

Mechanism Design with Predictions*

Chenyang Xu¹, Pinyan Lu^{2,3,†}

¹College of Computer Science, Zhejiang University

²ITCS, Shanghai University of Finance and Economics

³Huawei TCS Lab

xcy1995@zju.edu.cn, lu.pinyan@mail.shufe.edu.cn

Abstract

Improving algorithms via predictions is a very active research topic in recent years. This paper initiates the systematic study of mechanism design in this model. In a number of well-studied mechanism design settings, we make use of imperfect predictions to design mechanisms that perform much better than traditional mechanisms if the predictions are accurate (consistency), while always retaining worst-case guarantees even with very imprecise predictions (robustness). Furthermore, we refer to the largest prediction error sufficient to give a good performance as the error tolerance of a mechanism, and observe that an intrinsic tradeoff among consistency, robustness and error tolerance is common for mechanism design with predictions.

1 Introduction

A recently popular trend in algorithmic design is augmenting (online) algorithms with imperfect predictions. This line of work suggests that there exists an opportunity to bypass the worst-case lower bounds of online problems, which are caused by the uncertainty of the future. In this setting, the algorithm is given access to error-prone predictions, and its performance is bounded in terms of the quality of the predictions. The algorithm should perform better than the worst-case bound with accurate predictions, and never perform much worse than the best pure online algorithm even if the prediction error is large. Many classic online problems have been considered in the context, such as ski rental [Purohit *et al.*, 2018; Gollapudi and Panigrahi, 2019; Anand *et al.*, 2020], caching [Lykouris and Vassilvitskii, 2018; Rohatgi, 2020], and scheduling [Purohit *et al.*, 2018; Lattanzi *et al.*, 2020; Im *et al.*, 2021; Li and Xian, 2021].

Mechanism design and online algorithm design share some similarities. Both of them need to deal with missing information. Due to the uncertainty about the future or the agents' private information, algorithms (mechanisms) have to be overly

cautious and thereby worst-case bounds arise. Thus, it is interesting to investigate if predictions can help with mechanism design. This paper proposes a study of mechanism design with predictions. For generality and simplicity, we directly use agents' private information as predictions. This is justifiable in many applications. For example, in repeated auctions, we may use historical bidding records as the predictions. We note that there are several related works [Medina and Vassilvitskii, 2017; Antoniadis *et al.*, 2020]. In [Medina and Vassilvitskii, 2017], repeated posted price auctions were considered. In [Antoniadis *et al.*, 2020], the authors mainly focused on developing an online bipartite matching algorithm with predictions and noticed that the algorithm can be converted to a truthful mechanism. In this paper, we consider mechanism design with predictions more systematically and investigate a number of different mechanism design settings.

1.1 Challenge

We follow the terminology in [Purohit *et al.*, 2018] which is now standard: say a mechanism's *consistency* and *robustness* are its approximation ratios for accurate predictions and for arbitrarily inaccurate predictions respectively. Due to the truthfulness requirement in mechanism design, it is subtle to use predictions to get a good consistency and robustness.

A mechanism consists of two algorithms: the allocation algorithm and the payment algorithm. Due to truthfulness, the allocation algorithm needs to satisfy a certain monotone property, which is a global property of the algorithm. It is usually infeasible to change the allocation for certain inputs based on the predictions without changing the allocation for other inputs. For example, a widely-used approach to ensure robustness in online algorithm design is switching to pure online algorithms when the predictions are found to be unreliable. However, in mechanism design, switching to traditional mechanisms when the prediction error is large may hurt truthfulness. The agent who benefits more from the traditional mechanism may misreport the private information such that the prediction error looks large.

Thus, we need to design the allocation algorithm with predictions as a whole mapping to satisfy the monotone property. This gives a big challenge to maintain a good consistency and robustness. In other words, it is much less flexible to design a truthful mechanism than an (online) algorithm.

*This work was supported by Science and Technology Innovation 2030 – “The Next Generation of Artificial Intelligence” Major Project No.2018AAA0100900.

[†]The corresponding author.

1.2 Our Contributions

We study four very different and well-studied mechanism design problems and observe that predictions are indeed helpful.

Revenue-Maximizing Single-Item Auction

Maximizing the revenue is one of the most fundamental problems in auction design. There is a rich literature (e.g. [Myerson, 1981; Goldberg *et al.*, 2001; Azar *et al.*, 2013]). We consider the revenue-maximizing single-item auction and compare the revenue to the highest bid, the most ambitious benchmark. It is well-known that there is no good mechanism in worst-case mechanism design with respect to this goal. If we assume that all bids belong to $[1, h]$, no deterministic truthful mechanism has an approximation ratio better than h ([Goldberg *et al.*, 2001]).

With perfect predictions, it is trivial to achieve 1-approximation since we can simply run anonymous pricing scheme and set the price as the highest predicted value. However, this mechanism is very fragile. The approximation ratio drops to infinity even if there is only a tiny error in the predictions. This is of course highly undesirable. To address the issue, we propose a notion of *error tolerance* to measure how much prediction error the mechanism can tolerate to get a reasonable good approximation. Let $\eta \geq 1$ be the relative prediction error, where $\eta = 1$ means that there is no error. We give a deterministic truthful mechanism with γ -consistent and h -robust, where the robustness ratio matches the worst-case bound of h in the traditional setting and the consistency ratio $\gamma \geq 1$ is a parameter we can choose. The approximation ratio smoothly increases as a function of $\gamma\eta$ when $\eta \leq \gamma$, and then has a big drop after $\eta > \gamma$. Therefore, there is a tradeoff between the consistency and the error tolerance. Moreover, we prove that such a tradeoff is necessary and in some sense optimal for all deterministic truthful mechanisms. Our mechanism is simple and practical. It is the second price auction with individual reserve prices, where these reserves are set based on the predictions.

