Chasing Convex Functions with Long-term Constraints

Adam Lechowicz [*] Nicolas Christianson [†] Bo Sun [‡] Noman Bashir [§]
Mohammad Hajiesmaili [¶] Adam Wierman [‖] Prashant Shenoy [**]

February 22, 2024

Abstract

We introduce and study a family of online metric problems with long-term constraints. In these problems, an online player makes decisions $\mathbf{x}_t$ in a metric space (X, d) to simultaneously minimize their hitting cost $f_t(\mathbf{x}_t)$ and switching cost as determined by the metric. Over the time horizon T, the player must satisfy a long-term demand constraint $\sum_t c(\mathbf{x}_t) \geq 1$, where $c(\mathbf{x}_t)$ denotes the fraction of demand satisfied at time t. Such problems can find a wide array of applications to online resource allocation in sustainable energy and computing systems. We devise optimal competitive and learning-augmented algorithms for specific instantiations of these problems, and further show that our proposed algorithms perform well in numerical experiments.

1 Introduction

This paper introduces and studies a novel class of online metric problems with *long-term demand constraints* motivated by emerging applications in the design of sustainable systems. In *convex function chasing with a long-term constraint*, an online player aims to satisfy a demand by making decisions in a normed vector space, paying a hitting cost based on time-varying convex cost functions which are revealed online, and switching cost defined by the norm. The player is constrained to ensure that the entire demand is satisfied at or before the time horizon T ends, and their objective is to minimize their total cost. The generality of this problem makes it applicable to a wide variety of online resource allocation problems; in this paper, we consider one such special case, discussing its connections to other online settings and suggestions towards broad new areas of inquiry in *online optimization with long-term constraints*.

Our motivation to introduce these problems is rooted in an emerging class of *carbon-aware* control problems for sustainable systems. A shared objective involves minimizing carbon emissions by shifting flexible workloads temporally and/or spatially to better leverage low-carbon electricity generation (e.g., renewables such as solar and wind). Examples which have recently seen significant interest include carbon-aware electric vehicle (EV) charging [CBS+22] and carbon-aware compute shifting [WBS+21; BGH+21; RKS+22; ALK+23; HLB+23].

[*]University of Massachusetts Amherst. Email: alechowicz@cs.umass.edu
[†]California Institute of Technology. Email: nchristianson@caltech.edu
[‡]University of Waterloo. Email: bo.sun@uwaterloo.ca
[§]Massachusetts Institute of Technology. Email: nbashir@mit.edu
[¶]University of Massachusetts Amherst. Email: hajiesmaili@cs.umass.edu
[‖]California Institute of Technology. Email: adamw@caltech.edu
[**]University of Massachusetts Amherst. Email: shenoy@cs.umass.edu

The problems we introduce in this paper build on a long line of related work in online algorithms. Most existing work can be roughly classified into two types: *online metric problems*, where many works consider multidimensional decision spaces and switching costs but do not consider long-term constraints [BLS92; Kou09; CGW18; BKL+19; BCL+21; BC22; BCR23], and *online search problems*, which feature long-term demand constraints but do not consider multidimensional decision spaces or switching costs [EFK+01; LPS08; MAS14; SZL+21].

We briefly review the direct precursors of our work below. In the online metric literature, the problem we study is an extension of *convex function chasing* (CFC) introduced by Friedman and Linial [FL93], where an online player makes online decisions $\mathbf{x}_t$ in a normed vector space $(X, \|\cdot\|)$ over a sequence of time-varying cost functions in order to minimize their total hitting and switching cost. In the online search literature, the problem we study is a generalization of *one-way trading* (OWT) introduced by El-Yaniv et al. [EFK+01], in which an online player must sell an entire asset in fractional shares over a sequence of time-varying prices while maximizing their profit.

Despite extensive existing work in the online metric and online search tracks, few works simultaneously consider long-term demand constraints (as in OWT) and movement/switching costs (as in CFC). The existing prior works [LCZ+23; LCS+24] that consider both components are restricted to unidimensional decision spaces, as is typical in the online search literature. However, generalizing from the unidimensional case is highly non-trivial; e.g., in convex function chasing with a long-term constraint, the problem cannot simply be decomposed over dimensions due to the shared capacity constraint and multidimensional switching cost. Thus, in this work we tackle the following question:

> *Is it possible to design online algorithms for the studied problems that operate in multidimensional decision spaces while simultaneously considering long-term constraints, hitting costs, and switching costs?*

Although the aforementioned literature focuses on competitive algorithms in adversarial settings, there has recently been significant interest in moving beyond worst-case analysis, which can result in overly pessimistic algorithms. The field of *learning-augmented algorithms* [LV18; PSK18] has emerged as a paradigm for designing and analyzing algorithms that incorporate untrusted machine-learned advice to improve average-case performance without sacrificing worst-case performance bounds. Such algorithms are evaluated through the metrics of *consistency and robustness* (see Def. 2.1). Recent studies have proposed learning-augmented algorithms for related problems, including convex function chasing [CHW22], one-way trading [SLH+21], metrical task systems [CSW23], and online search [LSH+24]. While the literature in each of these tracks considers a spectrum of different advice models, their results prompt a natural open question:

> *Can we design algorithms for online metric problems with long-term constraints that effectively utilize untrusted advice (such as machine-learned predictions) to improve performance while preserving worst-case competitive guarantees?*

Contributions. Despite extensive prior literature on adjacent problems, the problems we propose in this paper are the first online settings to combine long-term demand constraints with multidimensional decision spaces and switching costs. We introduce convex function chasing with a long-term constraint, and a special case called *online metric allocation with a long-term constraint.* The general forms of both are independently interesting for further study.

We obtain positive results for both of the questions posed above under problem instantiations that are especially relevant for motivating applications. We provide the first competitive results for online problems of this form in Section 3, and show that our proposed algorithm (Algorithm 1) achieves the best possible competitive ratio. In Section 4, we propose a learning-augmented algorithm, CLIP (Algorithm 2), and show it achieves the provably optimal trade-off between consistency and robustness.

To achieve these results, the proposed algorithms must tackle technical challenges distinct from prior work studying adjacent problems. We build on a generalization of the threshold-based designs used for simple decision spaces in the online search literature called *pseudo-cost minimization*. We introduce a novel application of this framework to multidimensional decision spaces (see Section 3), and show that it systematically addresses the competitive drawbacks of typical algorithm designs for online metric problems. We evaluate our proposed algorithms in numerical experiments and show that our algorithms outperform a set of baseline heuristics on synthetic instances of convex function chasing with a long-term constraint.

Our learning-augmented algorithm CLIP (see Section 4) introduces a novel *projected consistency constraint* which is designed to guarantee $(1+\epsilon)$-consistency against the provided advice ADV by continuously comparing their solutions in terms of the cost incurred so far, the switching cost trajectories, and the projected worst-case cost required to complete the long-term constraint. To solve *both* convex function chasing and online metric allocation with long-term constraints, we derive a transformation result that directly relates the performance of an algorithm on the former problem with its performance on the latter (see Section 2).

2 Problem Formulation and Preliminaries

This section formalizes convex function chasing and online metric allocation with long-term constraints, motivating them with a sustainability application. We also provide preliminaries used throughout the paper, and give initial results to build algorithmic connections between both problems.

Convex function chasing with a long-term constraint. A player chooses decisions $\mathbf{x}_t \in X \subseteq \mathbb{R}^d$ online from a normed vector space $(X, \|\cdot\|)$ in order to minimize their total cost $\sum_{t=1}^{T} f_t(\mathbf{x}_t) + \sum_{t=1}^{T+1} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|$, where $f_t(\cdot) : X \to \mathbb{R}$ is a convex "hitting" cost that is revealed just before the player chooses $\mathbf{x}_t$, and $\|\mathbf{x}_t - \mathbf{x}_{t-1}\|$ is a switching cost associated with changing decisions between rounds. Additionally, the player must satisfy a long term constraint of the form $\sum_{t=1}^{T} c(\mathbf{x}_t) = 1$, where $c(\mathbf{x}) : X \to [0,1]$ gives the fraction of the constraint satisfied by a decision $\mathbf{x}$. We denote the *utilization* at time t by $z^{(t)} = \sum_{\tau=1}^{t} c(\mathbf{x}_\tau)$, which gives the total fraction of the long-term constraint satisfied up to and including time t. The offline version of this problem can be formalized as follows:

$$\min_{\{\mathbf{x}_t\}_{t\in[T]}} \underbrace{\sum_{t=1}^{T} f_t(\mathbf{x}_t)}_{\text{Convex hitting cost}} + \underbrace{\sum_{t=1}^{T+1} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|}_{\text{Switching cost}} \quad \text{s.t.} \quad \underbrace{\sum_{t=1}^{T} c(\mathbf{x}_t) \geq 1,}_{\text{Long-term constraint}} \quad \mathbf{x}_t^i \in [0,1] \; \forall i \in [d], \; \forall t \in [T]. \quad (1)$$

Assumptions. Here, we describe the precise variant of convex function chasing with a long-term constraint for which we design algorithms in the remainder of the paper. *Let $\|\mathbf{x} - \mathbf{x}'\| := \|\mathbf{x} - \mathbf{x}'\|_{\ell_1(\mathbf{w})}$, where $\|\cdot\|_{\ell_1(\mathbf{w})}$ denotes the weighted ℓ_1 norm with weight vector $\mathbf{w} \in \mathbb{R}^d$.*

We define the long-term constraint such that $c(\mathbf{x}) := \|\mathbf{x}\|_{\ell_1(\mathbf{c})}$, i.e., the weighted ℓ_1 norm with weight vector $\mathbf{c} \in \mathbb{R}^d$. Then let the metric space X be the ℓ_1 ball defined by $X := \{\mathbf{x} \in \mathbb{R}^d : c(\mathbf{x}) \leq 1\}$. For all cost functions $f_t(\cdot) : X \to \mathbb{R}$, we assume bounded gradients such that $L \leq [\nabla f_t]^i / c^i \leq U \; \forall i \in [d], t \in [T]$, where i denotes the ith dimension of the corresponding vector, and L, U are known positive constants.

Letting $\mathbf{0}$ denote the origin in $\mathbb{R}^d$ (w.l.o.g), we have the property $f_t(\mathbf{0}) = 0$ for all $t \in [T]$, i.e., that "satisfying none of the long-term constraint costs nothing", since $c(\mathbf{0}) = 0$. We assume the player starts and ends at the origin, i.e., $\mathbf{x}_0 = \mathbf{0}$ and $\mathbf{x}_{T+1} = \mathbf{0}$, to enforce switching "on" and "off." These assumptions are intuitive and reasonable in practice, e.g., in our example motivating application below.

For analysis, it will be useful to establish a shorthand for the magnitude of the switching cost. Let $\beta := \max \left(w^i / c^i \right)$, which gives the greatest magnitude of the switching cost coefficient when normalized by *the constraint function. We assume that β is bounded on the interval $[0, {}^{U-L}/_{2})$; if β is "large" (i.e., $> {}^{U-L}/_{2}$), we can show that the player should prioritize minimizing the switching cost.*[1]

Recall the player must fully satisfy the long-term constraint before the sequence ends. If the player has satisfied $z^{(t)}$ fraction of the constraint at time t, we assume a compulsory trade begins at time j as soon as $(T-(j+1)) \cdot \mathbf{c}^i < \left(1 - z^{(j)}\right) \ \forall i \in [d]$ (i.e., when the time steps after j are not sufficient to satisfy the constraint). During this compulsory trade, a cost-agnostic algorithm takes over, making maximal decisions to satisfy the constraint. To ensure that the problem remains technically interesting, we assume that the compulsory trade is a small portion of the sequence.[2]

For brevity, we henceforth use CFL to refer to the variant of convex function chasing with a long-term constraint under the assumptions outlined above.

An example motivating application. CFL can model a variety of applications, including specific applications that motivate this study. Consider a *carbon-aware temporal load shifting* application with heterogeneous servers. Here, each of the d dimensions corresponds to one of d heterogeneous servers. An algorithm makes decisions $\mathbf{x}_t \in \mathbb{R}^d$, where $\mathbf{x}_t^i \in [0,1]$ denotes the load of the ith server at time t. The long-term constraint $\sum_{t=1}^{T} c(\mathbf{x}_t) \geq 1$ enforces that an entire workload should be finished before time T, and each coefficient $\mathbf{c}^i$ represents the throughput of the ith server. Each cost function $f_t(\mathbf{x}_t)$ represents the carbon emissions due to the electricity usage of the servers configured according to $\mathbf{x}_t$, and the switching cost $\|\cdot\|_{\ell_1(\mathbf{w})}$ captures the carbon emissions overhead (e.g., extra latency) of pausing, resuming, scaling, and moving the workload between servers.

Online metric allocation with a long-term constraint. Bansal and Coester [BC22] introduced the online metric allocation problem (MAP), which connects several online metric problems. MAP on a star metric is equivalent to CFC when cost functions are separable over dimensions and supported on the unit simplex Δ_n.[3] Furthermore, the randomized metrical task systems problem (MTS) is a special case of MAP when cost functions are linear and increasing.

We build on this formulation in our setting and introduce *online metric allocation with a long-term constraint*, which captures a particularly interesting special case of CFL. The general version of the problem considers an n-point metric space (X, d), and a unit resource which can be allocated in arbitrary fractions to the points of X. At each time $t \in [T]$, convex cost functions $f_t^a(\cdot) : [0,1] \to \mathbb{R}$ arrive at each point a in the metric space. The online player chooses an allocation x_t^a to each point a in the metric space, such that $\sum_{a=1}^{n} x_t^a = 1$ for all $t \in [T]$. When changing this allocation between time steps, the player pays a switching cost defined by $d(a,b)$ for any distinct points $a, b \in X$. As in CFL, the long-term constraint enforces that $\sum_{t=1}^{T} c(\mathbf{x}_t) \geq 1$, where $c(\mathbf{x})$ is a linear and separable function of the form $c(\mathbf{x}) = \sum_{a=1}^{n} \mathbf{c}^a x^a$. As previously, the player's objective is to minimize the total cost (hitting plus switching costs) incurred while satisfying the long-term constraint.

Assumptions. In the rest of the paper, we consider an instantiation of online metric allocation with a long-term constraint on *weighted star metrics* that is particularly relevant to a wide class of resource allocation problems.
To ensure the long-term constraint is non-trivial, we denote at least one point a' in the metric space as the "OFF state", where $\mathbf{c}^{a'} = 0$ and $f_t^{a'}(x) = 0 \ \forall t \in [T], \forall x \in [0,1]$. For all other cost functions, we carry forward the assumptions that $L \leq {}^{df_t^a}/_{dx^a} \leq U$, $f_t^a(0) = 0 \ \forall t \in [T]$. We define $\beta := \max_{a',a} d(a', a)$, i.e., the maximum

[1] As brief justification for the bounds on β, consider that a feasible solution may have objective value $L + 2\beta$. If $\beta > {}^{U-L}/_{2}$, $L + 2\beta > U$, and we argue that the incurred switching cost is more important than the cost functions accepted.

[2] We assume the first time j' where $(T-(j'+1))\,\mathbf{c}^i < 1 \ \forall i$ satisfies $j' \gg 1$, which implies that T and $\mathbf{c}$ are both sized appropriately for the constraint. This is reasonable for an application such as carbon-aware load shifting, since short deadlines (small T) or low throughput (small $\mathbf{c}^i \ \forall i$) imply that even offline solutions suffer a lack of flexibility in reducing the overall cost.

[3] Given metric space X, consider $\Delta(X)$, which represents the set of probability measures over the points of X. Since X is finite, we have that $|X| = n$ and $\Delta(X)$ is denoted as Δ_n.

distance between the OFF *state and any other state in the weighted star, inheriting the same assumption that* $\beta \in [0, {U-L}/2)$. For brevity, we henceforth use MAL to refer to the problem on weighted star metrics with the assumptions described above.

Competitive analysis. Our goal is to design an algorithm that guarantees a small *competitive ratio* [MMS88; BLS92], i.e., performs nearly as well as the offline optimal solution. Formally, let $\mathcal{I} \in \Omega$ denote a valid input sequence, where Ω is the set of all feasible inputs for the problem. Let $\texttt{OPT}(\mathcal{I})$ denote the cost of an optimal offline solution for instance $\mathcal{I}$, and let $\texttt{ALG}(\mathcal{I})$ denote the cost incurred by running an online algorithm ALG over the same instance. The competitive ratio is then defined as $\text{CR}(\texttt{ALG}) := \max_{\mathcal{I} \in \Omega} {\texttt{ALG}(\mathcal{I})}/{\texttt{OPT}(\mathcal{I})} = \eta$, and ALG is said to be η-**competitive**. Note that $\text{CR}(\texttt{ALG})$ is always ≥ 1, and a *lower* competitive ratio implies that the online algorithm is guaranteed to be *closer* to the offline optimal solution.

Learning-augmented consistency and robustness. In the emerging literature on learning-augmented algorithms, competitive analysis is interpreted via the notions of *consistency* and *robustness*, introduced by [LV18; PSK18].

Definition 2.1. *Let* LALG *denote a learning-augmented online algorithm provided with advice denoted by* ADV*. Then* LALG *is said to be* b-***consistent*** *if it is* b*-competitive with respect to* ADV*. Conversely,* LALG *is* r-***robust*** *if it is* r*-competitive with respect to* OPT *when given any* ADV *(i.e., regardless of the performance of* ADV*).*

A connection between CFL **and** MAL**.** Below we state two useful results connecting the CFL and MAL settings that influence our algorithm design for each problem.

Lemma 2.2. *For any* MAL *instance on a weighted star metric* (X, d)*, there is a corresponding* CFL *instance on* $(\mathbb{R}^{n-1}, \|\cdot\|_{\ell_1(\mathbf{w}')})$ *which preserves* $f_t^a(\cdot)\ \forall t, c(\cdot)\ \forall a \in X$*, and upper bounds* $d(a, b),\ \forall (a, b) \in X$*.*

Leveraging Lemma 2.2, the following result explicitly connects the competitive results of the CFL and MAL settings.

Proposition 2.3. *Given an algorithm* ALG *for* CFL*, any competitive bound for* ALG *gives an identical competitive bound for* MAL *with parameters corresponding to the* CFL *instance constructed in Lemma 2.2.*

The proofs of both are deferred to Appendix B.3. At a high-level, Proposition 2.3 shows that if ALG is η-competitive against OPT which pays no switching cost, Lemma 2.2 implies it is also η-competitive on MAL. In the next section, our proposed algorithms will be presented using CFL notation, but these results provide the necessary condition which allows them to solve MAL as well.

3 Designing Competitive Algorithms

In this section, we present our robust algorithm design. We start by discussing some inherent challenges in the problem, highlighting reasons why existing algorithms (e.g., for CFC) fail. Next, we introduce a generalization of existing techniques from online search called pseudo-cost minimization, which underpins our competitive algorithm, ALG1 (Algorithm 1). Finally, we state (and prove in Appendix B) two bounds, which jointly imply that ALG1 achieves the optimal competitive ratio for CFL and MAL.

