R_{\text{left}}}$, we have

$$\inf_{x \in [0, \widehat{x}]} \mathrm{CP}_o(\hat{p}(x); (x, 0)) > \widetilde{R_{\text{left}}},$$

  which implies that $\mathrm{CP}_o(\hat{p}(x_1); (x_1, 0)) > \widetilde{R_{\text{left}}}$. By Lemma 9, we have $\hat{p}(x_1) < \widetilde{l}(x_1; C)$, which shows that $\hat{p}$ is not a feasible PL function and forms a contradiction.

□

D.2. Result 2: $p^\star$ is an Optimal Solution

Second, it is trivial that Algorithm 1 presents an optimal solution to Problem (C-Pareto-Trans). The reason is by Theorem 4, we have $p_{\text{right}}(x)$ achieves R_{right} for Problem (C-Pareto-right) and by Theorem 5, we have $p_{\text{left}}(x)$ achieves R_{left} for Problem (C-Pareto-left). By Equation (20), we know that $\textsc{robust}(p^\star(\cdot)) = \min\{R_{\text{right}}, R_{\text{left}}\}$, and we just showed that $\min\{R_{\text{right}}, R_{\text{left}}\} = R^\star$. The remaining thing is to show that $p^\star(\cdot)$ is feasible and valid. As we have shown in Theorems 4 and 5 that $p_{\text{right}}(\cdot)$ and $p_{\text{left}}(\cdot)$ are both feasible and valid for any $x \geqslant \bar{x}$ and $x \in [0, \bar{x}]$, respectively. Further, $p_{\text{right}}(\bar{x}) = p_{\text{left}}(\bar{x})$, which means that $p^\star(\cdot)$ is continuous. Therefore, $p^\star(\cdot)$ is feasible.

D.3. Result 3: No Algorithm Can Outperform $p^\star$

Here, we show that a PLA with the PL function of $p^\star(\cdot)$ is an optimal solution to Problem (3). That is, among any online algorithms Π, the aforementioned algorithm maximizes the robust ratio while ensuring its consistent ratio is at least C. Recall that we just showed

$$R^\star = \min\{R_{\text{right}}, R_{\text{left}}\} = \min\{\text{CP}_u(p^\star(\max\{m,\bar{x}\}); (\max\{m,\bar{x}\}, m)), \text{CP}_o(p^\star(\bar{x}), (\bar{x}, 0)), \inf_{x\in[0,\bar{x}]} \text{CP}_o(p^\star(x); (x, 0))\}. \tag{40}$$

Then, we split the proof into three parts, where in each parts, we discuss each term in Equation (40) is the minimum value.

Part 1: $R^\star = CP_u(p^\star(\max\{m,\bar{x}\}); (\max\{m,\bar{x}\}, m))$. Here, we show that no deterministic or randomized algorithm can achieve a robust ratio more than $\text{CP}_u(p^\star(\max\{m,\bar{x}\}); (\max\{m,\bar{x}\}, m))$. We define two (ordered) input sequences: In the first input sequence, I_1, $\bar{x}_u \leqslant \bar{x}$ low-reward requests arrive first, followed with $\underline{h}(\bar{x}_u)$ high-reward requests. In the second input sequence, I_2, $\max\{m,\bar{x}\}$ low-reward requests arrive first, followed m high-reward requests. Before receiving $\bar{x}_u$ low reward requests, any deterministic or randomized algorithm cannot differentiate the two input sequences and has to decide to accept how many low-reward requests in expectation. If there exists a deterministic or randomized algorithm $\mathcal{A}$, which can achieve a consistent ratio of at least C, and a robust ratio higher than $\text{CP}_u(p^\star(\max\{m,\bar{x}\}); (\max\{m,\bar{x}\}, m))$, it should satisfy

$$\frac{\mathbf{E}[\text{Rew}(\mathcal{A}, I_1)]}{\textsc{opt}(I_1)} \geqslant C,$$

Let the expected total amount of high-reward, low-reward requests being accepted by $\mathcal{A}$ be $h(\mathcal{A}, I_1)$, $\ell(\mathcal{A}, I_1)$, respectively. Then,

$$\frac{\mathbf{E}[\text{Rew}(\mathcal{A}, I_1)]}{\textsc{opt}(I_1)} = \frac{h(\mathcal{A}, I_1) r_h + \ell(\mathcal{A}, I_1) r_\ell}{\underline{h}(\bar{x}_u) r_h + \min\{\bar{x}_u, m - \underline{h}(\bar{x}_u)\} r_\ell} \geqslant C.$$

By the definition of $u(\bar{x}_u; C)$, we have $u(\bar{x}_u; C) = \sup\{p : \text{CP}_o(p; (\bar{x}_u, \underline{h}(\bar{x}_u))) = C\}$, and by Equation (6), $\frac{\min\{\underline{h}(\bar{x}_u), m\} r_h + \min\{\bar{x}_u, m - u(\bar{x}_u; C)\} r_\ell}{\min\{\underline{h}(\bar{x}_u), m\} r_h + \min\{\bar{x}_u, (m - \underline{h}(\bar{x}_u))^+\} r_\ell} = C$, and by Lemma 19, we have $\frac{\min\{\underline{h}(\bar{x}_u), m\} r_h + (m - u(\bar{x}_u; C)) r_\ell}{\min\{\underline{h}(\bar{x}_u), m\} r_h + \min\{\bar{x}_u, (m - \underline{h}(\bar{x}_u))^+\} r_\ell} = C$. Therefore, we have

$$\frac{h(\mathcal{A}, I_1) r_h + \ell(\mathcal{A}, I_1) r_\ell}{\min\{\underline{h}(\bar{x}_u), m\} r_h + \min\{\bar{x}_u, (m - \underline{h}(\bar{x}_u))^+\} r_\ell} \geqslant \frac{\min\{\underline{h}(\bar{x}_u), m\} r_h + (m - u(\bar{x}_u; C)) r_\ell}{\min\{\underline{h}(\bar{x}_u), m\} r_h + \min\{\bar{x}_u, (m - \underline{h}(\bar{x}_u))^+\} r_\ell},$$

as $h(\mathcal{A}, I_1) \leqslant \min\{\underline{h}(\bar{x}_u), m\}$, we have $\ell(\mathcal{A}, I_1) \geqslant m - u(\bar{x}_u; C)$, which implies that we should accept at least $m - u(\bar{x}_u; C)$ low-reward requests in expectation.

However, to achieve a robust ratio higher than $\mathrm{CP}_u(p^\star(\max\{m,\bar{x}\});(\max\{m,\bar{x}\},m))$ for sequence I_2, we have

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_2)]}{\mathrm{OPT}(I_2)} > \mathrm{CP}_u(p^\star(\max\{m,\bar{x}\});(\max\{m,\bar{x}\},m)).$$

By Equation (7), we have

$$\begin{aligned}\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_2)]}{\mathrm{OPT}(I_2)} &= \frac{h(\mathcal{A},I_2)r_h+\ell(\mathcal{A},I_2)r_\ell}{mr_h} > \mathrm{CP}_u(p^\star(\max\{m,\bar{x}\});(\max\{m,\bar{x}\},m)) \\ &= \frac{p^\star(\max\{m,\bar{x}\})r_h+(m-p^\star(\max\{m,\bar{x}\}))r_\ell}{mr_h},\end{aligned}$$

as $h(\mathcal{A},I_2)\leqslant(m-\ell(\mathcal{A}),I_2)$, we have $\ell(\mathcal{A},I_2)<m-p^\star(\max\{m,\bar{x}\})$, which implies that we should accept less than $m-p^\star(\max\{m,\bar{x}\})$ low-reward requests in expectation. However, when $R^\star=\mathrm{CP}_u(p^\star(\max\{m,\bar{x}\});(\max\{m,\bar{x}\},m))$, Part 1 of the proof of Theorem 4 shows that $p^\star(\max\{m,\bar{x}\})=p_{\mathrm{right}}(\max\{m,\bar{x}\})=u(\bar{x};C)$. By Equation (13), we have $u(\bar{x};C)=u(\bar{x}_u;C)$. Therefore, under sequence I_2, we have $\ell(\mathcal{A},I_2)<m-u(\bar{x}_u;C)$, which is a contradiction because once we observed $\bar{x}_u$ low-reward requests, we have already accept at least $m-u(\bar{x}_u;C)$ of them in expectation.

□

Part 2: $R^\star=CP_o(p^\star(\bar{x}),(\bar{x},0))$. To show the result, we split the analysis into two sub parts based on the value of $p^\star(\bar{x})$ and $\widetilde{l}(x_{-1};C)$. First, we show that when $l(x_{-1};C)>p^\star(\bar{x})$, no deterministic or randomized algorithm can achieve a robust ratio more than $\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))$. Here, we recall that $x_{-1}=\sup\{x\in[x_H,\bar{x}]:\frac{\partial l(x^-;C)}{\partial x}\leqslant -1\}$. Then, we show the same result when $l(x_{-1};C)\leqslant p^\star(\bar{x})$.

For the case where $l(x_{-1};C)>p^\star(\bar{x})$, we again define two (ordered) input sequences: In the first input sequence, I_1, x_{-1} low-reward requests arrive first, followed with $\bar{h}(x_{-1})$ high-reward requests. In the second input sequence, I_2, $\bar{x}$ low-reward requests arrive first, followed by 0 high-reward requests. Before receiving x_{-1} low reward requests, any deterministic or randomized algorithm cannot differentiate the two input sequences and has to decide to accept how many low-reward requests in expectation. If there exists a deterministic or randomized algorithm $\mathcal{A}$, which can achieve a consistent ratio of at least C, and a robust ratio more than $\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))$, it should satisfy

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_1)]}{\mathrm{OPT}(I_1)}\geqslant C,$$

Let the expected total amount of high-reward, low-reward requests being accepted by $\mathcal{A}$ be $h(\mathcal{A},I_1)$, $\ell(\mathcal{A},I_1)$, respectively. Then,

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_1)]}{\mathrm{OPT}(I_1)}=\frac{h(\mathcal{A},I_1)r_h+\ell(\mathcal{A},I_1)r_\ell}{\bar{h}(x_{-1})r_h+\min\{x_{-1},m-\bar{h}(x_{-1})\}r_\ell}\geqslant C.$$

By Equation (15), we have $\mathrm{CP}_u(l(x_{-1};C);(x_{-1},\bar{h}(x_{-1}))=C$. By Equation (7), we have

$$\mathrm{CP}_u(l(x_{-1};C);(x_{-1},\bar{h}(x_{-1}))=\frac{\max\{l(x_{-1};C),\min\{\bar{h}(x_{-1}),m-x_{-1}\}\}r_h+\min\{x_{-1},m-l(x_{-1};C)\}r_\ell}{\bar{h}(x_{-1})r_h+\min\{\bar{x}_u,m-\bar{h}(x_{-1})\}r_\ell}.$$

By Lemma 19, we have $\max\{l(x_{-1};C),\min\{\bar{h}(x_{-1}),m-x_{-1}\}\}=l(x_{-1};C)$ and $\min\{x_{-1},m-l(x_{-1};C)\}=m-l(x_{-1};C)$. Therefore, we have

$$\mathrm{CP}_u(l(x_{-1};C);(x_{-1},\bar{h}(x_{-1}))=\frac{l(x_{-1};C)r_h+(m-l(x_{-1};C))r_\ell}{\bar{h}(x_{-1})r_h+\min\{\bar{x}_u,m-\bar{h}(x_{-1})\}r_\ell}=C,$$

which implies that

$$\frac{h(\mathcal{A},I_1)r_h+\ell(\mathcal{A},I_1)r_\ell}{\bar{h}(x_{-1})r_h+\min\{\bar{x}_u,m-\bar{h}(x_{-1})\}r_\ell}\geqslant\frac{l(x_{-1};C)r_h+(m-l(x_{-1};C))r_\ell}{\bar{h}(x_{-1})r_h+\min\{\bar{x}_u,m-\bar{h}(x_{-1})\}r_\ell}.$$

As $h(\mathcal{A},I_1)\leqslant m-\ell(\mathcal{A},I_1)$, we have $\ell(\mathcal{A},I_1)\leqslant m-l(x_{-1};C)$, which implies that it should accept no more than $m-l(x_{-1};C)$ low-reward requests in expectation. However, to achieve a robust ratio more than $\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))$ for sequence I_2, we have

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_2)]}{\textsc{opt}(I_2)}>\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0)).$$

Let the expected total amount of low-reward requests being accepted under I_2 by $\mathcal{A}$ be $\ell(\mathcal{A},I_2)$, respectively. Then,

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_1)]}{\textsc{opt}(I_1)}=\frac{\ell(\mathcal{A})}{\bar{x}}>\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0)).$$

By Equation (6), we have

$$\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))=\frac{\min\{\bar{x},m-p^\star(\bar{x})\}}{\bar{x}}.$$

If $\bar{x}\leqslant m-p^\star(\bar{x})$, we have $\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))=1$, which means no algorithm can achieve a higher robust ratio. Otherwise, we have

$$\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))=\frac{m-p^\star(\bar{x})}{\bar{x}},$$

and we have

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_1)]}{\textsc{opt}(I_1)}=\frac{\ell(\mathcal{A},I_2)}{\bar{x}}>\frac{m-p^\star(\bar{x})}{\bar{x}},$$

which implies that $\ell(\mathcal{A})>m-p^\star(\bar{x})$, and we should accept more than $m-p^\star(\bar{x})$ low-reward requests in expectation. However, when $R^\star=\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))$, by the proof of Theorem 4, we have in this case $p^\star(\bar{x})=p_{\mathrm{right}}(\bar{x})=\widetilde{l}(\bar{x};C)$. However, given that $l(x_{-1};C)>p^\star(\bar{x})=\widetilde{l}(\bar{x};C)$, we have $x_{-1}<\bar{x}$. Otherwise, $l(x_{-1};C)=l(\bar{x};C)=\widetilde{l}(\bar{x};C)$. Between x_{-1} and $\bar{x}$, there are at most $\bar{x}-x_{-1}$ low-reward requests arriving, and any algorithm can accept at most $\bar{x}-x_{-1}$ low-reward requests. But we accept no more than $m-l(x_{-1};C)$ low-reward requests in expectation under I_1 and more than $m-p^\star(\bar{x})$ low-reward requests in expectation under I_2, and since $\widetilde{l}(x;C)$ is a line with slope -1 for $x\geqslant x_{-1}$, we have $m-p^\star(\bar{x})-(m-l(x_{-1};C))=\widetilde{l}(x_{-1};C)-\widetilde{l}(\bar{x};C)=\bar{x}-x_{-1}$, which is a contradiction.

