Proceedings of Machine Learning Research vol 178:1–42, 2022 35th Annual Conference on Learning Theory

Chasing Convex Bodies and Functions with Black-Box Advice

Nicolas Christianson NCHRISTIANSON@CALTECH.EDU
Tinashe Handina THANDINA@CALTECH.EDU
Adam Wierman ADAMW@CALTECH.EDU
California Institute of Technology

Editors: Po-Ling Loh and Maxim Raginsky

Abstract

We consider the problem of convex function chasing with black-box advice, where an online decision-maker aims to minimize the total cost of making and switching between decisions in a normed vector space, aided by black-box advice such as the decisions of a machine-learned algorithm. The decision-maker seeks cost comparable to the advice when it performs well, known as *consistency*, while also ensuring worst-case *robustness* even when the advice is adversarial. We first consider the common paradigm of algorithms that switch between the decisions of the advice and a competitive algorithm, showing that no algorithm in this class can improve upon 3-consistency while staying robust. We then propose two novel algorithms that bypass this limitation by exploiting the problem's convexity. The first, INTERP, achieves $(\sqrt{2}+\epsilon)$-consistency and $\mathcal{O}(\frac{C}{\epsilon^2})$-robustness for any $\epsilon > 0$, where C is the competitive ratio of an algorithm for convex function chasing or a subclass thereof. The second, BDINTERP, achieves $(1+\epsilon)$-consistency and $\mathcal{O}(\frac{CD}{\epsilon})$-robustness when the problem has bounded diameter D. Further, we show that BDINTERP achieves near-optimal consistency-robustness trade-off for the special case where cost functions are α-polyhedral.

Keywords: Convex body chasing, online optimization, learning-augmented algorithms

1. Introduction

We study the problem of convex function chasing (CFC), in which a player chooses decisions $\boldsymbol{x}_t$ online from a normed vector space $\mathcal{X} = (X, \|\cdot\|)$ in order to minimize the total cost $\sum_{t=1}^{T} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|$, where each f_t is a convex "hitting" cost function that is revealed prior to the player's selection of $\boldsymbol{x}_t$, and the term $\|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|$ penalizes changing decisions between rounds. A number of subclasses of CFC have been discussed in the literature, characterized by various restrictions on the class of cost functions f_t. Of particular note is the special case of convex body chasing (CBC), in which each cost function f_t is the $\{0, \infty\}$ indicator of a convex set K_t, so that each decision $\boldsymbol{x}_t$ must reside strictly within K_t. Algorithms for CFC and its special cases are judged on the basis of their competitive ratio, i.e., the worst-case ratio in cost between the algorithm and the hindsight optimal sequence of decisions (Definition 1).

Convex body chasing and function chasing were introduced by Friedman and Linial (1993) as continuous versions of several fundamental problems in online algorithms, including Metrical Task Systems (Borodin et al. (1992)) and the k-server problem (Koutsoupias and Papadimitriou (1995)). CFC has also been studied recently as the problem of "smoothed online convex optimization" (SOCO), introduced by Lin et al. (2012). The basic premise of CFC/SOCO, of choosing decisions online to optimize per-round costs with minimal movement between decisions, has seen wide application in a number of domains, including data center load-balancing (Lin et al. (2012)) and right-sizing (Lin et al. (2013); Albers and Quedenfeld (2018)), electric vehicle charging (Kim and Giannakis (2014)), and control (Goel and Wierman (2019); Li et al. (2021)).

In high-dimensional settings, the performance of algorithms for CBC and CFC can be arbitrarily poor: Friedman and Linial (1993) showed a $\sqrt{d}$ lower bound on the competitive ratio of any algorithm for CBC (and thus CFC) in d-dimensional Euclidean space, which Bubeck et al. (2019) extended to an $\Omega(\max\{\sqrt{d}, d^{1-\frac{1}{p}}\})$ lower bound in $\mathbb{R}^d$ with the ℓ^p norm. Prospects are poor even for subclasses of CFC with additional restrictions on the functions f_t. For instance, CFC with α-*polyhedral* cost functions, i.e. where each f_t has a unique minimizer away from which it grows with slope at least $\alpha > 0$, has been studied widely in the SOCO literature. State-of-the-art algorithms in this setting achieve competitive ratio $\mathcal{O}(\alpha^{-1})$, which grows arbitrarily large in the $\alpha \to 0$ limit (Chen et al. (2018); Zhang et al. (2021)).

The modern tools of machine learning wield great promise for improving upon these pessimistic performance guarantees. That is, for practical applications, there is often large amounts of data recorded from past problem instances, enabling the training of machine learning models that can outperform traditional, conservative online algorithms. However, these machine-learned algorithms are "black boxes," in the sense that they lack rigorous, worst-case performance guarantees. Such black-box algorithms might *typically* outperform robust online algorithms, but their lack of uncertainty quantification can lead to arbitrarily poor performance in the worse case, if they are deployed on held-out problem instances or under distribution shift.

Thus, a natural question arises: *is it possible to develop algorithms that achieve both the worst-case guarantees of traditional online algorithms for* CFC *and the average-case performance of machine-learned algorithms or other sources of black-box "advice"?*

These desiderata are naturally encoded in the notions of *robustness* and *consistency* introduced by Lykouris and Vassilvtiskii (2018) in the context of competitive caching. In this framework, a *consistent* algorithm is one with a competitive ratio with respect to the black-box advice, implying that when the advice is accurate, the algorithm will perform well; on the other hand, a *robust* algorithm is one that has a finite competitive ratio, regardless of advice performance. Our goal is to develop algorithms with tunable robustness and consistency guarantees, so that a decision-maker can decide in advance the trade-off they wish to make between exploiting good advice performance and ensuring worst-case robustness in the case that advice performs poorly.

1.1. Contributions

We answer the question above by proposing novel algorithms with tunable robustness and consistency bounds for CFC and any subclass thereof. In particular, we reduce the problem of designing robust and consistent algorithms for CFC to the design of *bicompetitive meta-algorithms* (Definitions 4, 5), which are unified "recipes" for combining black-box advice with a robust algorithm in a manner that guarantees a competitive ratio with respect to both ingredients. These "recipes" are very general – they can be used to combine advice with *any* algorithm for any subclass of CFC to obtain a customized robustness and consistency guarantee for that subclass without explicit knowledge of the algorithm or advice design.

More specifically, our contributions are twofold. We first consider the class of "switching" algorithms, which switch between the decisions of the advice and a robust algorithm. This class of algorithms has received considerable attention in the literature on robustness and consistency, and in particular, all prior algorithms for CFC with black-box advice in dimension greater than one have been switching algorithms. We prove a fundamental limit on the robustness and consistency of any switching algorithm for CFC, showing that *no* switching algorithm for CFC can improve on

3-consistency while obtaining finite robustness (Theorem 8). We give a switching meta-algorithm SWITCH (Appendix B.1, Algorithm 4) achieving this fundamental limit, obtaining $(3 + \mathcal{O}(\epsilon))$-consistency and $\mathcal{O}(\frac{C}{\epsilon^2})$-robustness for any $\epsilon > 0$, where C is the competitive ratio of any algorithm for CFC or a subclass thereof. We further show that the fundamental limit on switching algorithms can be broken in the special case of nested CBC, in which successive bodies are nested. In this setting, we provide an algorithm NESTEDSWITCH (Algorithm 1) achieving $(1 + \epsilon)$-consistency along with $\mathcal{O}(\frac{d}{\epsilon})$-robustness for nested CBC in d dimensions (Proposition 9).

Second, galvanized by the limitations of switching algorithms, we develop algorithms exploiting the convexity of the CFC problem to obtain improved robustness and consistency bounds. We propose a meta-algorithm INTERP (Algorithm 2) that, given a C-competitive algorithm for a subclass of CFC, achieves $(\mu(\mathcal{X}) + \epsilon)$-consistency and $\mathcal{O}(\frac{C}{\epsilon^2})$-robustness for any desired $\epsilon > 0$, where $\mu(\mathcal{X})$ is a geometric constant depending on the structure of the normed space $\mathcal{X}$ that is $\sqrt{2}$ in any Hilbert space and is strictly less than 3 in any ℓ^p space, $p \in (1, \infty)$ (Theorem 11). Moreover, under the additional assumption that the advice and the C-competitive algorithm are never farther apart than some distance D, we give a meta-algorithm BDINTERP (Algorithm 3) that achieves $(1 + \epsilon)$-consistency and $\mathcal{O}(\frac{CD}{\epsilon})$-robustness (Theorem 12). In particular, BDINTERP gives nearly-optimal consistency and robustness for the problem of CFC with α-polyhedral cost functions when $D = O(1)$ in α.

A key feature of our results is their generality: our main results on bicompetitive meta-algorithms (Proposition 7, Theorems 11, 12) hold in vector spaces with any norm and arbitrary, even infinite, dimension. This enables application to problems where the decisions $\boldsymbol{x}_t$ are infinite-dimensional objects such as probability measures, which could arise in settings such as iterated games. Moreover, these meta-algorithms enable the design of customized robust and consistent algorithms for *any* subclass of CFC, since they are agnostic to the specific algorithms used. We illustrate this by giving specific robustness and consistency results for the cases of CFC, CBC, and CFC restricted to α-polyhedral hitting cost functions; we give further examples in Appendix F.

1.2. Related work

Our work contributes to the literatures on CBC, CFC, and SOCO as well as the emerging literature on online algorithms with black-box advice. We discuss each in turn below.

1.2.1. CBC, CFC, AND SOCO

The problems of convex body chasing and function chasing were introduced by Friedman and Linial (1993), who gave a competitive algorithm for CBC in 2-dimensional Euclidean space. The problem in general dimension d has been largely settled in the last few years. In the setting where subsequent bodies are nested, Argue et al. (2019) gave an $\mathcal{O}(d \log d)$-competitive algorithm in any norm, and Bubeck et al. (2019) later gave an $\mathcal{O}(\min\{d, \sqrt{d \log T}\})$-competitive algorithm in the Euclidean setting that uses the geometric Steiner point of the convex bodies. Later, Argue et al. (2021) and Sellke (2020) concurrently obtained $\mathcal{O}(d)$-competitive algorithms for general CFC. The latter work builds upon the methods of Bubeck et al. (2019), developing a "functional" Steiner point algorithm that is d-competitive for CBC and $(d + 1)$-competitive for CFC in any normed space, matching the lower bound of d in the ℓ^∞ norm setting.

Several special cases of CFC/SOCO with restrictions on hitting cost structure have been studied in the literature to the end of obtaining "dimension-free" competitive ratios for these subclasses. Chen et al. (2018) obtained the first such bound for the subclass of CFC where hitting cost functions

Problem	Algorithm Name	Competitive Ratio	Setting
CFC	Functional Steiner Point	$d+1$	$\mathbb{R}^d$ with any norm
CBC	Functional Steiner Point	d	$\mathbb{R}^d$ with any norm
αCFC	Greedy	$\max\left\{1, \frac{2}{\alpha}\right\}$	Any normed vector space
αCFC	Greedy OBD	$\mathcal{O}\left(\frac{1}{\alpha^{1/2}}\right)$	$\mathbb{R}^d$ with ℓ^2 norm

Table 1: Competitive ratios for state-of-the-art algorithms for CFC, CBC, and αCFC.

f_t are α-polyhedral, which we call αCFC. The authors propose an algorithm, "Online Balanced Descent" (OBD), which achieves a competitive ratio $\mathcal{O}(\frac{1}{\alpha})$. This upper bound has been successively refined, with the most recent entry a simple greedy algorithm from Zhang et al. (2021) that achieves competitive ratio $\max\{1, \frac{2}{\alpha}\}$ in any normed vector space of arbitrary (even infinite) dimension by moving to the minimizer of each hitting cost function. The $\mathcal{O}(\frac{1}{\alpha})$ upper bound has been broken by Lin (2022) in the finite-dimensional Euclidean setting with an $\mathcal{O}(\frac{1}{\alpha^{1/2}})$-competitive algorithm, Greedy OBD, that is optimal within the class of memoryless, rotation- and scale-invariant algorithms. Another subclass of CFC that has received attention is that with (κ, γ)-well-centered hitting cost functions, which generalize well-conditioned functions. Argue et al. (2020) propose an algorithm achieving an $\mathcal{O}(2^{\gamma/2}\kappa)$ competitive ratio for this subclass, and an improved algorithm achieving competitive ratio $\mathcal{O}(\sqrt{\kappa})$ for the particular class of κ-well-conditioned functions along with a nearly matching $\Omega(\kappa^{1/3})$ lower bound. We summarize the state-of-the-art algorithms and competitive ratios that we refer to in our later results in Table 1, giving an extended version of the table in Appendix A, Table 2.

1.2.2. ONLINE ALGORITHMS WITH BLACK-BOX ADVICE

The idea of using machine-learned, black-box advice to improve online algorithms was first proposed by Mahdian et al. (2012) to design algorithms for online ad allocation, load balancing, and facility location. Formal notions of *robustness* and *consistency* were later coined by Lykouris and Vassilvtiskii (2018) in the context of designing learning-augmented algorithms for caching. The last few years have seen a surge in the application of the robustness and consistency paradigm in designing online algorithms augmented with black-box advice for a multitude of problems, for example ski rental and non-clairvoyant scheduling (Purohit et al. (2018); Wei and Zhang (2020)), energy generation scheduling (Lee et al. (2021)), bidding and bin-packing (Angelopoulos et al. (2020)), and Q-learning (Golowich and Moitra (2021)).

Closest to our work are the recent papers of Antoniadis et al. (2020) and Rutten et al. (2022). The former considers the problem of designing algorithms for *metrical task systems* (MTS) with black-box advice. MTS can be thought of as (non-convex) function chasing on general metric spaces, and hence their results also give robustness/consistency guarantees for CFC. They apply two classical results on combining k-server algorithms (Fiat et al. (1994); Baeza-Yates et al. (1993)) and on combining MTS algorithms via k-experts algorithms (Blum and Burch (1997); Freund and Schapire (1997)) to devise, in our parlance, bicompetitive meta-algorithms for MTS. In particular, their first algorithm switches between the advice and a C-competitive algorithm for MTS, and achieves 9-consistency and $9C$-robustness. They also propose a randomized switching algorithm that, under the assumption that the metric space has bounded diameter D, obtains cost bounded in expectation by $\min\{(1+\epsilon)\mathrm{C}_{\mathrm{ADV}} + \mathcal{O}(\frac{D}{\epsilon}), (1+\epsilon)C \cdot \mathrm{C}_{\mathrm{OPT}} + \mathcal{O}(\frac{D}{\epsilon})\}$, where $\mathrm{C}_{\mathrm{ADV}}$ is the cost of the advice and $\mathrm{C}_{\mathrm{OPT}}$ is the optimal cost. However, the large $\mathcal{O}(\frac{D}{\epsilon})$ additive factors in their result preclude $(1+\epsilon)$-consistency, since when $\mathrm{C}_{\mathrm{ADV}} = \mathcal{O}(1)$, the consistency bound will be $1 + \epsilon + \Omega(\frac{D}{\epsilon})$. This is to be expected, since as we show in Section 3, no deterministic switching algorithm can improve on 3-consistency while having finite robustness. Moreover, their results due not allow tuning robustness and consistency, i.e., neither algorithm allows trading-off robustness in order to obtain consistency arbitrarily close to 1.

On the other hand, Rutten et al. (2022) considers the problem of CFC with α-polyhedral hitting costs (αCFC), but with the convexity assumption dropped from the hitting costs f_t. They obtain a $(1+\epsilon)$-consistent, $2^{\tilde{\mathcal{O}}(\frac{1}{\alpha\epsilon})}$-robust algorithm in this setting, together with a lower bound showing that this exponential trade-off between robustness and consistency is necessary due to their non-convex setting. Their algorithm is a switching algorithm and crucially depends on the α-polyhedral structure of the hitting cost functions, and hence cannot be extended to general CFC. The authors also propose an algorithm for CFC in the 1-dimensional case (where $X = \mathbb{R}$) that achieves $(1+\epsilon)$-consistency and $\mathcal{O}(\frac{1}{\epsilon^2})$ robustness, and they prove a lower bound of $(1+\epsilon)$-consistency and $\mathcal{O}(\frac{1}{\epsilon})$ robustness on *any* algorithm for CFC with black-box advice. They leave open the broader problem of developing robust and consistent algorithms for CFC and its many subclasses in the higher-dimensional setting.

1.3. Notation

Throughout this paper X refers to a real vector space of arbitrary dimension. When a norm $\|\cdot\|$ is distinguished, $B(\boldsymbol{x}, r)$ is the closed $\|\cdot\|$-ball of radius $r \geq 0$ centered at $\boldsymbol{x}$, and $\Pi_K \boldsymbol{x}$ is a metric projection of the point $\boldsymbol{x} \in X$ onto a closed convex set K. For $\boldsymbol{x}, \boldsymbol{y} \in X$, we define $[\boldsymbol{x}, \boldsymbol{y}] \coloneqq \{\boldsymbol{z} \in X : \boldsymbol{z} = \lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}, \lambda \in [0,1]\}$ as the convex span of $\boldsymbol{x}$ and $\boldsymbol{y}$. The non-negative reals are denoted by $\mathbb{R}_+$, and for $T \in \mathbb{N}$, we write $[T] \coloneqq \{1, \ldots, T\}$. Asymptotic notation involving the variable $\epsilon > 0$ reflects the asymptotic regime $\epsilon \to 0$.

2. Preliminaries

We consider the general problem of *convex function chasing* (CFC) on a real normed vector space $\mathcal{X} = (X, \|\cdot\|)$. In particular, we make no assumption on either the dimension of $\mathcal{X}$ or on the choice of norm $\|\cdot\|$. In CFC, a decision-maker begins at some initial point $\boldsymbol{x}_0 \in X$, and at each time $t \in \mathbb{N}$ is handed a convex function $f_t : X \to \mathbb{R}_+$ and must choose some $\boldsymbol{x}_t \in X$, paying both the *hitting cost* $f_t(\boldsymbol{x}_t)$, as well as the *movement* or *switching cost* $\|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|$ induced by the norm. Crucially, $\boldsymbol{x}_t$ is chosen prior to the revelation of any future cost functions f_k, $k > t$, i.e., decisions are made *online*. The game ends at some time $T \in \mathbb{N}$, which is unknown to the decision-maker in advance. We refer to a tuple $(\boldsymbol{x}_0, f_1, \ldots, f_T)$ as an *instance* of the CFC problem. The total cost incurred by the decision-maker on a problem instance is $\sum_{t=1}^{T} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|$.

Informally, an *online algorithm* for CFC is an algorithm that, on a given instance of CFC, produces decisions online. We denote by ALG_t the t^{th} decision made by an online algorithm ALG; by convention, $\mathrm{ALG}_0 \coloneqq \boldsymbol{x}_0$, the starting point of the instance. Then the cost $\mathrm{C}_{\mathrm{ALG}}$ incurred by ALG

on an instance is

$$\mathrm{C}_{\mathrm{ALG}} = \sum_{t=1}^{T} f_t(\mathrm{ALG}_t) + \|\mathrm{ALG}_t - \mathrm{ALG}_{t-1}\|.$$

We also introduce the partial cost notation $\mathrm{C}_{\mathrm{ALG}}(t, t') = \sum_{i=t}^{t'} f_i(\mathrm{ALG}_i) + \|\mathrm{ALG}_i - \mathrm{ALG}_{i-1}\|$, defined for $1 \leq t \leq t' \leq T$. We refer to the set of all online algorithms for CFC as $\mathcal{A}_{\mathsf{CFC}}$.

We typically compare online algorithms for CFC against OPT, the offline optimal algorithm that chooses the hindsight optimal sequence of decisions for any problem instance. Its cost is the optimal value of the following convex program:

$$\mathrm{C}_{\mathrm{OPT}} = \mathrm{C}_{\mathrm{OPT}}(\boldsymbol{x}_0, f_1, \ldots, f_T) := \min_{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T \in X} \sum_{t=1}^{T} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\|$$

and its decisions are determined by the optimal solution. To evaluate the performance of an online algorithm for CFC, we consider the *competitive ratio*, which measures the worst case ratio in costs between an algorithm and OPT. In the following, we define both the conventional competitive ratio as well as a generalization that allows for comparing against arbitrary benchmark algorithms.

Definition 1 *Let* $\mathrm{ALG}^{(1)}$ *be an online algorithm for* CFC*, and let* $\mathrm{ALG}^{(2)}$ *be another (not necessarily online) algorithm for* CFC.[1] $\mathrm{ALG}^{(1)}$ *is defined to be* C***-competitive with respect to*** $\mathrm{ALG}^{(2)}$ *if, regardless of problem instance,* $\mathrm{C}^{(1)}_{\mathrm{ALG}} \leq C \cdot \mathrm{C}^{(2)}_{\mathrm{ALG}}$*. In particular, if* $\mathrm{ALG}^{(2)} = \mathrm{OPT}$*, we simply say that* $\mathrm{ALG}^{(1)}$ *is* C***-competitive****, or has* ***competitive ratio*** C*.*

2.1. Subclasses of Convex Function Chasing

CFC is a broad set of problems and many subclasses have received attention in the literature. We consider several subclasses of the general CFC problem in this work, distinguished by different assumptions on the hitting cost functions. In this section, we briefly define the subclasses of CFC which we refer to in our later results in the main text. We give more detailed definitions of these and several other subclasses of CFC in Appendix A.1.