Algorithm 1 Pseudo-cost minimization algorithm (ALG1)

input: long-term constraint function $c(\cdot)$, distance metric $\|\cdot\|_{\ell_1(\mathbf{w})}$, pseudo-cost threshold function $\phi(z)$
initialize: $z^{(0)} = 0$;
while cost function $f_t(\cdot)$ is revealed and $z^{(t-1)} < 1$ **do**
solve pseudo-cost minimization problem:

$$\mathbf{x}_t = \underset{\mathbf{x}\in X: c(\mathbf{x}) \le 1 - z^{(t-1)}}{\arg\min} \; f_t(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x})} \phi(u)du \tag{2}$$

update utilization $z^{(t)} = z^{(t-1)} + c(\mathbf{x}_t)$

Challenges. Canonical algorithms for CFC [CGW18; Sel20; ZJL+21] make decisions that attempt to minimize (or nearly minimize) the hitting cost of cost functions $f_t(\cdot)$ and switching cost across all time steps. As discussed in the introduction, the structure of the problem with a long-term constraint means that such myopic cost-minimization algorithms will fail in general. To illustrate this, consider the actions of a minimizer-driven algorithm on an arbitrary sequence with length T. For each $t < T$, the algorithm chooses a point at or near $\mathbf{0}$, since $\mathbf{0}$ is the minimizer of each f_t. However, since $c(\mathbf{0}) = 0$, such an algorithm must subsequently satisfy all or almost all of the long-term constraint during the *compulsory trade*, incurring an arbitrarily bad hitting cost.

This challenge motivates an algorithm design that balances between the two extremes of finishing the long-term constraint "immediately" (i.e., at the first or early time steps), and finishing the long-term constraint "when forced to" (i.e., during the compulsory trade). Both extremes result in a poor competitive ratio. Many algorithms in the online search literature (e.g., online knapsack, OWT) leverage a *threshold-based design* to address precisely this problem, as in [ZCL08; SZL+21; LCS+24]. However, such threshold-based algorithms are traditionally derived for single-dimensional decision spaces with no switching costs. In what follows, we describe a pseudo-cost minimization approach, which generalizes the threshold-based design to operate in the setting of CFL.

Algorithm description. Recall that $z^{(t)}$ gives the fraction of the long-term constraint satisfied at time t. Building off of the intuition of threshold-based design, we define a function ϕ, which will be used to compute a *pseudo-cost minimization* problem central to our robust algorithm.

Definition 3.1 (Pseudo-cost threshold function ϕ for CFL). *For any utilization $z \in [0, 1]$, ϕ is defined as:*

$$\phi(z) = U - \beta + (U/\alpha - U + 2\beta) \exp(z/\alpha), \tag{3}$$

where α is the competitive ratio and is defined in (4).

Then our algorithm (Algorithm 1, referred to as ALG1) solves the pseudo-cost minimization problem defined in (2) to obtain a decision $\mathbf{x}_t$ at each time step. At a high level, the inclusion of ϕ in this pseudo-cost problem enforces that, upon arrival of a cost function, the algorithm satisfies "just enough" of the long-term constraint. Concretely, the structure of the ϕ function enforces that $\phi(z^{(t)}) - \beta$ corresponds to the "best cost function seen so far". Then, if a good cost function arrives, the pseudo-cost minimization problem solves for the $\mathbf{x}_t$ which guarantees a competitive ratio of α against the current estimate of OPT.

At a glance, it is not obvious that the minimization problem in (2) is tractable; however, in Appendix B.1, we show that the problem is *convex*, implying that it can be solved efficiently. In Theorem 3.2, we state the competitive result for ALG1. We discuss the significance of the result below, and relegate the full proof to Appendix B.2.

Theorem 3.2. ALG1 *is α-competitive for* CFL, *where α is the solution to $\frac{U-L-2\beta}{U-U/\alpha-2\beta} = \exp(1/\alpha)$, given by*

$$\alpha := \left[W\left(\left(\frac{2\beta}{U} + \frac{L}{U} - 1 \right) e^{\frac{2\beta}{U}-1} \right) - \frac{2\beta}{U} + 1 \right]^{-1}, \tag{4}$$

where W is the Lambert W function [CGH+96].

Intuitively, parameters of CFL (L, U, and β) appear in the competitive bound. While results for OWT and CFC are not directly comparable, we discuss connections and the relative order of α. When $\beta \to 0$, α matches the optimal competitive ratio of $\left[W\left((L/U - 1)\, e^{-1}\right) + 1\right]^{-1}$ for the minimization variant of OWT [LPS08; SLH+21]. In the intermediate case (i.e., when $\beta \in (0, U-L/2)$), CFL adds a new *linear* dependence on β compared to OWT. Furthermore, when $\beta \to U-L/2$, α approaches U/L, which is the competitive ratio achievable by e.g., a myopic cost minimization algorithm. Since α does not feature a dependence on the dimension d of the vector space, we note a connection with CFC: it is known that "dimension-free" bounds are achievable in CFC with structural assumptions on the hitting cost [CGW18; AGG20] that are evocative of our bounded gradient assumptions in CFL.

Via Proposition 2.3, we obtain an immediate corollary to Theorem 3.2 which gives the following competitive bound when ALG1 is used to solve MAL. The full proof of Corollary 3.3 can be found in Appendix B.3.

Corollary 3.3. ALG1 *is α-competitive for* MAL.

On the tightness of competitive ratios. It is important to highlight that the bounds in Theorem 3.2 and Corollary 3.3 are the first competitive bounds for any variant of convex function chasing or online metric allocation imbued with long-term constraints. A natural follow-up question concerns whether any online algorithm for CFL (or MAL) can achieve a better competitive bound. In the following, we answer this question in the negative, showing that ALG1's competitive ratio is the best that any deterministic online algorithm for CFL and/or MAL can achieve. We state the result here, and defer the full proof to Appendix B.4.

Theorem 3.4. *There exists a family of* CFL *instances such that any deterministic online algorithm for* CFL *is at least α-competitive, where α is as defined in* (4).

Since ALG1 is α-competitive by Theorem 3.2, this implies that ALG1 achieves the optimal competitive ratio for CFL. Furthermore, by leveraging Lemma 2.2, this result gives an immediate corollary result in the MAL setting by constructing a corresponding family of MAL instances, which forces any algorithm to achieve a competitive ratio of α. We state the result here, deferring the full proof to Appendix B.5.

Corollary 3.5. *The* CFL *instances in Theorem 3.4 correspond to instances of* MAL *such that any deterministic online algorithm for* MAL *is at least α-competitive.*

As previously, since ALG1 is α-competitive by Corollary 3.3, it achieves the optimal competitive ratio for MAL. We note that beyond the settings of CFL and MAL considered in this paper, Theorem 3.4 and Corollary 3.5 are the first lower bound results for convex function chasing and online metric allocation with long-term constraints, and may thus give useful insight into the achievable competitive bounds for different or more general settings of these problems.

4 Learning-augmented Algorithms

In this section, we leverage techniques from the growing literature on *learning-augmented algorithms* to consider how *untrusted black-box advice* can help improve the average-case performance of an algorithm

Algorithm 2 **C**onsistency **Li**mited **P**seudo-cost minimization (CLIP)

input: consistency parameter ϵ, long-term constraint function $c(\cdot)$, pseudo-cost threshold $\phi^{\epsilon}(\cdot)$
initialize: $z^{(0)} = 0$; $p^{(0)} = 0$; $A^{(0)} = 0$; $\texttt{CLIP}_0 = 0$; $\texttt{ADV}_0 = 0$
while cost function $f_t(\cdot)$ is revealed, untrusted advice $\mathbf{a}_t$ is revealed, and $z^{(t-1)} < 1$ **do**
update advice cost $\texttt{ADV}_t = \texttt{ADV}_{t-1} + f_t(\mathbf{a}_t) + \|\mathbf{a}_t - \mathbf{a}_{t-1}\|_{\ell_1(\mathbf{w})}$ and advice utilization $A^{(t)} = A^{(t-1)} + c(\mathbf{a}_t)$
solve *constrained* pseudo-cost minimization problem:

$$\mathbf{x}_t = \underset{\mathbf{x} \in X : c(\mathbf{x}) \leq 1 - z^{(t-1)}}{\arg\min} f_t(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{p^{(t-1)}}^{p^{(t-1)} + c(\mathbf{x})} \phi^{\epsilon}(u) du \tag{5}$$

such that

$$\texttt{CLIP}_{t-1} + f_t(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{x} - \mathbf{a}_t\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - z^{(t-1)} - c(\mathbf{x}))L + \max((A^{(t)} - z^{(t-1)} - c(\mathbf{x})), 0)(U - L) \tag{6}$$
$$\leq (1 + \epsilon)[\texttt{ADV}_t + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A^{(t)})L]$$

update cost $\texttt{CLIP}_t = \texttt{CLIP}_{t-1} + f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})}$ and utilization $z^{(t)} = z^{(t-1)} + c(\mathbf{x}_t)$
solve *unconstrained* pseudo-cost minimization problem:

$$\bar{\mathbf{x}}_t = \underset{\mathbf{x} \in X : c(\mathbf{x}) \leq 1 - z^{(t-1)}}{\arg\min} f_t(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} - \int_{p^{(t-1)}}^{p^{(t-1)} + c(\mathbf{x})} \phi^{\epsilon}(u) du \tag{7}$$

update pseudo-utilization $p^{(t)} = p^{(t-1)} + \min(c(\bar{\mathbf{x}}_t), c(\mathbf{x}_t))$

for CFL and MAL while retaining worst-case guarantees. We first consider a sub-optimal "baseline" algorithm that directly combines advice with a robust algorithm such as ALG1. We then propose a unified algorithm called CLIP, which integrates advice more efficiently and achieves the optimal trade-off between *consistency* and *robustness* (Definition 2.1).

Advice model. For a CFL or MAL instance $\mathcal{I} \in \Omega$, let ADV denote untrusted black-box decision advice, i.e., $\texttt{ADV} := \{\mathbf{a}_t \in X : t \in [T]\}$. If the advice is correct, it achieves the optimal objective value (i.e., $\texttt{ADV}(\mathcal{I}) = \texttt{OPT}(\mathcal{I})$).

A simple baseline. Lechowicz et al. [LCS+24] show that a straightforward "fixed-ratio" learning-augmented approach works well in practice for unidimensional online search with switching costs. Here we show that a similar technique (playing a convex combination of the solutions chosen by the advice and a robust algorithm) achieves bounded but sub-optimal consistency and robustness for CFL.

Let $\texttt{ROB} := \{\tilde{\mathbf{x}}_t : t \in [T]\}$ denote the actions of a robust algorithm for CFL (e.g., ALG1). For any value $\epsilon \in (0, \alpha - 1]$, the fixed-ratio algorithm (denoted as Baseline for brevity) sets a fixed combination factor $\lambda := \frac{\alpha - 1 - \epsilon}{\alpha - 1}$. Then at each time step, Baseline chooses a combination decision according to $\mathbf{x}_t = \lambda \mathbf{a}_t + (1 - \lambda)\tilde{\mathbf{x}}_t$. We present consistency and robustness results for Baseline below, deferring the full proof to Appendix C.1.

Lemma 4.1. *Letting* ROB *denote the actions of* ALG1 *and setting a parameter* $\epsilon \in (0, \alpha - 1]$, Baseline *is* $(1 + \epsilon)$*-consistent and* $\left(\frac{(U + 2\beta)/L(\alpha - 1 - \epsilon) + \alpha\epsilon}{(\alpha - 1)}\right)$*-robust for* CFL.

Although this fixed-ratio algorithm verifies that an algorithm for CFL can utilize untrusted advice to improve performance, it remains an open question of whether the trade-off between consistency and robustness given in Lemma 4.1 is optimal. Thus, we study whether a learning-augmented algorithm for CFL can be designed which *does* achieve the provably optimal consistency-robustness trade-off. In the next section, we start by considering a more sophisticated method of incorporating advice into an algorithm design.

An optimal learning-augmented algorithm. We present CLIP (**C**onsistency-**Li**mited **P**seudo-cost minimization, Algorithm 2) which achieves the optimal trade-off between consistency and robustness for CFL. To start, for any $\epsilon \in (0, \alpha - 1]$, we define a corresponding *target robustness factor* γ^ϵ, which is defined as the unique positive solution to the following:

$$\gamma^\epsilon = \epsilon + \frac{U}{L} - \frac{\gamma^\epsilon}{L}(U - L) \ln \left(\frac{U - L - 2\beta}{U - U/\gamma^\epsilon - 2\beta} \right). \tag{8}$$

Note that $\gamma^{\alpha-1} = \alpha$, and $\gamma^0 = U/L$. We use γ^ϵ to define a pseudo-cost threshold function ϕ^ϵ which will be used in a minimization problem to choose a decision at each step of the CLIP algorithm.

Definition 4.2 (Pseudo-cost threshold function ϕ^ϵ)**.** *Given γ^ϵ from (8), $\phi^\epsilon(p)$ for $p \in [0, 1]$ is defined as:*

$$\phi^\epsilon(p) = U - \beta + (U/\gamma^\epsilon - U + 2\beta) \exp(p/\gamma^\epsilon). \tag{9}$$

For each time step $t \in [T]$, we define a *pseudo-utilization* $p^{(t)} \in [0, 1]$, where $p^{(t)} \leq z^{(t)}\ \forall t$, and $p^{(t)}$ describes the fraction of the long-term constraint which been satisfied "robustly" (as defined by the pseudo-cost) at time t.

Then CLIP (see Algorithm 2) solves a *constrained* pseudo-cost minimization problem (defined in (5)) to obtain a decision $\mathbf{x}_t$ at each time step. The objective of this problem is mostly inherited from ALG1, but the inclusion of a *consistency constraint* allows the framework to accommodate untrusted advice for bounded consistency and robustness.

The high-level intuition behind this consistency constraint (defined in (6)) is to directly compare the solutions of CLIP and ADV *so far*, while *"hedging"* against worst-case scenarios which may cause CLIP to violate the desired $(1 + \epsilon)$-consistency. We introduce some notation to simplify the expression of the constraint. We let $\texttt{CLIP}_t$ denote the cost of CLIP up to time t, i.e., $\texttt{CLIP}_t := \sum_{\tau=1}^t f_\tau(\mathbf{x}_\tau) + \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|_{\ell_1(\mathbf{w})}$. Similarly, we let $\texttt{ADV}_t := \sum_{\tau=1}^t f_\tau(\mathbf{a}_\tau) + \|\mathbf{a}_\tau - \mathbf{a}_{\tau-1}\|_{\ell_1(\mathbf{w})}$ denote the cost of ADV up to time t. Additionally, we let $A^{(t)}$ denote the utilization of ADV at time t, i.e., $A^{(t)} := \sum_{\tau=1}^t c(\mathbf{a}_\tau)$

The constraint defined in (6) considers the cost of both CLIP and ADV so far, and the current hitting and switching cost $f_t(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})}$, ensuring that $(1 + \epsilon)$-consistency is preserved. Both sides of the constraint also include terms which consider the cost of potential future situations. First, $\|\mathbf{x} - \mathbf{a}_t\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})}$ ensures that if CLIP pays a switching cost to follow ADV and/or pays a switching cost to "switch off" (move to $\mathbf{0}$) in e.g., the next time step, that cost has been paid for "in advance". As $\mathbf{x}_{T+1} = \mathbf{0}$, the constraint also charges ADV in advance for the mandatory switching cost at the end of the sequence $\left(\|\mathbf{a}_t\|_{\ell_1(\mathbf{w})}\right)$; this ensures that there is always a feasible setting of $\mathbf{x}_t$.

In the term $\left(1 - A^{(t)}\right) L$, the consistency constraint assumes that ADV can satisfy the rest of the long-term constraint at the best marginal cost L. Respectively, in the term $(1 - z^{(t-1)} - c(\mathbf{x}))L + \max((A^{(t)} - z^{(t-1)} - c(\mathbf{x})),\ 0)(U - L)$, the constraint assumes CLIP can satisfy *up to* $\left(1 - A^{(t)}\right)$ of the remaining long-term constraint at the best cost L, but any excess (i.e., $(A^{(t)} - z^{(t)})$) must be satisfied at the worst cost U (e.g., during the compulsory trade). This worst-case assumption ensures that when actual hitting costs replace the above terms, the desired $(1 + \epsilon)$-consistency holds.

At each step, CLIP also solves an *unconstrained* pseudo-cost minimization problem to obtain $\bar{\mathbf{x}}_t$, which updates the pseudo-utilization $p^{(t)}$. This ensures that when ADV has accepted a cost function which would not be accepted by the unconstrained pseudo-cost minimization, the threshold function ϕ^ϵ can "start from zero" in subsequent time steps.

At a high level, CLIP's consistency constraint combined with the pseudo-cost minimization generates decisions which are *as robust as possible* while preserving consistency. In Theorem 4.3, we state the consistency and robustness of CLIP; we relegate the full proof to Appendix C.2.

Theorem 4.3. *For any $\epsilon \in [0, \alpha - 1]$,* CLIP *is $(1+\epsilon)$-consistent and γ^ϵ-robust for* CFL *(γ^ϵ as defined in (8)).*

The previous result gives an immediate corollary when CLIP is used to solve MAL, which we state below. The full proof of Corollary 4.4 can be found in Appendix C.3.

Corollary 4.4. *For any $\epsilon \in [0, \alpha - 1]$,* CLIP *is $(1+\epsilon)$-consistent and γ^ϵ-robust for* MAL.

Optimal trade-offs between robustness and consistency. Although the trade-off given by CLIP implies that achieving 1-consistency requires a large robustness bound of U/L in the worst-case, in the following theorem we show that this is the best we can obtain from any consistent and robust algorithm. We state the result and discuss its significance here, deferring the full proof to Appendix C.4.

Theorem 4.5. *Given untrusted advice* ADV *and $\epsilon \in (0, \alpha - 1]$, any $(1+\epsilon)$-consistent learning-augmented algorithm for* CFL *is at least γ^ϵ-robust, where γ^ϵ is defined in (8).*

This result implies that CLIP achieves the *optimal* trade-off between consistency and robustness for CFL. Furthermore, via Lemma 2.2, this result immediately gives Corollary 4.6, which we state here and prove in Appendix C.5.

Corollary 4.6. *Any $(1+\epsilon)$-consistent learning-augmented algorithm for* MAL *is at least γ^ϵ-robust (γ^ϵ defined by (8)).*

As previously, this implies CLIP achieves the optimal consistency-robustness trade-off for MAL. Beyond the settings of CFL and MAL, these Pareto-optimality results may give useful insight into the achievable consistency-robustness trade-offs for more general settings.

5 Numerical Experiments

In this section, we conduct numerical experiments on synthetic CFL instances. We evaluate ALG1 and CLIP against the offline optimal solution, three heuristics adapted from related work, and the learning-augmented Baseline.

Setup. Here we give an overview of our experiment setup and comparison algorithms. We construct a d-dimensional decision space, where d is picked from the set $\{5, 7, \dots, 21\}$. The competitive ratio of our proposed algorithms depends on both U/L and $\beta = \max_i \mathbf{w}^\mathbf{i}$, as the switching cost. Hence, we evaluate their performance over the range of these parameters. We set different cost fluctuation ratios $U/L \in \{50, 150, \dots, 1250\}$ by setting L and U accordingly, and β is picked from the set $\beta \in \{0, 5, \dots, U/2.5\}$. We also set a parameter $\sigma \in \{0, 10, \dots, U/2\}$, which controls the dimension-wise variability of generated cost functions f_t. Across all experiments, $c(\mathbf{x}) = \|\mathbf{x}\|_1$.