Frugal Path Auction

Path auction is a reverse auction, where the auctioneer needs to buy a path and pays the edges in the path. The goal here is to minimize the total payment and the benchmark is the second cheapest path. This is a classic problem in frugal mechanism design. The problem was coined by [Archer and Tardos, 2002]. They showed that the VCG mechanism obtains an approximation ratio of $\Theta(n)$, where n is the number of agents, and the ratio is the best possible [Elkind *et al.*, 2004]. We obtain a deterministic truthful mechanism with 2γ -consistent and (n^2/γ) -robust. In terms of error tolerance, the approximation ratio smoothly increases as a function of $\gamma(1 + \eta)$ as long as $\eta \leq \gamma$. Here we observe a three way tradeoff among consistency, robustness and error tolerance. The mechanism is the generalized VCG mechanism (a.k.a affine maximizer), where the weights for different agents are set based on the predictions.

Besides the VCG mechanism, [Karlin *et al.*, 2005] proposed $\sqrt{\cdot}$ -mechanism for frugal path auction. The approximation ratio of $\sqrt{\cdot}$ -mechanism is still $\Theta(n)$, but it can outperform the VCG mechanism in some graphs. Later, this technique

was generalized to more problems in frugal mechanism design [Chen *et al.*, 2010]. We can also apply our technique to $\sqrt{\cdot}$ -mechanism to get a similar improvement when predictions are given.

Truthful Job Scheduling

Truthful mechanism for scheduling unrelated machines is the center problem of the very first algorithmic game theory paper [Nisan and Ronen, 2001], whose goal is to minimize the makespan. This problem is very different from the previous two settings in two aspects: it is a multidimensional mechanism design problem and the objective is not related to the payment. In [Nisan and Ronen, 2001], the authors showed that the VCG mechanism gives an approximation ratio of m , where m is the number of machines, and proved a lower bound of 2 for any deterministic truthful mechanism. They conjectured that no deterministic truthful mechanism has an approximation ratio better than m . Many papers worked on closing the gap [Christodoulou *et al.*, 2007; Koutsoupias and Vidali, 2007; Ashlagi *et al.*, 2009; Giannakopoulos *et al.*, 2020; Dobzinski and Shaulker, 2020; Christodoulou *et al.*, 2020]. The closest result so far proves a lower bound of $\Omega(\sqrt{m})$ [Christodoulou *et al.*, 2020].

For this problem, we give a deterministic truthful mechanism with approximation ratio of $O(\min\{\gamma\eta^2, \frac{m^3}{\gamma^2}\})$, where again γ is the consistency parameter we can choose and η is the prediction error. Here, we compute an (approximate) optimal allocation based on the predicted information and use that allocation as a guide for the mechanism.

Two-Facility Game on a Line

Finally, we consider a mechanism without money: two-facility game on a line. This setting was coined by [Procaccia and Tennenholtz, 2009], where the authors gave an upper bound of $n - 2$ and a lower bound of 1.5 for deterministic truthful mechanisms. The lower bound was later improved to 2 [Lu *et al.*, 2009] and $(n - 1)/2$ [Lu *et al.*, 2010]. Finally, [Fotakis and Tzamos, 2014] showed a tight lower bound of $n - 2$.

Since the space of truthful mechanism without money is more restricted, it is even more difficult to make use of predictions here. We get a deterministic truthful mechanism with $(1 + n/2)$ -consistent and $(2n - 1)$ -robust, whose consistency ratio is slightly better than the best known mechanism. Whether there is a mechanism with $o(n)$ -consistent and a bounded robustness is a very interesting open question.

2 Preliminaries

This section introduces the terminology necessary to understand the paper. An expert can skip it directly. We take the single-item sealed-bid auction for an example. In the auction, there is a seller that has a single good and several bidders who are interested in buying the good. Each bidder i has a private value v_i representing the maximum willingness-to-pay for the item. Each bidder i privately tells the auctioneer a bid b_i , while the auctioneer decides who is the winner (i.e. *the allocation rule*) and how much he needs to pay (i.e. *the payment rule*). Say the bidder who sets $b_i = v_i$ is a truthtelling

bidder. If a bidder is the winner, his utility is $v_i - p_i$, where p_i is the price he needs to pay. Otherwise, the utility is 0.

Definition 1. ([Roughgarden and Iwama, 2017]) *A mechanism is truthful if for any bidder i , setting $b_i = v_i$ always maximizes his utility regardless of other bidders' bids, and the utility of any truth-telling bidder is non-negative.*

Now we state a widely-used theorem proposed by [Myerson, 1981].