2.1.1. CONVEX BODY CHASING

In the problem of convex body chasing (CBC), the decision-maker must choose each decision $\boldsymbol{x}_t$ from a convex body $K_t \subseteq X$ that is revealed online. This can be seen as a special case of CFC where f_t is 0 on K_t and ∞ elsewhere; see Appendix A.1.1 for more details on this equivalence. A notable special case of CBC is the problem of *nested* convex body chasing (NCBC), in which subsequent bodies are nested, i.e., $K_t \supseteq K_{t+1}$ for each t. We define $\mathcal{A}_{\mathsf{CBC}}$ as the set of all online algorithms for CBC that are *feasible*, i.e., that produce decisions within the convex body K_t at each time. We define $\mathcal{A}_{\mathsf{NCBC}}$ similarly as the set of *feasible* online algorithms for NCBC.

2.1.2. α-POLYHEDRAL CONVEX FUNCTION CHASING

Several subclasses of CFC have been studied in the literature with hitting cost functions f_t restricted so as to enable dimension-free competitive ratios. One of the most well-studied such subclasses is

1. Like OPT, the decision of $\mathrm{ALG}^{(2)}$ at some time t is allowed to depend on problem instance data revealed after time t.

the problem of α-polyhedral convex function chasing (αCFC), e.g. Chen et al. (2018); Zhang et al. (2021), in which each hitting cost function f_t is restricted to be globally α-polyhedral, meaning intuitively that it has a unique minimizer, away from which it grows with slope at least $\alpha > 0$.

Definition 2 *Let $(X, \|\cdot\|)$ be a normed vector space, and let $\alpha > 0$. A function $f : X \to \mathbb{R}_+$ is **globally α-polyhedral** if it has unique minimizer $\boldsymbol{x}^* \in X$, and in addition,*

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) + \alpha\|\boldsymbol{x} - \boldsymbol{x}^*\| \qquad \text{for all } \boldsymbol{x} \in X.$$

2.2. Using Black-Box Advice: Robustness, Consistency, and Bicompetitive Analysis

In this work, we seek algorithms for CFC and its subclasses that can exploit the good performance of a black-box advice algorithm, such as a reinforcement learning model, while maintaining rigorous worst-case performance guarantees. More specifically, we strive for algorithms that can obtain cost not much worse than optimal when the black-box advice is perfect, yet which have uniformly bounded competitive ratio when the advice is arbitrarily bad or even adversarial. This dual objective is naturally formulated in terms of *robustness* and *consistency*, which were introduced by Lykouris and Vassilvtiskii (2018) and are defined as follows.

Definition 3 *Let* ALG *be an online algorithm for* CFC*, and let* ADV *be a black-box advice algorithm.* ALG *is said to be c-**consistent** if it is c-competitive with respect to* ADV*. On the other hand,* ALG *is defined to be r-**robust** if it is r-competitive, independent of the performance of* ADV.

Our precise goal is to design algorithms achieving $(1+\epsilon)$-consistency and $R(\epsilon)$-robustness for CFC and its subclasses, where $\epsilon > 0$ is a hyperparameter chosen by the decision-maker that encodes confidence in the advice. The dependence of the robustness $R(\epsilon)$ on ϵ anticipates a trade-off between exploiting advice and worst-case robustness. We ideally seek algorithms with robustness $R(\epsilon)$ as small as possible, so that the trade-off between consistency and robustness is tight.

Our methodology for designing robust and consistent algorithms is very general, in the sense that we do not restrict to any special cases of CFC and do not consider in our analysis the explicit behavior of the advice or of any specific algorithm for CFC. This is in contrast to the work of Rutten et al. (2022), whose main robustness and consistency guarantees depend crucially upon the α-polyhedral setting. Rather, we approach the task of designing robust and consistent algorithms via a more general problem of designing *bicompetitive meta-algorithms* for CFC, which, informally, are "recipes" for combining two CFC algorithms to produce a single algorithm with competitive guarantees with respect to both input algorithms. More formally, we give the following definitions.

Definition 4 *An online algorithm* ALG *for* CFC *is (c, r)-**bicompetitive** with respect to a pair of algorithms $(\text{ALG}^{(1)}, \text{ALG}^{(2)})$ if* ALG *is simultaneously c-competitive with respect to $\text{ALG}^{(1)}$ and r-competitive with respect to $\text{ALG}^{(2)}$. Equivalently, the cost of* ALG *can be bounded as*

$$\mathrm{C}_{\text{ALG}} \leq \min\left\{c \cdot \mathrm{C}_{\text{ALG}^{(1)}}, r \cdot \mathrm{C}_{\text{ALG}^{(2)}}\right\}.$$

Definition 5 *A **meta-algorithm*** META *for* CFC *is a mapping* META $: \mathcal{A}_{\mathsf{CFC}} \times \mathcal{A}_{\mathsf{CFC}} \to \mathcal{A}_{\mathsf{CFC}}$. *That is,* META *takes as input two online algorithms for* CFC *and returns a single online algorithm for the problem.* META *is said to be (c, r)-**bicompetitive** if its output is always (c, r)-bicompetitive with respect to its inputs.*

It follows immediately from the previous two definitions that if META is (c, r)-bicompetitive, ADV is the advice, and ROB is a b-competitive algorithm for (a subclass of) CFC, then META(ADV, ROB) is c-consistent and rb-robust. We discuss this observation in more detail in Appendix A.2. Thus bicompetitive meta-algorithms give a general approach for designing robust and consistent algorithms for CFC and its subclasses.

The idea of approaching robust and consistent algorithm design via the design of bicompetitive meta-algorithms has been considered to some extent in the literature on other online problems, e.g. in the work of Antoniadis et al. (2020) on combining algorithms for MTS. To our knowledge, however, our specific terminology has not seen wide use in the literature.

3. Warmup: Switching Algorithms and Their Fundamental Limits

A natural first approach for designing bicompetitive meta-algorithms for CFC is to consider the class of switching algorithms, whose decisions switch between two other algorithms:

Definition 6 *A meta-algorithm* META *is a* ***switching meta-algorithm*** *if, at each time t, the decision* $\text{META}_t(\text{ALG}^{(1)}, \text{ALG}^{(2)})$ *made by* META *resides in the set* $\{\text{ALG}_t^{(1)}, \text{ALG}_t^{(2)}\}$.

Switching algorithms have garnered significant attention in the literature on robustness and consistency in recent years, e.g., Antoniadis et al. (2020); Lee et al. (2021); Angelopoulos (2021); Rutten et al. (2022). In particular, the only robust and consistent algorithms for CFC or subclasses thereof in general dimension are the switching algorithms of Antoniadis et al. (2020) for MTS and Rutten et al. (2022) for αCFC. In the following proposition, we refine these prior results, showing the existence of a switching meta-algorithm for general CFC with tunable bicompetitive bound.

Proposition 7 *Suppose* ADV, ROB *are algorithms for* CFC *and* $C_{\text{ROB}} \geq 1$. *There is a switching meta-algorithm* SWITCH *(Appendix B, Algorithm 4) that is* $\left(3 + \mathcal{O}(\epsilon), 5 + \mathcal{O}(\frac{1}{\epsilon^2})\right)$*-bicompetitive with respect to the inputs* (ADV, ROB)*, where* $\epsilon > 0$ *is an algorithm hyperparameter.*

Our proof of Proposition 7 follows closely that of (Antoniadis et al., 2020, Theorem 1), extending it via the recent result of (Angelopoulos, 2021, Theorem 5) on linear search with a "hint"; we present a proof in Appendix B.2.

If ADV is an advice algorithm and ROB is a C-competitive algorithm for (a subclass of) CFC, then SWITCH yields $(3 + \mathcal{O}(\epsilon))$-consistency and $(5 + \mathcal{O}(\frac{1}{\epsilon^2}))C$-robustness. Notably, this does not appear to allow for *arbitrary* consistency: specifically, SWITCH cannot attain consistency less than 3 while maintaining finite robustness. This limitation is unsurprising, since an identical lower bound holds on the algorithm for linear search on which SWITCH is based (Angelopoulos, 2021, Theorem 7), which can be extended to a lower bound on the bicompetitiveness of switching meta-algorithms. It is natural, then, to ask whether this lower bound also applies to the robustness and consistency of CFC algorithms. That is, can we devise a switching algorithm that, so long as it is provided with some non-adversarial, competitive algorithm ROB, beats 3-consistency while staying robust?

In the following theorem, which we prove in Appendix B.3, we show that robustness and consistency also face this fundamental limit: any algorithm that switches between black-box advice and an advice-agnostic competitive algorithm cannot beat 3-consistency while preserving robustness. We prove the theorem in the ℓ^2 setting, though the result extends to other norms such as the ℓ^∞ norm.

Algorithm 1: $\textsc{NestedSwitch}(\textsc{Adv}, \textsc{Rob}; \epsilon, r)$

Input: Algorithms $\textsc{Adv}, \textsc{Rob} \in \mathcal{A}_{\mathsf{NCBC}}$; hyperparameters $\epsilon, r > 0$

Output: Decisions $\boldsymbol{x}_1 \in K_1, \ldots, \boldsymbol{x}_T \in K_T$ chosen online

for $t = 1, 2, \ldots, T$ **do**

  Observe K_t, $\tilde{\boldsymbol{x}}_t := \textsc{Adv}_t$, and $\boldsymbol{s}_t := \textsc{Rob}_t$

  if $\epsilon \cdot \mathrm{C}_{\textsc{Adv}}(1, t) \geq r(d+2)$ **then**

    $\boldsymbol{x}_t \leftarrow \boldsymbol{s}_t$

  else

    $\boldsymbol{x}_t \leftarrow \tilde{\boldsymbol{x}}_t$

end

Theorem 8 *Consider the ℓ^2 norm setting. Let* ADV *be an advice algorithm, and let* ROB *be any (deterministic) competitive algorithm for* CBC *that is advice-agnostic. Let* ALG *be an online algorithm that switches between* ADV *and* ROB*. If* ALG *is c-consistent with $c < 3$, then* ALG *cannot have finite robustness.*

This lower bound implies that to obtain finite robustness alongside consistency $c < 3$ for general CFC, one must venture beyond the realm of switching algorithms. This is exactly the focus of Section 4, where we approach this task by exploiting the convexity of CFC. First, though, we ask: are there any special cases of CFC in which switching algorithms *can* obtain $(1+\epsilon)$-consistency and finite robustness for any $\epsilon > 0$? The answer is affirmative for αCFC (Rutten et al. (2022)), and as we show in the next proposition, such an algorithm also exists for NCBC. Specifically, we propose an algorithm, NESTEDSWITCH (Algorithm 1), which can achieve a $(1+\epsilon)$-consistent, $\mathcal{O}(\frac{d}{\epsilon})$-robust trade-off for NCBC by using a simple threshold-based rule for switching between the advice and the Steiner point algorithm of Bubeck et al. (2019). We prove Proposition 9 in Appendix B.4.

Proposition 9 *Consider the problem of* NCBC *on $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$, where the initial body K_1 resides in some ball $B(\boldsymbol{y}, r)$ of radius r containing $\boldsymbol{x}_0$, and $\mathrm{C}_{\mathrm{OPT}} \geq 1$. If* ROB *is the Steiner point algorithm (Bubeck et al. (2019)) that chooses the Steiner point of K_t at each time t, then* NESTEDSWITCH *(Algorithm 1) is $(1+\epsilon)$-consistent and $\left(1 + \frac{1}{\epsilon}\right) r(d+2)$-robust, where $\epsilon > 0$ is a hyperparameter.*

4. Beyond Switching Algorithms: Exploiting Convexity to Break 3-Consistency

In this section, we present our main results: two novel meta-algorithms that transcend the limitations of switching algorithms by exploiting the convexity of CFC. The key insight that enables this improved performance is that *hedging* between ADV and ROB, i.e., choosing a decision that is a convex combination of the two, allows for more nuanced algorithm behavior than switching permits.

4.1. INTERP: a $(\sqrt{2}+\epsilon, \mathcal{O}(\epsilon^{-2}))$-Bicompetitive Meta-Algorithm

We preface our first algorithm with some definitions from the geometry of real normed vector spaces $\mathcal{X} = (X, \|\cdot\|)$ which we employ in the algorithm's statement and performance bound. We present abridged introductions of these notions here, giving more detail in Appendix C.

We begin by introducing the *rectangular constant* $\mu(\mathcal{X})$ of a normed vector space $\mathcal{X}$, which is bounded between $\sqrt{2}$ and 3, with $\mu(\mathcal{X}) = \sqrt{2}$ when $\mathcal{X}$ is Hilbert and $\mu(\ell^p) < 3$ for any $p \in (1, \infty)$ (Joly (1969); Baronti et al. (2021)). Next, we define the *radial retraction*.

Definition 10 (Rieffel (2006)) *On a normed vector space $\mathcal{X} = (X, \|\cdot\|)$, the* ***radial retraction*** *$\rho(\cdot\,; r) : X \to B(\mathbf{0}, r)$ is the metric projection onto the closed ball of radius $r \geq 0$:*

$$\rho(\boldsymbol{x}; r) = \begin{cases} \boldsymbol{x} & \text{if } \|\boldsymbol{x}\| \leq r \\ r\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} & \text{if } \|\boldsymbol{x}\| > r. \end{cases}$$

On a fixed normed space $\mathcal{X}$, the collection of radial retractions $\rho(\cdot\,; r)$ with $r > 0$ share a Lipschitz constant, which we call $k(\mathcal{X})$. It is known that $1 \leq k(\mathcal{X}) \leq 2$ (Thele (1974)), and moreover $k(\mathcal{X}) \leq \mu(\mathcal{X})$ (Appendix C, Proposition 22).

With these definitions at our disposal, we now proceed to the main result of this section. We propose Algorithm 2, a meta-algorithm INTERP that takes as input two algorithms ADV, ROB for (a subclass of) CFC, and hyperparameters $\epsilon, \gamma, \delta > 0$ satisfying $2\gamma + 2\delta = \epsilon$. INTERP works as follows: at each time t, if the cost of ROB so far is a substantial fraction of the cost of ADV, then INTERP can move to ADV_t while staying competitive with respect to ROB, and it does so (line 4). Otherwise, INTERP moves to a point $\boldsymbol{x}_t$ determined by the series of radial projections (lines 6, 7, and 8), which intuitively guide INTERP to take a "greedy step" toward the decision made by ROB while still remaining close enough to ADV so as to maintain a consistency guarantee.

Algorithm 2: $\text{INTERP}(\text{ADV}, \text{ROB}; \epsilon, \gamma, \delta)$

Input: Algorithms ADV, ROB; hyperparameters $\epsilon > 0$ and $\gamma > 0, \delta > 0$ satisfying $2\gamma + 2\delta = \epsilon$
Output: Decisions $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ chosen online

```
1 for t = 1, 2, ..., T do
2     Observe f_t, x̃_t := ADV_t, and s_t := ROB_t
3     if C_ROB(1, t) ≥ δ · C_ADV(1, t) then
4         x_t ← x̃_t
5     else
6         y_t ← s_{t-1} + ρ(x̃_t − s_{t-1}; ‖x_{t-1} − s_{t-1}‖)
7         z_t ← s_{t-1} + ρ(y_t − s_{t-1}; max{‖y_t − s_{t-1}‖ − γ · C_ADV(t, t), 0})
8         x_t ← s_t + ρ(x̃_t − s_t; ‖z_t − s_{t-1}‖)
9 end
```

We characterize the bicompetitive performance of INTERP in the following theorem, which holds in any normed vector space $\mathcal{X}$ of arbitrary dimension.

Theorem 11 INTERP *(Algorithm 2) is*

$$\left(\mu(\mathcal{X}) + \epsilon, 1 + \frac{k(\mathcal{X})}{\gamma} + \frac{\mu(\mathcal{X}) + \epsilon + 1 + \frac{k(\mathcal{X})}{\gamma}}{\delta}\right)\text{–bicompetitive}$$

with respect to (ADV, ROB). *With γ, δ chosen optimally, the bound is* $(\mu(\mathcal{X}) + \epsilon, \mathcal{O}(\epsilon^{-2}))$.

In particular, if ADV *is advice and* ROB *is C-competitive for (a subclass of)* CFC, *then* INTERP *is $(\mu(\mathcal{X}) + \epsilon)$-consistent and $\mathcal{O}(C\epsilon^{-2})$-robust.*

Notably, INTERP (Algorithm 2) strictly improves on the 3-consistent lower bound for switching meta-algorithms in any ℓ^p space with $1 < p < \infty$, in which it holds that $\mu(\ell^p) < 3$. Moreover, it obtains consistency $(\sqrt{2} + \epsilon)$ in any Hilbert space. We prove Theorem 11 and give details regarding optimal selection of the parameters γ, δ in Appendix D. The proof employs two potential function arguments with different potential functions for the bounds with respect to ADV and ROB. Moreover, the generality of the theorem's setting requires the development of several geometric results characterizing the radial projection and its relation to the rectangular constant in arbitrary-dimensional normed vector spaces, which we present in Appendix D.1 prior to the main proof. These results are crucial for enabling INTERP's robustness and consistency in the general setting and elucidate the presence of the constants $\mu(\mathcal{X})$ and $k(\mathcal{X})$ in its bicompetitive bound.

We also detail robustness and consistency corollaries of Theorem 11 for multiple subclasses of CFC in Appendix F. In particular, Theorem 11 and Table 1 imply an algorithm for CFC and CBC on $\mathbb{R}^d$ with any norm that is $(\mu(\mathbb{R}^d, \|\cdot\|) + \epsilon)$-consistent and $\mathcal{O}(\frac{d}{\epsilon^2})$-robust; we also obtain an algorithm for αCFC that is $(\mu(\mathcal{X}) + \epsilon)$-consistent and $\mathcal{O}(\frac{1}{\alpha\epsilon^2})$-robust for αCFC on any normed vector space $\mathcal{X}$.

4.2. Attaining $(1 + \epsilon)$-Consistency in Bounded Instances with BDINTERP

In the preceding section, we proved that in the Hilbert space setting, INTERP (Algorithm 2) obtains consistency $(\sqrt{2} + \epsilon)$ while remaining competitive with respect to ROB. While this is a significant improvement on the limit of 3-consistency faced by switching algorithms, the question remains: can we devise an algorithm that achieves $(1 + \epsilon)$-consistency and $R(\epsilon) < \infty$ competitiveness with respect to ROB for *any* $\epsilon > 0$, in *any* normed vector space? In this section, we provide a simple sufficient condition under which this is possible: if there exists some constant $D \in \mathbb{R}_+$ for which $\|\text{ADV}_t - \text{ROB}_t\| \le D$ for all t, then there is a meta-algorithm that is $(1 + \epsilon, \mathcal{O}(\frac{D}{\epsilon}))$-bicompetitive with respect to (ADV, ROB). We call this condition *D-boundedness* of ADV and ROB; it arises naturally in a number of settings, for example in any CBC instance in which the diameter of each body K_t is bounded by D.

We present the algorithm achieving this bicompetitive bound, BDINTERP, in Algorithm 3. Just like INTERP, BDINTERP takes as input two algorithms ADV, ROB for (a subclass of) CFC, and hyperparameters $\epsilon, \gamma, \delta > 0$ satisfying $2\gamma + 2\delta = \epsilon$. At a high level, BDINTERP operates similarly to INTERP, though it takes smaller greedy steps toward ROB, enabling it to maintain $(1 + \epsilon)$-consistency. Specifically, BDINTERP works as follows: if the cost of ROB is a sufficient fraction of the cost of ADV, then BDINTERP moves to ADV_t (line 4). Otherwise, it selects an auxiliary point $\boldsymbol{y}_t$ as the point along the segment $[\boldsymbol{s}_t, \tilde{\boldsymbol{x}}_t]$ with the same relative position as $\boldsymbol{x}_{t-1}$ on the segment $[\boldsymbol{s}_{t-1}, \tilde{\boldsymbol{x}}_{t-1}]$ (line 7), and then chooses $\boldsymbol{x}_t$ by taking a greedy step toward $\boldsymbol{s}_t$ from $\boldsymbol{y}_t$ (line 8).

We present the performance result for BDINTERP in Theorem 12; like Theorem 11, the result holds in any normed vector space $\mathcal{X}$ of arbitrary dimension.

Theorem 12 *Suppose that* ADV *and* ROB *are D-bounded, i.e.,* $\|\text{ADV}_t - \text{ROB}_t\| \le D$ *for all* $t \in [T]$*; and assume that* $\text{C}_{\text{ROB}} \ge 1$. *Then* BDINTERP *(Algorithm 3) is*

$$\left(1 + \epsilon, D + \frac{D}{\gamma} + \frac{1 + \epsilon}{\delta}\right)\text{–}\textit{bicompetitive}$$

with respect to (ADV, ROB). *With* γ, δ *chosen optimally, the bound is* $(1 + \epsilon, \mathcal{O}(\frac{D}{\epsilon}))$.

