THE PRIMAL-DUAL PREDICTION AUGMENTED ALGORITHM FOR PARKING PERMIT PROBLEM WITH THREE PERMIT TYPES

YAROSLAV KHARCHENKO AND **ALEXANDER KONONOV**

Abstract: We consider a parking permit problem with three permit types. We propose a randomized primal-dual algorithm that achieves better competitive ratio than any deterministic algorithm. We propose a prediction-augmented modification for the randomized algorithm, that allows both utilizing uncertain information about the future while still providing worst-case guarantees. We prove consistency and robustness bounds for this modification.

Keywords: online, prediction-augmented, randomized algorithm, competitive analysis.

1 Introduction

Online algorithms make decisions using only the information available at the moment, without being able to account for information revealed later. Because of this uncertainty the algorithms are usually analysed for the worst-case scenario, resulting in pessimistic quality metrics.

In the last decade studies have used machine-learning based predictions to improve online algorithms' quality. This new approach is made possible with the latest achievements of artificial intelligence and machine learning methods, enabling predictions of input data for optimization problems. This additional information can improve online algorithms, as is evident with semi-online algorithms. However, semi-online algorithms assume total correctness of this additional data. This assumption often cannot be achieved in practice. On the opposite, predictions are rarely guaranteed to be correct. Thus, prediction accuracy should not be relied upon when designing prediction-augmented algorithms. The algorithms are instead required to be consistent and robust, i.e.,

- if the prediction is close to accurate, the solution found should be close to either the best offline solution or to the optimal solution,
- if the prediction is arbitrarily inaccurate, the solution found should be close to the classic online solution.

The difficulty in designing such an algorithm arises in finding a balance between these qualities. Following the prediction blindly can lead to a bad solution. On the other hand, if the algorithm doesn't trust the prediction at all, it cannot benefit from a good prediction. This approach was first described in [22] and [23], where such algorithms were named "learning-augmented" or "prediction-augmented". Similar algorithms were later developed for other combinatorial optimization problems, such as the ski rental problem [2, 9, 17, 27], scheduling problems [3, 4, 5, 10, 17, 18, 28] and many others.

Our paper considers the parking permit problem, which was first proposed in [24], and is a generalization of the ski rental problem. In the literature on online algorithms, the problem under consideration is classified as a "rent or buy" problem. Although "rent or buy" problems have been studied intensively over the last two decades, the results obtained are mainly related to two cases. In the first case, authors consider a model in which the choice is between two options: either buying or renting the required resource. The ski rental problem, the most famous of these problems, has been studied from various angles [11, 21, 14], including the analysis of online prediction algorithms [17, 9, 2]. In the second case, an arbitrary number k of options are considered [24, 19]. As a rule, the authors obtain interesting theoretical results, but the competitiveness of their algorithms asymptotically depends on k. In this case, the constant for the function of k is ignored, which makes it difficult to evaluate these algorithms from a practical point of view.

The purpose of our study is to examine the competitiveness of algorithms for a problem slightly more complex than the ski rental problem in order to understand how much the accuracy of the algorithm deteriorates with an increase in the number of types of parking permits. We propose a randomized primal-dual algorithm for the restricted version of the problem with three permit types, and present a bound for its' competitive ratio. We also propose parametrized prediction-augmented algorithms with two prediction types, and show the consistency and robustness bounds. Additionally, we note that the predictions differ significantly and can be interpreted as local and global, which is not the case in different modifications of the ski rental problem.

1.1. Problem Definition. We consider the following problem, denoted as $\mathcal{P}$. There are three parking permit types. For permit type $k, k = 1..3$, its cost C_k and duration D_k are known. For each k the entire time interval of the problem consists of disjoint intervals of length D_k. Permit is only valid during the time interval it was purchased on. W.l.o.g. we suppose that

- $C_1 = 1, D_1 = 1$,
- $C_2 = B, D_2 = d$, we designate this time interval as a "week",
- $C_3 = A, D_3 = nd$, we designate this time interval as a "year",

where all numbers are positive integers.

We also suppose that $B < d$ and $A < Bn$. Otherwise, buying a weekly or annual permit does not make sense. The schedule consists of days, some of which are marked as rainy. The days are revealed to the algorithm one at a time. If the new day is rainy and no purchased permit covers it, algorithm must choose, which type of permit to purchase. The algorithm cannot use information about subsequent rainy days. Moreover, no assumption on the distribution of rainy days is imposed. Purchased permits cannot be refunded. Algorithm has to find the set of permits with the minimum total cost.