For a given setting of d, U/L, and β, we generate 1,000 random instances as follows. First, each term of the weight vector $\mathbf{w}$ for the weighted ℓ_1 norm is drawn randomly from the uniform distribution on $[0, \beta]$. Next, the time horizon T is generated randomly from a uniform distribution on $[6, 24]$. For each time $t \in [T]$, a cost function is generated as follows: Let $f_t(\mathbf{x}) = \mathbf{f}_t^\intercal \mathbf{x}$, where $\mathbf{f}_t$ is a d-dimensional *cost vector*. To generate $\mathbf{f}_t$, we first draw μ_t from the uniform distribution on $[L, U]$, and then draw each term of $\mathbf{f}_t$ from a normal distribution centered at μ_t with standard deviation σ (i.e., $\mathbf{f}_t^i \sim \mathcal{N}(\mu_t, \sigma)$). Any terms which are outside the assumed interval $[L, U]$ (i.e. $\mathbf{f}_t^i < L$ or $\mathbf{f}_t^i > U$) are truncated appropriately. For each instance, we report the empirical competitive ratios as the evaluation metric, comparing the tested algorithms against an offline optimal benchmark. We give results for the average empirical competitive ratio in the main body, with supplemental results for the 95$^{\text{th}}$ percentile ("worst-case") empirical competitive ratio in Appendix A.1.

In the setting with advice, we construct simulated advice as follows: Let $\xi \in [0,1]$ denote an *adversarial factor*. When $\xi = 0$, ADV gives the optimal solution, and when $\xi = 1$, ADV is fully adversarial. Formally, letting $\{\mathbf{x}_t^\star : t \in [T]\}$ denote the decisions made by an optimal solution, and letting $\{\check{\mathbf{x}}_t : t \in [T]\}$ represent the decisions made by a solution which maximizes the objective (rather than minimizing it), we have that $\texttt{ADV} = \{(1-\xi)\mathbf{x}_t^\star + \xi\check{\mathbf{x}}_t : t \in [T]\}$. We note that although $\{\check{\mathbf{x}}_t : t \in [T]\}$ is adversarial from the perspective of the objective, it is still a feasible solution for the problem (i.e., it satisfies the long-term constraint).

Comparison algorithms. We use CVXPY [DB16] to compute the offline optimal solution for each instance using a convex optimization solver with access to all cost functions in advance. This provides the empirical competitive ratio for each algorithm. We consider three online heuristic techniques based on the literature for related problems. The first technique is termed "**agnostic**", which chooses the minimum dimension of the cost function in the first time step $t = 1$ (i.e., $k = \arg\min_{i\in[d]} \mathbf{c}_1^i$), sets $\mathbf{x}_1^k = 1$, and $\mathbf{x}_t = \mathbf{0}\ \forall t > 1$. The second technique is termed "**move to minimizer**", which takes inspiration from algorithms for CFC [ZJL⁺21] and satisfies $1/T$ fraction of the long-term constraint at each time step by moving to the minimum dimension of each cost function. Formally, at each time step t, letting $k_t = \arg\min_{i\in[d]} \mathbf{c}_t^i$, "move to minimizer" sets $\mathbf{x}_t^{k_t} = 1/T$. Finally, the third technique is termed "**simple threshold**", which takes inspiration from algorithms for online search [EFK⁺01]. This algorithm sets a fixed threshold $\psi = \sqrt{UL}$, and completes the long-term constraint at the first time step and dimension where the hitting cost does not exceed ψ. Formally, at the first time step τ satisfying $\exists\ k \in [d] : \mathbf{f}_\tau^k \le \psi$, "simple threshold" sets $\mathbf{x}_\tau^k = 1$. Importantly, *none* of these heuristics are accompanied by traditional competitive guarantees, since our work is the first to consider CFL. In the setting with advice, we compare our proposed CLIP learning-augmented algorithm against the Baseline learning-augmented algorithm described in Section 4 (e.g., Lemma 4.1).

Experimental results. Figure 1 summarizes the main results for ALG1, the comparison algorithms, and one setting of CLIP ($\epsilon = 2$) in a CDF plot of the empirical competitive ratios across several experiments. Here we fix $U/L = 250$, $\xi = 0$, $\sigma = 50$, while varying β and d. ALG1 outperforms in both average-case and worst-case performance, improving on the closest "simple threshold" by an average of 18.2%, and outperforming "agnostic" and "move to minimizer" by averages of 56.1% and 71.5%, respectively. With *correct* advice, CLIP sees significant performance gains everywhere.

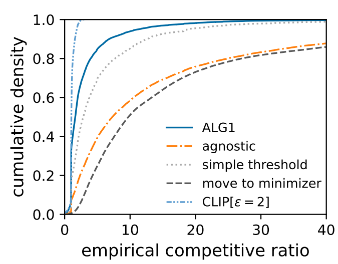

Figure 1: CDFs of empirical competitive ratios for various algorithms.

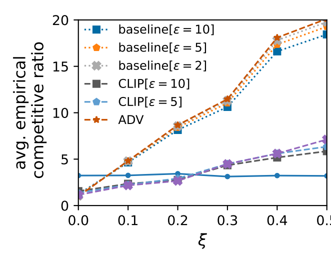

Figure 2: Varying adversarial factor ξ, with $U/L = 250$, $\beta = 50$, $d = 5$, and $\sigma = 50$.

In Figure 3-6, we investigate the impact of parameters on the average empirical competitive ratio for each algorithm. In Appendix A.1, we give corresponding plots for the 95th percentile ("worst-case") results. Figure 3 plots competitive ratios for different values of U/L. We fix $\beta = U/5$, $d = 5$, $\xi = 0$, $\sigma = U/5$, while varying U/L. Since there is a dependence on U/L in our competitive results, the performance of ALG1 degrades as U/L grows, albeit at a favorable pace compared to the heuristics. Figure 4 plots competitive ratios for different values of β. We fix $U/L = 250$, $d = 5$, $\xi = 0$, $\sigma = 50$. As β grows, the "agnostic" and "move to minimizer" heuristics improve because the switching cost paid by OPT grows.

In Figure 5, we plot competitive ratios for different values of d. We fix $U/L = 250$, $\beta = 50$, $\xi = 0$, $\sigma = 50$, while varying d. As d grows, ALG1 and CLIP's performance degrades slower compared to the heuristics, as predicted by their *dimension-free* theoretical bounds. Finally, Figure 6 plots competitive ratios for different

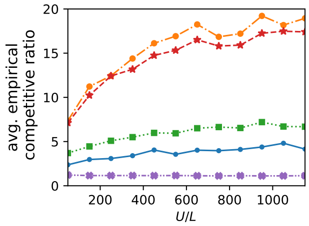

Figure 3: Varying U/L, with $\beta = U/5, d = 5, \xi = 0$, and $\sigma = U/5$.

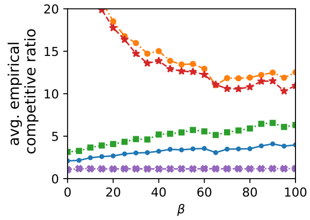

Figure 4: Varying β, with $U/L = 250, d = 5, \xi = 0$, and $\sigma = 50$.

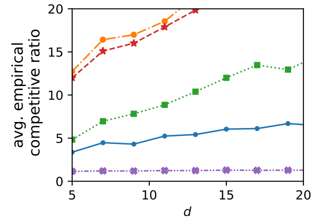

Figure 5: Varying d with $\beta = 50, U/L = 250, \sigma = 50$, and ξ=0.

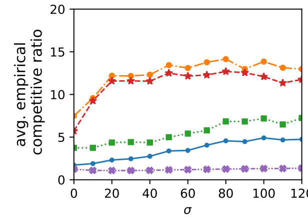

Figure 6: Varying σ, with $\beta = 50, U/L = 250, d = 5$, and $\xi = 0$.

values of σ. We fix $U/L = 250, \beta = 50, d = 5, \xi = 0$, while varying σ. As cost functions become more variable, the performance of all algorithms degrades, with the exception of CLIP. There is a plateau as σ grows, because a large σ implies that more terms in each $\mathbf{f}_t$ must be truncated to the interval $[L, U]$.

Figure 2 plots the effect of prediction error on the learning-augmented algorithms CLIP and Baseline. We test several values of $\xi \in [0, 1/2]$ (recall that $\xi = 0$ recovers correct advice), while fixing $U/L = 250, \beta = 50, d = 5$, and $\sigma = 50$. We also test Baseline and CLIP for several values of $\epsilon \in \{2, 5, 10\}$ (note that ADV corresponds to Baseline and CLIP with $\epsilon = 0$). Notably, we find that CLIP significantly outperforms the Baseline algorithm as ξ grows, showing an average improvement of 60.8% when $\xi > 0.1$. This result implies that CLIP is more empirically robust to prediction errors than the simple fixed ratio technique of Baseline.

6 Conclusion

We study online metric problems with long-term constraints, motivated by emerging problems in sustainability. These are the first such problems to concurrently incorporate multidimensional decision spaces, switching costs, and long-term demand constraints. Our main results instantiate the CFL and MAL problems towards a motivating application. We design competitive and learning-augmented algorithms, show that their performance bounds are tight, and validate them in numerical experiments. Several interesting open questions are prompted by our work. Specifically, (i) what is achievable in non-ℓ_1 vector spaces e.g., the Euclidean setting, and (ii) can our results for MAL inform algorithm designs for e.g., tree metrics, and by extension, arbitrary metric spaces?

References

[AGG20] C. J. Argue, Anupam Gupta, and Guru Guruganesh. "Dimension-Free Bounds for Chasing Convex Functions". In: *Proceedings of Thirty Third Conference on Learning Theory*. PMLR, July 2020, pp. 219–241. (Visited on 02/04/2022).

[ALK+23] Bilge Acun, Benjamin Lee, Fiodar Kazhamiaka, Kiwan Maeng, Udit Gupta, Manoj Chakkaravarthy, David Brooks, and Carole-Jean Wu. "Carbon Explorer: A Holistic Framework for Designing Carbon Aware Datacenters". In: *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*. ASPLOS 2023. Vancouver, BC, Canada: Association for Computing Machinery, 2023, pp. 118–132. ISBN: 9781450399166. DOI: 10.1145/3575693.3575754. URL: https://doi.org/10.1145/3575693.3575754.

[BC22] Nikhil Bansal and Christian Coester. "Online Metric Allocation and Time-Varying Regularization". In: *30th Annual European Symposium on Algorithms (ESA 2022)*. Ed. by Shiri Chechik, Gonzalo Navarro, Eva Rotenberg, and Grzegorz Herman. Vol. 244. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022, 13:1–13:13. ISBN: 978-3-95977-247-1. DOI: 10.4230/LIPIcs.ESA.2022.13. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ESA.2022.13.

[BCL+21] Sébastien Bubeck, Michael B. Cohen, James R. Lee, and Yin Tat Lee. "Metrical Task Systems on Trees via Mirror Descent and Unfair Gluing". In: *SIAM Journal on Computing* 50.3 (Jan. 2021), pp. 909–923. ISSN: 0097-5397, 1095-7111. DOI: 10.1137/19M1237879.

[BCR23] Sébastien Bubeck, Christian Coester, and Yuval Rabani. "The Randomized k-Server Conjecture Is False!" In: *Proceedings of the 55th Annual ACM Symposium on Theory of Computing (STOC 2023)*. STOC 2023. Orlando, FL, USA: Association for Computing Machinery, 2023, pp. 581–594. ISBN: 9781450399135. DOI: 10.1145/3564246.3585132. URL: https://doi.org/10.1145/3564246.3585132.

[BGH+21] Noman Bashir, Tian Guo, Mohammad Hajiesmaili, David Irwin, Prashant Shenoy, Ramesh Sitaraman, Abel Souza, and Adam Wierman. "Enabling Sustainable Clouds: The Case for Virtualizing the Energy System". In: *Proceedings of the ACM Symposium on Cloud Computing*. SoCC '21. Seattle, WA, USA: Association for Computing Machinery, 2021, pp. 350–358. ISBN: 9781450386388. DOI: 10.1145/3472883.3487009. URL: https://doi.org/10.1145/3472883.3487009.

[BKL+19] Sébastien Bubeck, Bo'az Klartag, Yin Tat Lee, Yuanzhi Li, and Mark Sellke. "Chasing Nested Convex Bodies Nearly Optimally". In: *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Proceedings. Society for Industrial and Applied Mathematics, Dec. 2019, pp. 1496–1508. DOI: 10.1137/1.9781611975994.91.

[BLS92] Allan Borodin, Nathan Linial, and Michael E. Saks. "An Optimal On-Line Algorithm for Metrical Task System". In: *J. ACM* 39.4 (Oct. 1992), pp. 745–763. ISSN: 0004-5411. DOI: 10.1145/146585.146588. URL: https://doi.org/10.1145/146585.146588.

[CBS+22] Kai-Wen Cheng, Yuexin Bian, Yuanyuan Shi, and Yize Chen. "Carbon-Aware EV Charging". In: *2022 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. 2022, pp. 186–192. DOI: 10.1109/SmartGridComm52983.2022.9960988.

[CGH+96] Robert M Corless, Gaston H Gonnet, David EG Hare, David J Jeffrey, and Donald E Knuth. "On the Lambert W function". In: *Advances in Computational mathematics* 5 (1996), pp. 329–359.

[CGW18] NiangJun Chen, Gautam Goel, and Adam Wierman. "Smoothed Online Convex Optimization in High Dimensions via Online Balanced Descent". In: *Proceedings of the 31st Conference On Learning Theory*. PMLR, July 2018, pp. 1574–1594.

[CHW22] Nicolas Christianson, Tinashe Handina, and Adam Wierman. "Chasing Convex Bodies and Functions with Black-Box Advice". In: *Proceedings of the 35th Conference on Learning Theory*. Vol. 178. PMLR, July 2022, pp. 867–908.

[CSW23] Nicolas Christianson, Junxuan Shen, and Adam Wierman. "Optimal robustness-consistency tradeoffs for learning-augmented metrical task systems". In: *International Conference on Artificial Intelligence and Statistics*. 2023.

[DB16] Steven Diamond and Stephen Boyd. "CVXPY: A Python-embedded modeling language for convex optimization". In: *Journal of Machine Learning Research* 17.83 (2016), pp. 1–5.

[EFK+01] Ran El-Yaniv, Amos Fiat, Richard M. Karp, and G. Turpin. "Optimal Search and One-Way Trading Online Algorithms". In: *Algorithmica* 30.1 (May 2001), pp. 101–139. DOI: 10.1007/s00453-001-0003-0. URL: https://doi.org/10.1007/s00453-001-0003-0.

[FL93] Joel Friedman and Nathan Linial. "On convex body chasing". In: *Discrete & Computational Geometry* 9.3 (Mar. 1993), pp. 293–321. DOI: 10.1007/bf02189324. URL: https://doi.org/10.1007/bf02189324.

[HLB+23] Walid A. Hanafy, Qianlin Liang, Noman Bashir, David Irwin, and Prashant Shenoy. "Carbon-Scaler: Leveraging Cloud Workload Elasticity for Optimizing Carbon-Efficiency". In: *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 7.3 (Dec. 2023). arXiv: 2302.08681 [cs.DC].

[Kou09] Elias Koutsoupias. "The k-server problem". In: *Computer Science Review* 3.2 (May 2009), pp. 105–118. DOI: 10.1016/j.cosrev.2009.04.002. URL: https://doi.org/10.1016/j.cosrev.2009.04.002.

[LCS+24] Adam Lechowicz, Nicolas Christianson, Bo Sun, Noman Bashir, Mohammad Hajiesmaili, Adam Wierman, and Prashant Shenoy. "Online Conversion with Switching Costs: Robust and Learning-augmented Algorithms". In: *Proceedings of the 2024 SIGMETRICS/Performance Joint International Conference on Measurement and Modeling of Computer Systems*. Venice, Italy: Association for Computing Machinery, June 2024. arXiv: 2310.20598 [cs.DS].

[LCZ+23] Adam Lechowicz, Nicolas Christianson, Jinhang Zuo, Noman Bashir, Mohammad Hajiesmaili, Adam Wierman, and Prashant Shenoy. "The Online Pause and Resume Problem: Optimal Algorithms and An Application to Carbon-Aware Load Shifting". In: *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 7.3 (Dec. 2023). arXiv: 2303.17551 [cs.DS].

[LPS08] Julian Lorenz, Konstantinos Panagiotou, and Angelika Steger. "Optimal Algorithms for k-Search with Application in Option Pricing". In: *Algorithmica* 55.2 (Aug. 2008), pp. 311–328. DOI: 10.1007/s00453-008-9217-8.

[LSH+24] Russell Lee, Bo Sun, Mohammad Hajiesmaili, and John C. S. Lui. "Online Search with Predictions: Pareto-optimal Algorithm and its Applications in Energy Markets". In: *Proceedings of the 15th ACM International Conference on Future Energy Systems*. e-Energy '24. Singapore, Singapore: Association for Computing Machinery, June 2024.

[LV18] Thodoris Lykouris and Sergei Vassilvtiskii. "Competitive Caching with Machine Learned Advice". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, July 2018, pp. 3296–3305. URL: https://proceedings.mlr.press/v80/lykouris18a.html.

[MAS14] Esther Mohr, Iftikhar Ahmad, and Günter Schmidt. "Online algorithms for conversion problems: A survey". In: *Surveys in Operations Research and Management Science* 19.2 (July 2014), pp. 87–104. DOI: 10.1016/j.sorms.2014.08.001. URL: https://doi.org/10.1016/j.sorms.2014.08.001.

[MMS88] Mark Manasse, Lyle McGeoch, and Daniel Sleator. "Competitive Algorithms for On-Line Problems". In: *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*. STOC '88. Chicago, Illinois, USA: Association for Computing Machinery, 1988, pp. 322–333. ISBN: 0897912640. DOI: 10.1145/62212.62243.

[MPF91] Dragoslav S. Mitrinovic, Josip E. Pečarić, and A. M. Fink. *Inequalities Involving Functions and Their Integrals and Derivatives*. Vol. 53. Springer Science & Business Media, 1991.

[PSK18] Manish Purohit, Zoya Svitkina, and Ravi Kumar. “Improving Online Algorithms via ML Predictions”. In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Vol. 31. Curran Associates, Inc., 2018.

[RKS+22] Ana Radovanovic, Ross Koningstein, Ian Schneider, Bokan Chen, Alexandre Duarte, Binz Roy, Diyue Xiao, Maya Haridasan, Patrick Hung, Nick Care, et al. “Carbon-Aware Computing for Datacenters”. In: *IEEE Transactions on Power Systems* (2022).

[Sel20] Mark Sellke. “Chasing Convex Bodies Optimally”. In: *Proceedings of the Thirty-First Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA ’20. USA: Society for Industrial and Applied Mathematics, Jan. 2020, pp. 1509–1518.