Algorithm 3: BDINTERP(ADV, ROB; ϵ, γ, δ)

Input: Algorithms ADV, ROB; hyperparameters $\epsilon > 0$ and $\gamma > 0, \delta > 0$ satisfying $2\gamma + 2\delta = \epsilon$
Output: Decisions $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ chosen online

for $t = 1, 2, \ldots, T$ **do**
  Observe f_t, $\tilde{\boldsymbol{x}}_t := \text{ADV}_t$, and $\boldsymbol{s}_t := \text{ROB}_t$
  if $\mathrm{C}_{\text{ROB}}(1, t) \geq \delta \cdot \mathrm{C}_{\text{ADV}}(1, t)$ **then**
    $\boldsymbol{x}_t \leftarrow \tilde{\boldsymbol{x}}_t$
  else
    $\nu \leftarrow \frac{\|\boldsymbol{x}_{t-1} - \boldsymbol{s}_{t-1}\|}{\|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{s}_{t-1}\|}$ if $\tilde{\boldsymbol{x}}_{t-1} \neq \boldsymbol{s}_{t-1}$, otherwise $\nu \leftarrow 0$
    $\boldsymbol{y}_t \leftarrow \nu \tilde{\boldsymbol{x}}_t + (1 - \nu) \boldsymbol{s}_t$
    $\boldsymbol{x}_t \leftarrow \boldsymbol{s}_t + \rho\left(\boldsymbol{y}_t - \boldsymbol{s}_t; \max\{\|\boldsymbol{y}_t - \boldsymbol{s}_t\| - \gamma \cdot \mathrm{C}_{\text{ADV}}(t, t), 0\}\right)$
end

In particular, if ADV *is advice and* ROB *is C-competitive for (a subclass of)* CFC*, then* BDINTERP *is $(1 + \epsilon)$-consistent and $\mathcal{O}(\frac{CD}{\epsilon})$-robust.*

Remarkably, Theorem 12 states that BDINTERP not only improves on the consistency of INTERP, but it also strictly improves upon INTERP's $\mathcal{O}(\epsilon^{-2})$ competitiveness with respect to ROB when $D = o(\epsilon^{-1})$. Moreover, BDINTERP substantially improves on the randomized switching algorithm of Antoniadis et al. (2020) in the D-bounded setting, providing deterministic and tunable robustness and consistency guarantees with no additive factor in the consistency term. We give a proof of Theorem 12, as well as details on optimal parameter selection, in Appendix E. The argument follows a similar line of reasoning as that of our proof of Theorem 11, though in the proof of competitiveness with respect to ROB (i.e., robustness), we employ a novel potential function constructed via the ratio between the respective distances of $\boldsymbol{x}_t$ and $\tilde{\boldsymbol{x}}_t$ to $\boldsymbol{s}_t$.

We detail robustness and consistency corollaries of Theorem 12 for multiple subclasses of CFC in Appendix F. In particular, Theorem 12 and Table 1 imply an algorithm for CFC and CBC with any norm that is $(1 + \epsilon)$-consistent and $\mathcal{O}(\frac{dD}{\epsilon})$-robust on D-bounded instances. We also obtain an algorithm for αCFC in the D-bounded finite-dimensional Euclidean setting that achieves $(1 + \epsilon)$-consistency and $\mathcal{O}(\frac{D}{\alpha^{1/2}\epsilon})$-robustness. This latter bound is nearly tight for αCFC: Rutten et al. (2022, Theorem 3.6) prove a lower bound of $\mathcal{O}(\frac{1}{\alpha})$ robustness for any $(1 + \alpha)$-consistent αCFC algorithm. Choosing $\epsilon = \alpha$, Theorem 12 gives us $(1 + \alpha)$-consistency and $\mathcal{O}(\frac{1}{\alpha^{3/2}})$-robustness when $D = O(1)$ in α, leaving a gap of just $\mathcal{O}(\alpha^{-1/2})$ between the upper and lower bounds. We leave to future work the question of whether the upper and lower bounds can be made tight and whether the factor of D (and more generally the D-boundedness assumption) can be dropped.

5. Conclusion

In this work, we examine the question of integrating black-box advice into algorithms for convex function chasing using the notions of robustness and consistency from the literature on online algorithms with machine-learned advice. We first propose an algorithm that switches between the decisions of an arbitrary C-competitive algorithm ROB and the advice, showing that it obtains $(3 + \mathcal{O}(\epsilon))$-consistency and finite robustness for any $\epsilon > 0$. We moreover show that this is optimal, in the sense that *no* switching algorithm can improve upon 3-consistency while maintaining finite robustness. We then move beyond switching algorithms, and propose two algorithms, INTERP and BDINTERP, which obtain improved robustness and consistency guarantees by exploiting the convexity inherent in the CFC problem. In particular, INTERP obtains $(\sqrt{2}+\epsilon)$-consistency and $\mathcal{O}(\frac{C}{\epsilon^2})$-robustness, and under the additional assumption of D-boundedness, BDINTERP can obtain $(1+\epsilon)$-consistency and $\mathcal{O}(\frac{CD}{\epsilon})$-robustness. We show that BDINTERP is nearly optimal for the problem of CFC with α-polyhedral hitting cost functions f_t, so long as $D = O(1)$ in α.

Several interesting questions remain open for future work: in particular, (a) the question of whether $(1+\epsilon)$-consistency and finite robustness can be obtained for general CFC without the D-boundedness assumption, and (b) the question of tight lower bounds on robustness and consistency for CFC and its many subclasses. Specifically, for the case of CFC in general, we pose the question of whether $(1+\epsilon)$-consistency is possible together with $\mathcal{O}(\frac{d}{\epsilon})$-robustness, or even whether the dependence on ϵ and d can be further improved in the robustness bound.

Acknowledgments

The authors thank Eitan Levin for several helpful discussions. The authors acknowledge support from an NSF Graduate Research Fellowship (DGE-1745301), NSF grants CNS-2146814, CPS-2136197, CNS-2106403, and NGSDI-2105648, and Amazon AWS.

References

Susanne Albers and Jens Quedenfeld. Optimal Algorithms for Right-Sizing Data Centers. In *Proceedings of the 30th on Symposium on Parallelism in Algorithms and Architectures*, SPAA '18, pages 363–372, New York, NY, USA, July 2018. Association for Computing Machinery. ISBN 978-1-4503-5799-9. doi: 10.1145/3210377.3210385. URL https://doi.org/10.1145/3210377.3210385.

Spyros Angelopoulos. Online Search with a Hint. In James R. Lee, editor, *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, volume 185 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 51:1–51:16, Dagstuhl, Germany, 2021. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. ISBN 978-3-95977-177-1. doi: 10.4230/LIPIcs.ITCS.2021.51. URL https://drops.dagstuhl.de/opus/volltexte/2021/13590. ISSN: 1868-8969.

Spyros Angelopoulos, Christoph Dürr, Shendan Jin, Shahin Kamali, and Marc Renault. Online Computation with Untrusted Advice. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*, volume 151 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 52:1–52:15, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-134-4. doi: 10.4230/LIPIcs.ITCS.2020.52. URL https://drops.dagstuhl.de/opus/volltexte/2020/11737. ISSN: 1868-8969.

Antonios Antoniadis, Christian Coester, Marek Elias, Adam Polak, and Bertrand Simon. Online metric algorithms with untrusted predictions. *arXiv:2003.02144 [cs]*, June 2020. URL http://arxiv.org/abs/2003.02144. arXiv: 2003.02144.

C. J. Argue, Sébastien Bubeck, Michael B. Cohen, Anupam Gupta, and Yin Tat Lee. A Nearly-Linear Bound for Chasing Nested Convex Bodies. In *Proceedings of the 2019 Annual ACM-SIAM*

Symposium on Discrete Algorithms (SODA), Proceedings, pages 117–122. Society for Industrial and Applied Mathematics, January 2019. doi: 10.1137/1.9781611975482.8. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611975482.8.

C. J. Argue, Anupam Gupta, and Guru Guruganesh. Dimension-Free Bounds for Chasing Convex Functions. In *Proceedings of Thirty Third Conference on Learning Theory*, pages 219–241. PMLR, July 2020. URL https://proceedings.mlr.press/v125/argue20a.html. ISSN: 2640-3498.

C. J. Argue, Anupam Gupta, Ziye Tang, and Guru Guruganesh. Chasing convex bodies with linear competitive ratio. *Journal of the ACM*, 68(5):1–10, 2021. doi: https://doi.org/10.1145/3450349.

R.A. Baeza-Yates, J.C. Culberson, and G.J.E. Rawlins. Searching in the Plane. *Information and Computation*, 106(2):234–252, October 1993. ISSN 0890-5401. doi: 10.1006/inco.1993.1054. URL https://doi.org/10.1006/inco.1993.1054.

M. Baronti, E. Casini, and P. L. Papini. Revisiting the rectangular constant in Banach spaces. *Bulletin of the Australian Mathematical Society*, pages 1–10, April 2021. ISSN 0004-9727, 1755-1633. doi: 10.1017/S0004972721000253. URL https://www.cambridge.org/core/product/identifier/S0004972721000253/type/journal_article.

Marcin Bienkowski, Jaroslaw Byrka, Marek Chrobak, Christian Coester, Lukasz Jez, and Elias Koutsoupias. Better Bounds for Online Line Chasing. In Peter Rossmanith, Pinar Heggernes, and Joost-Pieter Katoen, editors, *44th International Symposium on Mathematical Foundations of Computer Science (MFCS 2019)*, volume 138 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 8:1–8:13, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-117-7. doi: 10.4230/LIPIcs.MFCS.2019.8. URL http://drops.dagstuhl.de/opus/volltexte/2019/10952. ISSN: 1868-8969.

Garrett Birkhoff. Orthogonality in linear metric spaces. *Duke Mathematical Journal*, 1(2):169–172, June 1935. ISSN 0012-7094, 1547-7398. doi: 10.1215/S0012-7094-35-00115-6. URL https://projecteuclid.org/journals/duke-mathematical-journal/volume-1/issue-2/Orthogonality-in-linear-metric-spaces/10.1215/S0012-7094-35-00115-6.full. Publisher: Duke University Press.

Avrim Blum and Carl Burch. On-line learning and the metrical task system problem. In *Proceedings of the tenth annual conference on Computational learning theory - COLT '97*, pages 45–53, Nashville, Tennessee, United States, 1997. ACM Press. ISBN 978-0-89791-891-6. doi: 10.1145/267460.267475. URL http://portal.acm.org/citation.cfm?doid=267460.267475.

Béla Bollobás. *Linear Analysis: An Introductory Course*. Cambridge Mathematical Textbooks. Cambridge University Press, Cambridge; New York, 2nd edition, 1999. ISBN 978-0-521-65577-4.

Allan Borodin, Nathan Linial, and Michael E. Saks. An optimal on-line algorithm for metrical task system. *Journal of the ACM*, 39(4):745–763, October 1992. ISSN 0004-5411. doi: 10.1145/146585.146588. URL https://doi.org/10.1145/146585.146588.

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, pages 1496–1508. Society for Industrial and Applied Mathematics, December 2019. doi: 10.1137/1.9781611975994.91. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611975994.91.

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*, pages 1574–1594. PMLR, July 2018. URL https://proceedings.mlr.press/v75/chen18b.html. ISSN: 2640-3498.

Jocelyn Desbiens. Constante rectangle et biais d'un espace de Banach. *Bulletin of the Australian Mathematical Society*, 42(3):465–482, December 1990. ISSN 0004-9727, 1755-1633. doi: 10.1017/S000497270002863X. URL https://www.cambridge.org/core/product/identifier/S000497270002863X/type/journal_article.

Amos Fiat, Yuval Rabani, and Yiftach Ravid. Competitive k-server algorithms. *Journal of Computer and System Sciences*, 48(3):410–428, June 1994. ISSN 0022-0000. doi: 10.1016/S0022-0000(05)80060-1. URL https://www.sciencedirect.com/science/article/pii/S0022000005800601.

Yoav Freund and Robert E Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997. ISSN 0022-0000. doi: 10.1006/jcss.1997.1504. URL https://www.sciencedirect.com/science/article/pii/S002200009791504X.

Joel Friedman and Nathan Linial. On convex body chasing. *Discrete & Computational Geometry*, 9(3):293–321, March 1993. ISSN 0179-5376, 1432-0444. doi: 10.1007/BF02189324. URL http://link.springer.com/10.1007/BF02189324.

Gautam Goel and Adam Wierman. An Online Algorithm for Smoothed Regression and LQR Control. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, pages 2504–2513. PMLR, April 2019. URL https://proceedings.mlr.press/v89/goel19a.html. ISSN: 2640-3498.

Noah Golowich and Ankur Moitra. Can Q-Learning be Improved with Advice? *arXiv:2110.13052 [cs, math, stat]*, October 2021. URL http://arxiv.org/abs/2110.13052. arXiv: 2110.13052.

Robert C. James. Orthogonality and linear functionals in normed linear spaces. *Transactions of the American Mathematical Society*, 61(2):265–292, 1947. ISSN 0002-9947, 1088-6850. doi: 10.1090/S0002-9947-1947-0021241-4. URL https://www.ams.org/tran/1947-061-02/S0002-9947-1947-0021241-4/.

Jean-Luc Joly. Caracterisations d'espaces hilbertiens au moyen de la constante rectangle. *Journal of Approximation Theory*, 2:301–311, 1969.

Seung-Jun Kim and Geogios B. Giannakis. Real-time electricity pricing for demand response using online convex optimization. In *ISGT 2014*, pages 1–5, February 2014. doi: 10.1109/ISGT.2014.6816447.

Elias Koutsoupias and Christos H. Papadimitriou. On the k-server conjecture. *Journal of the ACM*, 42(5):971–983, September 1995. ISSN 0004-5411. doi: 10.1145/210118.210128. URL https://doi.org/10.1145/210118.210128.

Russell Lee, Jessica Maghakian, Mohammad Hajiesmaili, Jian Li, Ramesh Sitaraman, and Zhenhua Liu. Online Peak-Aware Energy Scheduling with Untrusted Advice. In *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*, pages 107–123, Virtual Event Italy, June 2021. ACM. ISBN 978-1-4503-8333-2. doi: 10.1145/3447555.3464860. URL https://dl.acm.org/doi/10.1145/3447555.3464860.

Yingying Li, Guannan Qu, and Na Li. Online Optimization With Predictions and Switching Costs: Fast Algorithms and the Fundamental Limit. *IEEE Transactions on Automatic Control*, 66(10): 4761–4768, October 2021. ISSN 1558-2523. doi: 10.1109/TAC.2020.3040249. Conference Name: IEEE Transactions on Automatic Control.

Minghong Lin, Zhenhua Liu, Adam Wierman, and Lachlan L. H. Andrew. Online algorithms for geographical load balancing. In *2012 International Green Computing Conference (IGCC)*, pages 1–10, June 2012. doi: 10.1109/IGCC.2012.6322266.

Minghong Lin, Adam Wierman, Lachlan L. H. Andrew, and Eno Thereska. Dynamic Right-Sizing for Power-Proportional Data Centers. *IEEE/ACM Transactions on Networking*, 21(5):1378–1391, October 2013. ISSN 1558-2566. doi: 10.1109/TNET.2012.2226216. Conference Name: IEEE/ACM Transactions on Networking.

Yiheng Lin. Personal correspondence. 2022.

Thodoris Lykouris and Sergei Vassilvtiskii. Competitive Caching with Machine Learned Advice. In *Proceedings of the 35th International Conference on Machine Learning*, pages 3296–3305. PMLR, July 2018. URL https://proceedings.mlr.press/v80/lykouris18a.html. ISSN: 2640-3498.

Mohammad Mahdian, Hamid Nazerzadeh, and Amin Saberi. Online Optimization with Uncertain Information. *ACM Transactions on Algorithms*, 8(1):1–29, January 2012. ISSN 1549-6325, 1549-6333. doi: 10.1145/2071379.2071381. URL https://dl.acm.org/doi/10.1145/2071379.2071381.

Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving Online Algorithms via ML Predictions. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/hash/73a427badebe0e32caa2e1fc7530b7f3-Abstract.html.

Marc A. Rieffel. Lipschitz extension constants equal projection constants. *Contemporary Mathematics*, 414:147–162, 2006. URL http://arxiv.org/abs/math/0508097. arXiv: math/0508097.

Daan Rutten, Nico Christianson, Debankur Mukherjee, and Adam Wierman. Online Optimization with Untrusted Predictions. *arXiv:2202.03519 [cs]*, February 2022. URL http://arxiv.org/abs/2202.03519. arXiv: 2202.03519.

Problem	State-of-the-Art Competitive Ratio	Setting
CFC	$d+1$	$\mathbb{R}^d$ with any norm
CBC	d	$\mathbb{R}^d$ with any norm
kCBC	$2k+1$	$\mathbb{R}^d$ with ℓ^2 norm
αCFC	$\max\left\{1, \frac{2}{\alpha}\right\}$	Any normed vector space
αCFC	$\mathcal{O}\left(\frac{1}{\alpha^{1/2}}\right)$	$\mathbb{R}^d$ with ℓ^2 norm
(κ,γ)CFC	$(2+2\sqrt{2})2^{\gamma/2}\kappa$	$\mathbb{R}^d$ with ℓ^2 norm

Table 2: Competitive ratios for state-of-the-art algorithms on various subclasses of CFC.

Mark Sellke. Chasing convex bodies optimally. In *Proceedings of the Thirty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '20, pages 1509–1518, USA, January 2020. Society for Industrial and Applied Mathematics.

R. L. Thele. Some Results on the Radial Projection in Banach Spaces. *Proceedings of the American Mathematical Society*, 42(2):483–486, 1974. ISSN 0002-9939. doi: 10.2307/2039529. URL https://www.jstor.org/stable/2039529. Publisher: American Mathematical Society.

Alexander Wei and Fred Zhang. Optimal robustness-consistency trade-offs for learning-augmented online algorithms. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 8042–8053. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/5bd844f11fa520d54fa5edec06ea2507-Paper.pdf.

Lijun Zhang, Wei Jiang, Shiyin Lu, and Tianbao Yang. Revisiting Smoothed Online Learning. *arXiv:2102.06933 [cs, math, stat]*, May 2021. URL http://arxiv.org/abs/2102.06933. arXiv: 2102.06933.

Appendix A. Preliminaries

In part A.1 of this appendix, we provide more detailed definitions of the subclasses of convex function chasing considered in this work, including both those introduced in Section 2.1 as well as several additional special cases which we refer to in the robustness and consistency results given in Appendix F. We also review state-of-the-art competitive algorithms for each subclass, which we summarize in Table 2, which is an extended version of Table 1 in the main text. Then, in part A.2, we elaborate on the claim made in Section 2.2 that a (c, r)-bicompetitive meta-algorithm for CFC, along with a b-competitive algorithm for a subclass of CFC, together yield a c-consistent and rb-robust algorithm for that subclass.

A.1. Subclasses of CFC

A.1.1. CONVEX BODY CHASING

In the problem of convex body chasing (CBC) on a normed vector space $(X, \|\cdot\|)$, at each time t a decision-maker is given a convex body $K_t \subseteq X$ and faces the requirement that their decision $\boldsymbol{x}_t$ must reside within K_t. A further special case of CBC is the problem of *nested* convex body chasing (NCBC), in which subsequent bodies are nested, i.e. $K_t \supseteq K_{t+1}$ for each t. We define the set of all online algorithms which are *feasible* for CBC, i.e. which produce decisions residing within the convex body K_t at each time, as $\mathcal{A}_{\mathsf{CBC}}$. We define $\mathcal{A}_{\mathsf{NCBC}}$ similarly as the set of all online algorithms which are feasible for NCBC. Sellke (2020) proved that an algorithm based on a functional generalization of the Steiner point of a convex body achieves competitive ratio d for CBC and $d+1$ for general CFC in $\mathbb{R}^d$ equipped with any norm.

The problem of convex body chasing can easily be seen as a special case of CFC in which each hitting cost f_t is the $\{0, \infty\}$ indicator of the convex set K_t. That is,

$$f_t(\boldsymbol{x}) = \begin{cases} 0 & \text{if } \boldsymbol{x} \in K_t \\ \infty & \text{otherwise.} \end{cases}$$

As noted in Sellke (2020), we need not even require hitting costs to take infinite values to recover convex body chasing from function chasing. Indeed, restricting to the finite-dimensional setting,[2] consider f_t defined as

$$f_t(\boldsymbol{x}) = 3 \cdot d(\boldsymbol{x}, K_t) = 3 \min_{\boldsymbol{y} \in K_t} \|\boldsymbol{x} - \boldsymbol{y}\|.$$

Then any algorithm $\textsc{Alg} \in \mathcal{A}_{\mathsf{CFC}}$ yields a set of decisions $\textsc{Alg}_1, \dots, \textsc{Alg}_T$ on the instance $(\boldsymbol{x}_0, f_1, \dots, f_T)$; and moreover, $\textsc{Alg}$ can be transformed into an algorithm $\textsc{Alg}' \in \mathcal{A}_{\mathsf{CFC}}$ with strictly improved cost, and which in particular incurs no hitting cost, by setting

$$\textsc{Alg}'_t = \begin{cases} \textsc{Alg}_t & \text{if } \textsc{Alg}_t \in K_t \\ \Pi_{K_t} \textsc{Alg}_t & \text{otherwise.} \end{cases}$$

Clearly each decision $\textsc{Alg}'_t$ resides in the convex body K_t; thus $\textsc{Alg}'$ is a feasible online algorithm for CBC, i.e., $\textsc{Alg}' \in \mathcal{A}_{\mathsf{CBC}}$. Moreover, the cost of $\textsc{Alg}'$ on the CBC instance is identical to its cost for the corresponding CFC instance, since it incurs no hitting cost. It follows that the competitive ratio of $\textsc{Alg}'$ for the CBC problem is at most the competitive ratio of $\textsc{Alg}$ as a CFC algorithm, since $\textsc{Opt}$ for a CBC instance and its corresponding CFC instance always coincide. In short, a C-competitive algorithm for CFC is also C-competitive for CBC.