An instance I of the problem $\mathcal{P}$ is defined by concrete values of A, B, d, n and a schedule Σ of rainy days. Denote the cost of algorithm's solution with $ALG(I)$ and the cost of optimal solution with $OPT(I)$ for the instance I. Value $R_{ALG} = \max\limits_{I} \frac{ALG(I)}{OPT(I)}$ is called the competitive ratio of the algorithm.

1.2. Prediction Model. In our work, we consider a new prediction model. At the start the algorithm is advised, whether it should buy the year-long permit. At the start of each week the algorithm is advised, whether it should buy the permit covering this week. We denote the entire set of these advice as the prediction Π. Thus, if the prediction Π is perfectly accurate, we can compute the cost of optimal solution of instance I of the problem $\mathcal{P}$. Let $\Sigma_{I,\Pi}$ be an arbitrary rainy day schedule for which the optimal strategy is to follow the forecast Π exactly. Denote by $Cost(I, \Pi)$ the cost of the optimal solution for such a schedule. In addition, we consider two forecast confidence parameters λ and μ, where $\lambda, \mu \in (0, 1)$. The values λ and μ reflect the degree of mistrust in the annual and weekly predictions, respectively. The closer the value of these parameters is to 0, the more we trust the prediction. Raising the values of λ and μ makes the algorithm more robust to prediction mistakes, but makes less use of the provided information. Denote the cost of algorithm's solution with $ALG(I, \Pi, \lambda, \mu)$ for the instance I and the prediction Π.

If for any instance I and any prediction Π

$$\frac{ALG(I, \Pi, \lambda, \mu)}{OPT(I)} \leq \beta(\lambda, \mu),$$

we say that the algorithm is $\beta(\lambda, \mu)$-robust.

If for any instance I and any prediction Π

$$\frac{ALG(I, \Pi, \lambda, \mu)}{Cost(I, \Pi)} \leq \gamma(\lambda, \mu),$$

we say that the algorithm is γ-consistent.

The consistency and robustness have the following intuitive interpretation. A consistent algorithm finds an approximation for the solution of the predicted schedule, and if the schedule was predicted correctly, the algorithm approximates the optimal solution well. Robustness is independent of the prediction, and guarantees the algorithm will approximate the optimal solution regardless of prediction errors. Thus, the consistency is the competitive ratio under perfect predictions whereas the robustness is the upper bound of competitive ratio when the predictions can be arbitrary bad.

1.3. Related Work. The ski rental problem requires you to decide whether to pay a small fee to rent skis on the current day or buy skis and ski on them on subsequent days. Unfortunately, you don't know how long you'll be skiing. It depends on the weather, health and a host of other circumstances. Thus, you are faced with the simplest online problem, first mentioned by Rudolph in the context of the work on competitive snoopy caching. By applying a simple rule: buy skis when the rental cost reaches the cost of the skis, we guarantee a 2-approximation solution in the worst case. This rule represents the best possible deterministic strategy. A randomized $\frac{e}{e-1}$-competitive algorithm was proposed in [13].

Due to the simplicity and universality of the ski rental problem, various formulations with predictions are intensively studied for it [3, 9, 16, 17, 27]. Note that in our work we use an approach based on the construction of a primal-dual algorithm proposed for online problems with predictions in [3].

The ski rental problem allows for many direct generalizations: the multi-shop ski rental problem [26], the parking permit problem [24], the Bahncard problem [8]. In addition, the issue of renting or buying is key for such complex online problems as snoopy caching [12, 13], TCP acknowledgment [6], total completion time scheduling [25] and others.

In this paper, we consider the parking permit problem proposed by [24]. In his paper, Meyerson considered the problem with k types of permits and presented k-competitive deterministic and $O(\log k)$-competitive randomized online algorithms. He also proved that an arbitrary deterministic algorithm has a competitive ratio of at least $k/3$, and an arbitrary randomized algorithm has a competitive ratio of at least $(\log k)/2$. Note that these results do not provide a reasonable lower bound on the competitive ratio for small values of k. For $k = 3$, a parameterized deterministic algorithm with extended prediction is presented in [15], and it is proven that no deterministic algorithm can achieve a competitiveness ratio less than 3.

2 A Primal-dual Online Algorithm

In this section we present a randomized algorithm for $\mathcal{P}$. The algorithm is based on the primal-dual approach introduced by [1]. The survey by [7] contains an extensive review of this approach. For each day the algorithm computes fractional values x, y_i, z_{ij}, that correspond to the probabilities of purchasing respective permit types on this day.

To apply the primal-dual method we formulate the parking permit problem with three permit types as an integer linear programming problem. Let D denote the set of pairs (i, j), where j is the number of a rainy day in the i-th week.