Now, let us consider the case where $l(x_{-1};C)\leqslant p^\star(\bar{x})$, where we recall that here by assumption $R^\star=\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))$. By the proof of Theorem 4, when $R_{\mathrm{right}}=\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))$, we have $p^\star(\bar{x})=p_{\mathrm{right}}(\bar{x})=\widetilde{l}(\bar{x};C)$. In addition, we have $l(x_{-1};C)=\widetilde{l}(x_{-1};C)$, and $l(x_{-1};C)\leqslant p^\star(\bar{x})$ is equivalent to $\widetilde{l}(x_{-1};C)\leqslant\widetilde{l}(\bar{x};C)$. Due to $\widetilde{l}(\cdot;C)$ is non-increasing, we have $x_{-1}=\bar{x}$ in this case.

To show the result, we again define two (ordered) input sequences: In the first input sequence, I_1, $\bar{x}$ low-reward requests arrive first, followed by $\bar{h}(\bar{x})$ high-reward requests. In the second input sequence, I_2, $\bar{x}$ low-reward requests arrive first, followed by 0 high-reward requests. If there exists a deterministic or randomized algorithm $\mathcal{A}$, which can achieve a consistent ratio of at least C, and a robust ratio more than $\mathrm{CP}_o(p^\star(\bar{x}),(\bar{x},0))$, it should satisfy

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_1)]}{\textsc{opt}(I_1)}\geqslant C.$$

Let the expected total amount of high-reward, low-reward requests being accepted by $\mathcal{A}$ be $h(\mathcal{A}, I_1)$, $\ell(\mathcal{A}, I_1)$, respectively. Then,

$$\frac{\mathbf{E}[\text{Rew}(\mathcal{A}, I_1)]}{\text{OPT}(I_1)} = \frac{h(\mathcal{A}, I_1)r_h + \ell(\mathcal{A}, I_1)r_\ell}{\bar{h}(\bar{x})r_h + \min\{\bar{x}, m - \bar{h}(\bar{x})\}r_\ell} \geqslant C.$$

By Equation (17), we have $\text{CP}_u(l(\bar{x}; C); (\bar{x}, \bar{h}(\bar{x}))) = C$. By Equation (7),

$$\text{CP}_u(l(\bar{x}; C); (\bar{x}, \bar{h}(\bar{x}))) = \frac{\max\{l(\bar{x}; C), \min\{\bar{h}(\bar{x}), m - \bar{x}\}r_h + \min\{\bar{x}, m - l(\bar{x}; C)\}r_\ell}{\bar{h}(\bar{x})r_h + \min\{\bar{x}, m - \bar{h}(\bar{x})\}r_\ell}.$$

As we have shown $\bar{x} = x_{-1}$ at the beginning of this case, by Lemma 19, we have

$\max\{l(\bar{x}; C), \min\{\bar{h}(\bar{x}), m - \bar{x}\} = l(\bar{x}; C)$ and $\min\{\bar{x}, m - l(\bar{x}; C)\} = m - l(\bar{x}; C)$. Therefore, we have

$$\frac{l(\bar{x}; C)r_h + (m - l(\bar{x}; C))r_\ell}{\bar{h}(\bar{x})r_h + \min\{\bar{x}, m - \bar{h}(\bar{x})\}r_\ell} = C,$$

and this implies

$$\frac{h(\mathcal{A}, I_1)r_h + \ell(\mathcal{A}, I_1)r_\ell}{\bar{h}(\bar{x})r_h + \min\{\bar{x}, m - \bar{h}(\bar{x})\}r_\ell} \geqslant \frac{l(\bar{x}; C)r_h + (m - l(\bar{x}; C))r_\ell}{\bar{h}(\bar{x})r_h + \min\{\bar{x}, m - \bar{h}(\bar{x})\}r_\ell}.$$

As $h(\mathcal{A}, I_1) = m - \ell(\mathcal{A}, I_1)$, we have $\ell(\mathcal{A}, I_1) \leqslant m - l(\bar{x}; C)$, which implies that we should accept no more than $m - l(\bar{x}; C)$ low-reward requests in expectation. However, to achieve a robust ratio more than the $\text{CP}_o(p^\star(\bar{x}), (\bar{x}, 0))$ for sequence I_2, we have

$$\frac{\mathbf{E}[\text{Rew}(\mathcal{A}, I_2)]}{\text{OPT}(I_2)} > \text{CP}_o(p^\star(\bar{x}), (\bar{x}, 0)).$$

This implies that we should accept more than $m - p^\star(\bar{x})$ low-reward requests in expectation, which is shown in the previous case. However, as is shown at the beginning of this case, here, $\bar{x} = x_{-1}$, and $p^\star(\bar{x}) = l(\bar{x}; C)$, which shows that upon receiving $\bar{x}$ low-reward requests, we should accept no more than $m - p^\star(\bar{x})$ low-reward requests and more than $m - p^\star(\bar{x})$ low-reward requests, which is obviously a contradiction.

Part 3: $R^\star = \inf_{x \in [0, \bar{x}]} CP_o(p^\star(x); (x, 0))$. Define $\widehat{x} = \text{argmin}_{x \in [0, \bar{x}]} \text{CP}_o(p^\star(x); (x, 0))$. If $\widehat{x}$ is not unique, we randomly the one with smallest x value. Let us first consider the case where $l(x_{-1}; C) > p^\star(\bar{x})$ and $\widehat{x} \in [x_{-1}, x_{-1} + l(x_{-1}; C) - p^\star(\bar{x})]$. In this case, we replace I_2 in the proof of Part 2 under the case where $l(x_{-1}; C) > p^\star(\bar{x})$ by: we define I_2 as $\widehat{x}$ low-reward requests arrive first, followed 0 high-reward requests, and we can have $p^\star(x)$ is optimal among all algorithms in this case.

For any other cases, we replace I_1, I_2 in the proof of Part 2 under the case where $l(x_{-1}; C) \leqslant p^\star(\bar{x})$ by: we define I_1 as $\widehat{x}$ low-reward requests arrive first, followed $\underline{h}(\widehat{x})$ high-reward requests; we define I_2 as $\widehat{x}$ low-reward requests arrive first, followed 0 high-reward requests, and we can have $p^\star(x)$ is optimal among all algorithms in this case.

Therefore, we have $p^\star(x)$ for $x \in [0, m]$ is optimal among all deterministic and randomized algorithms.

□

D.4. Lemma 7 and its Proof

LEMMA 7. *Given an arbitrary valid PL function $p(\cdot)$, we have for any $x \in [0, \max\{m, \bar{x}\}]$, $CP_u(p(x); (x, m))$ is a non-incresing function in x. That is, for any $x \in [0, \max\{m, \bar{x}\}]$, with $p(x) \leqslant m$,*

$$CP_u(p(m); (\max\{m, \bar{x}\}, m)) \leqslant CP_u(p(x); (x, m)).$$

Proof of Lemma 7 Take any valid PL function $p(x)$ for $x \in [0,m]$, by Equation (7),

$$\begin{aligned}\mathrm{CP}_u(p(x);(x,m)) &= \frac{\max\{p(x),\min\{m,m-x\}\}r_h + \min\{x,m-p(x)\}r_\ell}{mr_h + \min\{x,m-m\}r_\ell} \\ &= \frac{\max\{p(x),m-x\}r_h + \min\{x,m-p(x)\}r_\ell}{mr_h}.\end{aligned}$$

Let $x_1 = \sup\{x : p(x) + x \leqslant m\}$, first, as a valid PL function satisfies that $p'(x) \geqslant -1$ a.e., and the line $x + y = m$ has slope -1, once $x + p(x) \geqslant m$, we have for all $x' > x$, we have $x' + p(x') \geqslant m$. Therefore, for $x \in [0, x_1]$, we have $p(x) + x \leqslant m$, and

$$\mathrm{CP}_u(p(x);(x,m)) = \frac{(m-x)r_h + xr_\ell}{mr_h},$$

which is a monotone decreasing function with x.

For $x \in [x_1, m]$,

$$\mathrm{CP}_u(p(x);(x,m)) = \frac{p(x)r_h + (m-p(x))r_\ell}{mr_h},$$

which is also a non-increasing function with x due to $p(x)$ is non-increasing. Therefore, it is monotone decreasing, and we have for any valid PL function $p(x)$,

$$\mathrm{CP}_u(p(m);(m,m)) \leqslant \mathrm{CP}_u(p(x);(x,m)),$$

for $x \in [0,m]$. If $\bar{x} > m$, by Lemma 10, we have $\mathrm{CP}_u(p(\bar{x});(\bar{x},m)) = \mathrm{CP}_u(p(x);(x,m))$. Therefore, we have

$$\mathrm{CP}_u(p(m);(\max\{m,\bar{x}\},m)) \leqslant \mathrm{CP}_u(p(x);(x,m)),$$

for any $x \in [0, \max\{m,\bar{x}\}]$. □

Appendix E: Proof of Statements in Section 6

In this section, we provide the proof of statements in Section 6.In Section E.1, we prove Proposition 1 and shows the computational complexity and accuracy of the bisection algorithm. In Section E.2, we prove Theorem 7, which provides several properties of an optimal $C^\star(\mathcal{R})$. In Section E.3, we prove Theorem 8, which states that Algorithm 5 returns $C^\star(\mathcal{R})$. Finally, in Section E.4, we prove Theorem 6, which shows that the optimal PL function is optimal among all deterministic and randomized algorithms.

E.1. Proof of Proposition 1

We first show that the feasibility (i.e., determining if for any $x \in [\underline{x}, \bar{x}]$, and a given C, we have $u(x;C) \geqslant \widetilde{l}(x;C)$) check can be performed by a polynomial time algorithm. For any $C \in [0,1]$, checking whether for any $x \in [\underline{x}, \bar{x}]$, $u(x;C) \geqslant \widetilde{l}(x;C)$ is equivalent to checking the following condition

$$\min_{x \in [\underline{x},\bar{x}]} u(x;C) - \widetilde{l}(x;C) \geqslant 0. \tag{41}$$

Notice that by definition, for $x \in [\underline{x}, \underline{x}_u]$, $u(x;C) = m$ and $u(x;C) - \widetilde{l}(x;C)$ is always non-negative and for $x \in [\tilde{x}_\ell, \bar{x}]$, $\widetilde{l}(x;C) = 0$ and $u(x;C) - \widetilde{l}(x;C)$ is always non-negative, where $\tilde{x}_\ell = \inf\{x_{-1} < x < \bar{x} : \widetilde{l}(x;C) = 0\}$. Therefore, in Equation (41), we can ignore these two intervals. For any $x \in [\underline{x}_u, \tilde{x}_\ell]$, by Lemmas 6 and 16, $u(\cdot;C)$ is convex and $\widetilde{l}(\cdot;C)$ is concave, which implies that $u(x;C) - \widetilde{l}(x;C)$ is convex. Therefore, Problem (41) is a convex optimization problem on a compact set, which can be solved by polynomial-time algorithms.

With this result, Algorithm 4 has exactly the same structure with the classical bisection method which is a root-finding method that applies to any continuous function. Here, we make an analogy to the classical bisection method for finding the root (a single zero point). Compare to the classical bisection method for finding a single zero point, we can treat $C^\star(\mathcal{R})$ as the zero point, and treat the feasibility check as whether the value is positive or negative. It is well known that the classical bisection method can return a $x \in [x_0 - \epsilon, x_0 + \epsilon]$ in $O(\log(1/\epsilon))$ time, where x_0 is the zero point. Therefore, we have Algorithm 4 can return a $C_0 \in [C^\star(\mathcal{R}) - \epsilon, C^\star(\mathcal{R}) + \epsilon]$ in $O(\log(1/\epsilon))$ time.

E.2. Proof of Theorem 7

We split the proof into three parts. In part 1, we show that if for any $x \in \mathcal{V}$, we have $\widetilde{l}(x;C) \leqslant u(x;C)$, then such a C is feasible. That is, $C^\star(\mathcal{R}) \leqslant C$. In part 2, we show that a feasible $C = C^\star(\mathcal{R})$ is optimal if and only if there exists $\widehat{x} \in [\underline{x}, \bar{x}]$, such that $\widetilde{l}(\widehat{x}; C^\star(\mathcal{R})) = u(\widehat{x}; C^\star(\mathcal{R}))$. In part 3, we show that such $\widehat{x}$ must belong to $\mathcal{V}$.