Remark 13 *The preceding line of reasoning can be extended to show that a C-competitive algorithm for* CFC *gives a C-competitive algorithm for* CFC *with strict decision constraints, i.e., where at time t, the decision $\boldsymbol{x}_t$ must both reside in some convex body K_t and also incurs a convex hitting cost $f_t(\boldsymbol{x}_t)$.*

2. This restriction is natural because no algorithm can be competitive for CBC in the infinite-dimensional setting. We impose the restriction in order to ensure existence of a metric projection onto K_t.

A.1.2. CHASING LOW-DIMENSIONAL CONVEX BODIES

A special case of convex body chasing that has received significant attention is the problem of chasing low-dimensional bodies in higher-dimensional space: indeed, the seminal work of Friedman and Linial (1993) began by addressing the problem of chasing lines in the Euclidean plane. Bienkowski et al. (2019) later presented a 3-competitive algorithm for chasing lines in $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$, and most recently Argue et al. (2020) gave an algorithm that is $(2k+1)$-competitive for chasing convex bodies lying in k-dimensional affine subspaces, regardless of the dimension d of the underlying Euclidean space. Motivated by this last result, we define the problem of k-dimensional convex body chasing (kCBC), comprised of all instances of CBC in which each body K_t lies within an affine subspace of dimension at most k – i.e., $\dim \operatorname{aff} K_t \leq k$ for all t.

A.1.3. α-POLYHEDRAL CONVEX FUNCTION CHASING

A class of functions that has been studied extensively in the literature on online optimization with switching costs (SOCO) is the class of globally α-polyhedral functions, e.g., Chen et al. (2018); Zhang et al. (2021), which are defined as follows.

Definition 14 *Let $(X, \|\cdot\|)$ be a normed vector space, and let $\alpha > 0$. A function $f : X \to \mathbb{R}_+$ is **globally α-polyhedral** if it has unique minimizer $\boldsymbol{x}^* \in X$, and in addition,*

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) + \alpha\|\boldsymbol{x} - \boldsymbol{x}^*\| \qquad \text{for all } \boldsymbol{x} \in X.$$

Roughly speaking, a globally α-polyhedral function has a unique minimizer, away from which it grows with slope at least α. For a fixed $\alpha > 0$, we define the problem of *α-polyhedral convex function chasing* (αCFC) comprised of all those problem instances of CFC in which, in addition to being convex, each function f_t is also globally α-polyhedral. αCFC has been widely studied due to its admitting algorithms with "dimension-free" competitive ratios: Zhang et al. (2021) showed that a greedy algorithm that simply moves to the minimizer $\boldsymbol{x}_t^*$ of each function f_t is $\max\{1, \frac{2}{\alpha}\}$-competitive for αCFC in any normed vector space. In the setting of $\mathbb{R}^d$ with the ℓ^2 norm, Lin (2022) gave an algorithm augmenting the Online Balanced Descent algorithm of Chen et al. (2018) to achieve a competitive ratio of $\mathcal{O}(\frac{1}{\alpha^{1/2}})$.

A.1.4. (κ, γ)-WELL-CENTERED CONVEX FUNCTION CHASING

Another class of functions that has received attention in the design of algorithms for subclasses of CFC with dimension-free competitive ratios is the set of (κ, γ)-well-centered functions, introduced by Argue et al. (2020):

Definition 15 *Let $(X, \|\cdot\|)$ be a normed vector space, and let $\kappa, \gamma \geq 1$. A function $f : \mathbb{R}^d \to \mathbb{R}_+$ with minimizer $\boldsymbol{x}^*$ is (κ, γ)**-well-centered** if there exists some $a > 0$ such that*

$$\frac{a}{2}\|\boldsymbol{x} - \boldsymbol{x}^*\|^\gamma \leq f(\boldsymbol{x}) \leq \frac{a\kappa}{2}\|\boldsymbol{x} - \boldsymbol{x}^*\|^\gamma \qquad \text{for all } \boldsymbol{x} \in X.$$

Intuitively, the growth rate of a (κ, γ)-well-centered function away from its minimizer (as measured with the "distance" $\|\cdot\|^\gamma$) is bounded above and below, and the ratio of these bounds is at most κ. For fixed $\kappa, \gamma \geq 1$, we define the problem of *(κ, γ)-well-centered convex function chasing* ((κ, γ)CFC) comprised of all those problem instances of CFC in which each f_t is (κ, γ)-well-centered. Argue et al. (2020) showed that the "Move towards Minimizer" algorithm is $(2 + 2\sqrt{2})2^{\gamma/2}\kappa$-competitive for (κ, γ)CFC on $\mathbb{R}^d$ equipped with the ℓ^2 norm.

A.2. Bicompetitive meta-algorithms give robust and consistent algorithms

In this section, we briefly justify the claim that if META is a (c, r)-bicompetitive meta-algorithm for CFC, ADV is an advice algorithm, and ROB is a b-competitive online algorithm for a subclass of CFC, then $\text{META}(\text{ADV}, \text{ROB})$ is c-consistent and rb-robust for that subclass. This is straightforward to see for non-CBC subclasses, or more generally, for any subclass of CFC which does not involve hard constraints on the decisions $\boldsymbol{x}_t$. In particular, ROB being b-competitive means that $\mathrm{C}_{\text{ROB}} \leq b \cdot \mathrm{C}_{\text{OPT}}$, and so (c, r)-bicompetitiveness of META implies that both $\mathrm{C}_{\text{META}(\text{ADV},\text{ROB})} \leq c \cdot \mathrm{C}_{\text{ADV}}$ and $\mathrm{C}_{\text{META}(\text{ADV},\text{ROB})} \leq r \cdot \mathrm{C}_{\text{ROB}} \leq rb \cdot \mathrm{C}_{\text{OPT}}$, as desired.

The only subclasses that require more careful justification are those, such as CBC, with hard constraints on the decisions. However, so long as the advice always gives feasible decisions – e.g., in the CBC case, $\text{ADV} \in \mathcal{A}_{\mathsf{CBC}}$, so $\text{ADV}_t \in K_t$ for each t – then we can obtain the same result by applying the reasoning from Appendix A.1.1 on equivalent CFC reformulations of instances with hard constraints. That is, on any instance of the subclass, we must simply run $\text{META}(\text{ADV}, \text{ROB})$ on its equivalent reformulation as a CFC instance, and we thereby obtain the same guarantees of c-consistency and rb-robustness for the subclass.

Appendix B. Switching algorithms

B.1. The meta-algorithm SWITCH

We give the meta-algorithm SWITCH, which takes as hyperparameters $b > 1$ and $\delta \in (0, 1]$, in Algorithm 4. In order to reduce the two hyperparameters b, δ to a single hyperparameter ϵ as in the statement of Proposition 7, we simply introduce an auxiliary variable γ and make the substitutions $\delta \leftarrow b\gamma^2 - b^{-1}$, $b \leftarrow \sqrt{\gamma^{-2} + 1}$, and $\gamma \leftarrow \sqrt{\frac{\epsilon}{4}}$.

Algorithm 4: $\text{SWITCH}(\text{ADV}, \text{ROB}; b, \delta)$

Input: Algorithms $\text{ADV}, \text{ROB} \in \mathcal{A}_{\mathsf{CFC}}$; hyperparameters $b > 1$, $\delta \in (0, 1]$
Output: Decisions $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ chosen online

```
i ← 0
while problem instance has not ended do
    if i ≡ 0 mod 2 then
        x_t ← ADV_t until the last time t that C_ADV(1, t) ≤ b^i
        i ← i + 1
    else
        x_t ← ROB_t until the last time t that C_ROB(1, t) ≤ δb^i
        i ← i + 1
end
```

B.2. Proof of Proposition 7

The proof follows the argument of (Angelopoulos, 2021, Theorem 5) and uses a similar line of reasoning as (Antoniadis et al., 2020, Theorems 1, 18) in applying an algorithm for linear search to CFC.

Each value of i encountered in the execution of Algorithm 4 is taken to refer to a *phase* of the algorithm; every decision $\boldsymbol{x}_t$ made during a particular value of i is said to take place during the i^{th} phase. Our strategy will be to bound the cost that the algorithm incurs in each phase i, including the cost it takes to switch from the last decision of the previous phase $i-1$.

As a base case, consider $i = 0$. The total cost incurred by the algorithm during this phase is bounded by $b^i = b^0 = 1$, by line 4 of the algorithm.

Now consider phase $i > 0$, and assume that i is odd; after proving the cost bound for phase i in the odd case, we will state the corresponding bound for the even case, which follows a nearly identical argument. Let $\underline{t}$ be the last timestep in the $(i-1)^{\text{th}}$ phase – that is, $\underline{t}$ is defined such that $\mathrm{C}_{\mathrm{ADV}}(1, \underline{t}+1) > b^{i-1}$. We will assume that $\mathrm{C}_{\mathrm{ADV}}(1, \underline{t}) \leq b^{i-1}$, i.e., the algorithm makes at least one decision during phase $(i-1)$, selecting $\boldsymbol{x}_{\underline{t}} = \mathrm{ADV}_{\underline{t}}$; but the upper bound we obtain will also apply to the case where the $(i-1)^{\text{th}}$ phase is vacuous. Let $\bar{t}$ be the last timestep corresponding to phase i, i.e., $\bar{t} \geq \underline{t}$ is defined such that $\mathrm{C}_{\mathrm{ROB}}(1, \bar{t}+1) > \delta b^i$. If $\bar{t} = \underline{t}$, then clearly no decisions are made during phase i, so no cost is incurred during this phase. On the other hand, if $\bar{t} > \underline{t}$, then certainly $\mathrm{C}_{\mathrm{ROB}}(1, \bar{t}) \leq \delta b^i$, so the cost incurred by SWITCH during phase i, starting from its position at time $\underline{t}$, can be bounded as

$$\begin{aligned}
\mathrm{C}_{\mathrm{SWITCH}}(\underline{t}+1, \bar{t}) &= \sum_{t=\underline{t}+1}^{\bar{t}} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| \\
&= f_{\underline{t}+1}(\boldsymbol{x}_{\underline{t}+1}) + \|\boldsymbol{x}_{\underline{t}+1} - \boldsymbol{x}_{\underline{t}}\| + \sum_{t=\underline{t}+2}^{\bar{t}} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| \\
&\leq f_{\underline{t}+1}(\boldsymbol{x}_{\underline{t}+1}) + \|\boldsymbol{x}_{\underline{t}+1} - \boldsymbol{x}_0\| + \|\boldsymbol{x}_{\underline{t}} - \boldsymbol{x}_0\| + \sum_{t=\underline{t}+2}^{\bar{t}} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| \\
&= f_{\underline{t}+1}(\mathrm{ROB}_{\underline{t}+1}) + \|\mathrm{ROB}_{\underline{t}+1} - \boldsymbol{x}_0\| + \|\mathrm{ADV}_{\underline{t}} - \boldsymbol{x}_0\| \\
&\quad + \sum_{t=\underline{t}+2}^{\bar{t}} f_t(\mathrm{ROB}_t) + \|\mathrm{ROB}_t - \mathrm{ROB}_{t-1}\| \\
&\leq \mathrm{C}_{\mathrm{ROB}}(1, \bar{t}) + \mathrm{C}_{\mathrm{ADV}}(1, \underline{t}) \\
&\leq \delta b^i + b^{i-1}.
\end{aligned}$$

where the first two bounds use the triangle inequality, and the last bound follows by construction of $\bar{t}$ and $\underline{t}$. By a very similar argument, if i is even, we can bound the cost incurred by SWITCH during phase i as $\delta b^{i-1} + b^i$. Then the total cost expenditure of SWITCH through the end of some phase $N > 0$ is at most

$$\begin{aligned}
& b^0 + \sum_{i=0}^{\lfloor \frac{N-1}{2} \rfloor} (\delta b^{2i+1} + b^{2i}) + \sum_{j=1}^{\lfloor \frac{N}{2} \rfloor} (\delta b^{2j-1} + b^{2j}) \\
= & \begin{cases} b^N + 2\sum_{i=0}^{\frac{N}{2}-1} (b^{2i} + \delta b^{2i+1}) & \text{if } N \text{ is even} \\ \delta b^N + 2b^{N-1} + 2\sum_{i=0}^{\frac{N-3}{2}} (b^{2i} + \delta b^{2i+1}) & \text{if } N \text{ is odd} \end{cases}
\end{aligned} \tag{1}$$

Suppose then that the instance ends at time T *during* phase N. We break into cases depending on the value of N.

First, if $N = 0$, then clearly $\mathrm{C}_{\text{SWITCH}} = \mathrm{C}_{\text{ADV}}$, so SWITCH is 1-competitive with respect to ADV. Moreover, since $\mathrm{C}_{\text{ADV}} \le b^0 = 1$, then by the assumption in the proposition statement that $\mathrm{C}_{\text{ROB}} \ge 1$, it follows that SWITCH is at most 1-competitive with respect to ROB.

Second, if $N = 1$, then $\mathrm{C}_{\text{SWITCH}} \le 2b^0 + \mathrm{C}_{\text{ROB}} = 2 + \mathrm{C}_{\text{ROB}}$. Since $\mathrm{C}_{\text{ADV}} > 1$ and $\mathrm{C}_{\text{ROB}} \le \delta b$ (due to the instance ending at phase $N = 1$), this means that SWITCH is at most $(2+\delta b)$-competitive with respect to ADV. Moreover, by assumption $\mathrm{C}_{\text{ROB}} \ge 1$, SWITCH is at most 3-competitive with respect to ROB.

Next, suppose $N > 1$ and N is even. Then we have

$$\mathrm{C}_{\text{SWITCH}} \le 2 \sum_{i=0}^{\frac{N}{2}-1} (b^{2i} + \delta b^{2i+1}) + \mathrm{C}_{\text{ADV}}$$

which follows by applying (1) to bound cost through phase $(N-1)$, and bounding the remaining cost by $\delta b^{N-1} + \mathrm{C}_{\text{ADV}}$, i.e., the cost to switch back to ADV and follow it until the instance ends. Then note that $\mathrm{C}_{\text{ADV}} \ge b^{N-2}$ by definition of phase; introducing the substitution $2k \coloneqq N-2$, we find that the competitive ratio of SWITCH with respect to ADV is bounded as

$$\begin{aligned}
\frac{\mathrm{C}_{\text{SWITCH}}}{\mathrm{C}_{\text{ADV}}} &\le 1 + 2\frac{\sum_{i=0}^{k}(b^{2i} + \delta b^{2i+1})}{b^{2k}} \\
&= 1 + 2\left(\frac{b^{2k+2}-1}{b^{2k}(b^2-1)} + \delta b \frac{b^{2k+2}-1}{b^{2k}(b^2-1)}\right) \\
&\le 1 + 2\left(\frac{b^2}{b^2-1} + \delta \frac{b^3}{b^2-1}\right).
\end{aligned}$$

On the other hand, we know that $\mathrm{C}_{\text{ADV}} \le b^N$ and $\mathrm{C}_{\text{ROB}} \ge \delta b^{N-1}$, so by similar reasoning the competitive ratio of SWITCH with respect to ROB is bounded as

$$\begin{aligned}
\frac{\mathrm{C}_{\text{SWITCH}}}{\mathrm{C}_{\text{ROB}}} &\le \frac{b}{\delta} + 2\frac{\sum_{i=0}^{k}(b^{2i} + \delta b^{2i+1})}{\delta b^{2k+1}} \\
&\le \frac{b}{\delta} + 2\left(\frac{b}{\delta(b^2-1)} + \frac{b^2}{b^2-1}\right).
\end{aligned}$$

Finally, consider $N > 1$ for odd N. Then

$$\mathrm{C}_{\text{SWITCH}} \le 2 \sum_{i=0}^{\frac{N-1}{2}} b^{2i} + \sum_{i=0}^{\frac{N-3}{2}} \delta b^{2i+1} + \mathrm{C}_{\text{ROB}}.$$

Noting that $\mathrm{C}_{\text{ROB}} \le \delta b^N$, $\mathrm{C}_{\text{ADV}} \ge b^{N-1}$, and making the substitution $2k = N-1$, we obtain that the competitive ratio of SWITCH with respect to ADV is bounded as

$$\begin{aligned}
\frac{\mathrm{C}_{\text{SWITCH}}}{\mathrm{C}_{\text{ADV}}} &\le \delta b + 2\frac{\sum_{i=0}^{k} b^{2i} + \sum_{i=0}^{k-1} \delta b^{2i+1}}{b^{2k}} \\
&\le \delta b + 2\left(\frac{b^2}{b^2-1} + \delta \frac{b}{b^2-1}\right).
\end{aligned}$$

On the other hand, we know that $\mathrm{C}_{\textsc{Rob}} \geq \delta b^{N-2} = \delta b^{2k-1}$. Thus the competitive ratio of SWITCH with respect to ROB is bounded as

$$\begin{aligned}\frac{\mathrm{C}_{\textsc{Switch}}}{\mathrm{C}_{\textsc{Rob}}} &\leq 1 + 2\frac{\sum_{i=0}^{k} b^{2i} + \sum_{i=0}^{k-1} \delta b^{2i+1}}{\delta b^{2k-1}} \\ &\leq 1 + 2\left(\frac{1}{\delta}\frac{b^3}{b^2-1} + \frac{b^2}{b^2-1}\right).\end{aligned}$$

Combining these various cases, we obtain that SWITCH is

$$\left(1 + 2\left(\frac{b^2}{b^2-1} + \delta\frac{b^3}{b^2-1}\right), 1 + 2\left(\frac{b^2}{b^2-1} + \frac{1}{\delta}\frac{b^3}{b^2-1}\right)\right)\text{–bicompetitive}$$

with respect to $(\textsc{Adv}, \textsc{Rob})$. Introducing an auxiliary parameter γ and making the substitutions $\delta \leftarrow b\gamma^2 - b^{-1}$, $b \leftarrow \sqrt{\gamma^{-2}+1}$, and $\gamma \leftarrow \sqrt{\frac{\epsilon}{4}}$, we arrive at the bicompetitive bound in terms of ϵ stated in the proposition. ■

B.3. Proof of Theorem 8

We consider the setting of $\mathbb{R}^d$ with the ℓ^2 norm, where the advice ADV is adversarial and ROB is an *arbitrary* b-competitive algorithm for CBC, with $b < \infty$; ALG is any algorithm that switches between ROB and ADV. For simplicity of presentation, we will assume that $\sqrt{d}$ is an integer. ROB is assumed to be advice-agnostic, i.e., the behavior of ADV does not impact the decisions made by ROB (nor does the behavior of ALG, since ALG itself depends on both ADV and ROB). We construct a lower bound in the spirit of the standard example of chasing faces of the hypercube. At a high level, the CBC instance we construct has two phases: the first is comprised of multiple subphases in which an affine subspace is chosen adversarially and is repeatedly served until ROB has "almost" stopped moving. This phase lasts either until $3\sqrt{d}$ subphases have concluded, or until the first time that ALG coincides with ADV at the end of a subphase, whichever happens sooner. If the former holds, i.e., if ALG ends each of the $3\sqrt{d}$ subphases at ROB, then the instance is done. Otherwise, the second phase begins: there are a few different cases, but generally, the same affine subspace is served repeatedly while ADV slowly drifts away from ROB until ALG switches back to ROB. Then, the final body is simply the last advice decision as a singleton, forcing ALG to move back to the advice, and the instance concludes.

We now describe the lower bound in more specific detail. Since ROB is advice-agnostic, we may begin by describing its behavior before specifying the behavior of ADV and reasoning about the switching algorithm ALG. We denote by $\boldsymbol{e}_j \in \mathbb{R}^d$, $j \in [d]$ the j^{th} standard unit basis vector, which is 1 in its j^{th} entry and 0 elsewhere. Choose any $\delta > 0$. The starting position is $\boldsymbol{x}_0 = \mathbf{0}$.