[SLH+21] Bo Sun, Russell Lee, Mohammad Hajiesmaili, Adam Wierman, and Danny Tsang. “Pareto-Optimal Learning-Augmented Algorithms for Online Conversion Problems”. In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan. Vol. 34. Curran Associates, Inc., 2021, pp. 10339–10350.

[SZL+21] Bo Sun, Ali Zeynali, Tongxin Li, Mohammad Hajiesmaili, Adam Wierman, and Danny H.K. Tsang. “Competitive Algorithms for the Online Multiple Knapsack Problem with Application to Electric Vehicle Charging”. In: *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 4.3 (June 2021). DOI: 10.1145/3428336. URL: https://doi.org/10.1145/3428336.

[WBS+21] Philipp Wiesner, Ilja Behnke, Dominik Scheinert, Kordian Gontarska, and Lauritz Thamsen. “Let’s Wait AWhile: How Temporal Workload Shifting Can Reduce Carbon Emissions in the Cloud”. In: *Proceedings of the 22nd International Middleware Conference*. New York, NY, USA: Association for Computing Machinery, 2021, pp. 260–272. DOI: 10.1145/3464298.3493399.

[ZCL08] Yunhong Zhou, Deeparnab Chakrabarty, and Rajan Lukose. “Budget Constrained Bidding in Keyword Auctions and Online Knapsack Problems”. In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2008, pp. 566–576.

[ZJL+21] Lijun Zhang, Wei Jiang, Shiyin Lu, and Tianbao Yang. *Revisiting Smoothed Online Learning*. 2021. arXiv: 2102.06933 [cs.LG]. URL: https://arxiv.org/abs/2102.06933.

Appendix

A Numerical Experiments (continued)

In this section, we give supplemental results examining the 95th percentile ("worst-case") empirical competitive ratio results, following the same general structure as in the main body.

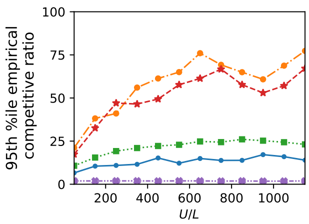

Figure 7: Varying U/L, with $\beta = U/5, d = 5, \xi = 0$, and $\sigma = U/5$.

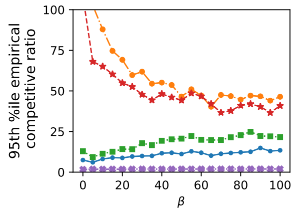

Figure 8: Varying β, with $U/L = 250, d = 5, \xi = 0$, and $\sigma = 50$.

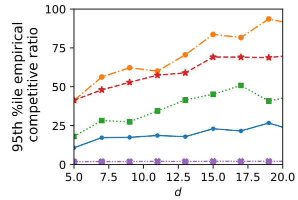

Figure 9: Varying d with $\beta = 50, U/L = 250, \sigma = 50$, and ξ=0.

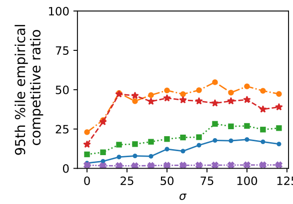

Figure 10: Varying σ, with $\beta = 50, U/L = 250, d = 5$, and $\xi = 0$.

A.1 Supplemental Results

To complement the results for the average empirical competitive ratio shown in Section 5, in this section we plot the 95th percentile empirical competitive ratios for each tested algorithm, which primarily serve to show that the improved performance of our proposed algorithm holds in both average-case and tail ("worst-case") scenarios.

In Figure 7-10, we investigate the impact of different parameters on the performance of each algorithm. In Figure 7, we plot 95th percentile empirical competitiveness for different values of U/L – in this experiment, we fix $\beta = U/5, d = 5, \xi = 0$, and $\sigma = U/5$, while varying $U/L \in \{50, \dots, 1250\}$. As observed in the average competitive ratio plot (Figure 3), the performance of ALG1 degrades as U/L grows, albeit at a favorable pace compared to the comparison algorithms. Figure 8 plots the 95th percentile empirical competitiveness for different values of β – in this experiment, we fix $U/L = 250, d = 5, \xi = 0$, and $\sigma = 50$. As previously in the average competitive results (Figure 4), "agnostic" and "move to minimizer" heuristics perform better when β grows, because the switching cost paid by the optimal solution grows as well.

In Figure 9, we plot the 95th percentile empirical competitiveness for different values of d – in this experiment, we fix $U/L = 250, \beta = 50, \xi = 0$, and $\sigma = 50$, while varying d. Mirroring the previous results (Figure 5), ALG1 and CLIP's competitive performance degrades slower as d grows compared to the comparison heuristics, as predicted by their *dimension-free* theoretical bounds. Finally, Figure 10 plots the 95th percentile empirical competitiveness for different values of σ, which is the dimension-wise variability of each cost function. Here we fix $U/L = 250, \beta = 50, d = 5$, and $\xi = 0$, while varying $\sigma \in \{0, \dots, U/2\}$. Intuitively, as cost functions become more variable, the competitive ratios of all tested algorithms degrade, with the exception of our learning-augmented algorithm CLIP. This degradation plateaus as σ grows, as a large standard deviation forces more of the terms of each cost vector $\mathbf{c}_t$ to be truncated to the interval $[L, U]$.

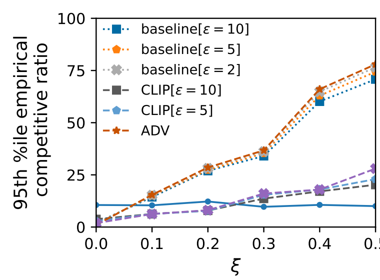

Figure 11: Varying adversarial factor ξ, with $U/L = 250, \beta = 50, d = 5, \sigma = 50$.

In Figure 11, we plot the 95th percentile empirical competitive ratio companion to Figure 2, which measures the effect of prediction error on the learning-augmented algorithms CLIP and Baseline. We test several values of $\xi \in [0, 1]$, the adversarial factor (recall that $\xi = 0$ implies the advice is correct), while fixing $U/L = 250, \beta = 50, d = 5, \sigma = 50$. We test `Baseline` and `CLIP` for several values of $\epsilon \in \{2, 5, 10\}$ (note that `ADV` corresponds to running either `Baseline` or `CLIP` with $\epsilon = 0$). Notably, in these 95th percentile "worst-case" results, we find that `CLIP` continues to significantly outperforms the `Baseline` algorithm as ξ grows, further validating that `CLIP` is more empirically robust to prediction errors than the simple fixed ratio technique of `Baseline`.

B Proofs for Section 3 (Competitive Algorithms)

B.1 Convexity of the pseudo-cost minimization problem in ALG1

In this section, we show that the pseudo-cost minimization problem central to the design of ALG1 is a convex minimization problem, implying that it can be solved efficiently.

Define $h_t(\mathbf{x}) : t \in [T]$ to represent the pseudo-cost minimization problem for a single arbitrary time step:

$$h_t(\cdot) = f_t(\mathbf{x}) + d(\mathbf{x}, \mathbf{x}_{t-1}) - \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x})} \phi(u)du. \tag{10}$$

Theorem B.1. *Under the assumptions of the* CFL *and* MAL *problem settings, $h_t(\cdot)$ is always convex.*

Proof. We prove the above statement by contradiction.

By definition, we know that the sum of two convex functions gives a convex function. Since we have that $d(\mathbf{x}, \mathbf{x}')$ is defined as some norm, by definition and by observing that $\mathbf{x}'$ is fixed, $d(\mathbf{x}, \mathbf{x}')$ is convex. We have also assumed as part of the problem setting that each $f_t(\mathbf{x})$ is convex. Thus, $f_t(\mathbf{x}) + d(\mathbf{x}, \mathbf{x}')$ must be convex.

We turn our attention to the term $-\int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x})} \phi(u)du$. Let $k(c(\mathbf{x})) = \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x})} \phi(u)du$. By the fundamental theorem of calculus, $\nabla k(c(\mathbf{x})) = \phi(z^{(t-1)} + c(\mathbf{x}))\nabla c(\mathbf{x})$

Let $g(c(\mathbf{x})) = \phi(z^{(t-1)} + c(\mathbf{x}))$. Then $\nabla^2 k(c(\mathbf{x})) = \nabla^2 c(\mathbf{x}) k(c(\mathbf{x})) + \nabla c(\mathbf{x}) g'(c(\mathbf{x})) \nabla c(\mathbf{x})^\intercal$. Since $c(\mathbf{x})$ is piecewise linear (CFL and MAL both assume it is linear), we know that $\nabla^2 c(\mathbf{x}) g(c(\mathbf{x})) = 0$. Since ϕ is monotonically decreasing on the interval $[0, 1]$, we know that $g'(c(\mathbf{x})) < 0$, and thus $\nabla c(\mathbf{x}) g'(c(\mathbf{x})) \nabla c(\mathbf{x})^\intercal$ is negative semidefinite. This implies that $k(c(\mathbf{x}))$ is concave in $\mathbf{x}$.

Since the negation of a concave function is convex, this causes a contradiction, because the sum of two convex functions gives a convex function.

Thus, $h_t(\cdot) = f_t(\mathbf{x}) + d(\mathbf{x}, \mathbf{x}_{t-1}) - \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x})} \phi(u)du$ is always convex under the assumptions of CFL and MAL. □

By showing that $h_t(\cdot)$ is convex, it follows that the pseudo-cost minimization (2) in ALG1 is a convex minimization problem (i.e., it can be solved efficiently using numerical methods).

B.2 Proof of Theorem 3.2

In this section, we prove Theorem 3.2, which shows that α as given by (4) is an upper bound on the worst-case competitive ratio of ALG1 (given by Algorithm 1) for the CFL problem.

Proof of Theorem 3.2. Let $z^{(j)} = \sum_{t\in[T]} c(\mathbf{x}_t)$ denote the fraction of the long-term constraint satisfied by ALG1 before the compulsory trade on an arbitrary CFL instance $\mathcal{I} \in \Omega$. Also note that $z^{(t)} = \sum_{m\in[t]} c(\mathbf{x}_m)$ is non-decreasing over n.

Lemma B.2. *The offline optimal solution* OPT$(\mathcal{I})$ *for any* CFL *instance $\mathcal{I} \in \Omega$ is lower bounded by $\phi(z^{(j)}) - \beta$.*

Proof of Lemma B.2. We prove this lemma by contradiction. Note that the offline optimum will stay at $\mathbf{0}$ whenever possible, and satisfy the long-term constraint using the cost functions with the minimum gradient (i.e., the best marginal cost). Assume that $\texttt{OPT}(\mathcal{I}) < \phi(z^{(j)}) - \beta$, and that $z^{(j)} < 1$ (implying that $\texttt{OPT}(\mathcal{I}) > L$).

Recall that any cost function $f_t(\cdot) : X \to \mathbb{R}$ is minimized exactly at $\mathbf{0}$, since $f_t(\mathbf{0}) = 0 \ \forall t \in [T]$. By convexity of the cost functions, this implies that the gradient of some cost function f_t is similarly minimized at the point $\mathbf{0}$, and thus the *best marginal cost* for f_t can be obtained by taking an infinitesimally small step away from $\mathbf{0}$ in at least one direction, which we denote (without loss of generality) as $i \in [d]$. For brevity, we denote this best marginal cost in f_t by $[\nabla f_t]^i$.

The assumption that $\texttt{OPT}(\mathcal{I}) < \phi(z^{(j)}) - \beta$ implies that instance $\mathcal{I}$ must contain a cost function $f_m(\cdot)$ at some arbitrary time step m ($m \in [T]$) which satisfies $[\nabla f_m]^i < \phi(z^{(j)}) - \beta$ for any dimension $i \in [d]$.

Prior work [LPS08; SZL+21] has shown that the worst-case for online search problems with long-term demand constraints occurs when cost functions arrive online in descending order, so we henceforth adopt this assumption. Recall that at each time step, ALG1 solves the pseudo-cost minimization problem defined in (2). Without loss of generality, assume that $z^{(m-1)} = z^{(j)}$, i.e. the cost function $f_m(\cdot)$ arrives when ALG1 has already reached its final utilization (before the compulsory trade). This implies that $\mathbf{x}_m = \mathbf{0}$, and further that $c(\mathbf{x}_m) = 0$. This implies that $f_m(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{m-1}\|_{\ell_1(\mathbf{w})} > \int_{z^{(m-1)}}^{z^{(m-1)}+c(\mathbf{x})} \phi(u)du$, since the pseudo-cost minimization problem should be minimized when ALG1 sets $\mathbf{x}_m = \mathbf{0}$.

The pseudo-cost minimization problem at time step m can be expressed as follows:

$$\mathbf{x}_m = \underset{\mathbf{x} \in \mathbb{R}^d : c(\mathbf{x}) \le 1 - z^{(m-1)}}{\arg\min} f_m(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{m-1}\|_{\ell_1(\mathbf{w})} - \int_{z^{(m-1)}}^{z^{(m-1)}+c(\mathbf{x})} \phi(u)du.$$

We note that $\|\mathbf{x} - \mathbf{x}_{m-1}\|_{\ell_1(\mathbf{w})}$ is upper bounded by $\beta(z^{(m-1)} + c(\mathbf{x}))$, since in the worst case, the previous online decision $\mathbf{x}_{m-1}$ built up all of ALG1's utilization ($z^{(m-1)}$) so far, and in the next step it will have to switch dimensions to ramp up to $\mathbf{x}$.

Since the function ϕ is monotonically decreasing on $z \in [0, 1]$, the $\mathbf{x}_m$ solving the true pseudo-cost minimization problem is lower-bounded by the $\check{\mathbf{x}}_m$ solving the following minimization problem (i.e., $c(\check{\mathbf{x}}_m) \le c(\mathbf{x}_m)$):

$$\check{\mathbf{x}}_m = \underset{\mathbf{x} \in \mathbb{R}^d : c(\mathbf{x}) \le 1 - z^{(m-1)}}{\arg\min} f_m(\mathbf{x}) + \beta(z^{(m-1)} + c(\mathbf{x})) - \int_{z^{(m-1)}}^{z^{(m-1)}+c(\mathbf{x})} \phi(u)du.$$

This further gives the following:

$$f_m(\mathbf{x}) + \beta(z^{(t)} + c(\mathbf{x})) - \int_{z^{(m-1)}}^{z^{(m-1)}+c(\mathbf{x})} \phi(u)du$$

$$f_m(\mathbf{x}) + \beta(z^{(t)} + c(\mathbf{x})) - \int_{z^{(m-1)}}^{z^{(m-1)}+c(\mathbf{x})} \left[U - \beta + \left(\frac{U}{\alpha} - U + 2\beta \right) \exp(u/\alpha) \right] du$$

$$f_m(\mathbf{x}) - (U - \beta)c(\mathbf{x}) + \beta(z^{(t)} + c(\mathbf{x})) - [U - U\alpha + 2\beta\alpha] \left(\exp\left(\frac{z^{(m-1)} + c(\mathbf{x})}{\alpha} \right) - \exp\left(\frac{z^{(m-1)}}{\alpha} \right) \right)$$

By assumption, since $f_m(\cdot)$ is convex and satisfies $[\nabla f_m]^i < \phi(z^{(j)}) - \beta$ at $\mathbf{x} = \mathbf{0}$, there must exist a dimension i in f_m where an incremental step away from $\mathbf{0}$ in direction i satisfies the following inequality: $f_m(\mathbf{x}) \lesssim [\nabla f_m]^i \cdot c(\mathbf{x}) < [\phi(z^{(j)}) - \beta] c(\mathbf{x})$ for some $\mathbf{x}$ where $c(\mathbf{x}) > 0$. Thus, we have the following in the pseudo-cost minimization problem:

$$([\nabla f_m]^i - U + \beta)c(\mathbf{x}) + \beta(z^{(t)} + c(\mathbf{x})) - [U - U\alpha + 2\beta\alpha] \left(\exp\left(\frac{z^{(m-1)} + c(\mathbf{x})}{\alpha} \right) - \exp\left(\frac{z^{(m-1)}}{\alpha} \right) \right)$$

Letting $c(x)$ be some scalar y (which is valid since we assume there is at least one dimension in $f_t(\cdot)$ where the cost function growth rate is at most ∇f_m), the pseudo-cost minimization problem finds the value y which minimizes the following quantity:

$$([\nabla f_m]^i - U + \beta)y + \beta(z^{(t)} + y) - [U - U\alpha + 2\beta\alpha]\left(\exp\left(\frac{z^{(m-1)} + y}{\alpha}\right) - \exp\left(\frac{z^{(m-1)}}{\alpha}\right)\right)$$

Taking the derivative of the above with respect to y yields the following:

$$\frac{d}{dy}\left[([\nabla f_m]^i - U + \beta)y + \beta(z^{(t)} + y) - [U - U\alpha + 2\beta\alpha]\left(\exp\left(\frac{z^{(m-1)} + y}{\alpha}\right) - \exp\left(\frac{z^{(m-1)}}{\alpha}\right)\right)\right] =$$

$$= [\nabla f_m]^i + 2\beta - U + \frac{(U\alpha - 2\alpha\beta - U)\exp\left(\frac{z^{(m-1)}+y}{\alpha}\right)}{\alpha}$$

If $y = 0$, we have the following by assumption that $[\nabla f_m]^i < \phi(z^{(j)}) - \beta$ and that $z^{(j)} = z^{(m-1)}$:

$$[\nabla f_m]^i + 2\beta - U + (U - 2\beta - U/\alpha)\exp\left(\frac{z^{(m-1)}}{\alpha}\right) < \phi(z^{(j)}) + \beta - U + (U - 2\beta - U/\alpha)\exp\left(\frac{z^{(m-1)}}{\alpha}\right)$$
$$< \phi(z^{(j)}) - \left(\phi(z^{(m-1)})\right) = 0$$

The above derivation implies that the derivative of the cost minimization problem at $c(\mathbf{x}) = 0$ (which corresponds to the case where $\mathbf{x} = \mathbf{0}$) is strictly less than 0. This further implies that $\check{\mathbf{x}}_m$ must be *non-zero*, since the minimizer must satisfy $c(\check{\mathbf{x}}_m) > 0$. Since $c(\check{\mathbf{x}}_m)$ lower bounds the true $c(\mathbf{x}_m)$, this causes a contradiction, as it was assumed that the utilization after time step m would satisfy $z^{(m)} = z^{(m-1)} = z^{(j)}$, but if $c(\mathbf{x}_m) > 0$, $z^{(m)}$ must satisfy $z^{(m)} > z^{(m-1)}$.

It then follows by contradiction that $\texttt{OPT}(\mathcal{I}) \geq \phi(z^{(j)}) - \beta$.

Lemma B.3. *The cost of* ALG1 *on any valid* CFL *instance* $\mathcal{I} \in \Omega$ *is upper bounded by*

$$\texttt{ALG1}(\mathcal{I}) \leq \int_0^{z^{(j)}} \phi(u)du + \beta z^{(j)} + (1 - z^{(j)})U. \tag{11}$$

Proof of Lemma B.3. First, recall that $z^{(t)} = \sum_{\tau\in[t]} c(\mathbf{x}_\tau)$ is non-decreasing over $t \in [T]$.