Part 1: Feasibility of C. By Lemma 4, C is feasible if and only if $u(x;C) \geqslant \widetilde{l}(x;C)$ for any $x \in [\underline{x}, \bar{x}]$. Here, we show that by only checking $u(x;C)$ and $\widetilde{l}(x;C)$ on $x \in \mathcal{V}$ is enough to check the feasibility of C given that $\mathcal{R}$ is a polyhedron. As $\mathcal{R}$ is a polyhedron, we have both $\bar{h}(\cdot)$ and $\underline{h}(\cdot)$ are piece-wise linear functions. By the first two properties of Lemma 6 and Equation (17), we have both $\widetilde{l}(\cdot;C)$ and $u(\cdot;C)$ are also piece-wise linear functions. We want to show that $\widetilde{l}(x;C) \leqslant u(x;C)$ for any $x \in \mathcal{V}$, implies that $\widetilde{l}(x;C) \leqslant u(x;C)$ for any $x \in [\underline{x}, \bar{x}]$

Suppose that for all $x \in \mathcal{V}$, $\widetilde{l}(x;C) \leqslant u(x;C)$. We take any $x_1 \in [\underline{x}, \bar{x}] - \mathcal{V}$. Let $[\underline{x}_1, \bar{x}_1]$ be the smallest interval contains x_1 such that $\underline{x}_1, \bar{x}_1 \in \mathcal{V}$. As it is the smallest interval, it does not contain any other x-vertices, which implies that for any $x \in [\underline{x}_1, \bar{x}_1]$, both $\widetilde{l}(x;C)$ and $u(x;C)$ are linear. As $\widetilde{l}(\underline{x}_1;C) \leqslant u(\underline{x}_1;C)$ and $\widetilde{l}(\bar{x}_1;C) \leqslant u(\bar{x}_1;C)$, we have $\widetilde{l}(x_1;C) \leqslant u(x_1;C)$ due to linearity and continuity of $\widetilde{l}(x;C)$ and $u(x;C)$.

Part 2: Optimality of C. We first prove the 'if' statement. That is, if there exists $\widehat{x} \in [\underline{x}, \bar{x}]$ and $C \in [\rho, 1]$, such that $\widetilde{l}(\widehat{x};C) = u(\widehat{x};C)$, then we have $C = C^\star(\mathcal{R})$ is optimal. We prove by contradiction. Contrary to our claim, suppose that there exists a feasible $\widehat{C} > C$ under which for any $x \in [\underline{x}, \bar{x}]$, we have $\widetilde{l}(x;\widehat{C}) \leqslant u(x;\widehat{C})$.

By the sixth property of Lemma 6 and Lemma 16, we have $\widetilde{l}(\widehat{x};\widehat{C}) \geqslant \widetilde{l}(\widehat{x};C) = u(\widehat{x};C) \geqslant u(\widehat{x};\widehat{C})$.

First, consider the case where $\widehat{x} \in [\underline{x}, \underline{x}_u]$, where we recall that $\underline{x}_u = \sup\{\underline{x} < x < \bar{x}_u : \mathrm{CP}_o(m;(x, \underline{h}(x))) \geqslant C\}$; that is, we have $u(\widehat{x};C) = m$ if and only if $x \in [\underline{x}, \underline{x}_u]$. We then have

$$\widetilde{l}(\widehat{x};\widehat{C}) \geqslant \widetilde{l}(\widehat{x};C) = u(\widehat{x};C) = m \geqslant u(\widehat{x};\widehat{C}),$$

which implies that $\widetilde{l}(\widehat{x};\widehat{C}) = \widetilde{l}(\widehat{x};C) = m$. However, by definition, we have $\mathrm{CP}_u(\widetilde{l}(\widehat{x};\widehat{C});(\widehat{x}, \bar{h}(\widehat{x}))) \geqslant \widehat{C}$ and $\mathrm{CP}_u(\widetilde{l}(\widehat{x};C);(\widehat{x}, \bar{h}(\widehat{x}))) \geqslant C$, and to be in the under-protecting case, given that $\widetilde{l}(\widehat{x};\widehat{C}) = \widetilde{l}(\widehat{x};C) = m$, we must have $\bar{h}(\widehat{x}) \geqslant m$. By putting $p = m$ and $y = m$ into Equation (7), we have $\mathrm{CP}_u(\widetilde{l}(\widehat{x};\widehat{C});(\widehat{x}, \bar{h}(\widehat{x}))) = \mathrm{CP}_u(\widetilde{l}(\widehat{x};C);(\widehat{x}, \bar{h}(\widehat{x}))) = 1$, which implies that $C = \widehat{C} = 1$, which contradicts to $C >$
$widehatC$.

Now consider the case where $\widehat{x} \in [\underline{x}_u, \bar{x}]$. We have $u(\widehat{x};C) < m$ and $\widetilde{l}(\widehat{x};\widehat{C}) \geqslant \widetilde{l}(\widehat{x};C) = u(\widehat{x};C) > u(\widehat{x};\widehat{C})$, where the last strict inequality is by Lemma 20 which states that $u(\widehat{x};\widehat{C}) = u(\widehat{x};C)$ only happens when both of them equals m. This forms a contradiction because the chain of inequalities implies $\widetilde{l}(\widehat{x};\widehat{C}) > u(\widehat{x};\widehat{C})$.

Next, we prove the 'only if' statement. That is, if $\widetilde{l}(x;C) < u(x;C)$ for all $x \in [\underline{x}, \bar{x}]$, then C is not optimal. That is, there exists $\widehat{C} > C$ such that $\widetilde{l}(x;C) \leqslant u(x;C)$ for all $x \in [\underline{x}, \bar{x}]$

In this case, by the sixth property of Lemma 6 and Lemma 16, for any $x \in [\underline{x}, \bar{x}]$, $\widetilde{l}(x;C)$ is continuously increasing in C and $u(x;C)$ is continuously decreasing in C. Therefore, by continuity of C, there exists $\delta > 0$ such that with $\widehat{C} = C + \delta$, we have $\widetilde{l}(x;\widehat{C}) < u(x;\widehat{C})$ for all $x \in [\underline{x}, \bar{x}]$. By Lemma 4, we have C is feasible and C is not optimal.

Part 3: $\widehat{x} \in \mathcal{V}$**.** . As we defined, $\mathcal{V}$ is the set containing the x value of all vertices of $\mathcal{R}$ (i.e., $\underline{h}(\cdot)$ and $\bar{h}(\cdot)$) and the set $\mathcal{R}_0$, where $\mathcal{R}_0 = \{(x, \underline{h}(x) : x \in [\underline{x}, \bar{x}]\} \cap \{(x,y) : x + y = m\}$. In Lemma 25, we show that all of x-vertices of $u(\cdot;C)$ and $\widetilde{l}(\cdot;C)$ are a subset of $\mathcal{V}$ and the elements of $\mathcal{R}_0$ might be x-vertices of $u(\cdot;C)$. By Lemma 27, we have there exists $\widehat{x} \in \mathcal{V}$ such that $u(\widehat{x};C^\star(\mathcal{R})) \geqslant \widetilde{l}(\widehat{x};C^\star(\mathcal{R}))$, which completes the proof.

□

E.3. Proof of Theorem 8

Given a polyhedron $\mathcal{R}$, let $C^\star(\mathcal{R})$ be the optimal consistent ratio among all PLAs. First, it is easy to see that the computational complexity is $O(|\mathcal{V}|^3)$ because we enumerate at most $|\mathcal{V}|(|\mathcal{V}|-1)/2$ pairs of x-vertices, and recall that

$$C^\star(\mathcal{R}) = \max\{C \in \mathcal{S} : \widetilde{l}(x;C) \leqslant u(x;C) \quad \text{for any } x \in \mathcal{V}\},$$

for each pair of vertices, if $\widetilde{l}(x;C) \leqslant u(x;C)$, for any $x \in \mathcal{V}$, we will add C into the set $\mathcal{S}$. By doing this, for each pair of vertices, we compare the value of $\widetilde{l}(x;C)$ and $u(x;C)$ for $x \in \mathcal{V}$ with at most $|\mathcal{V}|$ complexity. To summarize, the total complexity is bounded by $|\mathcal{V}|(|\mathcal{V}|-1)/2 \cdot |\mathcal{V}|$, which is $O(|\mathcal{V}|^3)$.

Second, Algorithm 5 cannot return an output $C > C^\star(\mathcal{R})$. This is because it will return

$$C^\star(\mathcal{R}) = \max\{C \in \mathcal{S} : \widetilde{l}(x;C) \leqslant u(x;C) \quad \text{for any } x \in \mathcal{V}\}.$$

and by Theorem 7, any C such that $\widetilde{l}(x;C) \leqslant u(x;C)$ for any $x \in \mathcal{V}$ implies $C \leqslant C^\star(\mathcal{R})$.

Finally, we show that by enumerating vertices as in Algorithm 5, we can find $C^\star(\mathcal{R})$. To show this, we need to split the proof into three cases according to the location of $\widehat{x}$, where $\widehat{x}$ is the intersection point of $u(\cdot;C^\star(\mathcal{R}))$ and $\widetilde{l}(\cdot;C^\star(\mathcal{R}))$. By Theorem 7, such $\widehat{x}$ always exists. If there are multiple intersection points, we take the one with the smallest x value. Before we divide into three cases, we highlight that enumerating vertices is only for reduce computational complexity, and all the following statements not related to $\mathcal{V}$ is correct for any general convex set $\mathcal{R}$.

- *Case 1:* $\widehat{x} \leqslant x_H$ In this case, we first show that $\widehat{x} = \bar{x}_u$ and $\bar{x}_u \leqslant x_H$. Second, as H is a vertex, we have $x_H \in \mathcal{V}$, and by Lemma 22, we have $\bar{x}_u \in \mathcal{V}$. We show that by balancing H and $(\bar{x}_u, \underline{h}(\bar{x}_u))$, we obtain a C_1 in Algorithm 5 according to Equation (28).

  For any $C_1 \leqslant C^\star(\mathcal{R})$, by the definition of $\widetilde{l}(\cdot;C_1)$, we have $\widetilde{l}(x;C_1) = \widetilde{l}(x_H;C_1)$ for $x \in [\underline{x}, x_H]$. As $u(\widehat{x};C_1) = \widetilde{l}(\widehat{x};C_1) = \widetilde{l}(x_H;C_1)$ and $u(x;C_1) \geqslant \widetilde{l}(x;C_1)$ for any $x \in [\underline{x}, \bar{x}]$, we have

  $$\widehat{x} \in \operatorname{argmin}_{x \in [\underline{x}, \bar{x}]} u(x;C_1).$$

  By Lemma 21, we have $\widehat{x} \in [\bar{x}_u, x_H]$. As we have defined that if there are multiple intersection points, we take $\widehat{x}$ as the one with the smallest x value. That is, $\widehat{x} = \bar{x}_u$. Moreover, we must have $\bar{x}_u \leqslant x_H$ in this case because otherwise, if $\bar{x}_u > x_H$, we have $u(x;C_1) = m$ for any $x \in [\underline{x}, x_H]$, and as $\widetilde{l}(x;C_1) < m$, we have there does not exist $\widehat{x}$ such that $u(\widehat{x};C_1) = \widetilde{l}(\widehat{x};C_1)$, which is a contradiction.

Next, we show that by balancing H and $(\bar{x}_u, \underline{h}(\bar{x}_u))$ according to Equation (28), we can get C_1. Recall that in Equation (28), given two points x_1 and x_2 (here x_H and $\bar{x}_u$), we find p (here $\widetilde{l}(x_H;C_1)$) such $\mathrm{CP}_u(p;(x_1,\bar{h}(x_1))) = \mathrm{CP}_o(p;(x_2,\underline{h}(x_2))$. This is because, by definition, we have $\mathrm{CP}_u(\widetilde{l}(x_H;C_1);H) = C_1$, and $\mathrm{CP}_o(\widetilde{l}(\bar{x}_u;C_1);(\bar{x}_u,\underline{h}(\bar{x}_u))) = C_1$, which implies that

$$\mathrm{CP}_u(\widetilde{l}(x_H;C_1);H) = \mathrm{CP}_o(\widetilde{l}(\bar{x}_u;C_1);(\bar{x}_u,\underline{h}(\bar{x}_u))) = C_1.$$

Finally, as $\widetilde{l}(\widehat{x};C_1) = \widetilde{l}(x_H;C_1)$, we have $\widetilde{l}(\widehat{x};C_1) = u(\widehat{x};C_1)$, and by Theorem 7, we know C_1 is optimal.

- *Case 2:* $x_H \leqslant \widehat{x} \leqslant x_{-1}$ In this case, by Theorem 7, we have $\widehat{x} \in \mathcal{V}$. We first show that by balancing $(\widehat{x},\bar{h}(\widehat{x}))$ and $(\widehat{x},\underline{h}(\widehat{x}))$, we obtain a C_1 in Algorithm 5 according to Equation (28). Moreover, by definition, we have $u(\widehat{x};C_1) = \widetilde{l}(\widehat{x};C_1)$, which shows the optimality of C_1 and the algorithm can return such a C_1. That is, let

$$\widehat{p} = \{p : \mathrm{CP}_u(p;(\widehat{x},\bar{h}(\widehat{x}))) = \mathrm{CP}_o(p;(\widehat{x},\underline{h}(\widehat{x})))\} \quad \text{and} \quad C_1 = \mathrm{CP}_u(\widehat{p};(\widehat{x},\bar{h}(\widehat{x}))).$$

We will show that $\widehat{p} = \widetilde{l}(\widehat{x};C_1)$.

As $x_H \leqslant \widehat{x} \leqslant x_{-1}$, by definition, we have $\mathrm{CP}_u(\widetilde{l}(\widehat{x};C_1);(\widehat{x},\bar{h}(\widehat{x}))) = C_1$. In addition, by Lemma 23, we have $\widehat{x} \in [\underline{x}_u, \bar{x}_u]$, and by definition, $\mathrm{CP}_o(u(\widehat{x};C_1);(\widehat{x},\underline{h}(\widehat{x}))) = C_1$, which implies that

$$\mathrm{CP}_u(\widetilde{l}(\widehat{x};C_1);(\widehat{x},\bar{h}(\widehat{x}))) = \mathrm{CP}_o(u(\widehat{x};C_1);(\widehat{x},\underline{h}(\widehat{x}))) = C_1.$$

This shows that $\widehat{p} = \widetilde{l}(\widehat{x};C_1) = u(\widehat{x};C_1)$, as desired.