Phase one. At time $t = 1$, the served body K_1 is the hyperplane forcing the first coordinate to be $z_1 \coloneqq 1$:

$$K_1 = \left\{\boldsymbol{x} : \boldsymbol{x}^\top \boldsymbol{e}_1 = z_1\right\}.$$

This same hyperplane K_1 is then repeatedly served until the time m_1 at which ROB is almost stationary. That is, the time $m_1 < \infty$ is chosen to satisfy the property that the cumulative cost incurred by ROB after time m_1, if K_1 were repeated indefinitely thereafter, is bounded above by δ. Such a time m_1 must exist, since ROB is b-competitive, the offline optimal cost for the instance comprised of repeated K_1s is 1, and the tail of a convergent series converges to zero. At time m_1, if ALG's decision coincides with that of ADV, i.e. if $\text{ALG}_{m_1} = \text{ADV}_{m_1}$ (note we will define the behavior of ALG and ADV later on), then we say that phase one is complete and we move on to phase two below. Otherwise, we continue to the next subphase in phase one as follows.

Let $z_2 := -\operatorname{sgn}(\text{ROB}_{m_1,2})$ be the negative of the sign of ROB's 2nd entry at time m_1 (defaulting to 1 if $\text{ROB}_{m_1,2} = 0$). At time $t = m_1 + 1$, we serve a new affine subspace K_2 defined as

$$K_2 = \left\{ \boldsymbol{x} : \boldsymbol{x}^\top \boldsymbol{e}_i = z_i, i = 1, 2 \right\}.$$

Note that this forces ROB to incur cost at least 1 at time $m_1 + 1$. This same body is repeated until the time m_2 at which ROB is almost stationary. That is, just as before, m_2 is defined as the time at which, if K_2 were repeated indefinitely from time $m_2 + 1$ onward, ROB would incur total cost no more than δ after time m_2. For the same reason as before, $m_2 < \infty$ is certain to exist by b-competitiveness of ROB. If $\text{ALG}_{m_2} = \text{ADV}_{m_2}$, then we say that phase one is complete and move on to phase two below. Otherwise, we continue to the next subphase in phase one.

The remaining subphases in phase one are constructed similarly: for each $j = 3, \ldots, 3\sqrt{d}$, we define $z_j := -\operatorname{sgn}(\text{ROB}_{m_{j-1},j})$ to be the negative of the sign of ROB's j^{th} entry at time m_{j-1}, and at time $t = m_{j-1} + 1$, we serve a new affine subspace K_j defined as

$$K_j = \left\{ \boldsymbol{x} : \boldsymbol{x}^\top \boldsymbol{e}_i = z_i, i = 1, \ldots, j \right\}, \tag{2}$$

which forces ROB to incur cost at least 1. This body K_j is then repeated until the time m_j at which ROB is almost stationary, i.e., after which it would incur cumulative cost no more than δ, were K_j to be repeated indefinitely. Then, if $\text{ALG}_{m_j} = \text{ADV}_{m_j}$, we say that phase one is complete and move on to phase two below. Otherwise, we remain in phase one and repeat this step with an incremented value of j. Once the subphase corresponding to $j = 3\sqrt{d}$ is completed, then the instance is concluded without moving on to phase two.

Behavior of the advice. We specify the behavior of ADV based on the behavior of ROB on the (possibly) counterfactual instance wherein phase one runs to termination without moving to phase two. That is, let $\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_{m_{3\sqrt{d}}}$ be the decisions of ROB on an auxiliary CBC instance where K_1 is served from time 1 through m_1, K_2 is served from time $m_1 + 1$ through m_2, and so on, terminating with $K_{3\sqrt{d}}$ being served from time $m_{3\sqrt{d}-1} + 1$ through $m_{3\sqrt{d}}$. Then define

$$\boldsymbol{a} = \underset{\boldsymbol{x} \in \{\pm 1\}^{d-3\sqrt{d}}}{\arg\max} \min_{j=1,\ldots,3\sqrt{d}} \|\boldsymbol{x} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2}, \tag{3}$$

where $\boldsymbol{r}_{m_j,3\sqrt{d}+1:}$ is the vector obtained by dropping the first $3\sqrt{d}$ entries in $\boldsymbol{r}_{m_j}$. Thus, $\boldsymbol{a}$ is the corner of the hypercube $\{\pm 1\}^{d-3\sqrt{d}}$ that is farthest (in ℓ^2) from any of the subvectors comprised of the last $d - 3\sqrt{d}$ entries of the decisions $\boldsymbol{r}_{m_1}, \ldots, \boldsymbol{r}_{m_{3\sqrt{d}}}$ made by ROB at the conclusion of the phase one subphases. Then we define the advice's phase one behavior simply as follows: at time 1, the advice immediately moves to the point

$$\hat{\boldsymbol{a}} = (z_1, \ldots, z_{3\sqrt{d}}, a_1, \ldots, a_{d-3\sqrt{d}}),$$

and it remains there until phase one is completed.

Phase two. Fix $\epsilon > 0$. Suppose that phase one terminates at time m_j, where $j < 3\sqrt{d}$ (since if $j = 3\sqrt{d}$, then the instance ends without moving on to phase two). Thus it is the case that $\text{ALG}_{m_j} = \text{ADV}_{m_j} = \hat{\boldsymbol{a}}$, and $\text{ROB}_{m_j} = \boldsymbol{r}_{m_j}$. Then the instance splits into two cases:

1.) Suppose that $\|\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2} \geq \sqrt{d - 3\sqrt{d}}$, and define $\boldsymbol{v} = \frac{\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}}{\|\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2}}$. Then at each time $t = m_j + 1, \dots, m_j + k$ (where k will be defined later), we serve the body K_j again. By our selection of m_j, the robust algorithm ROB will remain δ-close to its decision ROB_{m_j}, since we are simply continuing to serve the same body. However, at each of these times, we make the advice move to the point

$$\text{ADV}_t = \left(z_1, \dots, z_{3\sqrt{d}}, a_1 + (t - m_j)\epsilon v_1, \dots, a_{d-3\sqrt{d}} + (t - m_j)\epsilon v_{d-3\sqrt{d}}\right).$$

That is, at each time $t = m_j + 1, \dots, m_j + k$, the advice takes a step of length ϵ in the direction $\boldsymbol{v}$ in its last $d - 3\sqrt{d}$ coordinates. Then k is chosen such that $m_j + k$ is the first time after m_j at which $\text{ALG}_{m_j+k} = \text{ROB}_{m_j+k}$, i.e., the first time at which the algorithm switches back to ROB after following the advice. Note that $k < \infty$, by the assumption that ALG has finite robustness. Then the final body is chosen as

$$K_{\text{fin}} = \left\{\text{ADV}_{m_j+k}\right\} = \left\{\left(z_1, \dots, z_{3\sqrt{d}}, a_1 + k\epsilon v_1, \dots, a_{d-3\sqrt{d}} + k\epsilon v_{d-3\sqrt{d}}\right)\right\},$$

which allows the advice to stay put while ROB and ALG must move back to coincide with it.

2.) Suppose that $\|\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2} < \sqrt{d - 3\sqrt{d}}$. Since $\boldsymbol{a}$ maximizes the objective of (3), then it must hold that

$$\min_{i=1,\dots,3\sqrt{d}} \|-\boldsymbol{a} - \boldsymbol{r}_{m_i,3\sqrt{d}+1:}\|_{\ell^2} < \sqrt{d - 3\sqrt{d}}. \tag{4}$$

Let i^* be the minimizing index in (4); note that $i^* \neq j$, since otherwise,

$$\begin{aligned} 2\sqrt{d - 3\sqrt{d}} &= \|2\boldsymbol{a}\| \\ &\leq \|\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2} + \|\boldsymbol{a} + \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2} \quad \text{by the triangle inequality} \\ &< 2\sqrt{d - 3\sqrt{d}} \end{aligned}$$

giving a contradiction. Then the instance splits into two further subcases:

(a) Suppose that $i^* < j$. Then, just as in case 1.), at each time $t = m_j + 1, \dots, m_j + k$, we serve the body K_j again. At each of these times, we make the advice move to the point

$$\text{ADV}_t = \left(z_1, \dots, z_{3\sqrt{d}}, a_1 + (t - m_j)\epsilon v_1, \dots, a_{d-3\sqrt{d}} + (t - m_j)\epsilon v_{d-3\sqrt{d}}\right),$$

where $\boldsymbol{v} = \frac{\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}}{\|\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2}}$ just as in case 1.). Just as in case 1.), k is chosen such that $m_j + k$ is the first time after m_j at which $\text{ALG}_{m_j+k} = \text{ROB}_{m_j+k}$, i.e., the first time at which the algorithm switches back to ROB after following the advice. Then the final body is chosen as

$$K_{\text{fin}} = \left\{\text{ADV}_{m_j+k}\right\} = \left\{\left(z_1, \dots, z_{3\sqrt{d}}, a_1 + k\epsilon v_1, \dots, a_{d-3\sqrt{d}} + k\epsilon v_{d-3\sqrt{d}}\right)\right\},$$

which allows the advice to stay put while ROB and ALG must move back to coincide with it.

(b) Suppose that $i^* > j$. Then for each $l = j+1, \ldots, i^*$, serve the body K_l as defined in (2) from time $m_{l-1}+1$ through m_l, while keeping the advice at the same point $\hat{\boldsymbol{a}}$. Since ROB is advice agnostic and this sequence of bodies coincides with the remainder of the phase one sequence of bodies, it will be the case that $\text{ROB}_{m_{i^*}} = \boldsymbol{r}_{m_{i^*}}$. Then, finally, we split into two further subcases.

(i) If $\text{ALG}_{m_{i^*}} = \text{ROB}_{m_{i^*}} = \boldsymbol{r}_{m_{i^*}}$, then simply choose the final body as

$$K_{\text{fin}} = \{\hat{\boldsymbol{a}}\},$$

which allows the advice to stay put while ROB and ALG must move back to coincide with it.

(ii) If $\text{ALG}_{m_{i^*}} = \text{ADV}_{m_{i^*}} = \hat{\boldsymbol{a}}$, then proceed similarly to subcase (a): for each time $t = m_{i^*}+1, \ldots, m_{i^*}+k$, we serve the body K_{i^*} again and make the advice move to the point

$$\text{ADV}_t = \left(z_1, \ldots, z_{3\sqrt{d}}, a_1 + (t-m_j)\epsilon v_1, \ldots, a_{d-3\sqrt{d}} + (t-m_j)\epsilon v_{d-3\sqrt{d}}\right),$$

where this time $\boldsymbol{v} = \frac{\boldsymbol{a}-\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}}{\|\boldsymbol{a}-\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}\|_{\ell^2}}$. Just as in subcase (a), k is chosen such that $m_{i^*}+k$ is the first time after m_{i^*} at which $\text{ALG}_{m_{i^*}+k} = \text{ROB}_{m_{i^*}+k}$, i.e., the first time at which the algorithm switches back to ROB after following the advice starting from time m_{i^*}. Then the final body is simply chosen as

$$K_{\text{fin}} = \left\{\text{ADV}_{m_j+k}\right\} = \left\{\left(z_1, \ldots, z_{3\sqrt{d}}, a_1 + k\epsilon v_1, \ldots, a_{d-3\sqrt{d}} + k\epsilon v_{d-3\sqrt{d}}\right)\right\},$$

which allows the advice to stay put while forcing ROB and ALG to move to coincide with it.

Cost analysis. Let us now tally costs for each of the cases of the instance to prove the result.

Consider the initial case where the instance never makes it out of phase one; this means that $3\sqrt{d}$ subphases occur in phase one, and ALG finishes each subphase at the ROB decision. Since $\|\boldsymbol{r}_{m_j} - \boldsymbol{r}_{m_{j-1}}\|_{\ell^2} \geq 1$ for each $j = 1, \ldots, 3\sqrt{d}$ (where $\boldsymbol{r}_{m_0} := \boldsymbol{r}_0 = \boldsymbol{x}_0$), this means that ALG incurs cost at least $3\sqrt{d}$, whereas the advice, which moves immediately to $\hat{\boldsymbol{a}} \in \{\pm 1\}^d$ and stays there throughout the entire instance, incurs cost $\sqrt{d}$. Thus ALG is at least 3-consistent, and we are done.

Now, we turn to each of the cases within which the instance makes it to phase two. First, consider case 1.). Since $\text{ALG}_{m_j} = \text{ADV}_{m_j} = \hat{\boldsymbol{a}}$, the cost incurred by ALG through time m_j is at least $\sqrt{d}$. Then from time m_j to m_j+k-1 while ALG is following the advice, ALG incurs cost $\|(k-1)\epsilon\boldsymbol{v}\|_{\ell^2} = (k-1)\epsilon$. At time m_j+k, ALG switches back to ROB, incurring cost at least

$$\begin{aligned}
\|\text{ADV}_{m_j+k-1} - \text{ROB}_{m_j+k}\|_{\ell^2} &\geq \|\text{ADV}_{m_j+k-1} - \text{ROB}_{m_j}\|_{\ell^2} - \|\text{ROB}_{m_j} - \text{ROB}_{m_j+k}\|_{\ell^2} \\
&\geq \|\text{ADV}_{m_j+k-1} - \text{ROB}_{m_j}\|_{\ell^2} - \delta \\
&\geq \|\text{ADV}_{m_j+k-1,3\sqrt{d}+1:} - \text{ROB}_{m_j,3\sqrt{d}+1:}\|_{\ell^2} - \delta \\
&= \|(\boldsymbol{a} + (k-1)\epsilon\boldsymbol{v}) - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2} - \delta \\
&= \|\boldsymbol{a} - \boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2} + (k-1)\epsilon - \delta \\
&\geq \sqrt{d - 3\sqrt{d}} + (k-1)\epsilon - \delta. \qquad (5)
\end{aligned}$$

Finally, by an analogous argument to (5), to switch back to ADV, ALG incurs a cost of at least $\sqrt{d-3\sqrt{d}}+k\epsilon-\delta$. In sum, ALG incurs a total cost of $\sqrt{d}+2\sqrt{d-3\sqrt{d}}+(3k-2)\epsilon-2\delta$. On the other hand, ADV incurs a total cost of $\sqrt{d}+k\epsilon$. Then the consistency of ALG is

$$\frac{\sqrt{d}+2\sqrt{d-3\sqrt{d}}+(3k-2)\epsilon-2\delta}{\sqrt{d}+k\epsilon}$$

which can be made arbitrarily close to 3 by choosing ϵ and δ small and taking d arbitrarily large.

Next, let's move to case 2.). First, we set up some preliminaries. Let's call $\rho=\sqrt{d-3\sqrt{d}}-\|\boldsymbol{a}-\boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2}$, and note that $\rho>0$. Since $\|\boldsymbol{a}-\boldsymbol{r}_{m_j,3\sqrt{d}+1:}\|_{\ell^2}\geq\|-\boldsymbol{a}-\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}\|_{\ell^2}$, we have that

$$\|-\boldsymbol{a}-\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}\|_{\ell^2}\leq\sqrt{d-3\sqrt{d}}-\rho \tag{6}$$

and hence

$$\begin{aligned}\|\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}\|_{\ell^2}&\geq\|-\boldsymbol{a}\|-\|-\boldsymbol{a}-\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}\|_{\ell^2}\\&\geq\rho \qquad (7)\end{aligned}$$

Then

$$\begin{aligned}\|\hat{\boldsymbol{a}}-\boldsymbol{r}_{m_{i^*}}\|_{\ell^2}&\geq\|\boldsymbol{a}-\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}\|_{\ell^2} &(8)\\&\geq\left\|\boldsymbol{a}-\Pi_{B(-\boldsymbol{a},\sqrt{d-3\sqrt{d}}-\rho)}\boldsymbol{a}\right\|_{\ell^2} &(9)\\&=\left\|2\boldsymbol{a}-\Pi_{B(\boldsymbol{0},\sqrt{d-3\sqrt{d}}-\rho)}2\boldsymbol{a}\right\|_{\ell^2} &(10)\\&=\left\|2\boldsymbol{a}-(\sqrt{d-3\sqrt{d}}-\rho)\frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_{\ell^2}}\right\|_{\ell^2} &(11)\\&=\left\|\boldsymbol{a}+\rho\frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_{\ell^2}}\right\|_{\ell^2}\\&=\sqrt{d-3\sqrt{d}}+\rho &(12)\end{aligned}$$

where $\Pi_K\boldsymbol{x}$ denotes the projection of the point $\boldsymbol{x}$ onto the convex body K, (8) follows from $\boldsymbol{a}$ and $\boldsymbol{r}_{m_{i^*},3\sqrt{d}+1:}$ being subvectors of $\hat{\boldsymbol{a}}$ and $\boldsymbol{r}_{m_{i^*}}$, respectively, (9) follows from (6) and non-expansivity of the projection, (10) follows from translation, (11) applies the fact that the projection onto an origin-centered ball is just a radial projection, and (12) follows from $\|\boldsymbol{a}\|_{\ell^2}=\sqrt{d-3\sqrt{d}}$.

Now, let's consider the subcases, starting with subcase (a). Since $i^*<j$, we know that $\text{ALG}_{m_{i^*}}=\text{ROB}_{m_{i^*}}=\boldsymbol{r}_{m_{i^*}}$. Then by (7), ALG incurs cost at least ρ to get to $\boldsymbol{r}_{m_{i^*}}$, and by (12) it incurs another cost of at least $\sqrt{d-3\sqrt{d}}+\rho$ to get to $\text{ADV}_{m_j}=\hat{\boldsymbol{a}}$. From time m_j to m_j+k-1 while ALG is following the advice, ALG incurs cost $(k-1)\epsilon$. Then at time m_j+k, ALG switches back to ROB, and by a similar analysis to that in (5) done for case 1.), it incurs cost at least $\sqrt{d-3\sqrt{d}}-\rho+(k-1)\epsilon-\delta$ to do so. Finally, to switch back to ADV, ALG incurs a cost of at least $\sqrt{d-3\sqrt{d}}-\rho+k\epsilon-\delta$. Then in sum, ALG has incurred a total cost of $3\sqrt{d-3\sqrt{d}}+(3k-2)\epsilon-2\delta$ in this instance case. On the other hand, ADV incurs a total cost of $\sqrt{d}+k\epsilon$, so ALG has consistency

$$\frac{3\sqrt{d-3\sqrt{d}}+(3k-2)\epsilon-2\delta}{\sqrt{d}+k\epsilon},$$

which can be made arbitrarily close to 3 by choosing ϵ and δ small and taking d arbitrarily large.

Now, we move to subcase (b), beginning first with (i). ALG spends $\sqrt{d}$ to get to $\text{ADV}_{m_j} = \hat{\boldsymbol{a}}$ in the first place, and then by (12) it spends cost at least $\sqrt{d - 3\sqrt{d}} + \rho$ to get to $\text{ROB}_{m_{i^*}}$ at time m_{i^*}. Finally, it spends at least another $\sqrt{d - 3\sqrt{d}} + \rho$ to get back to $\hat{\boldsymbol{a}}$ for the final timestep. Thus in sum, ALG incurs cost $\sqrt{d} + 2(\sqrt{d - 3\sqrt{d}} + \rho)$, whereas ADV incurs cost $\sqrt{d}$, giving a consistency of

$$\frac{\sqrt{d} + 2(\sqrt{d - 3\sqrt{d}} + \rho)}{\sqrt{d}},$$

which even for arbitrarily small $\rho > 0$ can be made arbitrarily close to 3 by choosing d sufficiently large.

Finally, we consider scenario (ii) in subcase (b). ALG first spends $\sqrt{d}$ to get to $\text{ADV}_{m_j} = \hat{\boldsymbol{a}}$, and then from time $m_{i^*} + 1$ through $m_{i^*} + k - 1$ it incurs cost $(k-1)\epsilon$ to follow the advice. Using (12) and reasoning analogous to that in (5), ALG incurs cost $\sqrt{d - 3\sqrt{d}} + \rho + (k-1)\epsilon - \delta$ to switch back to ROB at time $m_{i^*} + k$, and finally, it incurs cost $\sqrt{d - 3\sqrt{d}} + \rho + k\epsilon - \delta$ to switch back to the advice in the final timestep. Thus in sum, ALG incurs cost at least $\sqrt{d} + 2(\sqrt{d - 3\sqrt{d}} + \rho) + (3k-2)\epsilon - 2\delta$, while ADV incurs cost $\sqrt{d} + k\epsilon$. Thus ALG has consistency

$$\frac{\sqrt{d} + 2(\sqrt{d - 3\sqrt{d}} + \rho) + (3k-2)\epsilon - 2\delta}{\sqrt{d} + k\epsilon},$$

which, even for very small $\rho > 0$, can be made arbitrarily close to 3 by choosing ϵ, δ small and taking d sufficiently large.

■

B.4. Proof of Proposition 9

Let us first recall Theorem 2.1 of Bubeck et al. (2019), which characterizes the cost incurred by moving to the Steiner point of each nested body.

Theorem 16 ((Bubeck et al., 2019, Theorem 2.1)) *Let $\boldsymbol{x}_0 = \mathbf{0}$ and $K_1 \subseteq B(\mathbf{0}, r)$ for some $r > 0$. Then following the Steiner point of each nested body K_t incurs total movement cost no more than rd.*

We now prove Proposition 9. For clarity, we abbreviate NESTEDSWITCH in this proof as NS.

If NS only ever follows ADV, then $\epsilon \cdot \mathrm{C}_{\text{ADV}} < r(d+2)$ and $\mathrm{C}_{\text{NS}} = \mathrm{C}_{\text{ADV}}$, so $\mathrm{C}_{\text{NS}} \leq \frac{r(d+2)}{\epsilon}$. Thus NS is 1-competitive with respect to ADV and $\frac{r(d+2)}{\epsilon}$-robust, since $\mathrm{C}_{\text{OPT}} \geq 1$.