Definition 2. (Monotonicity) *An allocation rule is monotone if the winner still wins when he increases the bid unilaterally.*

Theorem 1. (Myerson's Lemma [Myerson, 1981]) *For a single-parameter environment, the mechanism is truthful only if its allocation rule is monotone, while for any monotone allocation rule, there exists a unique payment rule which makes the mechanism truthful. Moreover, such a payment rule can be given by an explicit formula.*

For single-item auctions, the unique payment is the winner's threshold bid.

Definition 3. (Threshold Bid) *Given a single-item auction mechanism and the bid of each bidder, the threshold bid of a bidder is the minimum bid that he could make and win the auction when all other bidders fix their bids.*

3 Revenue-Maximizing Single-Item Auction

In this section, we consider revenue-maximizing single-item auction. There is one item and $n > 1$ bidders. Each bidder i has a private value $v_i^* \in [1, h]$ and reports a bid $b_i \in [1, h]$. Use a n -dimensional vector X to denote the allocation of the item, where each entry x_i is either 1 or 0 and it equals 1 only if bidder i wins. Each bidder i has an utility $u_i = x_i \cdot v_i^* - p_i$, where p_i is the payment. The auctioneer aims to design a truthful mechanism that maximizes the selling price, and the benchmark is the highest private value. In mechanism design with predictions, we are given access to the predictions of bidders' private values, denoted by $\hat{V} = \{\hat{v}_i\}_{i \in [n]}$. The predictions are erroneous and we define a natural prediction error $\eta := \max_{i \in [n]} \left\{ \frac{v_i^*}{\hat{v}_i}, \frac{\hat{v}_i}{v_i^*} \right\}$.

Theorem 2. *There exists a deterministic truthful mechanism parameterized by $\gamma \geq 1$ with approximation ratio at most $\min\{f(\gamma), h\}$ where*

$$f(\gamma) := \begin{cases} \gamma\eta & \eta \leq \gamma \\ \max\{\gamma^2\eta^2, \frac{h\eta}{\gamma^2}\} & \eta > \gamma. \end{cases}$$

From Theorem 2, the claimed mechanism has a robustness ratio h , which is the best possible in the traditional setting. The consistency ratio of the mechanism is determined by a parameter γ . Notice that the function $f(\gamma)$ is piecewise and has a "jump" when $\eta = \gamma$. We refer to the length of the first piece in $f(\gamma)$ as the error tolerance. Later in this section, we will prove that this jump point is unavoidable for any deterministic truthful mechanism. In order to communicate our main ideas more clearly, this section simplifies the algorithm and the analysis by assuming that the number of bidders is at least 3. In this paper's full version, we show that removing this assumption still gets the claimed approximation ratio.

Algorithm 1 Single-Item Auction with Predictions

Input: The predicted private values $\hat{V} = \{\hat{v}_i\}_{i \in [n]}$, the bids $B = \{b_i\}_{i \in [n]}$ and a parameter $\gamma \geq 1$.

Output: The winner and the payment.

- 1: Reindex the bidders in the decreasing order of their predicted values.
 - 2: Assign a bar $br(i)$ to each bidder i . Initially, $br(i) \leftarrow 1 \forall i \in [n]$.
 - 3: **if** $\forall i \in [n], \hat{v}_i \geq \hat{v}_1/\gamma^2$ **then**
 - 4: Update the bar of bidder 1: $br(1) \leftarrow \max\{\hat{v}_1/\gamma, 1\}$.
 - 5: **else**
 - 6: Update the bar of bidder 1: $br(1) \leftarrow \max\{\hat{v}_1/\gamma, 1\}$.
 - 7: For each bidder i with $\hat{v}_i \geq \hat{v}_1/\gamma^2$, update the bar: $br(i) \leftarrow \max\{\hat{v}_1/\gamma^2, 1\}$.
 - 8: **end if**
 - 9: Define a bidder set $S := \{i \in [n] | b_i \geq br(i)\}$.
 - 10: **return** the bidder $j \in S$ with the highest bid and the threshold bid $\theta(j)$.
-

3.1 Mechanism

We start by giving some intuitions. For simplicity of notation, reindex the bidders in the decreasing order of their predicted values. So now bidder 1 has the largest predicted value \hat{v}_1 . Notice that if the predictions are error-free, letting bidder 1 win and charging him \hat{v}_1 gives the optimal solution. Inspired by this, a natural idea is the following. If the reported bid b_1 is no less than the predicted value \hat{v}_1 , let bidder 1 win. Otherwise, ignore bidder 1 and let the remaining bidder with the highest bid win.

Clearly, this allocation rule is monotone. Then due to Theorem 1, we can ensure the truthfulness by charging the winner the threshold bid. More precisely, if bidder 1 wins, the payment is \hat{v}_1 , while if bidder i ($\neq 1$) wins, the payment is the second highest remaining bid. The mechanism has a consistency ratio of 1, meaning that it gets the optimal value \hat{v}_1 when $\eta = 1$. However, the approximation ratio will increase to h immediately even if η is only slightly larger than 1. Thus, although the robustness ratio of the mechanism is h which is the best possible in the traditional setting, we cannot refer to it as an ideal mechanism due to the tiny error tolerance.

One way to get a mechanism with a larger error tolerance is setting a lower bar for bidder 1. For example, let bidder 1 win if b_1 is at least \hat{v}_1/γ , where γ is a parameter ≥ 1 . The mechanism will have a comparable performance when $\eta \leq \gamma$, because bidder 1 will always win and the threshold bid is at least \hat{v}_1/γ . We build on this idea to give our mechanism. Moreover, we refine the way we handle other bidders such that the approximation ratio will not increase to h immediately once the prediction error becomes larger than γ . The mechanism is described in Algorithm 1.