Observe that whenever $c(\mathbf{x}_t) > 0$, we know that $f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} < \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x}_t)} \phi(u)du$. Then, if $c(\mathbf{x}_t) = 0$, which corresponds to the case when $\mathbf{x}_t = \mathbf{0}$, we have the following:

$$f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} - \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x}_t)} \phi(u)du = 0 + \|-\mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} - 0 \leq \beta c(\mathbf{x}_{t-1})$$

This gives that for any time step where $c(\mathbf{x}_t) = 0$, we have the following inequality:

$$f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} \leq \beta c(\mathbf{x}_{t-1}), \forall t \in [T] : c(\mathbf{x}_t) = 0. \tag{12}$$

And thus, since any time step where $c(\mathbf{x}_t) > 0$ implies $f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} < \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x}_t)} \phi(u)du$, we have the following inequality for all time steps (*i.e., an upper bound on the excess cost not accounted for in the pseudo-cost threshold function or compulsory trade*)

$$f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} - \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x}_t)} \phi(u)du \leq \beta c(\mathbf{x}_{t-1}), \forall t \in [T]. \tag{13}$$

Thus, we have

$$\beta z^{(j)} = \sum_{t\in[j]} \beta c(\mathbf{x}_{t-1}) \geq \sum_{t\in[j]} \left[f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} - \int_{z^{(t-1)}}^{z^{(t-1)}+c(\mathbf{x}_t)} \phi(u)du \right] \tag{14}$$

$$= \sum_{t\in[j]} \left[f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell 1(\mathbf{w})} \right] - \int_0^{z^{(j)}} \phi(u)du \tag{15}$$

$$= \mathtt{ALG1} - (1 - z^{(j)})U - \int_0^{z^{(j)}} \phi(u)du. \tag{16}$$

Combining Lemma B.2 and Lemma B.3 gives

$$\frac{\mathtt{ALG1}(\mathcal{I})}{\mathtt{OPT}(\mathcal{I})} \leq \frac{\int_0^{z^{(j)}} \phi(u)du + \beta z^{(j)} + (1 - z^{(j)})U}{\phi(z^{(j)}) - \beta} \leq \alpha, \tag{17}$$

where the last inequality holds since for any $z \in [0, 1]$

$$\int_0^z \phi(u)du + \beta z + (1 - z)U = \int_0^z [U - \beta + (U/\alpha - U + 2\beta)\exp(z/\alpha)] + \beta z + (1 - z)U \tag{18}$$

$$= (U - \beta)z + \alpha(U/\alpha - U + 2\beta)[\exp(z/\alpha) - 1] + \beta z + (1 - z)U \tag{19}$$

$$= \alpha(U/\alpha - U + 2\beta)[\exp(z/\alpha) - 1] + U \tag{20}$$

$$= \alpha\left[U - 2\beta + (U/\alpha - U + 2\beta)\exp(z/\alpha)\right] \tag{21}$$

$$= \alpha[\phi(z) - \beta]. \tag{22}$$

Thus, we conclude that ALG1 is α-competitive for CFL. □

B.3 Proof of Corollary 3.3

In this section, we prove Corollary 3.3, which shows that the worst-case competitive ratio of ALG1 for MAL is again upper bounded by α as defined in (4).

Proof of Corollary 3.3. To show this result, we first prove a result stated in the main body, namely Lemma 2.2, which states the following: For any MAL instance on a weighted star metric (X, d), there is a corresponding CFL instance on $(\mathbb{R}^{n-1}, \|\cdot\|_{\ell_1(\mathbf{w}')})$ which preserves $f_t^a(\cdot)\ \forall t, c(\cdot)\ \forall a \in X$, and upper bounds $d(a, b)\ \forall(a, b) \in X$.

Before the proof, we note that Bansal and Coester [BC22] showed online metric allocation on a weighted star metric (X, d) is identical to convex function chasing (with separable cost functions) on the normed vector space $(\Delta_n, \|\cdot\|_{\ell_1(\mathbf{w})})$, where Δ_n is the n-point simplex in $\mathbb{R}^n$ and $\|\cdot\|_{\ell_1(\mathbf{w})}$ is the weighted ℓ_1 norm, with weights given by the corresponding edge weight in the underlying star metric as follows:

$$\|\mathbf{x}\|_{\ell_1(\mathbf{w})} = \sum_{a\in X} \mathbf{w}^a |\mathbf{x}^a|.$$

Proof of Lemma 2.2. Recall that by assumption, the MAL instance contains at least one OFF point denoted by $a' \in X$ in the MAL instance, where $\mathbf{c}^{a'} = 0$. Without loss of generality, let the *first dimension* in Δ_n correspond to this OFF point.

We define a linear map $\Phi : \Delta_n \to \mathbb{R}^{n-1}$, where Φ has $n - 1$ rows and n columns, and is specified as follows:

$$\Phi_{i,j} = \begin{cases} 1 \text{ if j = i+1} \\ 0 \text{ otherwise} \end{cases}$$

It is straightforward to see that $\Phi\mathbf{x} \in \mathbb{R}^{n-1}$, $\forall \mathbf{x} \in \Delta_n$.

Recall that a CFL decision space is the ℓ_1 ball defined by the long-term constraint function in $\mathbb{R}^{n-1}$. For any MAL instance with constraint function $c(\mathbf{x}) : \Delta_n \to \mathbb{R}$, we can define a long-term constraint function $c'(\mathbf{x}') : \mathbb{R}^{n-1} \to \mathbb{R}$ as follows. The MAL constraint function $c(\mathbf{x})$ is defined as $\|\cdot\|_{\ell_1(\mathbf{c})}$ for some vector $\mathbf{c} \in \mathbb{R}^{n-1}$. Then

$$\mathbf{c}' = \Phi\mathbf{c}$$
$$c'(\mathbf{x}') = \|\mathbf{x}'\|_{\ell_1(\mathbf{c}')} \; \forall \mathbf{x}' \in \mathbb{R}^{n-1}$$

Furthermore, for any $z \in [0,1]$, let $\mathbf{x} \in \Delta_n : c(\mathbf{x}) < 1-z$. Then it follows that $\Phi\mathbf{x}$ is in $\mathbb{R}^{n-1} : c'(\mathbf{x}') < 1-w$. Recall that cost functions in the MAL instance are convex and linearly separable as follows:

$$f_t(\mathbf{x}) = \sum_{a \in X} f_t^a(\mathbf{x}^a)$$

Next, again letting $\mathbf{x} \in \Delta_n$, note that the ith term in $\mathbf{x}$ is identical to the $(i-1)$th term in $\Phi\mathbf{x}$ (excluding the first term in $\mathbf{x}$). Then we can construct cost functions in the CFL instance as follows:

$$f_t'(\mathbf{x}') = \sum_{i \in [n-1]} f_t^{i+1}(\mathbf{x}^i)$$

Under the mapping Φ, note that it is straightforward to show that $f_t(\mathbf{x}) = f_t'(\Phi\mathbf{x})$ for any $\mathbf{x} \in \Delta_n$.

Finally, consider the distances in the MAL instance's weighted star metric, which can be expressed as a weighted ℓ_1 norm defined by $\mathbf{w}$, where the terms of $\mathbf{w}$ correspond to the weighted edges of the star metric. Recall that $\beta \coloneqq \max_{a',a} \|a' - a\|_{\ell_1(\mathbf{w})}$, i.e., the maximum distance between the OFF point and any other point in the weighted star.

Then we define a corresponding distance metric in the CFL instance, which is an ℓ_1 norm weighted by $\mathbf{w}' \in \mathbb{R}^{n-1}$, which is defined as follows:

$$\mathbf{w}'^i = \mathbf{w}^{i+1} + \mathbf{w}^0.$$

Note that $\mathbf{w}^0$ is the edge weight associated with the OFF point. Then for any $(\mathbf{x}, \mathbf{y}) \in \Delta_n$, it is straightforward to show the following:

$$\|\mathbf{x} - \mathbf{y}\|_{\ell_1(\mathbf{w})} \leq \|\Phi\mathbf{x} - \Phi\mathbf{y}\|_{\ell_1(\mathbf{w}')}$$

This follows since for any $(\mathbf{x}, \mathbf{y}) \in \Delta_n$ where $\mathbf{x}^0 = 0$ and $\mathbf{y}^0 = 0$ (i.e., allocations which do not allocate anything to the OFF point), $\|\Phi\mathbf{x} - \Phi\mathbf{y}\|_{\ell_1(\mathbf{w}')} = \|\mathbf{x} - \mathbf{y}\|_{\ell_1(\mathbf{w})} + \|\mathbf{x} - \mathbf{y}\|_{\ell_1} \cdot \mathbf{w}^0$.

Conversely, if either $\mathbf{x}$ or $\mathbf{y}$ have $\mathbf{x}^0 > 0$ or $\mathbf{y}^0 > 0$, we have $\|\mathbf{x} - \mathbf{y}\|_{\ell_1(\mathbf{w})} \leq \|\Phi\mathbf{x} - \Phi\mathbf{y}\|_{\ell_1(\mathbf{w}')} \leq \|\mathbf{x} - \mathbf{y}\|_{\ell_1(\mathbf{w})} + \|\mathbf{x} - \mathbf{y}\|_{\ell_1} \cdot \mathbf{w}^0$. Finally, supposing that (without loss of generality) $\mathbf{x}$ has $\mathbf{x}^0 = 1$, we have that $\|\mathbf{x} - \mathbf{y}\|_{\ell_1(\mathbf{w})} = \|\Phi\mathbf{x} - \Phi\mathbf{y}\|_{\ell_1(\mathbf{w}')}$.
Thus, $\|\Phi\mathbf{x} - \Phi\mathbf{y}\|_{\ell_1(\mathbf{w}')}$ upper bounds $\|\mathbf{x} - \mathbf{y}\|_{\ell_1(\mathbf{w})}$. Furthermore, the constructed distance metric preserves β, i.e. given $(a', a) = \arg\max_{a',a} \|a' - a\|_{\ell_1(\mathbf{w})}$, we have that $\|\Phi a' - \Phi a\|_{\ell_1(\mathbf{w}')} = \beta$.

Next, we show that the transformation Φ is bijective. We define the affine map $\Phi^{-1} : \mathbb{R}^{n-1} \to \Delta_n$ as follows: Φ^{-1} has n rows and $n-1$ columns, where the first row is all -1, and the bottom n rows are the $n \times n$ identity matrix. Let $\mathbf{b} \in \mathbb{R}^{n-1}$ denote the vector with $\mathbf{b}^0 = 1$ and all other terms are zero, i.e., $\mathbf{b}_i = 0 \; \forall i \geq 1$.

For any $\mathbf{x}' \in \mathbb{R}^{n-1} : c'(\mathbf{x}') \leq 1$, it is straightforward to show that $\Phi^{-1}\mathbf{x}' + \mathbf{b}$ is in Δ_n, since by definition we have that $\sum_{i \in [n+1]} \left(\Phi^{-1}\mathbf{x}' + \mathbf{b}\right)_i = 1$. Furthermore, by definition of $c'(\mathbf{x}')$, we have that $c(\Phi^{-1}\mathbf{x}' + \mathbf{b}) = c'(\mathbf{x}')$, because the ith term (excluding the first term) of $\Phi^{-1}\mathbf{x}' + \mathbf{b}$ is identical to the $(i-1)$th term of $\mathbf{x}'$. Similarly, by definition of f_t', we have that $f_t(\Phi^{-1}\mathbf{x}' + \mathbf{b}) = f_t'(\mathbf{x}')$.

Finally, considering the distance metric, we have that for any $(\mathbf{x}', \mathbf{y}') \in \mathbb{R}^{n-1} : c'(\mathbf{x}') \leq 1$:

$$\|(\Phi^{-1}\mathbf{x}' + \mathbf{b}) - (\Phi^{-1}\mathbf{y}' + \mathbf{b})\|_{\ell_1(\mathbf{w})} \leq \|\mathbf{x}' - \mathbf{y}'\|_{\ell_1(\mathbf{w}')}.$$

This follows by considering that for any $\mathbf{x}'$, $\Phi^{-1}\mathbf{x}' + \mathbf{b}$ adds a dimension (corresponding to the OFF point) and sets $(\Phi^{-1}\mathbf{x}' + \mathbf{b}) = 1 - \|\mathbf{x}'\|_1$. Then the distance between any two points which allocate a non-negative fraction to the OFF point in Δ_n is $\leq$ the distance in $\mathbb{R}^{n-1}$ by definition of the weight vector $\mathbf{w}'$, and the distance between e.g., the allocation fully in the OFF point (a') and any other allocation is exactly preserved.

Furthermore, note that if $\mathbf{w}^0 = 0$ (i.e., the weight of the OFF state in the weighted star metric is 0), Φ is a bijective isometry between $(\Delta_n, \|\cdot\|_{\ell_1(\mathbf{w})})$ and $(\mathbb{R}^{n-1}, \|\cdot\|_{\ell_1(\mathbf{w}')})$. ◻

The transformation defined by Φ in Lemma 2.2 allows us to put decisions on the CFL instance $(\mathbb{R}^{n-1}, \|\cdot\|_{\ell_1(\mathbf{w}')})$ in one-to-one correspondence with decisions in $(\Delta_n, \|\cdot\|_{\ell_1(\mathbf{w})})$.

Below, we formalize this by proving a result stated in the main body (Proposition 2.3) which states the following: Given an algorithm ALG for CFL, any performance bound on ALG which assumes OPT does not pay any switching cost will translate to an identical performance bound for MAL whose parameters depend on the corresponding CFL instance constructed according to Lemma 2.2.

Proof of Proposition 2.3. The cost of ALG on the CFL instance is an upper bound on the cost of the ALG's decisions mapped into the MAL instance. This follows since the cost functions are preserved exactly between the two instances, the long-term constraint function is preserved exactly, and the CFL switching cost is by definition an upper bound on the MAL switching cost.

If the CFL performance bound assumes that OPT does not pay any switching cost (e.g., as in Theorem 3.2), lower bounding the cost of OPT on the CFL instance is equivalent to lower bounding the cost of OPT on the MAL instance, as the cost functions and constraint functions are preserved exactly.

Thus, we have that any such performance bound for ALG on the CFL instance constructed appropriately (as in Lemma 2.2) immediately gives an identical performance bound for the MAL instance, yielding the result. ◻

By Lemma 2.2, we have that since ALG1 is α-competitive for CFL (Theorem 3.2), ALG1 is α-competitive for any CFL instance constructed based on a MAL instance. Furthermore, by Proposition 2.3, ALG1 is also α-competitive on the underlying MAL instance, where α is given by (4). ◻

B.4 Proof of Theorem 3.4

In this section, we prove Theorem 3.4, which shows that α as given by (4) is the best competitive ratio achievable for CFL.

To show this lower bound, we first define a family of special adversaries, and then show that the competitive ratio for any deterministic algorithm is lower bounded under the instances provided by these adversaries.

Prior work has shown that difficult instances for online search problems with a minimization objective occur when inputs arrive at the algorithm in an decreasing order of cost [EFK+01; LPS08; SZL+21; LCZ+23]. For CFL, we additionally consider how an adaptive adversary can essentially force an algorithm to incur a large switching cost in the worst-case. We now formalize such a family of adversaries $\{\mathcal{A}_y\}_{y\in[L,U]}$, where $\mathcal{A}_y$ is called a *y-adversary*.

Definition B.4 (y-adversary for CFL)**.** Let $w, m \in \mathbb{Z}$ be sufficiently large, and $\delta := {}^{(U-L)}/_{w}$.

Without loss of generality, let $k = \arg\max_{i\in[d]} \mathbf{w}_i$, where $\mathbf{w}$ is the weight vector for $\|\cdot\|_{\ell_1(\mathbf{w})}$, and let $\beta = \max_{i\in[d]} \mathbf{w}_i$. For $y \in [L, U]$, an adaptive adversary $\mathcal{A}_y$ sequentially presents two types of cost functions $f_t(\cdot)$ to both ALG and OPT.

These types of cost functions are $\mathbf{Up}(\mathbf{x}) = U\mathbf{1}\mathbf{x}^\intercal$, and $\mathbf{Down}^i(\mathbf{x}) = \sum_{j\neq k}^d U\mathbf{x}^j + (U - i\delta)\mathbf{x}^k$.

The adversary sequentially presents cost functions from these two types in an alternating, "continuously decreasing" order. Specifically, they start by presenting cost function $\mathbf{Up}(\mathbf{x})$, up to m times.

Then, they present $\mathbf{Down}^1(\mathbf{x})$, which has linear cost coefficient U in every direction except direction k, which has cost coefficient $(U - 1 \cdot \delta)$. $\mathbf{Down}^1(\mathbf{x})$ is presented up to m times. If ALG ever "accepts" a cost function $\mathbf{Down}^1(\mathbf{x})$ (i.e., if ALG makes a decision $\mathbf{x}$ where $c(\mathbf{x}) > 0$), the adaptive adversary immediately presents $\mathbf{Up}(\mathbf{x})$ starting in the next time step until either ALG moves to the origin (i.e. online decision $\mathbf{x} = \mathbf{0}$) or ALG's utilization $z = 1$.

The adversary continues alternating in this manner, presenting $\mathbf{Down}^2(\mathbf{x})$ up to m times, followed by $\mathbf{Up}(\mathbf{x})$ if ALG accepts anything, followed by $\mathbf{Down}^3(\mathbf{x})$ up to m times, and so on. This continues until the adversary presents $\mathbf{Down}^{w_y}(\mathbf{x})$, where $y := (U - w_y\delta)$, up to m times. After presenting $\mathbf{Down}^{w_y}(\mathbf{x})$, $\mathcal{A}_y$ will present $\mathbf{Up}(\mathbf{x})$ until either ALG moves to the origin or has utilization $z = 1$. Finally, the adversary presents exactly m cost functions of the form $\sum_{j\neq k}^d U\mathbf{x}^j + (y + \varepsilon)\mathbf{x}^k$, followed by m cost functions $\mathbf{Up}(\mathbf{x})$.

The mechanism of this adaptive adversary is designed to present "good cost functions" (i.e., $\mathbf{Down}^i(\mathbf{x})$) in a worst-case decreasing order, interrupted by blocks of "bad cost functions" $\mathbf{Up}(\mathbf{x})$ which force a large switching cost in the worst case.

$\mathcal{A}_U$ is simply a stream of m cost functions U, and the final cost functions in any y-adversary instance are always $\mathbf{Up}(\mathbf{x})$.

Proof of Theorem 3.4. Let $g(y)$ denote a *conversion function* $[L, U] \rightarrow [0, 1]$, which fully describes the progress towards the long-term constraint (before the compulsory trade) of a deterministic ALG playing against adaptive adversary $\mathcal{A}_y$. Note that for large w, the adaptive adversary $\mathcal{A}_{y-\delta}$ is equivalent to first playing $\mathcal{A}_y$ (besides the last two batches of cost functions), and then processing batches with cost functions $\mathbf{Down}^{w_y+1}(\mathbf{x})$ and $\mathbf{Up}(\mathbf{x})$. Since ALG is deterministic and the conversion is unidirectional (irrevocable), we must have that $g(y - \delta) \geq g(y)$, i.e. $g(y)$ is non-increasing in $[L, U]$. Intuitively, the entire capacity should be satisfied if the minimum possible price is observed, i.e $g(L) = 1$.