- *Case 3:* $\widehat{x} > x_{-1}$ In this case, by Theorem 7, we have $\widehat{x} \in \mathcal{V}$, and by Lemma 24, we have $x_{-1} \in \mathcal{V}$. First, we show that by balancing $(\widehat{x},\underline{h}(\widehat{x}))$ and $(x_{-1},\bar{h}(x_{-1}))$, we obtain a C_1 in Algorithm 5 according to Equation (29). Finally, we show that under this C_1, $u(\widehat{x};C_1) = \widetilde{l}(\widehat{x};C_1)$, which shows the optimality of C_1 and the algorithm can return such a C_1. Let

$$\widehat{p} = \{p : \mathrm{CP}_u(p;(x_{-1},\bar{h}(x_{-1}))) = \mathrm{CP}_o(p - (\widehat{x} - x_{-1});(\widehat{x},\underline{h}(\widehat{x})))\} \quad \text{and} \quad \mathrm{CP}_u(\widehat{p};(x_{-1},\bar{h}(x_{-1}))) = C_1\,.$$

We will show that $\widehat{p} = u(\widehat{x};C_1) + (\widehat{x} - x_{-1}) = \widetilde{l}(\widehat{x};C_1) + (\widehat{x} - x_{-1}) = \widetilde{l}(x_{-1};C_1)$.

By Lemma 23, we have $\widehat{x} \in [\underline{x}_u, \bar{x}_u]$, and by definition, $\mathrm{CP}_o(u(\widehat{x};C_1);(\widehat{x},\underline{h}(\widehat{x}))) = C_1$. We futher observe that, by Equation (17), we have $\widetilde{l}(x_{-1};C_1) = l(x_{-1};C_1)$, and we have

$$\mathrm{CP}_u(l(x_{-1};C_1);(x_{-1},\bar{h}(x_{-1}))) = \mathrm{CP}_u(\widetilde{l}(x_{-1};C_1);(x_{-1},\bar{h}(x_{-1}))) = C_1,$$

which implies that

$$\mathrm{CP}_u(\widetilde{l}(x_{-1};C_1);(x_{-1},\bar{h}(x_{-1}))) = \mathrm{CP}_o(u(\widehat{x};C_1);(\widehat{x},\underline{h}(\widehat{x}))) = C_1.$$

Finally, as $\widetilde{l}(\widehat{x};C_1)) = \widetilde{l}(x_{-1};C_1) - (\widehat{x} - x_{-1}) = u(\widehat{x};C_1)$, we have $\widehat{p} = u(\widehat{x};C_1) + (\widehat{x} - x_{-1}) = \widetilde{l}(\widehat{x};C_1) + (\widehat{x} - x_{-1}) = \widetilde{l}(x_{-1};C_1)$.

E.4. Proof of Theorem 6

To show that the optimal consistent ratio among all PLAs is optimal among any deterministic and randomized algorithm, we still split the proof into three cases which are the same three cases as in the Proof of Theorem 8. Before presenting the proof, we highlight that although we used many properties from Theorem 7, here in the proof of Theorem 8, we do NOT assume $\mathcal{R}$ is a polyhedron.

Case 1: $\widehat{x} \leqslant x_H$ In this case, in case 1 of the Proof of Theorem 8, we have established that $\widehat{x} = \bar{x}_u$, and $C^\star(\mathcal{R}) = \mathrm{CP}_u(p; H) = \mathrm{CP}_o(p; (\widehat{x}, \underline{h}(\widehat{x})))$, where $p = l(x_H; C^\star(\mathcal{R})) = u(\bar{x}_u; C^\star(\mathcal{R}))$. Our goal here is to show that there does not exist any deterministic or randomized algorithm with a consistent ratio greater than $C^\star(\mathcal{R})$. Let us define two (ordered) input sequences: In the first input sequence, I_1, $\bar{x}_u$ low-reward requests arrive first, followed with $\underline{h}(\bar{x}_u)$ high-reward requests. In the second input sequence, I_2, x_H low-reward requests arrive first, followed y_H high-reward requests. We note that $\widehat{x} = \bar{x}_u < x_H$ and $\underline{h}(\widehat{x}) < y_H$ because H is the highest point of $\mathcal{R}$.

Observe that before receiving $\widehat{x}$ low-reward requests, any algorithm cannot differentiate the two input sequences. Hence, any algorithm should decide how many low-reward requests to accept in expectation among the first $\widehat{x}$ ones.

Next, we prove by contradiction. Suppose that there exists an algorithm $\mathcal{A}$, which can be either deterministic or randomized, and has a consistent ratio larger than $C^\star(\mathcal{R})$. Then, we have

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A}, I_1)]}{\mathrm{OPT}(I_1)} > C^\star(\mathcal{R}),$$

where the expectation is taken on the randomization of the algorithm. Let the expected total amount of high-reward, low-reward requests being accepted by $\mathcal{A}$ be $h(\mathcal{A}, I_1)$, $\ell(\mathcal{A}, I_1)$, respectively. Replace C by $C^\star(\mathcal{R})$ in Part 1 of Section D.3 where we show that any algorithm facing this instance should accept at least $m - u(\bar{x}_u; C)$ low-reward requests, we have $\ell(\mathcal{A}, I_1) > m - u(\bar{x}_u; C^\star(\mathcal{R}))$.

Next, if $\mathcal{A}$ achieves a consistent ratio greater than $C^\star(\mathcal{R})$, it should also satisfy that

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A}, I_2)]}{\mathrm{OPT}(I_2)} > C^\star(\mathcal{R}).$$

Let the expected total amount of high-reward, low-reward requests being accepted by $\mathcal{A}$ be $h(\mathcal{A}, I_2)$, $\ell(\mathcal{A}, I_2)$, respectively. Then,

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A}, I_2)]}{\mathrm{OPT}(I_2)} = \frac{h(\mathcal{A}, I_2) r_h + \ell(\mathcal{A}, I_2) r_\ell}{y_H r_h + \min\{x_H, m - y_H\} r_\ell} > C^\star(\mathcal{R}).$$

Recall that by the definition of $l(\cdot; C^\star(\mathcal{R}))$, we have $\mathrm{CP}_u(l(x_H; C^\star(\mathcal{R})); H) = C^\star(\mathcal{R})$ because $x_H \in [x_H, x_{-1}]$. Then, by Equation (7), we have

$$\mathrm{CP}_u(l(x_H; C^\star(\mathcal{R})); H) = \frac{\max\{l(x_H; C^\star(\mathcal{R})), \min\{y_H, m - x_H\}\} r_h + \min\{x_H, m - l(x_H; C^\star(\mathcal{R}))\} r_\ell}{y_H r_h + \min\{x_H, m - y_H\} r_\ell},$$

and by Lemma 19, we have $\min\{x_H, m - l(x_H; C^\star(\mathcal{R}))\} = m - l(x_H; C^\star(\mathcal{R}))$ and $\max\{l(x_H; C^\star(\mathcal{R})), \min\{y_H, m - x_H\} = l(x_H; C^\star(\mathcal{R}))$. Therefore, we have

$$\frac{l(x_H; C^\star(\mathcal{R})) r_h + (m - l(x_H; C^\star(\mathcal{R}))) r_\ell}{y_H r_h + \min\{x_H, m - y_H\} r_\ell} = C^\star(\mathcal{R}).$$

This implies that

$$\frac{h(\mathcal{A},I_2)r_h+\ell(\mathcal{A},I_2)r_\ell}{y_H r_h+\min\{x_H,m-y_H\}r_\ell}>\frac{l(x_H;C^\star(\mathcal{R}))r_h+(m-l(x_H;C^\star(\mathcal{R})))r_\ell}{y_H r_h+\min\{x_H,m-y_H\}r_\ell}.$$

As $h(\mathcal{A},I_2)\leqslant m-\ell(\mathcal{A},I_2)$, we have $\ell(\mathcal{A},I_2)<m-l(x_H;C^\star(\mathcal{R}))$. Recall that as mentioned at the beginning, we have $l(x_H;C^\star(\mathcal{R}))=u(\bar{x}_u;C^\star(\mathcal{R}))$. Therefore, we have

$$\ell(\mathcal{A},I_2)<m-l(x_H;C^\star(\mathcal{R}))=m-u(\bar{x}_u;C^\star(\mathcal{R}))<\ell(\mathcal{A},I_1),$$

which is a contradiction, because before receiving $\widehat{x}$ low-reward requests, any algorithm cannot differentiate the two input sequences, and this implies that $\ell(\mathcal{A},I_2)\geqslant\ell(\mathcal{A},I_1)$.

Case 2: $x_H\leqslant\widehat{x}\leqslant x_{-1}$. We replace the instances in the proof of case 1 to get the proof for this case. We replace the first instance by: $\widehat{x}$ low-reward requests arrive first, followed with $\underline{h}(\widehat{x})$ high-reward requests. We replace the second instance by: $\widehat{x}$ low-reward requests arrive first, followed with $\bar{h}(\widehat{x})$ high-reward requests. Then, we use similar arguments in case 1 to show the result.

Case 3: $\widehat{x}>x_{-1}$. In this case, as we showed in case 3 of the Proof of Theorem 8, we have that

$$C^\star(\mathcal{R})=\mathrm{CP}_u(p;(x_{-1},\bar{h}(x_{-1})))=\mathrm{CP}_o(p-(\widehat{x}-x_{-1});(\widehat{x},\underline{h}(\widehat{x}))),$$

where $p=l(x_{-1};C^\star(\mathcal{R}))$, and $u(\widehat{x};C^\star(\mathcal{R}))=p-(\widehat{x}-x_{-1})$. Let us define two (ordered) input sequences: In the first input sequence, I_1, x_{-1} low-reward requests arrive first, followed with $\bar{h}(x_{-1})$ high-reward requests. In the second input sequence, I_2, $\widehat{x}$ low-reward requests arrive first, followed $\underline{h}(\widehat{x})$ high-reward requests. In this case, we have $x_{-1}<\widehat{x}$. Before receiving x_{-1} low-reward requests, any deterministic or randomized algorithm cannot differentiate the two input sequences. Hence, any algorithm should decide how many low-reward requests to accept among the first x_{-1} ones in expectation.

Next, we prove by contradiction. Suppose that there exists an algorithm $\mathcal{A}$ which has a consistent ratio larger than $C^\star(\mathcal{R})$. As

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_1)]}{\textsc{opt}(I_1)}>C^\star(\mathcal{R}),$$

we have $\mathcal{A}$ should accept less than $m-\widetilde{l}(x_{-1};C^\star(\mathcal{R}))$ low-reward requests in expectation. Let the expected total amount of high-reward, low-reward requests being accepted by $\mathcal{A}$ be $h(\mathcal{A},I_1)$, $\ell(\mathcal{A},I_1)$, respectively. Replace C by $C^\star(\mathcal{R})$ in Part 2 of Section D.3 which shows that any algorithm facing this instance should accept less than $m-l(x_{-1};C)$ low-reward requests, we have $\ell(\mathcal{A},I_1)<m-l(x_{-1};C^\star(\mathcal{R}))$.

Next, if $\mathcal{A}$ achieves a consistent ratio greater than $C^\star(\mathcal{R})$, it should also satisfy that

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_2)]}{\textsc{opt}(I_2)}>C^\star(\mathcal{R}).$$

Let the expected total amount of high-reward, low-reward requests being accepted by $\mathcal{A}$ be $h(\mathcal{A},I_2)$, $\ell(\mathcal{A},I_2)$, respectively. Then,

$$\frac{\mathbf{E}[\mathrm{Rew}(\mathcal{A},I_2)]}{\textsc{opt}(I_2)}=\frac{h(\mathcal{A},I_2)r_h+\ell(\mathcal{A},I_2)r_\ell}{\underline{h}(\widehat{x})r_h+\min\{\widehat{x},m-\underline{h}(\widehat{x})\}r_\ell}>C^\star(\mathcal{R}).$$

As is mentioned at the beginning of this case, we have $\mathrm{CP}_o(u(\widehat{x};C^\star(\mathcal{R}));(\widehat{x},\underline{h}(\widehat{x})))=C^\star(\mathcal{R})$. By Equation (6), we have

$$\mathrm{CP}_o(u(\widehat{x};C^\star(\mathcal{R}));(\widehat{x},\underline{h}(\widehat{x})))=\frac{\underline{h}(\widehat{x})r_h+\min\{\widehat{x},m-u(\widehat{x};C^\star(\mathcal{R}))\}r_\ell}{\underline{h}(\widehat{x})r_h+\min\{\widehat{x},m-\underline{h}(\widehat{x})\}r_\ell}.$$

Here, we must have $\min\{\widehat{x}, m-u(\widehat{x};C^\star(\mathcal{R}))\} = m-u(\widehat{x};C^\star(\mathcal{R}))$ because otherwise, if $\min\{\widehat{x}, m-u(\widehat{x};C^\star(\mathcal{R}))\} = \widehat{x}$, then due to $u(\widehat{x};C^\star(\mathcal{R})) \geqslant \underline{h}(\widehat{x})$, we have $\min\{\widehat{x}, m-\underline{h}(\widehat{x})\} = \widehat{x}$, and

$\mathrm{CP}_o(u(\widehat{x};C^\star(\mathcal{R}));(\widehat{x},\underline{h}(\widehat{x}))) = 1 \neq C^\star(\mathcal{R})$. Therefore, by taking $\min\{\widehat{x}, m-u(\widehat{x};C^\star(\mathcal{R}))\} = m-u(\widehat{x};C^\star(\mathcal{R}))$, we obtain

$$\frac{\underline{h}(\widehat{x})r_h + (m-u(\widehat{x};C^\star(\mathcal{R})))r_\ell}{\underline{h}(\widehat{x})r_h + \min\{\widehat{x}, m-\underline{h}(\widehat{x})\}r_\ell} = C^\star(\mathcal{R}).$$

This implies that

$$\frac{h(\mathcal{A},I_2)r_h + \ell(\mathcal{A},I_2)r_\ell}{\underline{h}(\widehat{x})r_h + \min\{\widehat{x}, m-\underline{h}(\widehat{x})\}r_\ell} > \frac{\underline{h}(\widehat{x})r_h + (m-u(\widehat{x};C^\star(\mathcal{R})))r_\ell}{\underline{h}(\widehat{x})r_h + \min\{\widehat{x}, m-\underline{h}(\widehat{x})\}r_\ell}.$$

As $h(\mathcal{A},I_2) \leqslant \underline{h}(\widehat{x})$, we have $\ell(\mathcal{A},I_2) > m-u(\widehat{x};C^\star(\mathcal{R}))$. However, in this case $\widehat{x} > x_{-1}$, between x_{-1} and $\widehat{x}$, there are at most $\widehat{x}-x_{-1}$ low-reward requests arriving, and this implies that $\ell(\mathcal{A},I_2) - \ell(\mathcal{A},I_1) \leqslant \widehat{x}-x_{-1}$.