3.2 Analysis

Observing that in Algorithm 1, increasing any bidder's bid will only improve the chance that he wins, the allocation rule is monotone. Hence, we have the following lemma.

Lemma 4. *Algorithm 1 is a truthful mechanism.*

Now we focus on the approximation ratio analysis. We first give a simple proof of the robustness ratio h , and then show that the approximation ratio is at most $f(\eta)$.

Lemma 5. *Algorithm 1 has an approximation ratio at most h .*

Proof. The upshot is that for any bidder i , $\theta(i) \geq br(i)$, because if $b_i < br(i)$, bidder i will not be included in S and lose the auction. Since $br(i) \geq 1$ for any bidder i , the winner's payment is at least 1. Note that h is the upper bound of all bids. We have the optimal selling price $\text{OPT} \leq h$, implying that the approximation ratio is at most h . \square

We distinguish two cases to analyze the other bound: (1) $\eta \leq \gamma$ and (2) $\eta > \gamma$. The main difference of the two cases is whether we can ensure that bidder 1 is in S or not.

Lemma 6. *The approximation ratio of Algorithm 1 is at most $\gamma\eta$ if $\eta \leq \gamma$.*

Proof. In this case, we can claim that bidder 1 is always in S , i.e., $b_1 \geq br(1)$. If $br(1) = 1$, b_1 is always at least $br(1)$. Otherwise, $br(1) = \hat{v}_1/\gamma$. Since $v_1^* \geq \hat{v}_1/\eta$ and $\eta \leq \gamma$, we still have $b_1 \geq br(1)$.

Suppose that the winner is bidder j and the optimal payment is the private value v_k^* of bidder k . If $j = 1$, the payment is $\theta(1) \geq br(1)$. Otherwise, the threshold bid of bidder j is at least $b_1 \geq br(1)$. Observing that regardless of which case, the payment is at least $br(1)$, we can bound the payment PAY as follows:

$$\text{PAY} \geq br(1) \geq \frac{\hat{v}_1}{\gamma} \geq \frac{\hat{v}_k}{\gamma} \geq \frac{v_k^*}{\gamma\eta} = \frac{\text{OPT}}{\gamma\eta},$$

completing the proof. \square

Lemma 7. *The approximation ratio of Algorithm 1 is at most $\max\{\gamma^2\eta^2, \frac{h\eta}{\gamma^2}\}$ if $\eta > \gamma$.*

Proof Sketch. In this case, we cannot ensure that $b_1 \geq br(1)$. Notice that if bidder 1 is still included in S , following the same analysis in Lemma 6 gives the ratio $\gamma\eta$. So we only need to focus on the case that $b_1 < br(1)$. We further distinguish two subcases: (1) $\forall i \in [n]$, $\hat{v}_i \geq \hat{v}_1/\gamma^2$, and (2) $\exists i \in [n]$, $\hat{v}_i < \hat{v}_1/\gamma^2$. Basically, the two terms in the target approximation ratio come from these two subcases respectively. For the first subcase, we observe that the ratio between any two bidders' predicted values is at least $1/\gamma^2$. This implies that for any two bidders, the ratio between their real values is at least $1/(\gamma^2\eta^2)$. Thus, using the second price mechanism should return a payment at least $\text{OPT}/(\gamma^2\eta^2)$. For the second subcase, the payment could drop to 1. But here, we can show that the optimal payment is at most $h\eta/\gamma^2$, which proves the claimed ratio. Due to space, the detailed proof is omitted. \square

Combining the above lemmas, we can prove Theorem 2.

3.3 Lower Bounds

This subsection gives two lower bounds of the auction problem. We only state the theorems here and provide the proofs in the full version of this paper.

Theorem 3. *For any deterministic truthful mechanism with a consistency ratio γ , the approximation ratio is at least h/η when $\eta > \gamma$.*

For any $\gamma \leq h^{1/5}$, when η is slightly larger than γ , our ratio $f(\eta)$ is close to h/γ , which matches the lower bound given by Theorem 3. To show the optimality of the mechanism's performance when $\eta \leq \gamma$, we assume that η is sampled uniformly from $[1, \gamma]$ and prove a lower bound of the expected approximation ratio.

Theorem 4. *For any deterministic truthful mechanism with a consistency ratio $\gamma (< \sqrt{h})$, supposing that the prediction error η is a random variable uniformly distributed on $[1, \gamma]$, the expected approximation ratio is at least $(\gamma+1)\gamma/2$, which is exactly the expected ratio of Algorithm 1.*

4 Frugal Path Auction

In frugal path auction, there is a graph $G = (\{s, t\} \cup V, E)$ with at least two edge-disjoint s - t paths. Each edge $e \in E$ is owned by an agent and the cost c_e^* is a secret known only to that agent. Denote the number of agents by n . Each agent e submits a sealed bid b_e . Then, the auctioneer selects an s - t path L as the winner and gives payment p_e to each $e \in L$. If an agent wins, the utility is $p_e - c_e^*$. Otherwise, the utility is 0. The goal is to design a truthful mechanism such that the total payment $\sum_{e \in E} p_e$ is minimized. In mechanism design with predictions, the auctioneer is given access to the edge cost predictions $\hat{C} = \{\hat{c}_e\}_{e \in E}$.