Note that for $\varepsilon \rightarrow 0$, the optimal solution for adversary $\mathcal{A}_y$ is $\texttt{OPT}(\mathcal{A}_y) = y + 2\beta/m$, and for m sufficiently large, $\texttt{OPT}(\mathcal{A}_y) \rightarrow y$.

Due to the adaptive nature of each y-adversary, any deterministic ALG incurs a switching cost proportional to $g(y)$, which gives the amount of utilization obtained by ALG before the end of $\mathcal{A}_y$'s sequence.

Whenever ALG accepts some cost function with coefficient $U - i\delta$ in direction k, the adversary presents $\mathbf{Up}(\mathbf{x})$ starting in the next time step. Any ALG which does not switch away immediately obtains a competitive ratio strictly worse than an algorithm which does switch away (if an algorithm accepts c fraction of a good price and switches away immediately, the switching cost it will pay is $2\beta c$. An algorithm may continue accepting c fraction of coefficient U in the subsequent time steps, but a sequence exists where this decision will take up too much utilization to recover when better cost functions are presented later. In the extreme case, if an algorithm continues accepting c fraction of these U coefficients, it might fill its utilization and then OPT can accept a cost function which is arbitrarily better).

Since accepting any price by a factor of c incurs a switching cost of $2\beta c$, the switching cost paid by ALG on adversary $\mathcal{A}_y$ is $2\beta g(y)$. We assume that ALG is notified of the compulsory trade, and does not incur a significant switching cost during the final batch.

Then the total cost incurred by an $\alpha^\star$-competitive online algorithm ALG on adversary $\mathcal{A}_y$ is $\texttt{ALG}(\mathcal{A}_y) = g(U/\alpha^\star) U/\alpha^\star - \int_{U/\alpha^\star}^{y} u dg(u) + 2\beta g(y) + (1 - g(y))U$, where $udg(u)$ is the cost of buying $dg(u)$ utilization at cost coefficient u, the last term is from the compulsory trade, and the second to last term is the switching cost incurred by ALG. Note that any deterministic ALG which makes conversions when the price is larger than $U/\alpha^\star$ can be strictly improved by restricting conversions to prices $\leq U/\alpha^\star$.

For any $\alpha^\star$-competitive online algorithm, the corresponding conversion function $g(\cdot)$ must satisfy $\texttt{ALG}(\mathcal{A}_y) \le \alpha^\star \texttt{OPT}(\mathcal{A}_y) = \alpha^\star y, \forall y \in [L, U]$. This gives a necessary condition which the conversion function must satisfy as follows:

$$\texttt{ALG}(\mathcal{A}_y) = g(U/\alpha^\star)U/\alpha^\star - \int_{U/\alpha^\star}^{y} u dg(u) + 2\beta g(y) + (1 - g(y))U \le \alpha^\star y, \quad \forall y \in [L, U].$$

By integral by parts, the above implies that the conversion function must satisfy $g(y) \ge \frac{U-\alpha^\star y}{U-y-2\beta} - \frac{1}{U-y-2\beta}\int_{U/\alpha^\star}^{y} g(u)du$. By Grönwall's Inequality [MPF91, Theorem 1, p. 356], we have that

$$\begin{aligned} g(y) &\ge \frac{U-\alpha^\star y}{U-y-2\beta} - \frac{1}{U-y-2\beta}\int_{U/\alpha^\star}^{y} \frac{U-\alpha^\star u}{U-u-2\beta} \cdot \exp\left(\int_u^y \frac{1}{U-r-2\beta}dr\right) du \\ &\ge \frac{U-\alpha^\star y}{U-y-2\beta} - \int_{U/\alpha^\star}^{y} \frac{U-\alpha^\star u}{(U-u-2\beta)^2} du \\ &\ge \frac{U-\alpha^\star y}{U-y-2\beta} - \left[\frac{U\alpha^\star - U - 2\beta\alpha^\star}{u+2\beta-U} - \alpha^\star \ln(u+2\beta-U)\right]_{U/\alpha^\star}^{y} \\ &\ge \alpha^\star \ln(y+2\beta-U) - \alpha^\star \ln(U/\alpha^\star + 2\beta - U), \quad \forall y \in [L, U]. \end{aligned}$$

$g(L) = 1$ by the problem definition – we can combine this with the above constraint to give the following condition for an $\alpha^\star$-competitive online algorithm:

$$\alpha^\star \ln(L+2\beta-U) - \alpha^\star \ln(U/\alpha^\star + 2\beta - U) \le g(L) = 1.$$

The optimal $\alpha^\star$ is obtained when the above inequality is binding, so solving for the value of $\alpha^\star$ which solves $\alpha^\star \ln(L+2\beta-U) - \alpha^\star \ln(U/\alpha^\star + 2\beta - U) = 1$ yields that the best competitive ratio for any $\texttt{ALG}$ solving CFL is $\alpha^\star \ge \left[W\left(\frac{e^{2\beta/U}(L/U + 2\beta/U - 1)}{e}\right) - \frac{2\beta}{U} + 1\right]^{-1}$. ◻

B.5 Proof of Corollary 3.5

In this section, we prove Corollary 3.5, which shows that α as given by (4) is the best competitive ratio achievable for MAL.

To show this lower bound, we build off of the family of adversaries in Definition B.4, which are designed to force an algorithm to incur a large switching cost while satisfying the long-term constraint. In Definition B.5 we define this family of adversarial instances tailored for MAL.

Definition B.5 (y-adversary for MAL)**.** Let $w, m \in \mathbb{Z}$ be sufficiently large, and $\delta := (U-L)/w$.

Recall that $\mathbf{w}$ denotes the vector of edge weights for each point in the weighted star metric X, and the $\texttt{OFF}$ point is defined (without loss of generality) as the point $a' \in X$ where $\mathbf{c}^{a'} = 0$ and $f_t^{a'}(\mathbf{x}^a) = 0 \forall t \in [T], \forall \mathbf{x}^a \in [0,1]$. We will assume that $\mathbf{c}^a = 1 \; \forall a \in X : a \neq a'$.

Then we set $\mathbf{w}^{a'} = 0$, i.e., the $\texttt{OFF}$ point is connected to the interior vertex of the weighted star with an edge of weight 0. Without loss of generality, we let $k = \arg\max_{a \in [n]} \mathbf{w}^a$ denote the largest edge weight of any other (non-$\texttt{OFF}$) point in the metric. By definition, recall that $\beta = \mathbf{w}^k$.

For $y \in [L, U]$, an adaptive adversary $\mathcal{A}_y$ sequentially presents two different sets of cost functions $f_t^a(\cdot)$ at each point in the metric space.

These sets of cost functions are $\mathbf{Up} = \{f^a(x) = U\mathbf{x}^a \;\; \forall a \in X \setminus \{a'\}\}$, and $\mathbf{Down}^i = \{f^k(\mathbf{x}^k) = (U - i\delta)\mathbf{x}^k\} \cap \{f^a(\mathbf{x}^a) = U\mathbf{x}^a \;\; \forall a \in X \setminus \{a', k\}\}$. Note that the adversary only ever presents cost functions with a coefficient $< U$ at the point k which corresponds to the largest edge weight.

The adversary sequentially presents either of these two sets of cost functions in an alternating, "continuously decreasing" order. Specifically, they start by presenting $\mathbf{Up}$, up to m times.

Then, they present **Down**, which has cost coefficient U in every point except point k, which has cost coefficient $(U - 1 \cdot \delta)$. $\textbf{Down}^1$ is presented up to m times. If ALG ever "accepts" a cost function in $\textbf{Down}^1$ (i.e., if ALG makes a decision x where $c(x) > 0$), the adaptive adversary immediately presents **Up** starting in the next time step until either ALG moves entirely to the OFF point (i.e. online decision $x^{a'} = 1$) or ALG's utilization $z = 1$.

The adversary continues alternating in this manner, presenting $\textbf{Down}^2$ up to m times, followed by **Up** if ALG accepts anything, followed by $\textbf{Down}^3$ up to m times, and so on. This continues until the adversary presents $\textbf{Down}^{w_y}$, where $y = (U - w_y\delta)$, up to m times. After presenting $\textbf{Down}^{w_y}$, $\mathcal{A}_y$ will present $\textbf{Up}(x)$ until either ALG moves to the OFF point or has utilization $z = 1$. Finally, the adversary presents the set of cost functions $\{f^k(\mathbf{x}^k) = (y + \varepsilon)\mathbf{x}^k\} \cap \{f^a(\mathbf{x}^a) = U\mathbf{x}^a \;\; \forall a \in X \setminus \{a', k\}\}$ m times, followed by **Up** m times.

The mechanism of this adaptive adversary is designed to present "good cost functions" (i.e., $\textbf{Down}^i$) in a worst-case decreasing order, interrupted by blocks of "bad cost functions" **Up** which force a large switching cost in the worst case.

As in Theorem 3.4, $\mathcal{A}_U$ is simply a stream of m **Up** sets of cost functions, and the final cost functions in any y-adversary instance are always **Up**.

Proof of Corollary 3.5. As previously, we let $g(y)$ denote a *conversion function* $[L, U] \to [0, 1]$, which fully describes the progress towards the long-term constraint (before the compulsory trade) of a deterministic ALG playing against adaptive adversary $\mathcal{A}_y$. Since ALG is deterministic and the conversion is unidirectional (irrevocable), $g(y)$ is non-increasing in $[L, U]$. Intuitively, the entire long-term constraint should be satisfied if the minimum possible price is observed, i.e $g(L) = 1$. For $\varepsilon \to 0$, the optimal solution for adversary $\mathcal{A}_y$ is $\texttt{OPT}(\mathcal{A}_y) = y + 2\beta/m$, and for m sufficiently large, $\texttt{OPT}(\mathcal{A}_y) \to y$.

As in Theorem 3.4, the adaptive nature of each y-adversary forces any deterministic ALG to incur a switching cost of $2\beta g(y)$ on adversary $\mathcal{A}_y$, and we assume that ALG does not incur a significant switching cost during the final batch (i.e., during the compulsory trade).

Then the total cost incurred by an $\alpha^\star$-competitive online algorithm ALG on adversary $\mathcal{A}_y$ is $\texttt{ALG}(\mathcal{A}_y) = g(U/\alpha^\star)U/\alpha^\star - \int_{U/\alpha^\star}^{y} u dg(u) + 2\beta g(y) + (1 - g(y))U$, where $udg(u)$ is the cost of buying $dg(u)$ utilization at cost coefficient u, the last term is from the compulsory trade, and the second to last term is the switching cost incurred by ALG. Note that this expression for the cost is exactly as defined in Theorem 3.4.

Thus by Theorem 3.4, for any $\alpha^\star$-competitive online algorithm, the conversion function $g(\cdot)$ must satisfy $\texttt{ALG}(\mathcal{A}_y) \leq \alpha^\star \texttt{OPT}(\mathcal{A}_y) = \alpha^\star y, \forall y \in [L, U]$. Via integral by parts and Grönwall's Inequality [MPF91, Theorem 1, p. 356], we have the following condition on $g(y)$:

$$g(y) \geq \alpha^\star \ln (y + 2\beta - U) - \alpha^\star \ln (U/\alpha^\star + 2\beta - U), \quad \forall y \in [L, U].$$

$g(L) = 1$ by the problem definition – combining this with the previous condition gives the following condition for an $\alpha^\star$-competitive online algorithm:

$$\alpha^\star \ln (L + 2\beta - U) - \alpha^\star \ln (U/\alpha^\star + 2\beta - U) \leq g(L) = 1.$$

As in Theorem 3.4, the optimal $\alpha^\star$ is obtained when the above inequality is binding, yielding that the best competitive ratio for any ALG solving MAL is $\alpha^\star \geq \left[W\left(\frac{e^{2\beta/U}(L/U + 2\beta/U - 1)}{e} \right) - \frac{2\beta}{U} + 1 \right]^{-1}$. □

C Proofs for Section 4 (Learning-Augmentation)

C.1 Proof of Lemma 4.1

In this section, we prove Lemma 4.1, which shows that the baseline fixed-ratio combination algorithm (Baseline) is $(1+\epsilon)$-consistent and $\left(\frac{(U+2\beta)/L(\alpha-1-\epsilon)+\alpha\epsilon}{(\alpha-1)} \right)$-robust for CFL, given any $\epsilon \in [0, \alpha - 1]$ and where

α is as defined in (4). Recall that Lemma 4.1 specifies ALG1 as the "robust algorithm" to use for the following analysis.

Proof of Lemma 4.1. Under the assumption that ADV satisfies the long-term constraint, (i.e., that $\sum_{t=1}^{T} c(\mathbf{a}_t) \geq 1$), we first observe that the online solution of Baseline must also satisfy the long-term constraint.

Under the assumptions of CFL, note that $c(\mathbf{x})$ is linear (i.e., a weighted ℓ_1 norm with weight vector $\mathbf{c}$). By definition, denoting the decisions of ALG1 by $\tilde{\mathbf{x}}_t$, we know that $\sum_{t=1}^{T} c(\tilde{\mathbf{x}}_t) \geq 1$.

Thus, we have the following:

$$\sum_{t=1}^{T} c(\mathbf{x}_t) = \sum_{t=1}^{T} c\left(\lambda \mathbf{a}_t + (1-\lambda)\tilde{\mathbf{x}}_t\right) = \lambda \sum_{t=1}^{T} c(\mathbf{a}_t) + (1-\lambda) \sum_{t=1}^{T} c\left(\tilde{\mathbf{x}}_t\right) \geq \lambda + (1-\lambda) = 1.$$

Let $\mathcal{I} \in \Omega$ be an arbitrary valid CFL sequence. We denote the *hitting* and *switching* costs of the robust advice by $\text{ALG1}_{\text{hitting}}$ and $\text{ALG1}_{\text{switch}}$, respectively. Likewise, the hitting and switching cost of the black-box advice ADV is denoted by $\text{ADV}_{\text{hitting}}$ and $\text{ADV}_{\text{switch}}$.

The total cost of Baseline is upper bounded by the following:

$$\begin{aligned}
\texttt{Baseline}(\mathcal{I}) &= \sum_{t=1}^{T} f_t(\mathbf{x}_t) + \sum_{t=1}^{T+1} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})}, \\
&= \sum_{t=1}^{T} f_t\left(\lambda \mathbf{a}_t + (1-\lambda)\tilde{\mathbf{x}}_t\right) + \sum_{t=1}^{T+1} \|\lambda \mathbf{a}_t + (1-\lambda)\tilde{\mathbf{x}}_t - \lambda \mathbf{a}_{t-1} - (1-\lambda)\tilde{\mathbf{x}}_{t-1}\|_{\ell_1(\mathbf{w})}, \\
&\leq \lambda \sum_{t=1}^{T} f_t(\mathbf{a}_t) + (1-\lambda) \sum_{t=1}^{T} f_t(\tilde{\mathbf{x}}_t) + \sum_{t=1}^{T+1} \|\lambda \mathbf{a}_t - \lambda \mathbf{a}_{t-1}\|_{\ell_1(\mathbf{w})} + \sum_{t=1}^{T+1} \|(1-\lambda)\tilde{\mathbf{x}}_t - (1-\lambda)\tilde{\mathbf{x}}_{t-1}\|_{\ell_1(\mathbf{w})}, \\
&\leq \lambda \text{ADV}_{\text{hitting}}(\mathcal{I}) + (1-\lambda)\text{ALG1}_{\text{hitting}}(\mathcal{I}) + \lambda \sum_{t=1}^{T+1} \|\mathbf{a}_t - \mathbf{a}_{t-1}\|_{\ell_1(\mathbf{w})} + (1-\lambda) \sum_{t=1}^{T+1} \|\tilde{\mathbf{x}}_t - \tilde{\mathbf{x}}_{t-1}\|_{\ell_1(\mathbf{w})}, \\
&\leq \lambda \text{ADV}_{\text{hitting}}(\mathcal{I}) + (1-\lambda)\text{ALG1}_{\text{hitting}}(\mathcal{I}) + \lambda \text{ADV}_{\text{switch}}(\mathcal{I}) + (1-\lambda)\text{ALG1}_{\text{switch}}(\mathcal{I}), \\
&\leq \lambda \text{ADV}(\mathcal{I}) + (1-\lambda)\text{ALG1}(\mathcal{I}).
\end{aligned}$$

Since $\text{ALG1} \leq \alpha \cdot \text{OPT} \leq \alpha \cdot \text{ADV}$, this gives the following:

$$\texttt{Baseline}(\mathcal{I}) \leq \lambda \text{ADV}(\mathcal{I}) + (1-\lambda)\alpha \text{ADV}(\mathcal{I}), \tag{23}$$

$$\texttt{Baseline}(\mathcal{I}) \leq (\lambda + (1-\lambda)\alpha) \cdot \text{ADV}(\mathcal{I}) \tag{24}$$

$$\texttt{Baseline}(\mathcal{I}) \leq (1+\epsilon) \cdot \text{ADV}(\mathcal{I}). \tag{25}$$

Furthermore, since $\text{ADV} \leq U + 2\beta \leq \frac{\text{OPT}}{L/(U+2\beta)}$, we have:

$$\texttt{Baseline}(\mathcal{I}) \leq \lambda \frac{\text{OPT}(\mathcal{I})}{L/(U+2\beta)} + (1-\lambda)\alpha \text{OPT}(\mathcal{I}), \tag{26}$$

$$\texttt{Baseline}(\mathcal{I}) \leq \left[\frac{\lambda(U+2\beta)}{L} + (1-\lambda)\alpha\right] \cdot \text{OPT}(\mathcal{I}), \tag{27}$$

$$\texttt{Baseline}(\mathcal{I}) \leq \left(\frac{(U+2\beta)/L(\alpha - 1 - \epsilon) + \alpha\epsilon}{(\alpha-1)}\right) \cdot \text{OPT}(\mathcal{I}). \tag{28}$$

By combining (25) and (28), we conclude that Baseline is $(1+\epsilon)$-consistent with respect to black-box advice ADV, and $\left(\frac{(U+2\beta)/L(\alpha-1-\epsilon)+\alpha\epsilon}{(\alpha-1)}\right)$-robust. ☐

C.2 Proof of Theorem 4.3

In this section, we prove Theorem 4.3, which shows that CLIP is $(1+\epsilon)$-consistent and γ^ϵ-robust for CFL, where γ^ϵ is defined as the solution to the following (as in (8)):

$$\gamma^\epsilon = \epsilon + \frac{U}{L} - \frac{\gamma^\epsilon}{L}(U-L)\ln\left(\frac{U-L-2\beta}{U - {}^{U}\!/_{\gamma^\epsilon} - 2\beta}\right).$$

Proof of Theorem 4.3. We show the above result by separately considering consistency (the competitive ratio when advice is correct) and robustness (the competitive ratio when advice is not correct) in turn.

Recall that the black-box advice ADV is denoted by a decision $\mathbf{a}_t$ at each time t. Throughout the following proof, we use shorthand notation $\texttt{CLIP}_t$ to denote the cost of CLIP up to time t, and $\texttt{ADV}_t$ to denote the cost of ADV up to time t. We start with the following lemma to prove consistency.