On the other hand, if NS only ever follows ROB, then $\mathrm{C}_{\text{NS}} = \mathrm{C}_{\text{ROB}}$ and $\epsilon \cdot \mathrm{C}_{\text{ADV}} \geq r(d+2)$. Since ROB just follows the Steiner point of each nested body, we have $\mathrm{C}_{\text{ROB}} \leq r + rd$, where the rd comes from Theorem 16 and the extra factor of r arises from the triangle inequality applied to the $t = 1$ movement:

$$\|\boldsymbol{s}_1 - \boldsymbol{x}_0\|_{\ell^2} \leq \|\boldsymbol{s}_1 - \boldsymbol{y}\|_{\ell^2} + \|\boldsymbol{y} - \boldsymbol{x}_0\|_{\ell^2} \leq \|\boldsymbol{s}_1 - \boldsymbol{y}\|_{\ell^2} + r.$$

Thus $\mathrm{C}_{\text{NS}} = \mathrm{C}_{\text{ROB}} \leq r(d+1) \leq (1+\epsilon)\mathrm{C}_{\text{ADV}}$, and the desired robustness also holds.

Finally, suppose NS switches to ROB at time $t \in [T]$; i.e., $\text{NS}_1 = \text{ADV}_1, \ldots, \text{NS}_{t-1} = \text{ADV}_{t-1}, \text{NS}_t = \text{ROB}_t, \ldots, \text{NS}_T = \text{ROB}_T$. We know that

$$\mathrm{C}_{\text{NS}}(1, t-1) = \mathrm{C}_{\text{ADV}}(1, t-1) < \frac{r(d+2)}{\epsilon}$$

and since $K_t \subseteq B(\boldsymbol{y}, r)$,

$$\mathrm{C}_{\text{NS}}(t,t) = \|\text{ROB}_t - \text{ADV}_{t-1}\|_{\ell^2} \leq 2r$$

and finally

$$\mathrm{C}_{\text{NS}}(t+1, T) = \mathrm{C}_{\text{ROB}}(t+1, T) \leq rd.$$

Thus in sum,

$$\mathrm{C}_{\text{NS}} \leq \frac{r(d+2)}{\epsilon} + rd + 2r = \left(1 + \frac{1}{\epsilon}\right) r(d+2).$$

This gives both the robustness and consistency bounds, since $\epsilon \cdot \mathrm{C}_{\text{ADV}} \geq r(d+2)$ and $\mathrm{C}_{\text{OPT}} \geq 1$. ■

Appendix C. Background from the geometry of normed vector spaces

In this appendix, we introduce some notions and results from the literature on the geometry of normed vector spaces, expanding on the brief definitions of the rectangular constant and the radial retraction given in the main text in Section 4.1. In the following definitions and results, $\mathcal{X} = (X, \|\cdot\|)$ is an arbitrary real normed vector space.

We begin by defining Birkhoff-James orthogonality, which generalizes the usual Hilbert space orthogonality.

Definition 17 ((Birkhoff, 1935, p.169); (James, 1947, p.265)) *$\boldsymbol{x} \in X$ is **Birkhoff-James orthogonal** to $\boldsymbol{y} \in X$, denoted $\boldsymbol{x} \perp \boldsymbol{y}$, if $\|\boldsymbol{x}\| \leq \|\boldsymbol{x} + \lambda \boldsymbol{y}\|$ for all $\lambda \in \mathbb{R}$.*

Note that, unlike orthogonality in Hilbert spaces, Birkhoff-James orthogonality is not generally symmetric. However, it is homogeneous.

Lemma 18 ((James, 1947, p.265); (Joly, 1969, Remark 1)) *If $\boldsymbol{x} \perp \boldsymbol{y}$, then $a\boldsymbol{x} \perp b\boldsymbol{y}$ for all $a, b \in \mathbb{R}$.*

Using Birkhoff-James orthogonality, we can formally define define the first constant we introduced in Section 4.1: the rectangular constant. It is motivated by the following observation: in a finite-dimensional inner product space, orthogonality of $\boldsymbol{x}$ and $\boldsymbol{y}$ implies that $\frac{\|\boldsymbol{x}\|+\|\boldsymbol{y}\|}{\|\boldsymbol{x}+\boldsymbol{y}\|} \leq \sqrt{2}$. In an arbitrary normed vector space, the upper bound $\sqrt{2}$ is replaced with the *rectangular constant*, defined as follows using Birkhoff-James orthogonality.

Definition 19 ((Desbiens, 1990, Definition 2); original from (Joly, 1969, Definition 2)) *The **rectangular constant** $\mu(\mathcal{X})$ of a real normed vector space $\mathcal{X}$ is defined as*

$$\mu(\mathcal{X}) = \sup_{\boldsymbol{x} \perp \boldsymbol{y}} \frac{\|\boldsymbol{x}\| + \|\boldsymbol{y}\|}{\|\boldsymbol{x} + \boldsymbol{y}\|}.$$

It is known that $\sqrt{2} \le \mu(\mathcal{X}) \le 3$ (Joly, 1969, Section II), and these bounds are tight: $\mu(\mathcal{X}) = \sqrt{2}$ for any Hilbert space (Joly, 1969, Example 1; Section III), and $\mu(\mathcal{X}) = 3$ for "nonuniformly nonsquare" spaces such as ℓ^1 and ℓ^∞ (Baronti et al. (2021)). Moreover, $\mu(\ell^p) < 3$ for all $p \in (1, \infty)$. In fact, tighter bounds are known for the ℓ^p spaces: we review these in the following theorem.

Theorem 20 ((Baronti et al., 2021, Theorems 5.2, 5.4, 5.5)) *For $1 < p \le 2$,*

$$\mu(\ell^p) \le \min \left\{ \left(1 + \left(2^{1/(p-1)} - 1\right)^{p-1}\right)^{1/p}, \sqrt{\frac{p}{p-1}} \right\}.$$

For $p \ge 2$,

$$\mu(\ell^p) \le \left(1 + \left(2^{p-1} - 1\right)^{1/(p-1)}\right)^{(p-1)/p}.$$

Together, these constitute an upper bound on $\mu(\ell^p)$ that attains a (tight) minimum of $\sqrt{2}$ at $p = 2$, and that continuously increases toward 3 as $p \to \infty$ and $p \to 1$.

We now reiterate the definition of the radial retraction and its Lipschitz constant given in Section 4.1.

Definition 21 (Rieffel (2006)) *On a normed vector space $\mathcal{X} = (X, \|\cdot\|)$, the radial retraction $\rho(\cdot\,; r) : X \to B(\mathbf{0}, r)$ is the metric projection onto the closed ball of radius $r \ge 0$:*

$$\rho(\boldsymbol{x}; r) = \begin{cases} \boldsymbol{x} & \text{if } \|\boldsymbol{x}\| \le r \\ r\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} & \text{if } \|\boldsymbol{x}\| > r. \end{cases}$$

We define $k(\mathcal{X})$ to be the Lipschitz constant of $\rho(\cdot\,; 1)$, i.e., the smallest real number satisfying

$$\|\rho(\boldsymbol{x}; 1) - \rho(\boldsymbol{y}; 1)\| \le k(\mathcal{X})\|\boldsymbol{x} - \boldsymbol{y}\|$$

for all $\boldsymbol{x}, \boldsymbol{y} \in X$.

It holds that $k(\mathcal{X})$ is bounded between 1 and 2 in any normed vector space $\mathcal{X}$ (Thele (1974)). Moreover, $k(\mathcal{X})$ is identically the Lipschitz constant of $\rho(\cdot\,; r)$ for any $r > 0$ (Rieffel (2006)). To see that this is the case, observe that $\rho(\boldsymbol{x}; r) = r \cdot \rho(\frac{\boldsymbol{x}}{r}; 1)$; it then follows that

$$\|\rho(\boldsymbol{x}; r) - \rho(\boldsymbol{y}; r)\| = r \left\| \rho\left(\frac{\boldsymbol{x}}{r}; 1\right) - \rho\left(\frac{\boldsymbol{y}}{r}; 1\right) \right\| \le k(\mathcal{X})\|\boldsymbol{x} - \boldsymbol{y}\|.$$

Thus $k(\mathcal{X})$ is an upper bound on the Lipschitz constant of $\rho(\boldsymbol{x}; r)$ for general $r > 0$. Similar reasoning shows that $\rho(\boldsymbol{x}; r)$ can have no smaller Lipschitz constant than $k(\mathcal{X})$; so $k(\mathcal{X})$ is the Lipschitz constant for all $\rho(\boldsymbol{x}; r)$, $r > 0$.

We conclude this section with a result relating $k(\mathcal{X})$ with $\mu(\mathcal{X})$.

Proposition 22 *On a real normed vector space $\mathcal{X} = (X, \|\cdot\|)$, it holds that $k(\mathcal{X}) \le \mu(\mathcal{X})$.*

Proof (Thele, 1974, Theorem 1) characterizes $k(\mathcal{X})$ as follows:

$$k(\mathcal{X}) = \sup_{\boldsymbol{x} \perp \boldsymbol{y}, \boldsymbol{y} \neq \boldsymbol{0}, \lambda \in \mathbb{R}} \frac{\|\boldsymbol{y}\|}{\|\boldsymbol{y} - \lambda \boldsymbol{x}\|}. \tag{13}$$

Since, by Lemma 18, Birkhoff-James orthogonality is homogeneous, it is straightforward to see that (13) can be equivalently expressed as

$$k(\mathcal{X}) = \sup_{\boldsymbol{x} \perp \boldsymbol{y}, \boldsymbol{y} \neq \boldsymbol{0}} \frac{\|\boldsymbol{y}\|}{\|\boldsymbol{x} + \boldsymbol{y}\|}.$$

Then it is clear that

$$k(\mathcal{X}) = \sup_{\boldsymbol{x} \perp \boldsymbol{y}, \boldsymbol{y} \neq \boldsymbol{0}} \frac{\|\boldsymbol{y}\|}{\|\boldsymbol{x} + \boldsymbol{y}\|} \leq \sup_{\boldsymbol{x} \perp \boldsymbol{y}} \frac{\|\boldsymbol{x}\| + \|\boldsymbol{y}\|}{\|\boldsymbol{x} + \boldsymbol{y}\|} = \mu(\mathcal{X}).$$

■

Appendix D. Proof of Theorem 11

D.1. Geometric lemmas

Before presenting the analysis of Algorithm 2, we take a brief foray into the geometric theory of normed vector spaces, presenting and proving some lemmas that will be helpful in proving the bi-competitive bound given in Theorem 11. The results in this section depend heavily on the definitions and results introduced in Appendix C.

The first lemma characterizes (a modified form of) the radial retraction as a metric projection onto the boundary of a closed ball.

Lemma 23 *Let $(X, \|\cdot\|)$ be a normed vector space, and consider arbitrary $r \geq 0$, $\boldsymbol{x} \in X$, and $\boldsymbol{y} \in X \setminus \{\boldsymbol{x}\}$. Define $\hat{\boldsymbol{y}} = \boldsymbol{x} + r\frac{\boldsymbol{y}-\boldsymbol{x}}{\|\boldsymbol{y}-\boldsymbol{x}\|}$. Then $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\| \leq \|\boldsymbol{y} - \boldsymbol{w}\|$ for all $\boldsymbol{w} \in \partial B(\boldsymbol{x}, r)$.*

Proof It suffices to consider the case when $\boldsymbol{x} = \boldsymbol{0}$. If $r = 0$, $\partial B(\boldsymbol{0}, r) = \{\boldsymbol{0}\}$, so the result is clear. Otherwise, fix arbitrary $\boldsymbol{w} \in \partial B(\boldsymbol{0}, r)$ and observe

$$\begin{aligned}
\|\boldsymbol{y} - \boldsymbol{w}\| &\geq |\|\boldsymbol{y}\| - \|\boldsymbol{w}\|| && \text{by the triangle inequality} \\
&= |\|\boldsymbol{y}\| - r| \\
&= |\|\boldsymbol{y}\| - \|\hat{\boldsymbol{y}}\|| \\
&= \|\boldsymbol{y} - \hat{\boldsymbol{y}}\|
\end{aligned}$$

where the last step follows from collinearity of $\boldsymbol{y}, \hat{\boldsymbol{y}}$, and $\boldsymbol{0}$. ■

The second lemma generalizes the following geometric fact in the Euclidean plane to an arbitrary normed vector spaces: given a triangle $\boldsymbol{abc}$ in $(\mathbb{R}^2, \|\cdot\|_{\ell^2})$, and points $\boldsymbol{x} \in [\boldsymbol{a}, \boldsymbol{b}], \boldsymbol{y} \in [\boldsymbol{a}, \boldsymbol{c}]$ with $\|\boldsymbol{x} - \boldsymbol{b}\|_{\ell^2} = \|\boldsymbol{y} - \boldsymbol{c}\|_{\ell^2}$, it holds that $\|\boldsymbol{x} - \boldsymbol{y}\|_{\ell^2} \leq \|\boldsymbol{b} - \boldsymbol{c}\|_{\ell^2}$. In the general setting, this becomes a statement about the distance between the radial retractions of a single point onto two balls of the same radius with different centers.

Lemma 24 *Let* $(X, \|\cdot\|)$ *be a normed vector space, and fix arbitrary* $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \in X$ *and* $r \geq 0$. *Define* $\boldsymbol{x} = \boldsymbol{b} + \rho(\boldsymbol{a} - \boldsymbol{b}; r)$ *and* $\boldsymbol{y} = \boldsymbol{c} + \rho(\boldsymbol{a} - \boldsymbol{c}; r)$. *Then* $\|\boldsymbol{x} - \boldsymbol{y}\| \leq \|\boldsymbol{b} - \boldsymbol{c}\|$.

Proof We may assume without loss of generality that $\|\boldsymbol{a} - \boldsymbol{b}\| \geq \|\boldsymbol{a} - \boldsymbol{c}\|$. If $\boldsymbol{b} = \boldsymbol{c}$, $\|\boldsymbol{x} - \boldsymbol{y}\| = 0 = \|\boldsymbol{b} - \boldsymbol{c}\|$. Thus we restrict to the case where $\boldsymbol{b} \neq \boldsymbol{c}$ and distinguish cases based on the value of r. We may further restrict to the case where $\|\boldsymbol{a} - \boldsymbol{c}\| > 0$, as the case $\boldsymbol{a} = \boldsymbol{c}$ is trivial.

If $r = 0$, then $\boldsymbol{x} = \boldsymbol{b}$ and $\boldsymbol{y} = \boldsymbol{c}$. Thus $\|\boldsymbol{x} - \boldsymbol{y}\| = \|\boldsymbol{b} - \boldsymbol{c}\|$. On the other hand, if $r \geq \|\boldsymbol{a} - \boldsymbol{b}\|$, then $r \geq \|\boldsymbol{a} - \boldsymbol{c}\|$ as well, so $\boldsymbol{x} = \boldsymbol{y} = \boldsymbol{a}$, and certainly $\|\boldsymbol{x} - \boldsymbol{y}\| = 0 \leq \|\boldsymbol{b} - \boldsymbol{c}\|$.

Next, suppose $\|\boldsymbol{a} - \boldsymbol{c}\| \leq r < \|\boldsymbol{a} - \boldsymbol{b}\|$. Then $\boldsymbol{y} = \boldsymbol{a}$, so $\|\boldsymbol{x} - \boldsymbol{y}\| = \|\boldsymbol{x} - \boldsymbol{a}\|$. Moreover, $\|\boldsymbol{x} - \boldsymbol{b}\| = r \geq \|\boldsymbol{a} - \boldsymbol{c}\|$. Then by the triangle inequality,

$$\begin{aligned}
\|\boldsymbol{b} - \boldsymbol{c}\| &\geq |\|\boldsymbol{a} - \boldsymbol{b}\| - \|\boldsymbol{a} - \boldsymbol{c}\|| \\
&= |\|\boldsymbol{a} - \boldsymbol{x}\| + \|\boldsymbol{x} - \boldsymbol{b}\| - \|\boldsymbol{a} - \boldsymbol{c}\|| \\
&\geq \|\boldsymbol{a} - \boldsymbol{x}\| \\
&= \|\boldsymbol{x} - \boldsymbol{y}\|.
\end{aligned}$$

Finally, suppose $0 < r < \|\boldsymbol{a} - \boldsymbol{c}\|$, and define $\lambda = 1 - \frac{r}{\|\boldsymbol{a} - \boldsymbol{c}\|}$. Since $\boldsymbol{y} + \boldsymbol{b} - \boldsymbol{c} = \boldsymbol{b} + r\frac{\boldsymbol{a} - \boldsymbol{c}}{\|\boldsymbol{a} - \boldsymbol{c}\|}$, we know that $\boldsymbol{y} + \boldsymbol{b} - \boldsymbol{c} \in \partial B(\boldsymbol{b}, r)$. Observe moreover that

$$\boldsymbol{z} \coloneqq \boldsymbol{y} + \lambda(\boldsymbol{b} - \boldsymbol{c}) = \boldsymbol{b} + r\frac{\boldsymbol{a} - \boldsymbol{b}}{\|\boldsymbol{a} - \boldsymbol{c}\|} \in [\boldsymbol{a}, \boldsymbol{b}]$$

and $\boldsymbol{z} \neq \boldsymbol{b}$ by assumption that $r > 0$. Thus $\frac{\boldsymbol{z} - \boldsymbol{b}}{\|\boldsymbol{z} - \boldsymbol{b}\|} = \frac{\boldsymbol{a} - \boldsymbol{b}}{\|\boldsymbol{a} - \boldsymbol{b}\|}$, so $\boldsymbol{x} = \boldsymbol{b} + r\frac{\boldsymbol{z} - \boldsymbol{b}}{\|\boldsymbol{z} - \boldsymbol{b}\|}$. Thus:

$$\begin{aligned}
\|\boldsymbol{b} - \boldsymbol{c}\| &= \|(\boldsymbol{y} + \boldsymbol{b} - \boldsymbol{c}) - \boldsymbol{y}\| \\
&= \|(\boldsymbol{y} + \boldsymbol{b} - \boldsymbol{c}) - \boldsymbol{z}\| + \|\boldsymbol{z} - \boldsymbol{y}\| && \text{by collinearity of } \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{y} + \boldsymbol{b} - \boldsymbol{c} \\
&\geq \|\boldsymbol{x} - \boldsymbol{z}\| + \|\boldsymbol{z} - \boldsymbol{y}\| && \text{applying Lemma 23} \qquad (14) \\
&\geq \|\boldsymbol{x} - \boldsymbol{y}\| && \text{by triangle inequality.}
\end{aligned}$$

where, in (14), $\boldsymbol{x}, \boldsymbol{y}, \hat{\boldsymbol{y}}, \boldsymbol{w}$, and r in Lemma 23 are instantiated during its invocation with this proof's $\boldsymbol{b}, \boldsymbol{z}, \boldsymbol{x}, (\boldsymbol{y} + \boldsymbol{b} - \boldsymbol{c})$, and r, respectively. ■

The next geometric lemma provides a bound on the total distance traveled first between two points on a sphere, and then from the second point to a scaled version thereof, in terms of the rectangular constant and the distance between the initial and final points.

Lemma 25 *Let* $(X, \|\cdot\|)$ *be a normed vector space, and let* $t > 1$, $r > 0$, *and* $\boldsymbol{x}, \boldsymbol{y} \in \partial B(\boldsymbol{0}, r)$. *Then*

$$\|\boldsymbol{y} - \boldsymbol{x}\| + (t - 1)\|\boldsymbol{y}\| \leq \mu(\mathcal{X})\|t\boldsymbol{y} - \boldsymbol{x}\|.$$

Proof By a corollary of the Hahn-Banach theorem (Bollobás, 1999, Chapter 3, Corollary 7), there exists a *support functional* $f \in X^*$ at $\boldsymbol{y}$, i.e., some bounded linear functional $f : X \to \mathbb{R}$, with $\|f\|_{X^*} = 1$, $f(\boldsymbol{y}) = \|\boldsymbol{y}\| = r$, and the property that the hyperplane $H(r) \coloneqq \{\boldsymbol{z} \in X : f(\boldsymbol{z}) = r\}$ contains no points in $\operatorname{int}(B(\boldsymbol{0}, r))$. Note that we can equivalently write $H(r)$ in affine subspace form $H(r) = \boldsymbol{y} + \ker(f) = \{\boldsymbol{z} \in X : \boldsymbol{z} = \boldsymbol{y} + \boldsymbol{h}, \boldsymbol{h} \in \ker(f)\}$, by linearity of f. The fact that $H(r)$ contains no points in the interior of $B(\boldsymbol{0}, r)$ means that $\boldsymbol{y} \perp \boldsymbol{h}$ for all $\boldsymbol{h} \in \ker(f)$.