For a frugal mechanism, a standard way to measure its performance is *frugal ratio*. The formal definition of frugal ratio is a long statement (see [Karlin *et al.*, 2005]). For simplicity, this section states an equivalent but more accessible definition specific to the path auction problem given in [Archer and Tardos, 2002]. Say a frugal path mechanism \mathcal{M} . For an instance \mathcal{I} , use $P(\mathcal{I})$ to denote the total payment of \mathcal{M} and $S(\mathcal{I})$ to denote the cost of the second cheapest s - t path. The frugal ratio of a mechanism \mathcal{M} is defined to be $\text{FR} = \sup_{\mathcal{I}} P(\mathcal{I})/S(\mathcal{I})$.

For the brevity of the algorithm's statement and analysis, we assume that the graph consists of just some parallel s - t paths and each edge has a positive cost. Notice that the assumptions do not affect the problem's hardness. The worst-case bound is still $\Omega(n)$ under the assumptions. The mechanism is described in Algorithm 2.

Theorem 5. *Algorithm 2 is truthful and has a frugal ratio at most $f(\eta)$, where the prediction error $\eta := \max_{e \in E} \{ \frac{c_e^*}{\hat{c}_e}, \frac{\hat{c}_e}{c_e^*} \}$ and*

$$f(\eta) := \begin{cases} \gamma(1 + \eta) & \eta \leq \gamma \\ \frac{n^2}{\gamma} & \eta > \gamma. \end{cases}$$

Mechanism Intuition. The basic intuition is the same as the single-item auction problem. That is, we set a bar for the path with the minimum predicted cost and treat it differently when the reported cost is below or above the bar. Recollect that in single-item auction, if b_1 is below the bar, the mechanism ignores bidder 1 directly. However, things are different in path auction. Ignoring a path may lead to an unbounded

Algorithm 2 Frugal Path Auction with Predictions

Input: A graph G , the predicted edge costs $\hat{C} = \{\hat{c}_e\}_{e \in E}$, the bids $B = \{b_e\}_{e \in E}$ and a parameter $\gamma \in [1, n^{1/3}]$.

Output: The winner and the payments.

- 1: Say path \hat{L} is the path with the minimum predicted cost.
 - 2: For any s - t path $L' \neq \hat{L}$, set its weight $w(L') \leftarrow 1$.
 - 3: **if** $\forall e \in \hat{L}, b_e \leq \gamma \hat{c}_e$ **then**
 - 4: Set $w(\hat{L}) \leftarrow \gamma/n$.
 - 5: **else**
 - 6: Set $w(\hat{L}) \leftarrow n/\gamma$.
 - 7: **end if**
 - 8: **return** the path L with the minimum weighted bid and the threshold bid $\theta(e)$ of each edge $e \in L$.
-

frugal ratio. For example, consider a graph with three edge-disjoint s - t paths. Their costs are 1, 1 and ∞ respectively. Say we ignore the first path. Then any truthful mechanism has to pay ∞ , but the second cheapest path cost is only 1. Hence, the frugal path mechanism needs a more careful design after setting up a bar. When the bid of the predicted cheapest path is far larger than the predicted cost, instead of ignoring it directly, we assign a large weight to it so that the path is still a candidate but less competitive. The proof sketch of Theorem 5 is stated in the following while the detailed proof is provided in the full version of this paper.

Proof Sketch of Theorem 5. The truthfulness proof is simple. We can easily show that the weight function guarantees the monotonicity, and thus the truthfulness. For the ratio analysis, we distinguish two cases based on whether η is at most γ . The reason is that when $\eta \leq \gamma$, we can ensure that the weight of \hat{L} is always γ/n , and thereby \hat{L} always wins. Analyzing the threshold bid of each edge in \hat{L} is a bit subtle, but the intuition is the following. If edge $e \in \hat{L}$ cannot win when its bid is larger than $\gamma \hat{c}_e$, its threshold bid is at most $\gamma \hat{c}_e$, implying that the total payment of all such edges is at most $\gamma \eta$ times the second cheapest path cost. Otherwise, the threshold bid is bounded by γ/n times the second cheapest path cost because the weight of \hat{L} becomes n/γ when $b_e > \gamma \hat{c}_e$. Then, the total payment of these edges should be at most γ times the second cheapest path cost. For the case that $\eta > \gamma$, we show that regardless of which path wins, the threshold bid of each winning edge is at most n/γ times the second cheapest path cost. Thus, the total payment gives a frugal ratio of n^2/γ . \square

Essentially, Algorithm 2 can be viewed as a variant of the VCG path mechanism which incorporates predictions. Besides the VCG path mechanism, there is another famous mechanism for path auction called \checkmark -mechanism [Karlin *et al.*, 2005]. Leveraging the idea of \checkmark -mechanism can improve the robustness slightly. In some graphs, the robustness ratio decreases by a factor of \sqrt{n} . See more details in the full version of this paper. In the following, we give a lower bound to show the optimality of Algorithm 2 when η is small.