Lemma C.1. CLIP *is* $(1+\epsilon)$*-consistent.*

Proof. First, we note that the constrained optimization enforces that the possible cost so far plus a compulsory term is always within $(1+\epsilon)$ of the advice. Formally, if time step $j \in [T]$ denotes the time step marking the start of the compulsory trade, we have that the constraint given by (6) holds for every time step $t \in [j]$.

Thus, to show $(1+\epsilon)$ consistency, we must resolve the cost during the *compulsory trade* and show that the final cumulative cost of CLIP is upper bounded by $(1+\epsilon)$ADV.

Let $\mathcal{I} \in \Omega$ be an arbitrary valid CFL sequence. If the compulsory trade begins at time step $j < T$, both CLIP and ADV must greedily fill their remaining utilization during the last m time steps $[j, T]$. This is assumed to be feasible, and the switching cost is assumed to be negligible as long as m is sufficiently large.

Let $(1 - z^{(j-1)})$ denote the remaining long-term constraint that must be satisfied by CLIP at the final time step, and let $(1 - A^{(j-1)})$ denote the remaining long-term constraint to be satisfied by ADV.

We consider the following two cases, which correspond to the cases where CLIP has *under-* and *over-*provisioned with respect to ADV, respectively.

Case 1: CLIP($\mathcal{I}$) **has "underprovisioned" ($(1 - z^{(j-1)}) > (1 - A^{(j-1)})$).** In this case, CLIP must satisfy *more* of the long-term constraint during the compulsory trade compared to ADV.

From the previous time step, we know that the following constraint holds: $\texttt{CLIP}_{j-1} + \|\mathbf{x}_{j-1} - \mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L + (A^{(j-1)} - z^{(j-1)})U \le (1+\epsilon)\left[\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L\right]$.

Let $\{\mathbf{x}_t\}_{t\in[j,T]}$ and $\{\mathbf{a}_t\}_{t\in[j,T]}$ denote the decisions made by CLIP and ADV during the compulsory trade, respectively. By definition, we have that $\sum_{t=j}^T c(\mathbf{x}_t) = (1 - z^{(j-1)})$ and $\sum_{t=j}^T c(\mathbf{a}_t) = (1 - A^{(j-1)})$.

Considering $\{f_t(\cdot)\}_{t\in[j,T]}$, we know that by definition $\sum_{t=j}^T f_t(\mathbf{a}_t) \ge L \sum_{t=j}^T c(\mathbf{a}_t)$, and by convex assumptions on the cost functions, $\sum_{t=j}^T f_t(\mathbf{x}_t) \le \sum_{t=j}^T f_t(\mathbf{a}_t) + U(\sum_{t=j}^T c(\mathbf{x}_t) - \sum_{t=j}^T c(\mathbf{a}_t))$.

Note that the worst case for CLIP occurs when $\sum_{t=j}^T f_t(\mathbf{a}_t) = L \sum_{t=j}^T c(\mathbf{a}_t)$, as ADV is able to satisfy the rest of the long-term constraint at the best possible price.

By the constraint in the previous time step, we have the following:

$$\begin{aligned}\texttt{CLIP}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L + (A^{(j-1)} - z^{(j-1)})U \\ \leq (1+\epsilon)[\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L],\\ \texttt{CLIP}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + L\sum_{t=j}^{T} c(\mathbf{a}_t) + U\left(\sum_{t=j}^{T} c(\mathbf{x}_t) - \sum_{t=j}^{T} c(\mathbf{a}_t)\right) \\ \leq (1+\epsilon)\left[\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + L\sum_{t=j}^{T} c(\mathbf{a}_t)\right],\\ \texttt{CLIP}(\mathcal{I}) \leq (1+\epsilon)\left[\texttt{ADV}(\mathcal{I})\right].\end{aligned}$$

Case 2: $\texttt{CLIP}(\mathcal{I})$ **has "overprovisioned" ($(1 - z^{(j-1)}) \leq (1 - A^{(j-1)})$).** In this case, CLIP must satisfy *less* of the long-term constraint during the compulsory trade compared to ADV.

From the previous time step, we know that the following constraint holds: $\texttt{CLIP}_{j-1} + \|\mathbf{x}_{j-1} - \mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L + (A^{(j-1)} - z^{(j-1)})U \leq (1+\epsilon)\left[\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L\right]$.

Let $\{\mathbf{x}_t\}_{t\in[j,T]}$ and $\{\mathbf{a}_t\}_{t\in[j,T]}$ denote the decisions made by CLIP and ADV during the compulsory trade, respectively. By definition, we have that $\sum_{t=j}^{T} c(\mathbf{x}_t) = (1 - z^{(j-1)})$ and $\sum_{t=j}^{T} c(\mathbf{a}_t) = (1 - A^{(j-1)})$.

Considering $\{f_t(\cdot)\}_{t\in[j,T]}$, we know that by definition, $\sum_{t=j}^{T} f_t(\mathbf{x}_t) \geq L\sum_{t=j}^{T} c(\mathbf{x}_t)$, and $\sum_{t=j}^{T} f_t(\mathbf{a}_t) \geq L\sum_{t=j}^{T} c(\mathbf{a}_t)$. By convexity, because $\sum_{t=j}^{T} c(\mathbf{x}_t) \leq \sum_{t=j}^{T} c(\mathbf{a}_t)$, $\sum_{t=j}^{T} f_t(\mathbf{x}_t) \leq \sum_{t=j}^{T} f_t(\mathbf{a}_t)$.

By the constraint in the previous time step, we have:

$$\frac{\texttt{CLIP}_{j-1} + \|\mathbf{x}_{j-1} - \mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - z^{(j-1)})L}{\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L} = \frac{\texttt{CLIP}_{j-1} + \|\mathbf{x}_{j-1} - \mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + L\sum_{t=j}^{T} c(\mathbf{x}_t)}{\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + L\sum_{t=j}^{T} c(\mathbf{a}_t)} \leq (1+\epsilon).$$

Let $y = \sum_{t=j}^{T} f_t(\mathbf{x}_t) - L\sum_{t=j}^{T} c(\mathbf{x}_t)$, and let $y' = \sum_{t=j}^{T} f_t(\mathbf{a}_t) - L\sum_{t=j}^{T} c(\mathbf{a}_t)$. By definition, $y \geq 0$ and $y' \geq 0$.

Note that $\texttt{CLIP}_{j-1} + \|\mathbf{x}_{j-1} - \mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - z^{(j-1)})L + y \geq \texttt{CLIP}(\mathcal{I})$ and $\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + L\sum_{t=j}^{T} c(\mathbf{a}_t) + y' = \texttt{ADV}(\mathcal{I})$.

Furthermore, by definition and convexity of the cost functions $f_t(\cdot)$, we have that $y \leq y'$.

Combined with the constraint from the previous time step, we have the following bound:

$$\frac{\texttt{CLIP}(\mathcal{I})}{\texttt{ADV}(\mathcal{I})} \leq \frac{\texttt{CLIP}_{j-1} + \|\mathbf{x}_{j-1} - \mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - z^{(j-1)})L + y}{\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + (1 - A^{(j-1)})L + y'} \leq \frac{\texttt{CLIP}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + L\sum_{t=j}^{T} c(\mathbf{x}_t)}{\texttt{ADV}_{j-1} + \|\mathbf{a}_{j-1}\|_{\ell_1(\mathbf{w})} + L\sum_{t=j}^{T} c(\mathbf{a}_t)} \leq (1+\epsilon).$$

Thus, by combining the bounds in each of the above two cases, the result follows, and we conclude that CLIP is $(1+\epsilon)$-consistent with accurate advice. □

Having proved the consistency of CLIP, we proceed to show robustness in the next lemma.

Lemma C.2. CLIP *is* γ^{ϵ}*-robust, where* γ^{ϵ} *is as defined in* (8).

Proof. Let $\epsilon \in (0, \alpha - 1]$ be the target consistency (recalling that CLIP is $(1+\epsilon)$ consistent), and let $\mathcal{I} \in \Omega$ denote an arbitrary valid CFL sequence.

To prove the robustness of CLIP, we consider two "bad cases" for the advice $\texttt{ADV}(\mathcal{I})$, and show that in the worst-case, CLIP's competitive ratio is bounded by γ^{ϵ}.

Case 1: $\texttt{ADV}(\mathcal{I})$ **is "inactive".** Consider the case where ADV accepts nothing during the main sequence and instead satisfies the entire long-term constraint in the final time step. In the worst-case, this gives that $\texttt{ADV}(\mathcal{I}) = U + 2\beta$.

Based on the consistency constraint (and using the fact that CLIP will always be "overprocuring" w.r.t. ADV throughout the main sequence), we can derive an upper bound on the amount that CLIP is allowed to accept from the robust pseudo-cost minimization. Recall the following constraint:

$$\texttt{CLIP}_{t-1} + f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{x}_t - \mathbf{a}_t\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - z^{(t-1)} - c(\mathbf{x}_t))L$$
$$\leq (1+\epsilon)\left[\texttt{ADV}_t + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A^{(t)})L\right].$$

Proposition C.3. *z_{PCM} is an upper bound on the amount that* CLIP *can accept from the pseudo-cost minimization without violating* $(1+\epsilon)$ *consistency, and is defined as:*

$$z_{PCM} = \gamma^\epsilon \ln\left[\frac{U - L - 2\beta}{U - {U}/{\gamma^\epsilon} - 2\beta}\right]$$

Proof. Consider an arbitrary time step t. When CLIP is *not* allowed to accept anything more from the robust pseudo-cost minimization, we have that $c(\mathbf{x}_t)$ is restricted to be 0 (recall that $\mathbf{a}_t = \mathbf{0}$ for any time steps before T, because the advice is assumed to be inactive).

By definition, since any cost functions accepted in $\texttt{CLIP}_{t-1}$ can be attributed to the robust pseudo-cost minimization, we have the following in the worst-case:

$$\texttt{CLIP}_{t-1} = \int_0^{z^{(t-1)}} \phi^\epsilon(u)du + \beta z^{(t-1)}.$$

Combining the above with the left-hand side of the consistency constraint, we have the following by observing that $\mathbf{x}_t = \mathbf{0}$ and $\mathbf{a}_t = \mathbf{0}$, and the switching cost to "ramp-up" is absorbed into the pseudo-cost ϕ:

$$\texttt{CLIP}_{t-1} + (1 - z^{(t-1)})L = \int_0^{z^{(t-1)}} \phi^\epsilon(u)du + \beta z^{(t-1)} + (1 - z^{(t-1)})L.$$

As stated, let $z^{(t-1)} = z_{\text{PCM}}$. Then by properties of the pseudo-cost,

$$\begin{aligned}
\texttt{CLIP}_{t-1} + (1 - z_{\text{PCM}})L &= \int_0^{z_{\text{PCM}}} \phi(u)du + \beta z_{\text{PCM}} + (1 - z_{\text{PCM}})U + (1 - z_{\text{PCM}})L - (1 - z_{\text{PCM}})U, \\
&= \gamma^\epsilon\left[\phi^\epsilon(z_{\text{PCM}}) - \beta\right] + (1 - z_{\text{PCM}})L - (1 - z_{\text{PCM}})U, \\
&= \gamma^\epsilon L + (L - U)\left(1 - \gamma^\epsilon \ln\left[\frac{U - L - 2\beta}{U - {U}/{\gamma^\epsilon} - 2\beta}\right]\right), \\
&= \gamma^\epsilon L + L - U - (L - U)\,\gamma^\epsilon \ln\left[\frac{U - L - 2\beta}{U - {U}/{\gamma^\epsilon} - 2\beta}\right].
\end{aligned}$$

Substituting for the definition of γ^ϵ, we obtain:

$$\begin{aligned}
\texttt{CLIP}_{t-1} + (1 - z_{\text{PCM}})L &= \gamma^\epsilon L + L - U - (L - U)\,\gamma^\epsilon \ln\left[\frac{U - L - 2\beta}{U - {U}/{\gamma^\epsilon} - 2\beta}\right], \\
&= \left[\epsilon L + U - \gamma^\epsilon (U - L)\ln\left[\frac{U - L - 2\beta}{U - {U}/{\gamma^\epsilon} - 2\beta}\right]\right] + L - U + (U - L)\,\gamma^\epsilon \ln\left[\frac{U - L - 2\beta}{U - {U}/{\gamma^\epsilon} - 2\beta}\right], \\
&= \epsilon L + L = (1+\epsilon)L.
\end{aligned}$$

This completes the proposition, since $(1+\epsilon)L$ is exactly the right-hand side of the consistency constraint (note that $(1+\epsilon)\left[\texttt{ADV}_t + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A_t)L\right] = (1+\epsilon)L$). ◻

If CLIP is constrained to use at most z_{PCM} of its utilization to be robust, the remaining $(1 - z_{\text{PCM}})$ utilization must be used for the compulsory trade and/or to follow ADV. Thus, we have the following worst-case competitive ratio for CLIP, specifically for Case 2:

$$\frac{\texttt{CLIP}(\mathcal{I})}{\texttt{OPT}(\mathcal{I})} \leq \frac{\int_0^{z_{\text{PCM}}} \phi^\epsilon(u)du + \beta z_{\text{PCM}} + (1 - z_{\text{PCM}})U}{L}$$

By the definition of $\phi^\epsilon(p)$, we have the following:

$$\frac{\texttt{CLIP}(\mathcal{I})}{\texttt{OPT}(\mathcal{I})} \leq \frac{\int_0^{z_{\text{PCM}}} \phi^\epsilon(u)du + \beta z_{\text{PCM}} + (1 - z_{\text{PCM}})U}{L}$$
$$\leq \frac{\gamma^\epsilon\left[\phi^\epsilon(z_{\text{PCM}}) - \beta\right]}{L} \leq \frac{\gamma^\epsilon\left[L + \beta - \beta\right]}{L} \leq \gamma^\epsilon.$$

Case 2: ADV$(\mathcal{I})$ **is "overactive".** We now consider the case where ADV accepts bad cost functions which which it "should not" accept (i.e. $\texttt{ADV}(\mathcal{I}) \gg \texttt{OPT}(\mathcal{I})$). Let $\texttt{ADV}(\mathcal{I}) = v \gg \texttt{OPT}_T$ (i.e. the final total hitting and switching cost of ADV is v for some $v \in [L, U + 2\beta]$, and this is much greater than the optimal solution).

This is without loss of generality, since we can assume that v is the "best cost function" accepted by ADV and the consistency ratio changes strictly in favor of ADV. Based on the consistency constraint, we can derive a lower bound on the amount that CLIP *must* accept from ADV in order to stay $(1 + \epsilon)$-consistent.

To do this, we consider the following sub-cases:

- **Sub-case 2.1**: Let $v \geq \frac{U+\beta}{1+\epsilon}$.

In this sub-case, CLIP can fully ignore the advice, because the following consistency constraint is never binding (note that $\texttt{ADV}_t \geq \frac{U+\beta}{1+\epsilon}A^{(t)}$):

$$\texttt{CLIP}_{t-1} + f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{x}_t - \mathbf{a}_t\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A^{(t)})L + (A^{(t)} - z^{(t-1)} - c(\mathbf{x}_t))U$$
$$\leq (1 + \epsilon)\left[\texttt{ADV}_t + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A^{(t)})L\right],$$
$$(1 - A^{(t)})L + (A^{(t)})U + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} \leq (1 + \epsilon)\left[\texttt{ADV}_t + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A^{(t)})L\right],$$
$$(1 - A^{(t)})L + UA^{(t)} + \beta A^{(t)} \leq (1 + \epsilon)\left[\frac{U + \beta}{1 + \epsilon}A^{(t)} + (1 - A^{(t)})L\right]$$

- **Sub-case 2.2**: Let $v \in (L, \frac{U+\beta}{1+\epsilon})$.

To remain $(1 + \epsilon)$ consistent, CLIP must accept some of these "bad cost functions" denoted by v in the worst-case. We would like to derive a lower bound $z_{\texttt{ADV}}$, such that $z_{\texttt{ADV}}$ describes the minimum amount that CLIP must accept from ADV in order to always satisfy the $(1 + \epsilon)$ consistency constraint.

Based on the consistency constraint, we have the following:

$$\texttt{CLIP}_{t-1} + f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{x}_t - \mathbf{a}_t\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A^{(t)})L + (A^{(t)} - z^{(t-1)} - c(\mathbf{x}_t))U$$
$$\leq (1 + \epsilon)\left[\texttt{ADV}_t + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} + (1 - A^{(t)})L\right].$$

We let $f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w})} + \|\mathbf{x}_t - \mathbf{a}_t\|_{\ell_1(\mathbf{w})} + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w})} \leq vc(\mathbf{x}_t)$ for any $\mathbf{x}_t : c(\mathbf{x}_t) < c(\mathbf{a}_t)$, which holds by convexity of the cost functions $f_t(\cdot)$ and a prevailing assumption that $c(\mathbf{x}_t) \leq c(\mathbf{a}_t)$ for the "bad cost functions" accepted by ADV. Note that $v - U$ is negative (by the condition of Sub-case 1.2):

$$\texttt{CLIP}_{t-1} + vc(\mathbf{x}_t) + L - LA^{(t)} + UA^{(t)} - Uz^{(t-1)} - U\mathbf{x}_t \leq (1+\epsilon)\left[vA^{(t-1)} + vc(\mathbf{a}_t) + L - LA^{(t)}\right],$$

$$vc(\mathbf{x}_t) - U\mathbf{x}_t \leq (1+\epsilon)\left[vA^{(t-1)} + vc(\mathbf{a}_t) + L - LA^{(t)}\right] - \texttt{CLIP}_{t-1} - L + LA^{(t)} - UA^{(t)} + Uz^{(t-1)},$$

$$vc(\mathbf{x}_t) - U\mathbf{x}_t \leq vA^{(t)} - UA^{(t)} - \texttt{CLIP}_{t-1} + Uz^{(t-1)} + \epsilon\left[vA^{(t-1)} + vc(\mathbf{a}_t) + L - LA^{(t)}\right],$$

$$\mathbf{x}_t \geq \frac{vA^{(t)} - UA^{(t)} - \texttt{CLIP}_{t-1} + Uz^{(t-1)} + \epsilon\left[vA^{(t)} + L - LA^{(t)}\right]}{v-U}.$$

In the event that $A^{(t-1)} = 0$ (i.e. nothing has been accepted so far by either ADV or CLIP), we have the following:

$$\mathbf{x}_t \geq \frac{vc(\mathbf{a}_t) - Uc(\mathbf{a}_t) + \epsilon\left[vc(\mathbf{a}_t) + L - Lc(\mathbf{a}_t)\right]}{v-U},$$

$$\mathbf{x}_t \geq \mathbf{a}_t - \frac{\epsilon\left[vc(\mathbf{a}_t) + L - Lc(\mathbf{a}_t)\right]}{U-v}.$$

Through a recursive definition, we can show that for any $A^{(t)}$, given that CLIP has accepted $z^{(t-1)}$ of ADV's suggested prices so far, it must set $\mathbf{x}_t$ such that:

$$z^{(t)} \geq z^{(t-1)} + \mathbf{a}_t - \frac{\epsilon\left[vc(\mathbf{a}_t) + L - Lc(\mathbf{a}_t)\right]}{U-v}.$$

Continuing the assumption that v is constant, if CLIP has accepted $z^{(t-1)}$ thus far, we have the following if we assume that the acceptance up to this point happened in a single previous time step m:

$$c(\mathbf{x}_t) \geq A^{(t)} + \frac{Uc(\mathbf{x}_m) - \texttt{CLIP}_{t-1} + \epsilon\left[vA^{(t)} + L - LA^{(t)}\right]}{v-U},$$

$$c(\mathbf{x}_t) \geq c(\mathbf{a}_t) + c(\mathbf{a}_m) + \frac{Uc(\mathbf{x}_m) - vc(\mathbf{x}_m) + \epsilon\left[v(c(\mathbf{a}_t) + c(\mathbf{a}_m)) + L - L(c(\mathbf{a}_t) + c(\mathbf{a}_m))\right]}{v-U},$$

$$c(\mathbf{x}_t) \geq c(\mathbf{a}_t) + c(\mathbf{a}_m) - \mathbf{x}_m + \frac{\epsilon\left[v(c(\mathbf{a}_t) + c(\mathbf{a}_m)) + L - L(c(\mathbf{a}_t) + c(\mathbf{a}_m))\right]}{v-U},$$

$$c(\mathbf{x}_t) + c(\mathbf{x}_m) \geq c(\mathbf{a}_t) + c(\mathbf{a}_m) + \frac{\epsilon\left[v(c(\mathbf{a}_t) + c(\mathbf{a}_m)) + L - L(c(\mathbf{a}_t) + c(\mathbf{a}_m))\right]}{v-U},$$

$$z^{(t)} \geq A^{(t)} + \frac{\epsilon\left[vA^{(t)} + L - LA^{(t)}\right]}{v-U}.$$

This gives intuition into the desired z_{ADV} bound. The above describes and motivates that the *aggregate* acceptance by CLIP at any given time step t must satisfy a lower bound. Consider that the worst case for Sub-case 1.2 occurs when all of the v prices accepted by ADV arrive first, before any prices which would be considered by the pseudo-cost minimization. Then let $A^{(t)} = 1$ for some arbitrary time step t, and we have the following lower bound on z_{ADV}:

$$z_{\mathsf{ADV}} \geq 1 - \frac{v\epsilon}{U-v}.$$

If CLIP is forced to use z_{ADV} of its utilization to be $(1+\epsilon)$ consistent against ADV, that leaves at most $(1 - z_{\mathsf{ADV}})$ utilization for robustness.