However, as is mentioned at the beginning, we have $u(\widehat{x};C^\star(\mathcal{R})) = l(x_{-1};C^\star(\mathcal{R})) - (\widehat{x}-x_{-1})$. This implies that

$$\begin{aligned}
\ell(\mathcal{A},I_2) - \ell(\mathcal{A},I_1) &> m-u(\widehat{x};C^\star(\mathcal{R})) - (m-l(x_{-1};C^\star(\mathcal{R}))) && (42)\\
&= m-(l(x_{-1};C^\star(\mathcal{R})) - (\widehat{x}-x_{-1})) - (m-l(x_{-1};C^\star(\mathcal{R}))) && (43)\\
&= \widehat{x}-x_{-1}, && (44)
\end{aligned}$$

which is a contradiction.

Appendix F: Preliminary Lemmas

Here, we introduce several small lemmas which will be used in several proofs in this paper.

F.1. Lemma 8 and its Proof

LEMMA 8 **(Not Enough Demand)**. *For any $A=(x,y)$ with $x+y<m$ and $p<y$, we have $CP_u(p;A=(x,y))=1$.*

Proof of Lemma 8 By Equation (7), we have

$$\begin{aligned}
\mathrm{CP}_u(p;A=(x,y)) &= \frac{\max\{p,\min\{y,m-x\}\}r_h + \min\{x,m-p\}r_\ell}{yr_h + \min\{x,m-y\}r_\ell}\\
&= \frac{\max\{p,y\}r_h + \min\{x,m-p\}r_\ell}{yr_h + xr_\ell} = \frac{yr_h + xr_\ell}{yr_h + xr_\ell} = 1\,,
\end{aligned}$$

where in the first equation, we used the assumption that $x+y<m$, and in the second equation, we used the assumption that $x+y<m$ and $p<y$. The last equation is the desired result.

F.2. Lemma 9 and its Proof

LEMMA 9 **(Monotonicity of the Compatible Ratio CP$(p;(x,y))$ w.r.t. p)**. *For any $0 \leqslant y \leqslant p_1 \leqslant p_2$, and any $x \geqslant 0$, we have*

$$CP_o(p_1;(x,y)) \geqslant CP_o(p_2;(x,y)).$$

Further, for any $p_1 \leqslant p_2 \leqslant y$, and any $x \geqslant 0$, we have

$$CP_u(p_2;(x,y)) \geqslant CP_u(p_1;(x,y)).$$

That is, the compatible ratio of a point (x,y) increases when the gap between the protection level p and y, i.e., $|p-y|$, gets smaller. In addition, if $p>m-x$, we have the strong monotonicity for both of $CP_o(p;(x,y))$ and $CP_u(p;(x,y))$.

Proof of Lemma 9 By the definition of CP_o in Equation (6) and the assumption that $y \leqslant p_1 \leqslant p_2$, we have

$$\mathrm{CP}_o(p_1;(x,y)) = \frac{\min\{y,m\}r_h + \min\{x, m-p_1\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell} \geqslant \frac{\min\{y,m\}r_h + \min\{x, m-p_2\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell} = \mathrm{CP}_o(p_2;(x,y))\,.$$

The last inequality is the desired result. Moreover, if $p_2 > p_1 > m-x$, we have the strong monotonicity.

By the definition of CP_u in Equation (7) and the assumption that $p_1 \leqslant p_2 \leqslant y$, we have

$$\begin{aligned}\mathrm{CP}_u(p_2;(x,y)) &= \frac{\max\{p_2, \min\{y,(m-x)^+\}\}r_h + \min\{x, m-p_2\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell} \\ &\geqslant \frac{\max\{p_1, \min\{y,(m-x)^+\}\}r_h + \min\{x, m-p_1\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell} = \mathrm{CP}_u(p_1;(x,y)),\end{aligned}$$

where the inequality holds because for $p_2 \geqslant p_1$, $\max\{p_2, \min\{y, m-x\}\} \geqslant \max\{p_1, \min\{y, m-x\}\}$ and $\min\{x, m-p_2\} \geqslant \min\{x, m-p_1\}$. Moreover, if $p_2 > p_1 > m-x$, we have the strong monotonicity.

F.3. Lemma 10 and its Proof

LEMMA 10. *For any $y > m$ and $p \in [0,m]$, we have*

$$CP_u(p;(x,y)) = CP_u(p;(x,m)).$$

For any $x > m$ and $p \in [0,m]$, we have

$$CP_o(p;(x,y)) = CP_o(p;(m,y)),$$

and

$$CP_u(p;(x,y)) = CP_u(p;(m,y)).$$

Proof of Lemma 10 We first prove that $\mathrm{CP}_u(p;(x,y)) = \mathrm{CP}_u(p;(x,m))$ for any $y > m$. By Equation (7), we have

$$\begin{aligned}\mathrm{CP}_u(p;(x,y)) &= \frac{\max\{p, \min\{y,(m-x)^+\}\}r_h + \min\{x, m-p\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell} \\ &= \frac{\max\{p, \min\{m,(m-x)^+\}\}r_h + \min\{x, m-p\}r_\ell}{mr_h} = \mathrm{CP}_u(p;(x,m)).\end{aligned}$$

Next, we show that $\mathrm{CP}_o(p;(x,y)) = \mathrm{CP}_o(p;(m,y))$ for any $x > m$. By Equation (6), we have

$$\begin{aligned}\mathrm{CP}_o(p;(x,y)) &= \frac{\min\{y,m\}r_h + \min\{x, m-p\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell} \\ &= \frac{\min\{y,m\}r_h + \min\{m, m-p\}r_\ell}{\min\{y,m\}r_h + \min\{m,(m-y)^+\}r_\ell} = \mathrm{CP}_o(p;(m,y)).\end{aligned}$$

By Equation (7), we have

$$\begin{aligned}\mathrm{CP}_u(p;(x,y)) &= \frac{\max\{p, \min\{y,(m-x)^+\}\}r_h + \min\{x, m-p\}r_\ell}{\min\{y,m\}r_h + \min\{x,(m-y)^+\}r_\ell} \\ &= \frac{\max\{p, \min\{y,0\}\}r_h + \min\{m, m-p\}r_\ell}{\min\{y,m\}r_h + \min\{m,(m-y)^+\}r_\ell} = \mathrm{CP}_u(p;(m,y)).\end{aligned}$$

□

F.4. Lemma 11 and its Proof

LEMMA 11. *Recall that* $\underline{h}(x)$*, defined in Equation* (5)*, is the lower envelop of* $\mathcal{R}$*. Let* $\underline{\mathcal{H}}(x) = \min\{\underline{h}(x), m\}$*. If there exists* $x_0 \in [\underline{x}, \bar{x}]$ *such that* $\underline{h}(x_0) < m$*, then either* $\underline{\mathcal{H}}(x) = \underline{h}(x)$ *for all* $x \in [\underline{x}, \bar{x}]$ *or there exists* $\underline{x} \leqslant x_1 < x_2 \leqslant \bar{x}$ *such that* $\underline{\mathcal{H}}(x) = m$ *for* $x \in [\underline{x}, x_1] \cup [x_2, \bar{x}]$ *and* $\underline{\mathcal{H}}(x) = \underline{h}(x)$ *for* $x \in [x_1, x_2]$*. In other words,* $\underline{\mathcal{H}}(x) = \underline{h}(x)$ *on a connected interval.*

Proof of Lemma 11 We will prove this statement by contradiction. Let us suppose that $\underline{\mathcal{H}}(x) = \underline{h}(x)$ on $[x_3, x_4] \cup [x_5, x_6]$. Then, let us randomly pick $x_7 \in (x_4, x_5)$. We recall that $\underline{\mathcal{H}}(x) = \min\{\underline{h}(x), m\}$.

Now, we observe that $\underline{h}(x_4) < m$, $\underline{h}(x_5) < m$, and $\underline{h}(x_7) \geqslant m$. However, this contradicts the fact that $\underline{h}(\cdot)$ is a convex function. Therefore, our initial assumption that $\underline{\mathcal{H}}(x) = \underline{h}(x)$ on $[x_3, x_4] \cup [x_5, x_6]$ is false

□

F.5. Lemma 12 and its Proof

LEMMA 12. *Recall that* $\bar{x}_u$ *is defined in Equation* (14)*, and* $\underline{x}_u = \sup\{\underline{x} < x < \bar{x}_u : CP_o(m; (x, \underline{h}(x))) \geqslant C\}$*. Then, for* $x \in (\underline{x}_u, \bar{x}_u)$*, we have* $\underline{\mathcal{H}}(x) = \underline{h}(x)$*.*

Proof of Lemma 12 By Lemma 11, we have either $\underline{\mathcal{H}}(x) = \underline{h}(x)$ for all $x \in [\underline{x}, \bar{x}]$ or there exists $\underline{x} \leqslant x_1 < x_2 \leqslant \bar{x}$ such that $\underline{\mathcal{H}}(x) = m$ for $x \in [\underline{x}, x_1] \cup [x_2, \bar{x}]$ and $\underline{\mathcal{H}}(x) = \underline{h}(x)$ for $x \in [x_1, x_2]$. In the first case, the statement is trivial.

In the second case, we prove by contradiction. Consider any $x \in (\underline{x}_u, \bar{x}_u)$ and assume that $x \in [\underline{x}, x_1]$. We then have $\underline{\mathcal{H}}(x) = m$, which implies that $\underline{h}(x) \geqslant m$. By Equation (6), we have

$$\mathrm{CP}_o(m; (x, \underline{h}(x))) = 1 \geqslant C,$$

which implies that $x < \underline{x}_u$, which contradicts the fact that $x \in (\underline{x}_u, \bar{x}_u)$.

Now, consider any $x \in (\underline{x}_u, \bar{x}_u)$ and assume that $x \in [x_2, \bar{x}]$. We then have $\underline{\mathcal{H}}(x) = m$, which implies that $\underline{h}(x) \geqslant m$. Recall that

$$\bar{x}_u = \begin{cases} x_L & \text{if } x_L + y_L \geqslant m; \\ \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\} & \text{Otherwise}, \end{cases} \tag{45}$$

In this case, as $\underline{h}(x) < m$ for $x \in [x_1, x_2]$, we have $x_L \in [x_1, x_2]$. Therefore, $\bar{x}_u \neq x_L$ because otherwise, we have $\bar{x}_u = x_L \leqslant x_2$, and there does not exist $x \in (\underline{x}_u, \bar{x}_u)$ and $x \in [x_2, \bar{x}]$. Then, we have $\bar{x}_u = \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\}$. As we assume that $\underline{h}(x) \geqslant m$ for $x \in [x_2, \bar{x}]$, then we have $\underline{\mathcal{H}}'(\bar{x}^-) = 0$ because $\underline{\mathcal{H}}(x) = m$ for $x \in [x_2, \bar{x}]$. Therefore, $\bar{x}_u = \bar{x}$. In addition, as we have $\underline{h}(\bar{x}) \geqslant m$, by Equation (6), we have

$$\mathrm{CP}_o(m; (\bar{x}, \underline{h}(\bar{x}))) = 1 \geqslant C,$$

which implies that $\underline{x}_u = \bar{x}_u$. Therefore, there does not exist $x \in (\underline{x}_u, \bar{x}_u)$, which is a contradiction. □

F.6. Lemma 13 and its Proof

LEMMA 13. *Fix any $C \in (0,1)$. Recall that $\underline{x}_u = \sup\{\underline{x} < x < \bar{x}_u : CP_o(m;(x,\underline{h}(x))) \geqslant C\}$, and $\bar{x}_l = \inf\{x_H < x < \bar{x} : CP_u(0;(x,\overline{h}(x))) \geqslant C\}$. Then, for $x \in [\bar{x}_\ell, \bar{x}]$, we have*

$$CP_u(0;(x,\overline{h}(x))) \geqslant C.$$

Similarly, for any $\underline{x} < x \leqslant \underline{x}_u$, we have

$$CP_o(m;(x,\underline{h}(x))) \geqslant C.$$

Proof of Lemma 13 *Part 1:* We first define the region $\mathcal{R}_1$ such that for any $(x,y) \in \mathcal{R}_1$, $\mathrm{CP}_u(p(x) = 0;(x,y)) = C$. Obviously, $\{x+y \leqslant m\} \not\subset \mathcal{R}_1$ because if $x+y \leqslant m$ and $p(x)=0$, by Lemma 8, $\mathrm{CP}_u(p(x) = 0;(x,y)) = 1$. Next, we find $(x,y) \in \{x+y > m\}$, which belongs to $\mathcal{R}_1$.

We solve the following equation: for $x+y>m$

$$\mathrm{CP}_u(0;(x,y)) = C.$$

Notice that for $x > x_H$, by the definition of H, we have $y < m$, so we can obtain

$$\frac{(m-x)r_h + xr_\ell}{yr_h + (m-y)r_\ell} = C,$$

which is equivalent as

$$(m-x)r_h + xr_\ell = C(yr_h + (m-y)r_\ell). \tag{46}$$

We take derivative on both side of Equation (46), and we get

$$y'(x) = -1/C.$$

Let $\mathcal{L}(x)$ be the line with slope $-1/C$ and across $(\bar{x}_\ell, \bar{h}(\bar{x}_\ell))$. We have $\mathrm{CP}_u(0;(x,\mathcal{L}(x))) = C$. Recall that $\bar{x}_\ell = \inf\{x_H < x < \bar{x} : \mathrm{CP}_u(0;(x,\overline{h}(x))) \geqslant C\}$. Then, for any $x < \bar{x}_\ell$, we have $\mathrm{CP}_u(0;(x,\bar{h}(x))) < C$. By Lemma 2, we have $\bar{h}(x) > \mathcal{L}(x)$, and by Lemma 26, we have for $x > \bar{x}_\ell$, $\bar{h}(x) < \mathcal{L}(x)$. By Lemma 2 again, we have $\mathrm{CP}_u(0;(x,\bar{h}(x)) \geqslant C$.