Theorem 6. *For any deterministic truthful mechanism, the frugal ratio is $\Omega(\eta^2)$ given any $\eta = O(\sqrt{n})$.*

Algorithm 3 Truthful Job Scheduling with Predictions

Input: Machines M , jobs J , the bid matrix $\{b_{ij}\}_{(i,j) \in M \times J}$, the predicted matrix $\{\hat{t}_{ij}\}_{(i,j) \in M \times J}$ and a parameter $\gamma \in [1, m]$.

Output: The allocation matrix X and the payment matrix P .

- 1: Construct a new processing time matrix \bar{T} as follows.
 - 2: **for** each job $j \in J$ **do**
 - 3: Define $\hat{t}_j^{(min)} := \min_{i \in M} \hat{t}_{ij}$.
 - 4: For each machine $i \in M$,
 set $\bar{t}_{ij} \leftarrow \min\{\hat{t}_{ij}, (m/\gamma) \cdot \hat{t}_j^{(min)}\}$.
 - 5: **end for**
 - 6: Compute the optimal allocation \bar{X} of the scheduling instance \bar{T} .
 - 7: **for** each job $j \in J$ **do**
 - 8: Use k to denote the machine with $\bar{x}_{kj} = 1$.
 - 9: **if** $\hat{t}_{kj} > \bar{t}_{kj}$ **then**
 - 10: Call job j a greedy job.
 - 11: Define $l := \arg \min_{i \in M} \hat{t}_{ij}$ and set up a weight function: $w_{lj} \leftarrow \gamma/m$ and $w_{ij} \leftarrow 1 \forall i \neq l$.
 - 12: **else**
 - 13: Call job j a non-greedy job.
 - 14: Set up a weight function: $w_{kj} \leftarrow \gamma^2/m^2$ and $w_{ij} \leftarrow 1 \forall i \neq k$.
 - 15: **end if**
 - 16: Let the machine z with the minimum weighted bid win and pay the threshold bid θ_{zj} for job j . Namely, set $x_{zj} \leftarrow 1, p_{zj} \leftarrow \theta_{zj}$ and $x_{ij}, p_{ij} \leftarrow 0 \forall i \neq z$.
 - 17: **end for**
 - 18: **return** $X = \{x_{ij}\}_{(i,j) \in M \times J}$ and $P = \{p_{ij}\}_{(i,j) \in M \times J}$.
-

When $\eta = \gamma - \epsilon$ for a tiny positive ϵ , the frugal ratio of Algorithm 2 approaches the lower bound.

5 Truthful Job Scheduling

In this section, we consider the truthful job scheduling on unrelated machines. In the problem, there are n jobs J and m machines M . Each machine i is owned by a selfish agent, and the processing time t_{ij}^* of each job j on machine i is a private value known only to the agent. Without loss of generality, assume that $t_{ij} > 0$. Each job has to be assigned to exactly one machine. The mechanism will ask each machine i to report a processing time b_{ij} for each job j , and then allocate jobs. Once job j is assigned to machine i , the mechanism pays the machine (agent) p_{ij} . Use $X = \{x_{ij}\}_{(i,j) \in M \times J}$ to denote the job allocation matrix, where x_{ij} is 1 if job j is assigned to machine i , otherwise 0. The makespan of an allocation X is the completion time of the last job, i.e., $MS := \max_{i \in M} \sum_{j \in J} x_{ij} t_{ij}^*$. The utility of each machine i is defined to be $\sum_{j \in J} (p_{ij} - x_{ij} t_{ij}^*)$. The goal is to design a truthful mechanism which minimizes the makespan. In mechanism design with predictions, we are given the predicted processing times $\hat{T} = \{\hat{t}_{ij}\}_{(i,j) \in M \times J}$. The prediction is error-prone and the prediction error $\eta := \max_{(i,j)} \{\frac{\hat{t}_{ij}}{t_{ij}^*}, \frac{t_{ij}^*}{\hat{t}_{ij}}\}$.

This problem is very different from the previous two problems. First, it is not a single-parameter environment. For

each machine i , the private value is a n -dimensional vector $\langle t_{i1}^*, \dots, t_{in}^* \rangle$. Second, the objective function is not directly related to the payment. The mechanism gives the winners money but the goal is to minimize the makespan. Thus, the payment rule in truthful scheduling is only for the purpose of ensuring the truthfulness. We describe the mechanism in Algorithm 3 and claim the following theorem.

Theorem 7. *Algorithm 3 is truthful and has an approximation ratio $O(\min\{\gamma\eta^2, \frac{m^3}{\gamma^2}\})$.*

Mechanism Intuition. We deal with each job independently. Thus, each job can be viewed as a single-item auction, and we can apply our technique without hurting the truthfulness. Similar with the previous two mechanisms, we retain the same job allocation when η is small, in order to build the consistency and the error tolerance. We first compute the optimal assignment for the predicted instance, and set up a weight function of machines such that selecting the machine with the minimum weighted bid for each job gets the optimal assignment. Then we decrease the weights of some machines by a factor. This ensures that when η is small, these machines still obtain the minimum weighted bids, and thereby the job assignment remains unchanged. For the robustness guarantee, we observe that if the ratio between any two machines' weights is at most c , the approximation ratio of the mechanism must be bounded by $c \cdot m$. Thus, we round the predicted instance such that after rounding, the ratio between any two machines' weights is bounded. Notice that using the rounded weights directly may hurt the consistency because the processing time of a job could be different in the predicted instance and the rounded instance. To maintain a good consistency and robustness, we only use the rounded weights to guide the jobs that share the same processing time in both of the two instances. For all other jobs, we set up their weights such that they can be assigned greedily when $\eta = 1$.