We define $z' = \min(1 - z_{\mathsf{ADV}}, z_{\mathrm{PCM}})$ and consider the following two cases.

• **Sub-case 2.2.1**: if $z' = z_{\text{PCM}}$, the worst-case competitive ratio is bounded by the following. Note that if $z' = z_{\text{PCM}}$, the amount of utilization that CLIP can use to "be robust" is exactly the same as in **Case 1**:

$$\frac{\texttt{CLIP}(\mathcal{I})}{\texttt{OPT}(\mathcal{I})} \leq \frac{\int_0^{z_{\text{PCM}}} \phi(u)du + \beta z_{\text{PCM}} + (1 - z_{\mathsf{ADV}} - z_{\text{PCM}})U + z_{\mathsf{ADV}} v}{L},$$
$$\leq \frac{\int_0^{z_{\text{PCM}}} \phi(u)du + \beta z_{\text{PCM}} + (1 - z_{\text{PCM}})U}{L} \leq \gamma^{\epsilon}.$$

• **Sub-case 2.2.2**: if $z' = 1 - z_{\mathsf{ADV}}$, the worst-case competitive ratio is bounded by the following. Note that CLIP *cannot* use z_{PCM} of its utilization for robustness, so the following bound assumes that the cost functions accepted by CLIP are bounded by the *worst* $(1 - z_{\mathsf{ADV}})$ *fraction* of the pseudo-cost threshold function ϕ^{ϵ} (which follows since ϕ^{ϵ} is non-decreasing on $z \in [0, 1]$):

$$\frac{\texttt{CLIP}(\mathcal{I})}{\texttt{OPT}(\mathcal{I})} \leq \frac{\int_0^{1-z_{\mathsf{ADV}}} \phi(u)du + \beta(1 - z_{\mathsf{ADV}}) + z_{\mathsf{ADV}} v}{L}.$$

Note that if $z' = 1 - z_{\mathsf{ADV}}$, we know that $1 - z_{\mathsf{ADV}} < z_{\text{PCM}}$, which further gives the following by definition of z_{ADV}:

$$1 - z_{\text{PCM}} < 1 - \frac{v\epsilon}{U - v},$$
$$v\epsilon < (U - v) z_{\text{PCM}},$$
$$v < \frac{U}{(1 + \frac{\epsilon}{z_{\text{PCM}}})}.$$

Since $z_{\mathsf{ADV}} v = \left(\frac{(1 - z_{\text{PCM}})U}{1 + \frac{\epsilon}{z_{\text{PCM}}}} \right)$, we have the following:

$$\frac{\texttt{CLIP}(\mathcal{I})}{\texttt{OPT}(\mathcal{I})} \leq \frac{\int_0^{1-z_{\mathsf{ADV}}} \phi(u)du + \beta(1 - z_{\mathsf{ADV}}) + \left(\frac{(1 - z_{\text{PCM}})U}{1 + \frac{\epsilon}{z_{\text{PCM}}}} \right)}{L},$$
$$\leq \frac{\int_0^{z_{\text{PCM}}} \phi(u)du + \beta z_{\text{PCM}} + (1 - z_{\text{PCM}})U}{L} \leq \gamma^{\epsilon}.$$

Thus, by combining the bounds in each of the above two cases, the result follows, and we conclude that CLIP is γ^{ϵ}-robust for any advice ADV. □

Having proven Lemma C.1 (consistency) and Lemma C.2 (robustness), the statement of Theorem 4.3 follows – CLIP is $(1 + \epsilon)$-consistent and γ^{ϵ}-robust given any advice for CFL. □

C.3 Proof of Corollary 4.4

In this section, we prove Corollary 4.4, which shows that CLIP is $(1+\epsilon)$-consistent and γ^{ϵ}-robust for MAL, where γ^{ϵ} is defined in (8).

Proof of Corollary 4.4. We show the above result by separately considering consistency (the competitive ratio when advice is correct) and robustness (the competitive ratio when advice is not correct), relying on the proof of Theorem 4.3.

Consistency. By definition, MAL on a weighted star metric is identical to an instance of convex function chasing with a long-term constraint on $(\Delta_n, \|\cdot\|_{\ell_1(\mathbf{w}')})$, where Δ_n is the n-point simplex in $\mathbb{R}^n$ and $\|\cdot\|_{\ell_1(\mathbf{w}')}$

is the weighted ℓ_1 norm, with weights $\mathbf{w}'$ given by the corresponding edge weight in the underlying star metric.

Observe that the consistency proof given in Lemma C.1 holds when the consistency constraint at each time step is defined as follows:

$$\texttt{CLIP}_{t-1} + f_t(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}_{t-1}\|_{\ell_1(\mathbf{w}')} + \|\mathbf{x} - \mathbf{a}_t\|_{\ell_1(\mathbf{w}')} + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w}')} + (1 - z^{(t-1)} - c(\mathbf{x}))L + \max((A^{(t)} - z^{(t-1)} - c(\mathbf{x})),\ 0)(U - L) \leq (1+\epsilon)[\texttt{ADV}_t + \|\mathbf{a}_t\|_{\ell_1(\mathbf{w}')} + (1 - A^{(t)})L], \tag{29}$$

where $\mathbf{x}$ and $\mathbf{a}$ denote decisions by CLIP and ADV (respectively) supported on Δ_n. Thus, since the consistency proof in Lemma C.1 exactly holds under the CFL vector space corresponding to MAL, we conclude that CLIP is $(1+\epsilon)$-consistent for MAL.

Robustness. First, we note that the robustness proof given in Lemma C.2 assumes OPT does not pay any switching cost. This implies that the proof of Lemma C.2 meets the conditions of Proposition 2.3, which states that any performance bound for an arbitrary ALG solving CFL which assumes OPT pays no switching cost translates to an identical bound for MAL, where the problem's parameters can be recovered by constructing a corresponding CFL instance according to Lemma 2.2.

Thus, by Proposition 2.3, we conclude that CLIP is γ^ϵ-robust for MAL, where γ^ϵ is defined in (8).

By combining the two results, the statement of Corollary 4.4 follows – CLIP is $(1+\epsilon)$-consistent and γ^ϵ-robust given any advice ADV for MAL. □

C.4 Proof of Theorem 4.5

In this section, we prove Theorem 4.5, which shows that any $(1+\epsilon)$-consistent algorithm for CFL is at least γ^ϵ-robust, where γ^ϵ is as defined in (8).

Proof of Theorem 4.5. To show this result, we leverage the same special family of y-adversaries for CFL defined in Definition B.4, where $y \in [L, U]$. Recall that $k = \arg\max_{i \in [d]} \mathbf{w}_i$, where $\mathbf{w}$ is the weight vector for $\|\cdot\|_{\ell_1(\mathbf{w})}$.

As in the proof of Theorem 3.4, we note that for adversary $\mathcal{A}_y$, the optimal offline solution is $\texttt{OPT}(\mathcal{A}_y) = y + {}^{2\beta}/_{m}$, and that as m grows large, $\texttt{OPT}(\mathcal{A}_y) \to y$.

Against these adversaries, we consider two types of advice – the first is *bad* advice, which sets $\mathbf{a}_t = \mathbf{0}$ for all time steps $t < T$ (i.e., before the compulsory trade), incurring a final cost of $U + 2\beta$.

On the other hand, *good* advice sets $\mathbf{a}_t = \mathbf{0}$ for all time steps up to the first time step when y is revealed, at which point it sets $\mathbf{a}_t^k = {}^{1}/_{m}$ to achieve final cost $\texttt{ADV}(\mathcal{A}_y) = \texttt{OPT}(\mathcal{A}_y) = y + {}^{2\beta}/_{m}$.

We let $g(y)$ denote a *robust conversion function* $[L, U] \to [0, 1]$, which fully quantifies the actions of a learning-augmented algorithm LALG playing against adaptive adversary $\mathcal{A}_y$, where $g(y)$ gives the progress towards the long-term constraint under the instance $\mathcal{A}_y$ before (either) the compulsory trade or the black-box advice sets $\mathbf{a}_t^k > 0$. Note that for large w, the adaptive adversary $\mathcal{A}_{y-\delta}$ is equivalent to first playing $\mathcal{A}_y$ (besides the last two batches of cost functions), and then processing batches with cost functions $\mathbf{Down}^{w_y+1}(\mathbf{x})$ and $\mathbf{Up}(\mathbf{x})$. Since LALG is deterministic and the conversion is unidirectional (irrevocable), we must have that $g(y-\delta) \geq g(y)$, i.e. $g(y)$ is non-increasing in $[L, U]$.

As in the proof of Theorem 3.4, the adaptive nature of each y-adversary forces any algorithm to incur a switching cost proportional to $g(y)$, specifically denoted by $2\beta g(y)$.

For any γ-robust online algorithm LALG given any arbitrary black-box advice, the following must hold:

$$\texttt{LALG}(\mathcal{A}_y) \leq \gamma \texttt{OPT}(\mathcal{A}_y) = \gamma y, \ \forall y \in [L, U].$$

The cost of LALG with conversion function g on an instance $\mathcal{A}_y$ is $\texttt{LALG}(\mathcal{A}_y) = g({}^{U}/_{\gamma}){}^{U}/_{\gamma} - \int_{U/\gamma}^{y} u dg(u) + 2\beta g(y) + (1 - g(y))U$, where $udg(u)$ is the cost of buying $dg(u)$ utilization at price u, the last term is from the compulsory trade, and the second to last term is the switching cost incurred by LALG.

This implies that $g(y)$ must satisfy the following:

$$g(U/\gamma)U/\gamma - \int_{U/\gamma}^{y} u dg(u) + 2\beta g(y) + (1 - g(y))U \leq \gamma y, \; \forall y \in [L, U].$$

By integral by parts, the above implies that the conversion function must satisfy $g(y) \geq \frac{U-\gamma y}{U-y-2\beta} - \frac{1}{U-y-2\beta} \int_{U/\gamma}^{y} g(u)du$. By Grönwall's Inequality [MPF91][Theorem 1, p. 356], we have that

$$g(y) \geq \frac{U-\gamma y}{U-y-2\beta} - \frac{1}{U-y-2\beta} \int_{U/\gamma}^{y} \frac{U-\gamma u}{U-u-2\beta} \cdot \exp\left(\int_{u}^{y} \frac{1}{U-r-2\beta} dr\right) du \tag{30}$$

$$\geq \frac{U-\gamma y}{U-y-2\beta} - \int_{U/\gamma}^{y} \frac{U-\gamma u}{(U-u-2\beta)^2} du \tag{31}$$

$$\geq \frac{U-\gamma y}{U-y-2\beta} - \left[\frac{U\gamma - U - 2\beta\gamma}{u+2\beta-U} - \gamma \ln(u+2\beta-U)\right]_{U/\gamma}^{y} \tag{32}$$

$$\geq \gamma \ln(y+2\beta-U) - \gamma \ln(U/\gamma + 2\beta - U), \quad \forall y \in [L, U]. \tag{33}$$

In addition, to simultaneously be η-consistent when the advice *is* correct, LALG must satisfy $\texttt{LALG}(\mathcal{A}_L) \leq \eta \texttt{OPT}(\mathcal{A}_L) = \eta L$. If the advice is correct (and m is sufficiently large), we assume that LALG pays no switching cost to satisfy the long-term constraint at the best cost functions L. It must still pay for switching incurred by the robust algorithm (recall that OPT pays no switching cost).

$$\int_{U/\gamma}^{L} g(u)du + 2\beta g(L) \leq \eta L - L. \tag{34}$$

By combining equations (33) and (34), the conversion function $g(y)$ of any γ-robust and η-consistent online algorithm must satisfy the following:

$$\gamma \int_{U/\gamma}^{L} \ln\left(\frac{u+2\beta-U}{U/\gamma+2\beta-U}\right) du + 2\beta\left[\gamma \ln\left(\frac{u+2\beta-U}{U/\gamma+2\beta-U}\right)\right] \leq \eta L - L. \tag{35}$$

When all inequalities are binding, this equivalently gives that

$$\eta \geq \gamma + 1 - \frac{U}{L} + \frac{\gamma(U-L)}{L} \ln\left(\frac{U-L-2\beta}{U-U/\gamma^{\epsilon}-2\beta}\right). \tag{36}$$

We define η such that $\eta \coloneqq (1+\epsilon)$. By substituting for η into (36), we recover the definition of γ^{ϵ} as given by (8), which subsequently completes the proof. Thus, we conclude that any $(1+\epsilon)$-consistent algorithm for CFL is at least γ^{ϵ}-robust. □

C.5 Proof of Corollary 4.6

In this section, we prove Corollary 4.6, which shows that any $(1+\epsilon)$-consistent algorithm for MAL is at least γ^{ϵ}-robust, where γ^{ϵ} is as defined in (8).

Proof of Corollary 4.6. To show this result, we leverage the same special family of y-adversaries for CFL defined in Definition B.5, where $y \in [L, U]$. Recall that $k = \arg\max_{a \in [n]} \mathbf{w}^a$, deonotes the largest edge weight of any (non-OFF) point in the metric space, and $\beta = \mathbf{w}^k$.

As in the proof of Theorem 3.4, we note that for adversary $\mathcal{A}_y$, the optimal offline solution is $\texttt{OPT}(\mathcal{A}_y) = y + 2\beta/m$, and that as m grows large, $\texttt{OPT}(\mathcal{A}_y) \to y$.

Against these adversaries, we consider two types of advice – the first is *bad* advice, which sets $\mathbf{a}_t^{a'} = 1$ (i.e., ADV stays in the OFF point) for all time steps $t < T$ (i.e., before the compulsory trade), incurring a final cost of $U + 2\beta$.

On the other hand, *good* advice sets $\mathbf{a}_t^{a'} = 1$ for all time steps up to the first time step when y is revealed, at which point it sets $\mathbf{a}_t^k = 1/m$ to achieve final cost $\texttt{ADV}(\mathcal{A}_y) = \texttt{OPT}(\mathcal{A}_y) = y + 2\beta/m$.

As previously, we let $g(y)$ denote a *robust conversion function* $[L, U] \rightarrow [0, 1]$, which fully quantifies the actions of a learning augmented algorithm LALG playing against adaptive adversary $\mathcal{A}_y$. Since LALG is deterministic and the conversion is unidirectional (irrevocable), $g(y)$ is non-increasing in $[L, U]$. Intuitively, the entire long-term constraint should be satisfied if the minimum possible price is observed, i.e $g(L) = 1$.

As in Theorem 4.5, the adaptive nature of each y-adversary forces any deterministic ALG to incur a switching cost of $2\beta g(y)$ on adversary $\mathcal{A}_y$, and we assume that ALG does not incur a significant switching cost during the final batch (i.e., during the compulsory trade).

For any γ-robust LALG given any arbitrary black-box advice, the following must hold:

$$\texttt{LALG}(\mathcal{A}_y) \leq \gamma \texttt{OPT}(\mathcal{A}_y) = \gamma y, \ \forall y \in [L, U].$$

The cost of LALG with conversion function g on an instance $\mathcal{A}_y$ is $\texttt{LALG}(\mathcal{A}_y) = g(U/\gamma)U/\gamma - \int_{U/\gamma}^{y} u dg(u) + 2\beta g(y) + (1 - g(y))U$, where $u dg(u)$ is the cost of buying $dg(u)$ utilization at price u, the last term is from the compulsory trade, and the second to last term is the switching cost incurred by LALG. Note that this expression for the cost is exactly as defined in Theorem 4.5.

Thus by Theorem 4.5, for any learning-augmented algorithm LALG which is simultaneously η-consistent and γ-robust, the conversion function $g(\cdot)$ must satisfy the following inequality (via integral by parts and Grönwall's Inequality [MPF91, Theorem 1, p. 356]):

$$\gamma \int_{U/\gamma}^{L} \ln \left(\frac{u + 2\beta - U}{U/\gamma + 2\beta - U} \right) du + 2\beta \left[\gamma \ln \left(\frac{u + 2\beta - U}{U/\gamma + 2\beta - U} \right) \right] \leq \eta L - L. \tag{37}$$

When all inequalities are binding, this equivalently gives that the optimal η and γ satisfy:

$$\eta \geq \gamma + 1 - \frac{U}{L} + \frac{\gamma(U - L)}{L} \ln \left(\frac{U - L - 2\beta}{U - U/\gamma^\epsilon - 2\beta} \right). \tag{38}$$

We define η such that $\eta \coloneqq (1 + \epsilon)$. By substituting for η into (36), we recover the definition of γ^ϵ as given by (8), which subsequently completes the proof. Thus, we conclude that any $(1 + \epsilon)$-consistent algorithm for CFL is at least γ^ϵ-robust. □