Part 2: We define the region $\mathcal{R}_2$ such that for any $(x,y) \in \mathcal{R}_2$, $\mathrm{CP}_o(p(x) = m;(x,y)) = C$. For $(x,y) \in \{x+y \geqslant m\}$, we solve

$$\mathrm{CP}_o(m;(x,y)) = C.$$

Notice that we must have $y < m$ because otherwise, $\mathrm{CP}_o(m;(x,y)) = 1 \neq C$. Then, we can obtain

$$\frac{yr_h}{yr_h + (m-y)r_\ell} = C,$$

and we can get $y = \frac{Cr_\ell}{r_h - C(r_h - r_\ell)} m$. Therefore, in the area $\{x+y \geqslant m\}$, $\mathcal{R}_2$ is a line with slope 0.

Next, we explore the part $(x,y) \in \{x+y < m\}$, this time

$$\mathrm{CP}_o(m;(x,y)) = C,$$

implies that

$$\frac{yr_h}{yr_h + xr_\ell} = C,$$

which is equivalent as

$$y(1-C)r_h = Cr_\ell x. \tag{47}$$

Then, we take derivative on both side of Equation (47), and we get

$$y'(x) = \frac{Cr_\ell}{(1-C)r_h},$$

which means that, in the area $\{x+y<m\}$, $\mathcal{R}_2$ is a line with positive slope $\frac{Cr_\ell}{(1-C)r_h}$, for $x < m - \frac{Cr_\ell}{r_h - C(r_h - r_\ell)}m$, and across $(m - \frac{Cr_\ell}{r_h - C(r_h-r_\ell)}m, \frac{Cr_\ell}{r_h - C(r_h-r_\ell)}m)$.

Combined with the results above, we start to prove this lemma. Let $\mathcal{L}(x)$ be a piecewise linear line segment such that $(x, \mathcal{L}(x)) \in \mathcal{R}_2$ for all $x \in [0, m]$. That is, for $x \in [0, m - \frac{Cr_\ell}{r_h - C(r_h-r_\ell)}m]$, $\mathcal{L}(x)$ has slope $\frac{Cr_\ell}{(1-C)r_h}$ and across $(m - \frac{Cr_\ell}{r_h - C(r_h-r_\ell)}m, \frac{Cr_\ell}{r_h - C(r_h-r_\ell)}m)$. For $x \in (m - \frac{Cr_\ell}{r_h - C(r_h-r_\ell)}m, m]$, $\mathcal{L}(x) = \frac{Cr_\ell}{r_h - C(r_h-r_\ell)}m$. Moreover, for any $x \in [0, m]$, we have $\mathrm{CP}_o(m; (x, \mathcal{L}(x))) = C$.

Recall that $\underline{x}_u = \sup\{\underline{x} < x < \bar{x}_u : \mathrm{CP}_o(m; (x, \underline{h}(x))) \geqslant C\}$. We first show the case where $\underline{x}_u \leqslant x_L$. As $\mathcal{L}$ is an non-decreasing function, and for $x \leqslant x_L$, $\underline{h}(\cdot)$ is a decreasing function. Therefore, as we have $\underline{h}(\underline{x}_u) = \mathcal{L}(\underline{x}_u)$, we have $\underline{h}(x) > \mathcal{L}(x)$ for any $x \in [\underline{x}, \underline{x}_u]$. Therefore, by Lemma 2, we have for any $x \in [\underline{x}, \underline{x}_u]$,

$$\mathrm{CP}_o(m; (x, \underline{h}(x))) \geqslant C.$$

Second, we show the case where $\underline{x}_u > x_L$. In this case, $\underline{h}(\cdot)$ is decreasing for $x < x_L$, and increasing for $x > x_L$. If there exists $x_1 \in [\underline{x}, \underline{x}_u]$ such that $\mathrm{CP}_o(m; (x_1, \underline{h}(x_1))) < C$, then, by Lemma 2, this implies that $\underline{h}(x_1) < \mathcal{L}(x_1)$. As the lowest point L is below the line $\mathcal{L}$ and $\underline{h}(x)$ is convex, we have there are at most two intersections of $\underline{h}(x)$ and $\mathcal{L}$. One is in the left of L and the other is $(\underline{x}_u, \underline{h}(\underline{x}_u))$. As $\underline{h}(x)$ is convex increasing for $x \in [x_L, \bar{x}_u]$ and given that $\mathcal{L}(x)$ is concave increasing for $x \in [x_L, \bar{x}_u]$, we have

$$\underline{h}'(\underline{x}_u^+) \geqslant \underline{h}'(\underline{x}_u^-) \geqslant \mathcal{L}(\underline{x}_u^+),$$

which implies that there exists $\epsilon > 0$, such that $\underline{h}(\underline{x}_u + \epsilon) \geqslant \mathcal{L}(\underline{x}_u + \epsilon)$. By Lemma 2, we have

$$\mathrm{CP}_o(m; (\underline{x}_u + \epsilon, \underline{h}(\underline{x}_u + \epsilon))) \geqslant \mathrm{CP}_o(m; (\underline{x}_u + \epsilon, \langle(\underline{x}_u + \epsilon))) = C,$$

which contradicts to the definition of $\underline{x}_u$. Therefore, for any $x \in [\underline{x}, \underline{x}_u]$,

$$\mathrm{CP}_o(m; (x, \underline{h}(x))) \geqslant C.$$

□

F.7. Lemma 14 and its Proof

LEMMA 14. *Fix any $C > 0$. For any $x \in [\underline{x}_u, \bar{x}_u]$, $CP_o(p; (x, \underline{h}(x))) = C$ has a solution $p \geqslant \min\{\underline{h}(x), m\}$. For any $x \in [x_H, \bar{x}_\ell]$, $CP_u(p; (x, \bar{h}(x))) = C$ has a solution $p \leqslant \min\{\bar{h}(x), m\}$.*

Proof of Lemma 14 Fix any $C > 0$. Take $p = \min\{\underline{h}(x), m\}$, by Equation (6), we have $\mathrm{CP}_o(p; (x, \underline{h}(x))) = 1$. If $\underline{h}(x) \geqslant m$, then we claim that $x \notin [\underline{x}_u, \bar{x}_u]$. This is because take $p = m$, by Lemma 10, we have $\mathrm{CP}_o(m; (x, \underline{h}(x))) = \mathrm{CP}_o(m; (x, m)) = 1$, which contradicts the definition of $\underline{x}_u$. Otherwise, if $\underline{h}(x) < m$, we have $\mathrm{CP}_o(m; (x, \underline{h}(x))) < C$ and $\mathrm{CP}_o(\underline{h}(x); (x, \underline{h}(x))) = 1$. Therefore, by mean value theorem, we have there exists $p_1 \in [\underline{h}(x), m]$ such that $\mathrm{CP}_o(p_1; (x, \underline{h}(x))) = C$.

Similarly, take $p = \min\{\bar{h}(x), m\}$, by Equation (7), we have $\mathrm{CP}_u(p;(x,\bar{h}(x))) = 1$. Take $p = 0$, by the definition of $\underline{x}_\ell = \sup\{\underline{x} < x < x_H : \mathrm{CP}_u(0;(x,\bar{h}(x))) = C\}$, $\bar{x}_l = \inf\{x_H < x < \bar{x} : \mathrm{CP}_u(0;(x,\bar{h}(x))) \geqslant C\}$, we have $\mathrm{CP}_u(p;(x,\bar{h}(x))) < C$. From Equation (7), we can simply check that $\mathrm{CP}_u(p;(x,\bar{h}(x)))$ is continuous in p for any x. Therefore, by mean value theorem, we have there exists $p_1 \in [0, \min\{\bar{h}(x), m\}]$ such that $\mathrm{CP}_u(p_1;(x,\bar{h}(x))) = C$.

F.8. Lemma 15 and its Proof

LEMMA 15. *Recall that*

$$\bar{x}_u = \begin{cases} x_L & \text{if } x_L + y_L \geqslant m; \\ \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\} & \text{Otherwise.} \end{cases} \tag{48}$$

Then, for any $x \in [\bar{x}_u, \bar{x}]$, we have $CP_o(u(x;C);(x,\underline{h}(x))) \geqslant C$.

Proof of Lemma 15 Define $\hat{u}(x;C)$ as the original $u(x;C)$ without forcing to be constant after $\bar{x}_u$. More precisely, for any $C \in [0,1]$, we define

$$\hat{u}(x;C) = \sup\left\{p \in [0,m] : \mathrm{CP}_o(p;(x,\underline{h}(x))) = C\right\} \qquad x \in [\underline{x}_u, \bar{x}] \tag{49}$$

while we set $\hat{u}(x;C) = m$ for any $x \in [0, \underline{x}_u]$. Note while $u(x;C) \neq \hat{u}(x;C)$ for any $x \in (\bar{x}_u, \bar{x}]$, we have $u(x;C) = \hat{u}(x;C)$ for any $x \in [0, \bar{x}_u]$. Then, we can check that for any $x \in [\underline{x}_u, \bar{x}]$, Equation (50) is satisfied, which means for any $x \in [\underline{x}_u, \bar{x})$ and $C \in [0,1]$, when $\underline{\mathcal{H}}'(x)$ exists, we have

$$\frac{\partial \hat{u}(x;C)}{\partial x} = \begin{cases} ((1-C)\frac{r_h}{r_\ell} + C)\underline{\mathcal{H}}'(x) & \text{if } x + \underline{\mathcal{H}}(x) \geqslant m; \\ (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x) - C & \text{if } x + \underline{\mathcal{H}}(x) < m\,, \end{cases} \tag{50}$$

where $\underline{\mathcal{H}}(x) = \min\{\underline{h}(x), m\}$. By the definition of $\bar{x}_u$, as $\frac{\partial \hat{u}(x;C)}{\partial x}|_{x=\bar{x}_u^-} < 0$ and $\frac{\partial \hat{u}(x;C)}{\partial x}|_{\bar{x}_u^+} \geqslant 0$, we obtain

$$\inf_{x \in [\underline{x}, \bar{x}]} \hat{u}(x;C) = \hat{u}(\bar{x}_u;C),$$

which is because the derivative of $\hat{u}(x;C)$ is linear, and $\frac{\partial \hat{u}(x;C)}{\partial x}|_{x=\bar{x}_u^-} < 0$ and $\frac{\partial \hat{u}(x;C)}{\partial x}|_{\bar{x}_u^+} \geqslant 0$ implies that $\frac{\partial \hat{u}(x;C)}{\partial x} \geqslant 0$ for all $x \geqslant \bar{x}_u$. Therefore, by any $x \geqslant \bar{x}_u$, $\hat{u}(x;C) \geqslant u(x;C)$ as we force $u(x;C)$ to be a constant value of $u(\bar{x}_u;C)$. The definition of $\hat{u}(\cdot;C)$ implies that $\mathrm{CP}_o(\hat{u}(x;C);(x,\underline{h}(x))) = C$. By Lemma 9, for $x \geqslant \bar{x}_u$, as $\hat{u}(x;C) \geqslant u(x;C)$, we have

$$\mathrm{CP}_o(u(x;C);(x,\underline{h}(x))) \geqslant \mathrm{CP}_o(\hat{u}(x;C);(x,\underline{h}(x))) \geqslant C.$$

F.9. Lemma 16 and its Proof

LEMMA 16. *For any $C \in [0,1]$, $\widetilde{l}(x;C)$ is decreasing and is concave for $x \in [\underline{x}, \tilde{x}_\ell]$, where $\tilde{x}_\ell = \inf\{x_{-1} < x < \bar{x} : \widetilde{l}(x;C) = 0\}$. For any $x \in [\underline{x}, \bar{x}]$, $\widetilde{l}(x;C)$ is continuously increasing in C.*

Proof of Lemma 16 By Equation (17), as $l(x;C)$ is decreasing for any $C \in [0,1]$, we have $\widetilde{l}(x;C)$ is also decreasing for any $C \in [0,1]$. Next, by Lemma 6, we have $l(x;C)$ is concave for $x \in [\underline{x}, \bar{x}_\ell]$. Therefore, $\frac{\partial l(x^+;C)}{\partial x}$ is non-increasing. As $\widetilde{l}(x;C) = l(x;C)$ for $x \in [\underline{x}, x_{-1}]$, we have $\frac{\partial \widetilde{l}(x^+;C)}{\partial x}$ is non-increasing for $x \in [\underline{x}, x_{-1}]$. By the definition of x_{-1}, we have $\frac{\partial \widetilde{l}(x_{-1}^-;C)}{\partial x} \geqslant \frac{\partial \widetilde{l}(x_{-1}^+;C)}{\partial x} = -1$, and for $x_{-1} < x \leqslant \tilde{x}$, we have $\widetilde{l}(x;C)$ is a line with slope -1 and $\frac{\partial \widetilde{l}(x^+;C)}{\partial x} = -1$, which is non-increasing. Therefore, $\widetilde{l}(x;C)$ is concave for $x \in [\underline{x}, \tilde{x}_\ell]$.