Define $s = f(\boldsymbol{x})$, and note that since $\|\boldsymbol{x}\| = r$ and $\|f\|_{X^*} = 1$, we must have $s \leq r$. Then define $\boldsymbol{z} = \frac{s}{r}\boldsymbol{y}$, and observe $H(s) = \boldsymbol{x} + \ker(f) = \boldsymbol{z} + \ker(f)$. Thus $\boldsymbol{x} = \boldsymbol{z} + \boldsymbol{h}$ for some specific $\boldsymbol{h} \in \ker(f)$. By homogeneity of Birkhoff-James orthogonality (Lemma 18), it follows that $(t - \frac{s}{r})\boldsymbol{y} \perp -\boldsymbol{h}$. As such,

$$\begin{aligned}
\|\boldsymbol{y} - \boldsymbol{x}\| + (t-1)\|\boldsymbol{y}\| &\leq \|\boldsymbol{y} - \boldsymbol{z}\| + \|\boldsymbol{x} - \boldsymbol{z}\| + (t-1)\|\boldsymbol{y}\| \\
&= \left(1 - \frac{s}{r}\right)\|\boldsymbol{y}\| + \|\boldsymbol{h}\| + (t-1)\|\boldsymbol{y}\| \\
&= \|-\boldsymbol{h}\| + \left(t - \frac{s}{r}\right)\|\boldsymbol{y}\| \\
&\leq \mu(\mathcal{X})\left\|-\boldsymbol{h} + \left(t - \frac{s}{r}\right)\boldsymbol{y}\right\| \\
&= \mu(\mathcal{X})\|t\boldsymbol{y} - \boldsymbol{x}\|.
\end{aligned}$$

■

Finally, we present a lemma building upon Lemma 25 that will be indispensable for the consistency analysis of Algorithm 2.

Lemma 26 *Let $(X, \|\cdot\|)$ be a normed vector space, and fix arbitrary $r \geq 0$, $\boldsymbol{w}, \boldsymbol{y} \in X$, and $\boldsymbol{x} \in X \setminus \operatorname{int}(B(\boldsymbol{w}, r))$. Define $\hat{\boldsymbol{x}} = \boldsymbol{w} + \rho(\boldsymbol{x} - \boldsymbol{w}; r)$ and $\hat{\boldsymbol{y}} = \boldsymbol{w} + \rho(\boldsymbol{y} - \boldsymbol{w}; r)$. Then*

$$\|\hat{\boldsymbol{y}} - \hat{\boldsymbol{x}}\| + \|\boldsymbol{y} - \hat{\boldsymbol{y}}\| \leq \mu(\mathcal{X})\|\boldsymbol{y} - \boldsymbol{x}\| + \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|.$$

Proof If $r = 0$, then $B(\boldsymbol{w}, r) = \{\boldsymbol{w}\}$, so $\hat{\boldsymbol{x}} = \hat{\boldsymbol{y}} = \boldsymbol{w}$, and the result follows from the triangle inequality, as $\mu(\mathcal{X}) \geq \sqrt{2}$. Thus we restrict to the case that $r > 0$.

It suffices to consider the case where $\boldsymbol{w} = \boldsymbol{0}$. Then $\hat{\boldsymbol{x}} = \rho(\boldsymbol{x}; r)$ and $\hat{\boldsymbol{y}} = \rho(\boldsymbol{y}; r)$. We distinguish two cases.

First, suppose $\boldsymbol{y} \in B(\boldsymbol{0}, r)$. Then $\hat{\boldsymbol{y}} = \boldsymbol{y}$, and by the triangle inequality,

$$\|\hat{\boldsymbol{y}} - \hat{\boldsymbol{x}}\| + \|\boldsymbol{y} - \hat{\boldsymbol{y}}\| = \|\boldsymbol{y} - \hat{\boldsymbol{x}}\| \leq \|\boldsymbol{y} - \boldsymbol{x}\| + \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \leq \mu(\mathcal{X})\|\boldsymbol{y} - \boldsymbol{x}\| + \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|.$$

Second, suppose $\boldsymbol{y} \in X \setminus B(\boldsymbol{0}, r)$. Then $\hat{\boldsymbol{y}} = r\frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}$. We distinguish two further subcases.

(i) Suppose that $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\| \leq \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|$. Then

$$\begin{aligned}
\|\hat{\boldsymbol{y}} - \hat{\boldsymbol{x}}\| + \|\boldsymbol{y} - \hat{\boldsymbol{y}}\| &\leq \|\hat{\boldsymbol{y}} - \hat{\boldsymbol{x}}\| + \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \\
&\leq k(\mathcal{X})\|\boldsymbol{y} - \boldsymbol{x}\| + \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \\
&\leq \mu(\mathcal{X})\|\boldsymbol{y} - \boldsymbol{x}\| + \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| && \text{by Proposition 22.}
\end{aligned}$$

(ii) On the other hand, suppose that $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\| > \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|$, or equivalently, $\|\boldsymbol{y}\| \geq \|\boldsymbol{x}\|$, and define $\boldsymbol{z} = \rho(\boldsymbol{y}; \|\boldsymbol{x}\|) = \|\boldsymbol{x}\|\frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}$. By collinearity, $\|\boldsymbol{z} - \hat{\boldsymbol{y}}\| = \|\boldsymbol{x}\| - r = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|$. Furthermore, we have that

$$\|\hat{\boldsymbol{y}} - \hat{\boldsymbol{x}}\| \leq \frac{\|\boldsymbol{x}\|}{r}\|\hat{\boldsymbol{y}} - \hat{\boldsymbol{x}}\| = \|\boldsymbol{z} - \boldsymbol{x}\|.$$

It then follows that

$$\begin{aligned}\|\hat{\boldsymbol{y}}-\hat{\boldsymbol{x}}\|+\|\boldsymbol{y}-\hat{\boldsymbol{y}}\| &\leq \|\boldsymbol{z}-\boldsymbol{x}\|+\|\boldsymbol{y}-\boldsymbol{z}\|+\|\boldsymbol{z}-\hat{\boldsymbol{y}}\| \\ &= \|\boldsymbol{z}-\boldsymbol{x}\|+\left(\frac{\|\boldsymbol{y}\|}{\|\boldsymbol{x}\|}-1\right)\|\boldsymbol{z}\|+\|\boldsymbol{x}-\hat{\boldsymbol{x}}\| \\ &\leq \mu(\mathcal{X})\|\boldsymbol{y}-\boldsymbol{x}\|+\|\boldsymbol{x}-\hat{\boldsymbol{x}}\|\end{aligned}$$

where the final inequality follows from Lemma 25, where $\boldsymbol{x}, \boldsymbol{z} \in \partial B(\mathbf{0}, \|\boldsymbol{x}\|)$ are (respectively) the points $\boldsymbol{x}, \boldsymbol{y}$ in that lemma's statement.

■

We have now presented all technical lemmas that will be employed in our proof of Theorem 11. Before moving on to this proof in the next section, we first provide several immediate corollaries of the preceding lemmas characterizing various steps of Algorithm 2.

Corollary 27 *In the specification of* INTERP *(Algorithm 2), if x_t is determined by Line 8, then* $\|\boldsymbol{x}_t-\boldsymbol{z}_t\| \leq \|\boldsymbol{s}_t-\boldsymbol{s}_{t-1}\|$.

Proof This follows immediately from Lemma 24 with the lemma's $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, and r respectively chosen as $\tilde{\boldsymbol{x}}_t$, $\boldsymbol{s}_{t-1}$, $\boldsymbol{s}_t$ and $\|\boldsymbol{z}_t-\boldsymbol{s}_{t-1}\|$. ■

Corollary 28 *In the specification of* INTERP *(Algorithm 2),*

$$\|\boldsymbol{y}_t-\boldsymbol{x}_{t-1}\| \leq k(\mathcal{X})\|\tilde{\boldsymbol{x}}_t-\tilde{\boldsymbol{x}}_{t-1}\|.$$

Proof This follows by definition of the Lipschitz constant $k(\mathcal{X})$ of the radial retraction, and the observation that $\boldsymbol{y}_t$ (respectively $\boldsymbol{x}_{t-1}$) is the radial retraction of $\tilde{\boldsymbol{x}}_t$ (respectively $\tilde{\boldsymbol{x}}_{t-1}$) onto the ball $B(\boldsymbol{s}_{t-1}, \|\boldsymbol{x}_{t-1}-\boldsymbol{s}_{t-1}\|)$. ■

Corollary 29 *In the specification of* INTERP *(Algorithm 2),*

$$\|\boldsymbol{y}_t-\boldsymbol{x}_{t-1}\|+\|\tilde{\boldsymbol{x}}_t-\boldsymbol{y}_t\| \leq \mu(\mathcal{X})\|\tilde{\boldsymbol{x}}_t-\tilde{\boldsymbol{x}}_{t-1}\|+\|\tilde{\boldsymbol{x}}_{t-1}-\boldsymbol{x}_{t-1}\|.$$

Proof This follows immediately from Lemma 26 with $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}$, and r in the lemma's statement chosen respectively as $\tilde{\boldsymbol{x}}_{t-1}$, $\tilde{\boldsymbol{x}}_t$, $\boldsymbol{s}_{t-1}$, and $\|\boldsymbol{x}_{t-1}-\boldsymbol{s}_{t-1}\|$, which in turn yields $\hat{\boldsymbol{y}}=\boldsymbol{y}_t$ and $\hat{\boldsymbol{x}}=\boldsymbol{x}_{t-1}$. ■

D.2. Proof of bicompetitive bound

We prove the bicompetitive bound of Theorem 11 in two parts: we will first show the competitive ratio with respect to ADV, and will follow with the competitive ratio with respect to ROB. Both results proceed via potential function arguments: the first uses the potential function $\|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\|$, and the second uses the potential function $c\|\boldsymbol{x}_t - \boldsymbol{s}_t\|$ (with c to be defined later on). The robustness and consistency claim then follows immediately from the bicompetitive bound and the observation in Appendix A.2.

Proof of competitiveness with respect to ADV. We define "phases" of the algorithm as follows: if $\boldsymbol{x}_t$ is determined by line 4 of the algorithm, then the advice is in the "ADV" phase. Otherwise, if $\boldsymbol{x}_t$ is determined by line 8, then the advice is in the "ROB" phase. We refer to the time indices in which the algorithm is in the "ROB" phase as $R_1, \dots, R_k \in [T]$ (where $k \leq T$, and $R_1 < \cdots < R_k$ are in increasing order). If the algorithm is never in the "ROB" phase, then $\boldsymbol{x}_t = \tilde{\boldsymbol{x}}_t \; \forall t \in [T]$, and thus INTERP is 1-competitive with respect to ADV. Thus we restrict to the case that there is at least one time index in which the algorithm is in the "ROB" phase. By design, for each $j \in [k]$, $\mathrm{C}_{\mathrm{ROB}}(1, R_j) \leq \delta \cdot \mathrm{C}_{\mathrm{ADV}}(1, R_j)$.

Now we break into two cases depending on the phase. First, suppose that INTERP is in the "ADV" phase. This means that $\boldsymbol{x}_t = \tilde{\boldsymbol{x}}_t$. Then

$$\begin{aligned} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| &= f_t(\tilde{\boldsymbol{x}}_t) + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_{t-1}\| \\ &\leq f_t(\tilde{\boldsymbol{x}}_t) + \|\tilde{\boldsymbol{x}}_t - \tilde{\boldsymbol{x}}_{t-1}\| + \|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\| \end{aligned} \tag{15}$$

follows immediately from the triangle inequality.

Second, consider the case that the algorithm is in the "ROB" phase. This means that $\boldsymbol{x}_t$ is determined by line 8 of the algorithm; and there exists some $\lambda \in [0, 1]$ for which $\boldsymbol{x}_t = \lambda \boldsymbol{s}_t + (1 - \lambda)\tilde{\boldsymbol{x}}_t$. In this case, observe

$$\begin{aligned} & f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| \\ \leq \quad & \lambda f_t(\boldsymbol{s}_t) + (1-\lambda) f_t(\tilde{\boldsymbol{x}}_t) + 2\|\boldsymbol{x}_t - \boldsymbol{z}_t\| + 2\|\boldsymbol{z}_t - \boldsymbol{y}_t\| + \|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{y}_t\| \qquad (16) \\ \leq \quad & 2 \cdot \mathrm{C}_{\mathrm{ROB}}(t, t) + 2\gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t, t) + f_t(\tilde{\boldsymbol{x}}_t) + \|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{y}_t\| \qquad (17) \end{aligned}$$

where (16) follows from convexity of f_t and the triangle inequality, and (17) follows from bounding $\|\boldsymbol{x}_t - \boldsymbol{z}_t\|$ via Corollary 27 and $\|\boldsymbol{z}_t - \boldsymbol{y}_t\|$ via line (7) of the algorithm. Invoking Corollary 29 gives the result

$$\begin{aligned} & f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| \\ \leq \quad & 2 \cdot \mathrm{C}_{\mathrm{ROB}}(t, t) + 2\gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t, t) + f_t(\tilde{\boldsymbol{x}}_t) + \mu(\mathcal{X})\|\tilde{\boldsymbol{x}}_t - \tilde{\boldsymbol{x}}_{t-1}\| + \|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\| \\ \leq \quad & 2 \cdot \mathrm{C}_{\mathrm{ROB}}(t, t) + (\mu(\mathcal{X}) + 2\gamma)\mathrm{C}_{\mathrm{ADV}}(t, t) + \|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\| \qquad (18) \end{aligned}$$

Summing (15) and (18) over time and noting that the left-hand side $\|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\|$ and right-hand side $\|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\|$ telescope, we obtain

$$\begin{aligned}
&\mathrm{C}_{\textsc{Interp}}(1,T) \\
\leq\ & \sum_{t=1}^{T} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_T - \boldsymbol{x}_T\| \\
\leq\ & \sum_{t \in \{R_j\}_{j=1}^k} 2 \cdot \mathrm{C}_{\textsc{Rob}}(t,t) + (\mu(\mathcal{X}) + 2\gamma)\mathrm{C}_{\textsc{Adv}}(t,t) + \sum_{t \in [T] \setminus \{R_j\}_{j=1}^k} \mathrm{C}_{\textsc{Adv}}(t,t) \\
& \leq 2 \cdot \mathrm{C}_{\textsc{Rob}}(1,R_k) + (\mu(\mathcal{X}) + 2\gamma)\mathrm{C}_{\textsc{Adv}}(1,T) \\
& \leq 2\delta \cdot \mathrm{C}_{\textsc{Adv}}(1,R_k) + (\mu(\mathcal{X}) + 2\gamma)\mathrm{C}_{\textsc{Adv}}(1,T) \\
& \leq (\mu(\mathcal{X}) + \epsilon)\mathrm{C}_{\textsc{Adv}}(1,T)
\end{aligned}$$

where the second to last inequality follows from the assumption that the algorithm is in the "ROB" phase at time R_k, implying $\mathrm{C}_{\textsc{Rob}}(1,R_k) \leq \delta \cdot \mathrm{C}_{\textsc{Adv}}(1,R_k)$; and in the last inequality we use the assumption on the parameters that $2\gamma + 2\delta = \epsilon$. This gives the competitive bound with respect to ADV. Note that we can repeat the same argument with truncated time horizon to obtain that INTERP is $(\mu(\mathcal{X}) + \epsilon)$-competitive with respect to ADV at every timestep. ■

Proof of competitiveness with respect to ROB. Define the potential function $\phi_t = c\|\boldsymbol{x}_t - \boldsymbol{s}_t\|$, with $c > 0$ to be determined later.

Let $t' \in \{0, \ldots, T\}$ be the last time interval in which the algorithm's decision is determined by line 4 of the algorithm, or equivalently, the greatest t such that $\mathrm{C}_{\textsc{Rob}}(1,t) \geq \delta \cdot \mathrm{C}_{\textsc{Adv}}(1,t)$. Applying the competitive bound of INTERP with respect to ADV to the subhorizon $t = 1, \ldots, t'$, we have $\mathrm{C}_{\textsc{Interp}}(1,t') \leq (\mu(\mathcal{X}) + \epsilon)\mathrm{C}_{\textsc{Adv}}(1,t')$. By the triangle inequality, and since all algorithms begin at the same starting point $\boldsymbol{x}_0$, we have $\|\tilde{\boldsymbol{x}}_{t'} - \boldsymbol{s}_{t'}\| \leq \mathrm{C}_{\textsc{Adv}}(1,t') + \mathrm{C}_{\textsc{Rob}}(1,t')$. Putting these together, we have

$$\begin{aligned}
\mathrm{C}_{\textsc{Interp}}(1,t') + \phi_{t'} &= \mathrm{C}_{\textsc{Interp}}(1,t') + c\|\tilde{\boldsymbol{x}}_{t'} - \boldsymbol{s}_{t'}\| \\
&\leq (\mu(\mathcal{X}) + \epsilon)\mathrm{C}_{\textsc{Adv}}(1,t') + c(\mathrm{C}_{\textsc{Adv}}(1,t') + \mathrm{C}_{\textsc{Rob}}(1,t')) \\
&\leq \left(\frac{\mu(\mathcal{X}) + \epsilon + c}{\delta} + c\right)\mathrm{C}_{\textsc{Rob}}(1,t') \qquad (19)
\end{aligned}$$

Now consider arbitrary $t \in \{t'+1, \ldots, T\}$. We distinguish two cases. First, suppose $\boldsymbol{x}_t = \boldsymbol{s}_t$. Then

$$\begin{aligned}
f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \phi_t - \phi_{t-1} &= f_t(\boldsymbol{s}_t) + \|\boldsymbol{s}_t - \boldsymbol{x}_{t-1}\| + c\|\boldsymbol{s}_t - \boldsymbol{s}_t\| - c\|\boldsymbol{s}_{t-1} - \boldsymbol{x}_{t-1}\| \\
&\leq f_t(\boldsymbol{s}_t) + \|\boldsymbol{s}_t - \boldsymbol{s}_{t-1}\| + \|\boldsymbol{s}_{t-1} - \boldsymbol{x}_{t-1}\| - c\|\boldsymbol{s}_{t-1} - \boldsymbol{x}_{t-1}\| \\
&\leq \mathrm{C}_{\textsc{Rob}}(t,t) \qquad (20)
\end{aligned}$$

where the final inequality holds so long as $c \geq 1$.

On the other hand, suppose $\boldsymbol{x}_t \neq \boldsymbol{s}_t$. Observe that

$$\begin{aligned}
\|\boldsymbol{x}_t - \boldsymbol{s}_t\| &\leq \|\boldsymbol{z}_t - \boldsymbol{s}_{t-1}\| && \text{by line 8 of the algorithm} \\
&= \|\boldsymbol{y}_t - \boldsymbol{s}_{t-1}\| - \gamma \cdot \mathrm{C}_{\textsc{Adv}}(t,t) && \text{by line 7 of the algorithm and } \boldsymbol{x}_t \neq \boldsymbol{s}_t \\
&\leq \|\boldsymbol{x}_{t-1} - \boldsymbol{s}_{t-1}\| - \gamma \cdot \mathrm{C}_{\textsc{Adv}}(t,t) && \text{by line 6 of the algorithm} \qquad (21)
\end{aligned}$$

Then noting that $\boldsymbol{x}_t = \lambda \boldsymbol{s}_t + (1-\lambda)\tilde{\boldsymbol{x}}_t$ for some $\lambda \in [0,1]$, we have

$$\begin{aligned}
& f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \phi_t - \phi_{t-1} \\
\leq\ & \lambda f_t(\boldsymbol{s}_t) + (1-\lambda) f_t(\tilde{\boldsymbol{x}}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| - c\gamma \cdot \mathrm{C}_{\textsc{Adv}}(t,t) && (22) \\
\leq\ & f_t(\boldsymbol{s}_t) + f_t(\tilde{\boldsymbol{x}}_t) + \|\boldsymbol{x}_t - \boldsymbol{z}_t\| + \|\boldsymbol{z}_t - \boldsymbol{y}_t\| + \|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| - c\gamma \cdot \mathrm{C}_{\textsc{Adv}}(t,t) && (23) \\
\leq\ & \mathrm{C}_{\textsc{Rob}}(t,t) + f_t(\tilde{\boldsymbol{x}}_t) + \gamma \cdot \mathrm{C}_{\textsc{Adv}}(t,t) + \|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| - c\gamma \cdot \mathrm{C}_{\textsc{Adv}}(t,t) && (24) \\
\leq\ & \mathrm{C}_{\textsc{Rob}}(t,t) + (k(\mathcal{X}) + \gamma - c\gamma)\mathrm{C}_{\textsc{Adv}}(t,t) && (25) \\
\leq\ & \mathrm{C}_{\textsc{Rob}}(t,t) && (26)
\end{aligned}$$

where (22) follows from convexity and (21), (23) follows from the triangle inequality, (24) follows from bounding $\|x_t - z_t\|$ via Corollary 27 and $\|z_t - y_t\|$ via algorithm line (7), (25) follows from the observation in Corollary 28, and (26) holds so long as $c \geq 1 + \frac{k(\mathcal{X})}{\gamma}$.