Proof Sketch of Theorem 7. The proofs of the truthfulness and the robustness are simple. For each job, the allocation rule is monotone and the ratio between any two machines' weights is bounded. The tricky part is to show the $O(\gamma\eta^2)$ bound. We first show that the consistency ratio is $O(\gamma)$. Suppose that $\eta = 1$. Partition all jobs into two sets: greedy jobs and non-greedy jobs. For non-greedy jobs, we claim that their assignment is the same as the optimal assignment of \bar{T} , and thus, their contribution to the makespan is at most the optimal makespan of \bar{T} . For greedy jobs, we show that although they are assigned differently, shifting each one to the machine with the minimum processing time will not increase the makespan too much. Summing over the contributions of the two job sets gives the $O(\gamma)$ consistency ratio. For erroneous predictions, we distinguish two cases based on whether η is at most $\sqrt{m/\gamma}$. For both of the two cases, we can obtain the $O(\gamma\eta^2)$ approximation ratio. \square

6 Two-Facility Game on a Line

In this section, we consider the problem of locating two facilities on a real line to serve a set of selfish agents. In the problem, there are n agents and each agent i has a private location

Algorithm 4 Two-Facility Game with Predictions

Input: The reported location profile $X = \{x_1, \dots, x_n\}$ and the predicted locations $\hat{V} = \{\hat{v}_1, \dots, \hat{v}_n\}$.

Output: The two facility locations.

- 1: Compute the optimal facility locations \hat{l}_1, \hat{l}_2 for \hat{V} . We assume w.l.o.g. that \hat{l}_1 is the facility which attracts more agents and there exists an agent \hat{i} with $\hat{l}_1 = \hat{v}_{\hat{i}}$.
 - 2: Set $l_1 \leftarrow x_{\hat{i}}$.
 - 3: Define $d_A := \max_{x_j \leq x_{\hat{i}}} (x_{\hat{i}} - x_j)$ and $d_B := \max_{x_j \geq x_{\hat{i}}} (x_j - x_{\hat{i}})$.
 - 4: **if** $d_A \leq d_B$ **then**
 - 5: Set $l_2 \leftarrow l_1 + \max\{2d_A, d_B\}$.
 - 6: **else**
 - 7: Set $l_2 \leftarrow l_1 - \max\{d_A, 2d_B\}$.
 - 8: **end if**
 - 9: **return** l_1 and l_2 .
-

v_i^* . The mechanism asks each agent $i \in [n]$ to report the location $x_i \in \mathbb{R}$, and then outputs two facility locations $l_1, l_2 \in \mathbb{R}$. Each agent i wants to minimize the connection cost c_i , which is the distance between v_i^* and the nearest facility. The goal is to design a truthful mechanism which minimizes the total connection cost. In mechanism design with predictions, we are given the predicted location \hat{v}_i of each agent $i \in [n]$. We show that in an environment without money, we can still use predictions to improve the approximation ratio slightly. The mechanism is described in Algorithm 4.

Theorem 8. *Algorithm 4 is a truthful mechanism with $(1 + n/2)$ -consistent and $(2n - 1)$ -robust.*

Mechanism Intuition. The mechanism builds on the line mechanism [Lu *et al.*, 2010]. We notice that in the line mechanism, there exists an arbitrarily selected dictator. Thus, a natural idea is leveraging predictions to select the dictator. An advantage of this idea is that the robustness can always be guaranteed because selecting any agent to be the dictator obtains an $O(n)$ approximation ratio. Then we check the structure of optimal solutions, and find that for any instance, there always exists an agent such that the line mechanism's performance can be improved if letting this agent be the dictator. Thus, setting the dictator to be such an agent in the predicted instance makes the mechanism consistent.

Proof Sketch of Theorem 8. Since Algorithm 4 is essentially the line mechanism, the truthfulness and the robustness are proved directly. The remaining part is the consistency analysis. For the case that $\eta = 1$, we partition agents into two sets based on the facilities that serve them in the optimal solution. Say S_1 is the agents served by facility \hat{l}_1 . For any agent in S_1 , the current connection cost is at most the connection cost in the optimal solution because facility \hat{l}_1 opens. According to the definition of \hat{l}_1 , at least half of agents are in S_1 . For the remaining agents, we show that each one's connection cost is at most OPT plus the connection cost in the optimal solution. Thus, the total connection cost is at most $(1 + n/2)$ OPT. \square