Fix any $x_1 \in [\underline{x}, \bar{x}]$, by Lemma 6, we know that $l(x_1;C)$ is a continuous increasing function in C. If $x_1 \in [\underline{x}, x_{-1}]$, as $\widetilde{l}(x_1;C) = l(x_1;C)$, we have $\widetilde{l}(x_1;C)$ is also continuously increasing in C. Otherwise, define $\mathcal{L}(x;C) = (-x + x_{-1} + l(x_{-1};C), 0)^+$. Then, $\widetilde{l}(x_1;C) = \max\{\mathcal{L}(x_1;C), l(x_1;C)\}$. As both $\mathcal{L}(x_1;C)$ and $l(x_1;C)$ are increasing continuous in C, we have $\widetilde{l}(x_1;C)$ is increasing and continuous in C.

F.10. Lemma 17 and its Proof

LEMMA 17. *Define $p_b(x)$ as a function which balances the compatible ratio of two points $(x, \underline{h}(x))$ and $(x, \bar{h}(x))$ for any $x \in [\underline{x}, \bar{x}]$; that is,*

$$CP_o(p_b(x); (x, \underline{h}(x))) = CP_u(p_b(x); (x, \bar{h}(x)))\,.$$

Then, $p_b(x)$ exists for any $x \in [\underline{x}, \bar{x}]$, and

$$CP_o(p_b(x); (x, \underline{h}(x))) = CP_u(p_b(x); (x, \bar{h}(x))) \geqslant C^\star(\mathcal{R})\,,$$

where $C^\star(\mathcal{R})$ is the maximum consistent ratio among all PLAs given that the ML advice $\mathcal{R}$.

Proof of Lemma 17 Fix any $x \in [\underline{x}, \bar{x}]$, define $f(p) = \mathrm{CP}_o(p; (x, \underline{h}(x))) - \mathrm{CP}_u(p; (x, \bar{h}(x)))$. As $\mathrm{CP}_o(p; (x, \underline{h}(x)))$ and $\mathrm{CP}_u(p; (x, \bar{h}(x)))$ are both continuous in p, we have $f(p)$ is continuous in p. Next, take $p = \min\{m, \underline{h}(x)\}$, then $\mathrm{CP}_o(p; (x, \underline{h}(x))) = 1$ and $\mathrm{CP}_u(p; (x, \bar{h}(x))) \leqslant 1$, and hence we have $f(\min\{m, \underline{h}(x)\}) \geqslant 0$. Take $p = \min\{m, \bar{h}(x)\}$, then $\mathrm{CP}_u(p; (x, \bar{h}(x))) = 1$ and $\mathrm{CP}_o(p; (x, \underline{h}(x))) \leqslant 1$, and hence we have $f(\min\{m, \bar{h}(x)\}) \leqslant 0$. Therefore, by mean value theorem, there must exist $p \in [\min\{m, \underline{h}(x)\}, \min\{m, \bar{h}(x)\}]$ such that $f(p) = 0$, i.e. $\mathrm{CP}_o(p_b(x); (x, \underline{h}(x))) = \mathrm{CP}_u(p_b(x); (x, \bar{h}(x)))$.

Next, we prove that $\mathrm{CP}_o(p_b(x); (x, \underline{h}(x))) = \mathrm{CP}_u(p_b(x); (x, \bar{h}(x))) \geqslant C^\star(\mathcal{R})$ for any $x \in [\underline{x}, \bar{x}]$. We prove by contradiction. Suppose that there exists $x_1 \in [\underline{x}, \bar{x}]$ such that $\mathrm{CP}_o(p_b(x); (x, \underline{h}(x))) = \mathrm{CP}_u(p_b(x); (x, \bar{h}(x))) < C^\star(\mathcal{R})$, then, if we set any $p < p_b(x)$, by Lemma 9, we have

$$\mathrm{CP}_u(p; (x, \bar{h}(x))) < \mathrm{CP}_u(p_b(x); (x, \bar{h}(x))) < C^\star(\mathcal{R})\,.$$

Similarly, if we set any $p > p_b(x)$, by Lemma 9, we have

$$\mathrm{CP}_o(p; (x, \underline{h}(x))) < \mathrm{CP}_o(p_b(x); (x, \underline{h}(x))) < C^\star(\mathcal{R})\,.$$

Therefore, there does not exist a PL function such that the consistent ratio is $C^\star(\mathcal{R})$, which is a contradiction. This is because here $C^\star(\mathcal{R})$ in is defined as an upper bound on the consistent ratio of any PLA.

F.11. Lemma 18 and its Proof

LEMMA 18. *For any fixed PL function p, we have $CP_o(p; (x, 0))$ is a decreasing function in x. That is, for any $x_1 \leqslant x_2$, we have*

$$CP_o(p; (x_1, 0)) \geqslant CP_o(p; (x_2, 0)).$$

Proof of Lemma 18 If $x_1 \leqslant x_2 \leqslant m$, by Equation (6), we have

$$\mathrm{CP}_o(p; A = (x, 0)) = \frac{0 \cdot r_h + \min\{x, m-p\} r_\ell}{0 \cdot r_h + \min\{x, m-0\} r_\ell} = \frac{\min\{x, m-p\}}{x}.$$

Take any $x_1 \leqslant x_2$, if $\min\{x_2, m-p\} = x_2$, then $\min\{x_1, m-p\} = x_1$, and we have

$$\frac{\min\{x_1, m-p\}}{x_1} = \frac{\min\{x_2, m-p\}}{x_2} = 1.$$

If $\min\{x_2, m-p\} = m-p$ and $\min\{x_1, m-p\} = m-p$, we have

$$\frac{\min\{x_2, m-p\}}{x_2} = \frac{m-p}{x_2} \leqslant \frac{m-p}{x_1} = \frac{\min\{x_1, m-p\}}{x_1}.$$

If $\min\{x_2, m-p\} = m-p$ and $\min\{x_1, m-p\} = x_1$, we have

$$\frac{\min\{x_2, m-p\}}{x_2} = \frac{m-p}{x_2} \leqslant 1 = \frac{x_1}{x_1} = \frac{\min\{x_1, m-p\}}{x_1}.$$

Therefore, we have for any $x_1 \leqslant x_2 \leqslant m$, we have

$$\mathrm{CP}_o(p;(x_1,0)) \geqslant \mathrm{CP}_o(p;(x_2,0)).$$

Next, for $x_1 \leqslant m \leqslant x_2$, by Lemma 10, we have

$$\mathrm{CP}_o(p;(x_1,0)) \geqslant \mathrm{CP}_o(p;(m,0)) = \mathrm{CP}_o(p;(x_2,0)).$$

For $m \leqslant x_1 \leqslant x_2$, again, by Lemma 10, we have

$$\mathrm{CP}_o(p;(m,0)) = \mathrm{CP}_o(p;(x_1,0)) = \mathrm{CP}_o(p;(x_2,0)).$$

F.12. Lemma 19 and its Proof

LEMMA 19. *For any $C \in (0,1)$. Let $\bar{x}_u$ be defined in Equation (14), we have*

$$\min\{\bar{x}_u, m-u(\bar{x}_u;C)\} = m-u(\bar{x}_u;C).$$

Recall that H is the point in set $\bar{\mathcal{R}}$ that has the highest low-reward demand, where $\bar{\mathcal{R}} = \{(x,y)\in\mathcal{R} : y = \sup_{(x',y')\in\mathcal{R}} \min\{y',m\}\}$ Further, we have

$$\min\{x_H, m-l(x_H;C^\star(\mathcal{R}))\} = m-l(x_H;C^\star(\mathcal{R})).$$

Let $x_{-1} = \sup\{x\in[x_H,\bar{x}] : \frac{\partial l(x^-;C)}{\partial x} \leqslant -1\}$. We have

$$x_{-1} \geqslant m - l(x_{-1};C)$$

.

Proof of Lemma 19 We prove by contradiction. If $\min\{\bar{x}_u, m-u(\bar{x}_u;C)\} = \bar{x}_u$, then by Equation (6), we have

$$\mathrm{CP}_o(u(\bar{x}_u;C);(\bar{x}_u,\underline{h}(\bar{x}_u))) = \frac{\min\{\underline{h}(\bar{x}_u),m\}r_h + \min\{m-u(\bar{x}_u;C),\bar{x}_u,\}r_\ell}{\min\{\underline{h}(\bar{x}_u),m\}r_h + \min\{(m-\underline{h}(\bar{x}_u))^+,\bar{x}_u\}r_\ell}.$$

Given that $\min\{\bar{x}_u, m-u(\bar{x}_u;C)\} = \bar{x}_u$, as $u(\bar{x}_u;C) \geqslant \min\{\underline{h}(\bar{x}_u),m\}$, we have $\min\{(m-\underline{h}(\bar{x}_u))^+,\bar{x}_u\} = \bar{x}_u$, and we have

$$\mathrm{CP}_o(u(\bar{x}_u;C);(\bar{x}_u,\underline{h}(\bar{x}_u))) = \frac{\min\{\underline{h}(\bar{x}_u),m\}r_h + \bar{x}_u r_\ell}{\min\{\underline{h}(\bar{x}_u),m\}r_h + \bar{x}_u r_\ell} = 1 > C,$$

which contradicts to the definition of $u(\cdot;C)$.

Next, to show the second statement, We still prove by contradiction. If $\min\{x_H, m-l(x_H;C^\star(\mathcal{R}))\} = x_H$, then by Equation (7), we have

$$\mathrm{CP}_u(l(x_H;C^\star(\mathcal{R}));H) = \frac{\max\{l(x_H;C^\star(\mathcal{R})),\min\{y_H,(m-x_H)^+\}\}r_h + \min\{x_H,m-l(x_H;C^\star(\mathcal{R}))\}r_\ell}{y_H r_h + \min\{x_H, m-y_H\}r_\ell}.$$

Given that $\min\{x_H, m-l(x_H;C^\star(\mathcal{R}))\} = x_H$, we have

$$\mathrm{CP}_u(l(x_H;C^\star(\mathcal{R}));H) = \frac{\min\{y_H,(m-x_H)^+\}r_h + x_H r_\ell}{y_H r_h + \min\{x_H, m-y_H\}r_\ell} = \mathrm{CP}_u(0;H),$$

which implies that $x_H \geqslant \bar{x}_\ell$, and this contradicts to the definition of $\bar{x}_\ell$ since $x_H \leqslant x_{-1} < \bar{x}_\ell$.

Finally, we show that $x_{-1} \geqslant m - l(x_{-1}; C)$. We still prove by contradiction. Suppose that $x_{-1} < m - l(x_{-1}; C)$. By Equation (7), we have

$$\mathrm{CP}_u(l(x_{-1};C);(x_{-1},\bar{h}(x_{-1}))) = \frac{\max\{l(x_{-1};C), \min\{\bar{h}(x_{-1}), (m-x_{-1})^+\}\}r_h + \min\{x_{-1}, m-l(x_{-1};C)\}r_\ell}{\min\{\bar{h}(x_{-1}), m\}r_h + \min\{x_{-1}, (m-\bar{h}(x_{-1}))^+\}r_\ell}.$$

If we have $x_{-1} < m - l(x_{-1}; C)$, then we have

$$C = \mathrm{CP}_u(l(x_{-1};C);(x_{-1},\bar{h}(x_{-1}))) = \frac{\min\{\bar{h}(x_{-1}), (m-x_{-1})^+\}r_h + x_{-1}r_\ell}{\min\{\bar{h}(x_{-1}), m\}r_h + \min\{x_{-1}, (m-\bar{h}(x_{-1}))^+\}r_\ell} = \mathrm{CP}_u(0;(x_{-1},\bar{h}(x_{-1}))),$$

which implies that $x_{-1} \geqslant \bar{x}_\ell$, which is a contradiction.

F.13. Lemma 20 and its Proof

Lemma 20. *Let $C^\star(\mathcal{R})$ be the optimal consistent ratio of $\mathcal{R}$. Take any $C > C^\star(\mathcal{R})$, then $u(\hat{x}; C) = u(\hat{x}; C^\star(\mathcal{R}))$ only if both of them are equal to m.*

Proof of Lemma 20 If $u(\hat{x}; C^\star(\mathcal{R})) < m$, then by definition of $\underline{x}_u$, we have $\hat{x} \in [\underline{x}_u, \bar{x}]$. For $\hat{x} \in [\underline{x}_u, \bar{x}_u]$, by Equation (13), we have $u(\hat{x}; C^\star(\mathcal{R})) = \sup\{p \in [0,m] : \mathrm{CP}_o(p; (\hat{x}, \underline{h}(\hat{x}))) = C^\star(\mathcal{R})\}$. For $C > C^\star(\mathcal{R})$, if $u(\hat{x}; C) = u(\hat{x}; C^\star(\mathcal{R}))$, then we have

$$\begin{aligned}\sup\{p \in [0,m] : \mathrm{CP}_o(p;(\hat{x},\underline{h}(\hat{x}))) = C\} &= u(\hat{x};C) = u(\hat{x};C^\star(\mathcal{R}))\\ &= \sup\{p \in [0,m] : \mathrm{CP}_o(p;(\hat{x},\underline{h}(\hat{x}))) = C^\star(\mathcal{R})\},\end{aligned}$$

which implies that $\mathrm{CP}_o(u(\hat{x};C);(x,\underline{h}(\hat{x}))) = C$ and $\mathrm{CP}_o(u(\hat{x};C);(x,\underline{h}(\hat{x}))) = C^\star(\mathcal{R})$, which is a contradiction. For $\hat{x} \in [\bar{x}_u, \bar{x}]$, we have $u(\hat{x}; C^\star(\mathcal{R})) = u(\bar{x}_u; C^\star(\mathcal{R}))$ and $u(\hat{x}; C) = u(\bar{x}_u; C)$. By the similar statement above, we have

$$\begin{aligned}\sup\{p \in [0,m] : \mathrm{CP}_o(p;(\hat{x},\underline{h}(\hat{x}))) = C\} &= u(\bar{x}_u;C) = u(\bar{x}_u;C^\star(\mathcal{R}))\\ &= \sup\{p \in [0,m] : \mathrm{CP}_o(p;(\hat{x},\underline{h}(\hat{x}))) = C^\star(\mathcal{R})\},\end{aligned}$$

which implies that $\mathrm{CP}_o(u(\bar{x}_u;C);(\hat{x},\underline{h}(\hat{x}))) = C$ and $\mathrm{CP}_o(u(\bar{x}_u;C);(\hat{x},\underline{h}(\hat{x}))) = C^\star(\mathcal{R})$, which is a contradiction.