Thus we set $c = 1 + \frac{k(\mathcal{X})}{\gamma}$; summing (20) and (26) over times $t'+1, \ldots, T$ and adding to (19), we obtain

$$\begin{aligned}
\mathrm{C}_{\textsc{Interp}}(1,T) &\leq \left(1 + \frac{k(\mathcal{X})}{\gamma} + \frac{\mu(\mathcal{X}) + \epsilon + 1 + \frac{k(\mathcal{X})}{\gamma}}{\delta}\right) \mathrm{C}_{\textsc{Rob}}(1,t') + \mathrm{C}_{\textsc{Rob}}(t'+1,T) \\
&\leq \left(1 + \frac{k(\mathcal{X})}{\gamma} + \frac{\mu(\mathcal{X}) + \epsilon + 1 + \frac{k(\mathcal{X})}{\gamma}}{\delta}\right) \mathrm{C}_{\textsc{Rob}}(1,T).
\end{aligned}$$

■

D.3. Parameter optimization

We conclude with a brief comment on the optimal selection of parameters γ, δ for INTERP. If we minimize the competitive bound of INTERP with respect to ROB over parameters $\gamma, \delta > 0$ satisfying $2\gamma + 2\delta = \epsilon$, then we obtain the following $\mathcal{O}(\frac{1}{\epsilon^2})$ bound on the competitive ratio with respect to ROB (with arguments of $\mu(\mathcal{X}), k(\mathcal{X})$ suppressed):

$$3 + \frac{2(\epsilon + k(\mathcal{X})(4+\epsilon) + \epsilon\mu(\mathcal{X})) + 4\sqrt{k(\mathcal{X})(2+\epsilon)(2k(\mathcal{X}) + \epsilon(1 + \epsilon + \mu(\mathcal{X})))}}{\epsilon^2}$$

which is obtained by setting

$$\gamma = \frac{\sqrt{k(\mathcal{X})(2+\epsilon)(2k(\mathcal{X}) + \epsilon(1+\epsilon+\mu(\mathcal{X})))} - k(\mathcal{X})(2+\epsilon)}{2(1 - k(\mathcal{X}) + \epsilon + \mu(\mathcal{X}))}$$

and

$$\delta = \frac{\epsilon}{2} - \gamma.$$

With parameters chosen optimally thus, INTERP is $(\mu(\mathcal{X}) + \epsilon, \mathcal{O}(\epsilon^{-2}))$-bicompetitive. Moreover, even if $\mu(\mathcal{X})$ and $k(\mathcal{X})$ are not known exactly, simply setting $\gamma = \delta = \frac{\epsilon}{4}$ gives an (up to a constant factor) identical $(\mu(\mathcal{X}) + \epsilon, \mathcal{O}(\epsilon^{-2}))$-bicompetitiveness.

Appendix E. Proof of Theorem 12

We prove Theorem 12 in two parts: we first prove the competitive ratio of BDINTERP with respect to ADV, and then we prove the competitive ratio with respect to ROB. The robustness and consistency claim then follows immediately from the bicompetitive bound and the observation in Appendix A.2.

Proof of competitiveness with respect to ADV. We define "phases" of the algorithm as follows: if $\boldsymbol{x}_t$ is determined by line 4 of the algorithm, then the advice is in the "ADV" phase. Otherwise, if $\boldsymbol{x}_t$ is determined by line 8, then the advice is in the "ROB" phase. We refer to the time indices in which the algorithm is in the "ROB" phase as $R_1, \ldots, R_k \in [T]$ (where $k \leq T$, and $R_1 < \cdots < R_k$ are in increasing order). If the algorithm is never in the "ROB" phase, then $\boldsymbol{x}_t = \text{ADV}_t\ \forall t \in [T]$, and thus BDINTERP is 1-competitive with respect to ADV. Thus we restrict to the case that there is at least one time index in which the algorithm is in the "ROB" phase. By design, for each $j \in [k]$, $\mathrm{C}_{\text{ROB}}(1, R_j) \leq \delta \cdot \mathrm{C}_{\text{ADV}}(1, R_j)$.

Now we break into two cases depending on the phase. First, suppose the BDINTERP is in the "ADV" phase. This means that $\boldsymbol{x}_t = \tilde{\boldsymbol{x}}_t$. Then

$$\begin{aligned} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| &= f_t(\tilde{\boldsymbol{x}}_t) + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_{t-1}\| \\ &\leq f_t(\tilde{\boldsymbol{x}}_t) + \|\tilde{\boldsymbol{x}}_t - \tilde{\boldsymbol{x}}_{t-1}\| + \|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\| \end{aligned} \tag{27}$$

follows immediately from the triangle inequality.

Second, consider the case that the algorithm is in the "ROB" phase. This means that $\boldsymbol{x}_t$ is determined by line 8 of the algorithm; and there exists some $\lambda \in [0, 1]$ for which $\boldsymbol{x}_t = \lambda \boldsymbol{s}_t + (1 - \lambda)\tilde{\boldsymbol{x}}_t$. In this case, observe

$$\begin{aligned} & f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| \\ \leq\ & \lambda f_t(\boldsymbol{s}_t) + (1 - \lambda) f_t(\tilde{\boldsymbol{x}}_t) + 2\|\boldsymbol{x}_t - \boldsymbol{y}_t\| + \|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{y}_t\| && (28) \\ \leq\ & f_t(\boldsymbol{s}_t) + f_t(\tilde{\boldsymbol{x}}_t) + 2\gamma \cdot \mathrm{C}_{\text{ADV}}(t, t) + \|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{y}_t\| && (29) \end{aligned}$$

where (28) follows from convexity of f_t and the triangle inequality, (29) follows via algorithm line 8. Observing that $\boldsymbol{x}_{t-1} = \nu \tilde{\boldsymbol{x}}_{t-1} + (1 - \nu)\boldsymbol{s}_{t-1}$, we can use the triangle inequality to obtain

$$\|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| \leq \nu \|\tilde{\boldsymbol{x}}_t - \tilde{\boldsymbol{x}}_{t-1}\| + (1 - \nu)\|\boldsymbol{s}_t - \boldsymbol{s}_{t-1}\|. \tag{30}$$

Moreover, observe

$$\begin{aligned} \|\tilde{\boldsymbol{x}}_t - \boldsymbol{y}_t\| &= (1 - \nu)\|\tilde{\boldsymbol{x}}_t - \boldsymbol{s}_t\| \\ &\leq (1 - \nu)(\|\tilde{\boldsymbol{x}}_t - \tilde{\boldsymbol{x}}_{t-1}\| + \|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{s}_{t-1}\| + \|\boldsymbol{s}_t - \boldsymbol{s}_{t-1}\|) \\ &= (1 - \nu)(\|\tilde{\boldsymbol{x}}_t - \tilde{\boldsymbol{x}}_{t-1}\| + \|\boldsymbol{s}_t - \boldsymbol{s}_{t-1}\|) + \|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\| \end{aligned} \tag{31}$$

where the final equality follows by definition of ν. Applying (30) and (31) to (29), we obtain

$$\begin{aligned} & f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| \\ \leq\ & 2 \cdot \mathrm{C}_{\text{ROB}}(t, t) + (1 + 2\gamma)\mathrm{C}_{\text{ADV}}(t, t) + \|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\| \end{aligned} \tag{32}$$

Summing (27) and (32) over time and noting that the left-hand side $\|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\|$ and right-hand side $\|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{x}_{t-1}\|$ telescope, we obtain

$$\begin{aligned}
&\mathrm{C}_{\text{BDINTERP}}(1,T) \\
\leq\ & \sum_{t=1}^{T} f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \|\tilde{\boldsymbol{x}}_T - \boldsymbol{x}_T\| \\
\leq\ & \sum_{t\in\{R_j\}_{j=1}^k} 2\cdot \mathrm{C}_{\text{ROB}}(t,t) + (1+2\gamma)\mathrm{C}_{\text{ADV}}(t,t) + \sum_{t\in[T]\setminus\{R_j\}_{j=1}^k} \mathrm{C}_{\text{ADV}}(t,t) \\
& \leq 2\cdot \mathrm{C}_{\text{ROB}}(1,R_k) + (1+2\gamma)\mathrm{C}_{\text{ADV}}(1,T) \\
& \leq 2\delta\cdot \mathrm{C}_{\text{ADV}}(1,R_k) + (1+2\gamma)\mathrm{C}_{\text{ADV}}(1,T) \\
& \leq (1+\epsilon)\mathrm{C}_{\text{ADV}}(1,T)
\end{aligned}$$

where the second to last inequality follows from the assumption that the algorithm is in the "ROB" phase at time R_k, implying $\mathrm{C}_{\text{ROB}}(1,R_k) \leq \delta\cdot\mathrm{C}_{\text{ADV}}(1,R_k)$; and in the last inequality we use the assumption on the parameters that $2\gamma + 2\delta = \epsilon$. This gives the competitive bound with respect to ADV. Note that we can repeat the same argument with truncated time horizon to obtain that BDINTERP is $(1+\epsilon)$-competitive with respect to ADV at every timestep. ■

Proof of competitiveness with respect to ROB. Define the potential function $\phi_t = c\frac{\|\boldsymbol{x}_t - \boldsymbol{s}_t\|}{\|\tilde{\boldsymbol{x}}_t - \boldsymbol{s}_t\|}$, with $c > 0$ to be determined later (we set $\phi_t := 0$ in the case that $\tilde{\boldsymbol{x}}_t = \boldsymbol{s}_t$).

Let $t' \in \{0,\ldots,T\}$ be the last time interval in which the algorithm's decision is determined by line 4 of the algorithm, or equivalently, the greatest t such that $\mathrm{C}_{\text{ROB}}(1,t) \geq \delta\cdot\mathrm{C}_{\text{ADV}}(1,t)$. Applying the competitive bound of BDINTERP with respect to ADV to the subhorizon $t = 1,\ldots,t'$, we have $\mathrm{C}_{\text{BDINTERP}}(1,t') \leq (1+\epsilon)\mathrm{C}_{\text{ADV}}(1,t')$. Thereby we obtain

$$\begin{aligned}
\mathrm{C}_{\text{BDINTERP}}(1,t') + \phi_{t'} &\leq (1+\epsilon)\mathrm{C}_{\text{ADV}}(1,t') + c \\
&\leq \frac{1+\epsilon}{\delta}\mathrm{C}_{\text{ROB}}(1,t') + c. \qquad (33)
\end{aligned}$$

where in the first inequality we have used the fact that $\|\boldsymbol{x}_t - \boldsymbol{s}_t\| \leq \|\tilde{\boldsymbol{x}}_t - \boldsymbol{s}_t\|$ for all t.

Now consider arbitrary $t \in \{t'+1,\ldots,T\}$. We distinguish two cases. First, suppose $\boldsymbol{x}_t = \boldsymbol{s}_t$ and $\tilde{\boldsymbol{x}}_{t-1} \neq \boldsymbol{s}_{t-1}$. Then

$$\begin{aligned}
f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \phi_t - \phi_{t-1} &= f_t(\boldsymbol{s}_t) + \|\boldsymbol{s}_t - \boldsymbol{x}_{t-1}\| - c\frac{\|\boldsymbol{x}_{t-1} - \boldsymbol{s}_{t-1}\|}{\|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{s}_{t-1}\|} \\
&\leq f_t(\boldsymbol{s}_t) + \|\boldsymbol{s}_t - \boldsymbol{s}_{t-1}\| + \|\boldsymbol{x}_{t-1} - \boldsymbol{s}_{t-1}\| - c\frac{\|\boldsymbol{x}_{t-1} - \boldsymbol{s}_{t-1}\|}{\|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{s}_{t-1}\|} \\
&\leq \mathrm{C}_{\text{ROB}}(t,t) \qquad (34)
\end{aligned}$$

where the final inequality holds so long as $c \geq D$, by D-boundedness of ADV and ROB. Clearly (34) will also hold in the case that $\tilde{\boldsymbol{x}}_{t-1} = \boldsymbol{s}_{t-1}$, since this will imply $\boldsymbol{x}_{t-1} = \boldsymbol{s}_{t-1}$.

On the other hand, suppose $\boldsymbol{x}_t \neq \boldsymbol{s}_t$. Thus we can assume that $\tilde{\boldsymbol{x}}_t \neq \boldsymbol{s}_t$ and $\tilde{\boldsymbol{x}}_{t-1} \neq \boldsymbol{s}_{t-1}$. First, note that

$$\begin{aligned}
\frac{\|\boldsymbol{x}_t - \boldsymbol{s}_t\|}{\|\tilde{\boldsymbol{x}}_t - \boldsymbol{s}_t\|} &= \frac{\|\boldsymbol{y}_t - \boldsymbol{s}_t\| - \gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t,t)}{\|\tilde{\boldsymbol{x}}_t - \boldsymbol{s}_t\|} \qquad (35)\\
&= \nu - \frac{\gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t,t)}{\|\tilde{\boldsymbol{x}}_t - \boldsymbol{s}_t\|}\\
&\leq \frac{\|\boldsymbol{x}_{t-1} - \boldsymbol{s}_{t-1}\|}{\|\tilde{\boldsymbol{x}}_{t-1} - \boldsymbol{s}_{t-1}\|} - \frac{\gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t,t)}{D} \qquad (36)
\end{aligned}$$

where (35) follows from line 8 of the algorithm and $\boldsymbol{x}_t \neq \boldsymbol{s}_t$, and (36) follows by definition of ν and the D-boundedness of ADV, ROB.

Then noting that by convexity, $\boldsymbol{x}_t = \lambda \boldsymbol{s}_t + (1-\lambda)\tilde{\boldsymbol{x}}_t$ for some $\lambda \in [0,1]$, we have

$$\begin{aligned}
& f_t(\boldsymbol{x}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| + \phi_t - \phi_{t-1}\\
\leq\;& \lambda f_t(\boldsymbol{s}_t) + (1-\lambda) f_t(\tilde{\boldsymbol{x}}_t) + \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}\| - c\frac{\gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t,t)}{D} \qquad (37)\\
\leq\;& f_t(\boldsymbol{s}_t) + f_t(\tilde{\boldsymbol{x}}_t) + \|\boldsymbol{x}_t - \boldsymbol{y}_t\| + \|\boldsymbol{y}_t - \boldsymbol{x}_{t-1}\| - c\frac{\gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t,t)}{D} \qquad (38)\\
\leq\;& f_t(\boldsymbol{s}_t) + f_t(\tilde{\boldsymbol{x}}_t) + \gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t,t) + \nu\|\tilde{\boldsymbol{x}}_t - \tilde{\boldsymbol{x}}_{t-1}\| + (1-\nu)\|\boldsymbol{s}_t - \boldsymbol{s}_{t-1}\| - c\frac{\gamma \cdot \mathrm{C}_{\mathrm{ADV}}(t,t)}{D} \qquad (39)\\
\leq\;& \mathrm{C}_{\mathrm{ROB}}(t,t) + \left(1 + \gamma - \frac{c\gamma}{D}\right)\mathrm{C}_{\mathrm{ADV}}(t,t)\\
\leq\;& \mathrm{C}_{\mathrm{ROB}}(t,t) \qquad (40)
\end{aligned}$$

where (37) follows from convexity and (36), (38) follows from the triangle inequality, and (39) follows from (30) and line 8 of the algorithm. The final inequality (40) holds as long as $c \geq D + \frac{D}{\gamma}$.

Thus we set $c = D + \frac{D}{\gamma}$; summing (34) and (40) over times $t'+1, \ldots, T$ and adding to (33), we obtain

$$\begin{aligned}
\mathrm{C}_{\mathrm{INTERP}}(1,T) &\leq \frac{1+\epsilon}{\delta}\mathrm{C}_{\mathrm{ROB}}(1,t') + D + \frac{D}{\gamma} + \mathrm{C}_{\mathrm{ROB}}(t'+1,T)\\
&\leq \frac{1+\epsilon}{\delta}\mathrm{C}_{\mathrm{ROB}}(1,T) + D + \frac{D}{\gamma}\\
&\leq \left(D + \frac{D}{\gamma} + \frac{1+\epsilon}{\delta}\right)\mathrm{C}_{\mathrm{ROB}}(1,T)
\end{aligned}$$

where in the final inequality we have used the assumption that $\mathrm{C}_{\mathrm{ROB}} \geq 1$. ■

E.1. Parameter optimization

To conclude, we briefly comment on the optimal selection of parameters γ, δ for BDINTERP. Optimizing the competitive bound of BDINTERP with respect to ROB over those $\gamma, \delta > 0$ satisfying $2\gamma + 2\delta = \epsilon$, we obtain the following $\mathcal{O}(\frac{D}{\epsilon})$-competitive bound with respect to ROB:

$$2 + D + \frac{2(1+D) + 4\sqrt{D(1+\epsilon)}}{\epsilon}$$

which is obtained by setting

$$\gamma = \frac{D\epsilon}{2(D + \sqrt{D(1+\epsilon)}}$$

and

$$\delta = \frac{\epsilon}{2} - \gamma.$$

With parameters chosen optimally thus, BDINTERP is $(1+\epsilon, \mathcal{O}(D\epsilon^{-1}))$-bicompetitive. Moreover, even if D is not known exactly *a priori*, simply setting $\gamma = \delta = \frac{\epsilon}{4}$ gives an (up to a constant factor) identical $(1+\epsilon, \mathcal{O}(D\epsilon^{-1}))$-bicompetitiveness.

Appendix F. Robustness and consistency corollaries of Theorems 11 and 12

In this section, we detail the upper bounds on robustness and consistency resulting from Theorems 11 and 12 on CFC and each of its subclasses defined in Appendix A.1. Each of these corollaries follows immediately upon instantiating the robust algorithm ROB provided as input to INTERP (Algorithm 2) or BDINTERP (Algorithm 3) with a competitive algorithm whose competitive ratio is listed in Table 2. We begin with the corollaries of Theorem 11.

Corollary 30

(i) INTERP *(Algorithm 2) with* ROB *chosen as the functional Steiner point algorithm (Sellke (2020)) is $(\mu(\mathbb{R}^d, \|\cdot\|)+\epsilon)$-consistent and $\mathcal{O}(\frac{d}{\epsilon^2})$-robust for* CFC *and* CBC *on $\mathbb{R}^d$ with any norm.*

(ii) INTERP *(Algorithm 2) with* ROB *chosen as the low-dimensional chasing algorithm of Argue et al. (2020) is $(\sqrt{2}+\epsilon)$-consistent and $\mathcal{O}(\frac{k}{\epsilon^2})$-robust for k*CBC *on $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$.*

(iii) INTERP *(Algorithm 2) with* ROB *chosen as the greedy algorithm (Zhang et al. (2021)) is $(\mu(\mathcal{X})+\epsilon)$-consistent and $\mathcal{O}(\frac{1}{\alpha\epsilon^2})$-robust for α*CFC *on any normed vector space $\mathcal{X}$.*

(iv) INTERP *(Algorithm 2) with* ROB *chosen as the greedy OBD algorithm (Lin (2022)) is $(\sqrt{2}+\epsilon)$-consistent and $\mathcal{O}(\frac{1}{\alpha^{1/2}\epsilon^2})$-robust for α*CFC *on $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$.*

(v) INTERP *(Algorithm 2) with* ROB *chosen as the Move towards Minimizer algorithm (Argue et al. (2020)) is $(\sqrt{2}+\epsilon)$-consistent and $\mathcal{O}(\frac{2^{\gamma/2}\kappa}{\epsilon^2})$-robust for (κ,γ)*CFC *on $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$.*

In particular, each of the consistency bounds is $\sqrt{2}+\epsilon$ in the case that the decision space is Hilbert.

We now present the corollaries of Theorem 12.

Corollary 31 *In each of the following, suppose that* (ADV, ROB) *are D-bounded and* $\mathrm{C}_{\mathrm{ROB}} \geq 1$.

(i) BDINTERP *(Algorithm 3) with* ROB *chosen as the functional Steiner point algorithm (Sellke (2020)) is $(1+\epsilon)$-consistent and $\mathcal{O}(\frac{dD}{\epsilon})$-robust for* CFC *and* CBC *on $\mathbb{R}^d$ with any norm.*

(ii) BDINTERP *(Algorithm 3) with* ROB *chosen as the low-dimensional chasing algorithm of Argue et al. (2020) is $(1+\epsilon)$-consistent and $\mathcal{O}(\frac{kD}{\epsilon})$-robust for k*CBC *on $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$.*

(iii) BDINTERP *(Algorithm 3) with* ROB *chosen as the greedy algorithm (Zhang et al. (2021)) is $(1+\epsilon)$-consistent and $\mathcal{O}(\frac{D}{\alpha\epsilon})$-robust for α*CFC *on any normed vector space.*

(iv) BDINTERP *(Algorithm 3) with* ROB *chosen as the Greedy OBD algorithm (Lin (2022)) is* $(1+\epsilon)$*-consistent and* $\mathcal{O}(\frac{D}{\alpha^{1/2}\epsilon})$*-robust for* $\alpha\mathsf{CFC}$ *on* $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$.

(v) BDINTERP *(Algorithm 3) with* ROB *chosen as the Move towards Minimizer algorithm (Argue et al. (2020)) is* $(1+\epsilon)$*-consistent and* $\mathcal{O}(\frac{2^{\gamma/2}\kappa D}{\epsilon})$*-robust for* $(\kappa,\gamma)\mathsf{CFC}$ *on* $(\mathbb{R}^d, \|\cdot\|_{\ell^2})$.