References

- [Anand *et al.*, 2020] Keerti Anand, Rong Ge, and Debmalya Panigrahi. Customizing ML predictions for online algorithms. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 303–313. PMLR, 2020.
- [Antoniadis *et al.*, 2020] Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. Secretary and online matching problems with machine learned advice. In *NeurIPS*, 2020.
- [Archer and Tardos, 2002] Aaron Archer and Éva Tardos. Frugal path mechanisms. In *SODA*, pages 991–999. ACM/SIAM, 2002.
- [Ashlagi *et al.*, 2009] Itai Ashlagi, Shahar Dobzinski, and Ron Lavi. An optimal lower bound for anonymous scheduling mechanisms. In *EC*, pages 169–176. ACM, 2009.
- [Azar *et al.*, 2013] Pablo Daniel Azar, Constantinos Daskalakis, Silvio Micali, and S. Matthew Weinberg. Optimal and efficient parametric auctions. In *SODA*, pages 596–604. SIAM, 2013.
- [Chen *et al.*, 2010] Ning Chen, Edith Elkind, Nick Gravin, and Fedor Petrov. Frugal mechanism design via spectral techniques. In *FOCS*, pages 755–764. IEEE Computer Society, 2010.
- [Christodoulou *et al.*, 2007] George Christodoulou, Elias Koutsoupias, and Angelina Vidali. A lower bound for scheduling mechanisms. In *SODA*, pages 1163–1170. SIAM, 2007.
- [Christodoulou *et al.*, 2020] George Christodoulou, Elias Koutsoupias, and Annamária Kovács. On the nisan-ronen conjecture for submodular valuations. In *STOC*, pages 1086–1096. ACM, 2020.
- [Dobzinski and Shaulker, 2020] Shahar Dobzinski and Ariel Shaulker. Improved lower bounds for truthful scheduling. *CoRR*, abs/2007.04362, 2020.
- [Elkind *et al.*, 2004] Edith Elkind, Amit Sahai, and Kenneth Steiglitz. Frugality in path auctions. In *SODA*, pages 701–709. SIAM, 2004.
- [Fotakis and Tzamos, 2014] Dimitris Fotakis and Christos Tzamos. On the power of deterministic mechanisms for facility location games. *ACM Trans. Economics and Comput.*, 2(4):15:1–15:37, 2014.
- [Giannakopoulos *et al.*, 2020] Yiannis Giannakopoulos, Alexander Hammerl, and Diogo Poças. A new lower bound for deterministic truthful scheduling. In *SAGT*, volume 12283 of *Lecture Notes in Computer Science*, pages 226–240. Springer, 2020.
- [Goldberg *et al.*, 2001] Andrew V. Goldberg, Jason D. Hartline, and Andrew Wright. Competitive auctions and digital goods. In *SODA*, pages 735–744. ACM/SIAM, 2001.
- [Gollapudi and Panigrahi, 2019] Sreenivas Gollapudi and Debmalya Panigrahi. Online algorithms for rent-or-buy with expert advice. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 2319–2327. PMLR, 2019.
- [Im *et al.*, 2021] Sungjin Im, Ravi Kumar, Mahshid Montazer Qaem, and Manish Purohit. Non-clairvoyant scheduling with predictions. In *SPAA*, pages 285–294. ACM, 2021.
- [Karlin *et al.*, 2005] Anna R. Karlin, David Kempe, and Tami Tamir. Beyond VCG: frugality of truthful mechanisms. In *FOCS*, pages 615–626. IEEE Computer Society, 2005.
- [Koutsoupias and Vidali, 2007] Elias Koutsoupias and Angelina Vidali. A lower bound of $1+\phi$ for truthful scheduling mechanisms. In *MFCSS*, volume 4708 of *Lecture Notes in Computer Science*, pages 454–464. Springer, 2007.
- [Lattanzi *et al.*, 2020] Silvio Lattanzi, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Online scheduling via learned weights. In *SODA*, pages 1859–1877. SIAM, 2020.
- [Li and Xian, 2021] Shi Li and Jiayi Xian. Online unrelated machine load balancing with predictions revisited. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 6523–6532. PMLR, 2021.
- [Lu *et al.*, 2009] Pinyan Lu, Yajun Wang, and Yuan Zhou. Tighter bounds for facility games. In *WINE*, volume 5929 of *Lecture Notes in Computer Science*, pages 137–148. Springer, 2009.
- [Lu *et al.*, 2010] Pinyan Lu, Xiaorui Sun, Yajun Wang, and Zeyuan Allen Zhu. Asymptotically optimal strategy-proof mechanisms for two-facility games. In *EC*, pages 315–324. ACM, 2010.
- [Lykouris and Vassilvitskii, 2018] Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 3302–3311. PMLR, 2018.
- [Medina and Vassilvitskii, 2017] Andres Muñoz Medina and Sergei Vassilvitskii. Revenue optimization with approximate bid predictions. In *NIPS*, pages 1858–1866, 2017.
- [Myerson, 1981] Roger B. Myerson. Optimal auction design. *Math. Oper. Res.*, 6(1):58–73, 1981.
- [Nisan and Ronen, 2001] Noam Nisan and Amir Ronen. Algorithmic mechanism design. *Games Econ. Behav.*, 35(1-2):166–196, 2001.
- [Procaccia and Tennenholtz, 2009] Ariel D. Procaccia and Moshe Tennenholtz. Approximate mechanism design without money. In *EC*, pages 177–186. ACM, 2009.
- [Purohit *et al.*, 2018] Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ML predictions. In *NeurIPS*, pages 9684–9693, 2018.
- [Rohatgi, 2020] Dhruv Rohatgi. Near-optimal bounds for online caching with machine learned advice. In *SODA*, pages 1834–1845. SIAM, 2020.
- [Roughgarden and Iwama, 2017] Tim Roughgarden and Kazuo Iwama. Twenty lectures on algorithmic game theory. *Bull. EATCS*, 122, 2017.