F.14. Lemma 21 and its Proof

Lemma 21. *Recall that $u(\cdot; C)$ is defined in Equation (13), and $\bar{x}_u$ is defined in Equation (14). We have $u(x; C)$ gets its minimum value at $x \in [\bar{x}_u, \bar{x}]$.*

Proof of Lemma 21 By the first property of Lemma 6, we have

$$\frac{\partial u(x;C)}{\partial x} = \begin{cases} ((1-C)\frac{r_h}{r_\ell} + C)\underline{\mathcal{H}}'(x) & \text{if } x + \underline{\mathcal{H}}(x) \geqslant m; \\ (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x) - C & \text{if } x + \underline{\mathcal{H}}(x) < m. \end{cases} \tag{51}$$

Recall that

$$\bar{x}_u = \begin{cases} x_L & \text{if } x_L + y_L \geqslant m; \\ \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\} & \text{Otherwise.} \end{cases} \tag{52}$$

Therefore, if $x_L + y_L \geqslant m$, we have $\bar{x}_u = x_L$. As L is the lowest point, which implies that $\underline{h}'(x_L^-) < 0$ and $\underline{h}'(x_L^+) > 0$. Recall that $\underline{\mathcal{H}}(x) = \min\{\underline{h}, m\}$, we have $\underline{\mathcal{H}}'(x_L^-) \leqslant 0$ and $\underline{\mathcal{H}}'(x_L^+) \geqslant 0$. For $x + \underline{\mathcal{H}}(x) \geqslant m$, as we have $\frac{\partial u(x;C)}{\partial x} = ((1-C)\frac{r_h}{r_\ell} + C)\underline{\mathcal{H}}'(x)$, we have $\frac{\partial u(x;C)}{\partial x} < 0$ for $x < x_L$ and $\frac{\partial u(x;C)}{\partial x} > 0$ for $x > x_L$, which implies that $\bar{x}_u = x_L$ is the point such that $u(x;C)$ achieves its lowest value. As we force $u(x;C) = u(\bar{x}_u;C)$, we have $u(x;C)$ gets its minimum value at $x \in [\bar{x}_u, \bar{x}]$.

In other cases, by taking $\bar{x}_u = \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\}$, we have $\frac{\partial u(x;C)}{\partial x} < 0$ for $x < \bar{x}_u$ and $\frac{\partial u(x;C)}{\partial x} > 0$ for $x > \bar{x}_u$, which implies that $\bar{x}_u$ is the point such that $u(x;C)$ achieves its lowest value. As we force $u(x;C) = u(\bar{x}_u;C)$, we have $u(x;C)$ gets its minimum value at $x \in [\bar{x}_u, \bar{x}]$.

F.15. Lemma 22 and its Proof

LEMMA 22. *Recall that $\bar{x}_u$ is defined in Equation (14), and $\mathcal{V}$ is the x-vertices set of a polyhedron $\mathcal{R}$ plus all elements of $\mathcal{R}_0$, where $\mathcal{R}_0 = \{(x, \underline{h}(x) : x \in [\underline{x}, \bar{x}]\} \cap \{(x,y) : x + y = m\}$, then $\bar{x}_u \in \mathcal{V}$.*

Proof of Lemma 22 Recall that

$$\bar{x}_u = \begin{cases} x_L & \text{if } x_L + y_L \geqslant m; \\ \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\} & \text{Otherwise}, \end{cases} \tag{53}$$

That is, $\bar{x}_u$ equals to either x_L or $\sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\}$. As L is a vertex of $\mathcal{R}$, we have $x_L \in \mathcal{V}$.

Then, we claim that $\bar{x}_u = \sup\{x \in [x_L, \bar{x}] : (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x^-) - C < 0\} \in \mathcal{V}$. As $\mathcal{R}$ is a polyhedron, we have $\underline{h}(\cdot)$ is a piecewise linear function. As $\underline{\mathcal{H}}(x) = \min\{\underline{h}(x), m\}$, we have $\underline{\mathcal{H}}(\cdot)$ is also a piecewise linear function. If $\bar{x}_u$ is not a x-vertex, then there exists $\epsilon > 0$ such that $\underline{\mathcal{H}}'(x^-) = \underline{\mathcal{H}}'((x+\epsilon)^-)$, and we have

$$(1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'((x+\epsilon)^-) - C < 0,$$

which contradicts the definition of $\bar{x}_u$.

F.16. Lemma 23 and its Proof

LEMMA 23. *If $\widetilde{l}(\hat{x}; C^\star(\mathcal{R})) = u(\hat{x}; C^\star(\mathcal{R}))$ for $x_H \leqslant \hat{x} \leqslant x_{-1}$, we have $\hat{x} \in [\underline{x}_u, \bar{x}_u]$.*

Proof of Lemma 23 As $u(\hat{x}; C^\star(\mathcal{R})) = \widetilde{l}(\hat{x}; C^\star(\mathcal{R})) < m$, we have $\hat{x} > \underline{x_u}$. Then, we show that $\hat{x} \leqslant \bar{x}_u$ by contradiction. If $\hat{x} > \bar{x}_u$, as $u(\cdot; C^\star(\mathcal{R}))$ is constant between $[\bar{x}_u, \hat{x}]$ and $\widetilde{l}(\cdot; C^\star(\mathcal{R}))$ is a decreasing function for $x \in [x_H, \hat{x}]$, we have there exists $\epsilon > 0$ such that $\widetilde{l}(\hat{x} - \epsilon; C^\star(\mathcal{R})) > u(\hat{x} - \epsilon; C^\star(\mathcal{R}))$, which is a contradiction. Therefore, we have $\hat{x} \in [\underline{x_u}, \bar{x}_u]$,

F.17. Lemma 24 and its Proof

LEMMA 24. *Recall that $x_{-1} = \sup\{x \in [x_H, \bar{x}] : \frac{\partial l(x^-; C^\star(\mathcal{R}))}{\partial x} \leqslant -1\}$. Given a polyhedron $\mathcal{R}$, recall that $\mathcal{V}$ is the x-vertices set of $\mathcal{R}$ plus all elements in $\mathcal{R}_0$, then $x_{-1} \in \mathcal{V}$.*

Proof of Lemma 24 As $\mathcal{R}$ is a polyhedron, we have $\bar{h}(\cdot)$ is a piecewise linear function. By the second property of Lemma 6, we have $l(\cdot; C)$ is also a piecewise linear function. Then, we prove by contradiction. Suppose that $x_{-1} \notin \mathcal{V}$. Then, there exists $\epsilon > 0$ such that $\frac{\partial l(x_{-1}^-; C^\star(\mathcal{R}))}{\partial x} = \frac{\partial l((x_{-1}+\epsilon)^-; C^\star(\mathcal{R}))}{\partial x}$. Then, we have $\frac{\partial l((x_{-1}+\epsilon)^-; C^\star(\mathcal{R}))}{\partial x} \leqslant -1$, which contradicts to the definition of x_{-1}.

F.18. Lemma 25 and its Proof

LEMMA 25. *Suppose that $\mathcal{R}$ is a polyhedron. Recall that $u(\cdot;C)$ and $\widetilde{l}(\cdot;C)$ are defined in Equations* (13) *and* (17) *respectively. Then, we have $u(\cdot;C)$ and $\widetilde{l}(\cdot;C)$ are piecewise linear functions and all of their x-vertices are a subset of $\mathcal{V}$. In addition, the elements of $\mathcal{R}_0$ are x-vertices of $u(\cdot;C)$, where $\mathcal{R}_0 = \{(x, \underline{h}(x) : x \in [\underline{x}, \bar{x}]\} \cap \{(x,y) : x+y=m\}$.*

Proof of Lemma 25 As $\mathcal{R}$ is a polyhedron, we have both $\bar{h}(\cdot)$ and $\underline{h}(\cdot)$ are piecewise linear. Since $\overline{\mathcal{H}}(x) = \min\{m, \bar{h}(x)\}$ and $\underline{\mathcal{H}}(x) = \min\{m, \underline{h}(x)\}$, we have both $\overline{\mathcal{H}}(\cdot)$ and $\underline{\mathcal{H}}(\cdot)$ are piecewise linear and both $\overline{\mathcal{H}}'(\cdot)$ and $\underline{\mathcal{H}}'(\cdot)$ are piecewise constant.

Recall that by Lemma 6, we have for any $x \in (x_H, \bar{x}_l)$ and $C \in [0,1]$,

$$\frac{\partial l(x;C)}{\partial x} = C\overline{\mathcal{H}}'(x).$$

Therefore, $\frac{\partial l(x;C)}{\partial x}$ is piecewise constant and $l(x;C)$ is piecewise linear. In addition, we can find that the x-vertices of $l(\cdot;C)$ are also ones of $\bar{h}(\cdot)$. By Equation (17) and Lemma 24, we have $\widetilde{l}(\cdot;C)$ is also piecewise linear with x-vertices belong to ones of $\bar{h}(\cdot)$.

Recall that by Lemma 6, we have for any $x \in (\underline{x}_u, \bar{x}_u)$ and $C \in [0,1]$,

$$\frac{\partial u(x;C)}{\partial x} = \begin{cases} ((1-C)\frac{r_h}{r_\ell} + C)\underline{\mathcal{H}}'(x) & \text{if } x + \underline{\mathcal{H}}(x) \geqslant m; \\ (1-C)\frac{r_h}{r_\ell}\underline{\mathcal{H}}'(x) - C & \text{if } x + \underline{\mathcal{H}}(x) < m. \end{cases} \tag{54}$$

Therefore, we have $\frac{\partial u(x;C)}{\partial x}$ is piecewise constant and $u(x;C)$ is piecewise linear. In addition, the x-vertices of $u(x;C)$ for $x + \underline{\mathcal{H}}(x) \geqslant m$ is a subset to x-vertices of $\overline{\mathcal{H}}(x)$. The x-vertices of $u(x;C)$ for $x + \underline{\mathcal{H}}(x) < m$ is also a subset to x-vertices of $\overline{\mathcal{H}}(x)$. Moreover, any point x such that $x + \underline{\mathcal{H}}(x) = m$, which is we defined as an element of $\mathcal{R}_0$, is also a vertex, and by our definition, $\mathcal{V}$ contains all elements of $\mathcal{R}_0$.

F.19. Geometric Lemmas

LEMMA 26. *Suppose that we have a line $\mathcal{L}(x)$ with any negative slope on an interval $\mathcal{I}$. $f(x)$ is a concave decreasing function which intersects $\mathcal{L}(x)$ at $(x_0, f(x_0))$. If there exists $x_1 < x_0$ such that $f(x_1) > f(x_0)$, then for all $x > x_0$, we have $f(x) < \mathcal{L}(x)$.*

Proof of Lemma 26 Suppose that there exists $x_2 > x_0$ such that $f(x_2) \geqslant \mathcal{L}(x_2)$. As $(x_1, f(x_1)$, $(x_2, f(x_2))$ are both above the line $\mathcal{L}$, if we connect $(x_1, f(x_1)$, $(x_2, f(x_2))$ by a line $\mathcal{L}_1(x)$, we have $\mathcal{L}_1(x) > \mathcal{L}(x)$ for $x \in [x_1, x_2]$. Therefore, $\mathcal{L}_1(x_0) > \mathcal{L}(x_0) = f(x_0)$ since $x_0 \in [x_1, x_2]$.

However, as $f(x)$ is a concave function, if we take $x_1 < x_2$ and connect $(x_1, f(x_1)$, $(x_2, f(x_2))$ by a line $\mathcal{L}_1(x)$, we should always have $f(x) \geqslant \mathcal{L}_1(x)$, which is a contradiction to $\mathcal{L}_1(x_0) > \mathcal{L}(x_0) = f(x_0)$.

□

LEMMA 27. *Let $f(x)$ and $g(x)$ be two piecewise linear functions defined on an interval I, with $f(x) \geqslant g(x)$ for any $x \in I$. If $\{x : f(x) = g(x)\}$ is not empty, we have there exists $x_0 \in \mathcal{V}$, which is an x-vertex of either $f(x)$ or $g(x)$, such that $f(x_0) = g(x_0)$.*

Proof of Lemma 27 We prove by contradiction. Suppose that any $x_1 \in \{x : f(x) = g(x)\}$ is not an x-vertex of either $f(x)$ or $g(x)$. Then, we have $f'(x_1^+) = f'(x_1^-)$ and $g'(x_1^+) = g'(x_1^-)$. As $f(x) \geqslant g(x)$ everywhere and $f(x_1) = g(x_1)$, we have $f'(x_1^-) \leqslant g'(x_1^-)$.

If $f'(x_1^-) < g'(x_1^-)$, we have $f'(x_1^+) < g'(x_1^+)$ and by $f(x_1) = g(x_1)$, we have there exists $\epsilon > 0$ such that $f(x_1 + \epsilon) < g(x_1 + \epsilon)$, which is a contradiction to $f(x) \geqslant g(x)$ everywhere.

If $f'(x_1^-) = g'(x_1^-)$, we define x_2 as $x_2 = \inf\{x : f(x) = g(x)$ for any $x \in [x, x_1]\}$. Then, we have $f'(x_2^+) = g'(x_2^+)$. By the definition of x_2, we know that for any $\epsilon_1 > 0$, we have $f(x_2 - \epsilon_1) > g(x_2 - \epsilon_1)$, which implies that $f(x_2^-) \neq g(x_2^-)$. As $f'(x_2^+) = g'(x_2^+)$, we have either $f(x_2^-) \neq f'(x_2^+)$ or $g(x_2^-) \neq g'(x_2^+)$, which implies that x_2 is an x-vertex for f or g, which is a contradiction.

□