The integer linear program is formalized as follows:

$$Ax + B\sum_{i=1}^{n} y_i + \sum_{(i,j)\in D} z_{ij} \to \min \tag{1}$$

$$x + y_i + z_{ij} \geq 1 \qquad \forall (i,j) \in D \tag{2}$$

$$x, y_i, z_{ij} \in \{0,1\} \tag{3}$$

The linear programming relaxation (LP) is:

$$Ax + B\sum_{i=1}^{n} y_i + \sum_{(i,j)\in D} z_{ij} \to \min \tag{4}$$

$$x + y_i + z_{ij} \geq 1 \qquad \forall (i,j) \in D \tag{5}$$

$$x \geq 0, \quad y_i \geq 0, \quad z_{ij} \geq 0 \tag{6}$$

The dual program (DP) is:

$$\sum_{(i,j)\in D} \xi_{ij} \to \max \tag{7}$$

$$\sum_{(i,j)\in D} \xi_{ij} \leq A \tag{8}$$

$$\sum_{j|(i,j)\in D} \xi_{ij} \leq B \qquad \forall i \tag{9}$$

$$0 \leq \xi_{ij} \leq 1 \qquad \forall (i,j) \in D \tag{10}$$

We note that our ILP-formulations requires perfect knowledge of the set of rainy days. In order to take into account the online formulation of the problem, the algorithm must satisfy the following properties.

- In online mode, the linear constraints defining a feasible LP solution are given to the algorithm one at a time on each new rainy day. In order to maintain a feasible solution for the current set of given constraints, the algorithm is allowed to change the values of the variables.

This results in a sequence of solutions $(x^{(i,j)}, y_i^{(i,j)}, z_{ij})$ that are updated with each new rainy day $(i,j) \in D$. Let's number the rainy days in the order they appear. Let $(x^{(k)}, y_i^{(k)}, z_{ij}) = (x^{(i,j)}, y_i^{(i,j)}, z_{ij})$, where $(i,j) \in D$ is the k-th rainy day. The next lemma slightly generalizes the result of [7] obtained for the ski rental problem.

Lemma 1. *Let $(x^{(k)}, y_i^{(k)}, z_{ij})$ be a solution of LP restricted to the linear constraints imposed after k raining days. Let $x^{(k)} + y_i^{(k)} \leq 1$ for all k. If the sequences $x^{(k)}$ and $y_i^{(k)}$ are non-decreasing then we can turn a fractional solution into an online randomized algorithm with expected cost no greater than the cost of the fractional solution.*

Proof. Let us arrange the increments of x on the interval $[0, 1]$ on the left, starting from the point 0, and the increments of y_i on the right, starting from the point 1 for all i. Choose $\alpha_0 \in [0, 1]$ uniformly at random. We are going to buy the year-long permit on the day corresponding to the increment x to which α belongs. If the year-long permit has not yet been purchased, then we buy the weekly permit on the day corresponding to the increment of y_i to which α belongs. The probability of buying the year-long permit

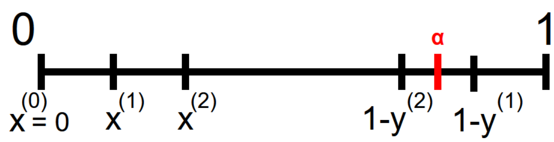

FIG. 1. Example of increment arrangement on the $[0, 1]$ interval. The value α belongs to the second increment of y, corresponding to the second day of the current week.

on the day corresponding to the increment of x is equal to the magnitude of that increment. It follows that the expected cost of buying this permit is precisely Ax. Similarly, the expected cost of buying the weekly permit y_i on week i is exactly By_i. Suppose we have neither a year-long nor a weekly permit covering the kth rainy day (i, j). The probability of this event is equal to $1 - x^{(k)} - y_i^{(k)}$. From (5) we have $1 - x^{(k)} - y_i^{(k)} \leq z_{ij}$. Thus, by linearity of expectation, for any number of rainy days, the expected cost of the randomized algorithm is at most the cost of the fractional solution. □

Denote $e_T(\alpha) = (1+\frac{1}{T})^{\alpha T}$. Note that when T increases, $e_T(\alpha)$ approaches e^α from below.

Algorithm 1 is a primal-dual algorithm, which is a modification of the algorithm for the ski rental problem presented in [6]. The algorithm starts with a zero solution to the primal problem. Each new rainy day $(i, j) \in D$ adds a new constraint $x + y_i + z_{ij} \geq 1$. To satisfy this constraint, the algorithm updates the primal and dual variables in such a way as to keep the ratio between the objective function values of the primal and dual problems as small as possible.

We will first prove a technical lemma about updating the variables.

Lemma 2. *Suppose an algorithm has a variable x which is equal to 0 at the start of the algorithm and gets updated multiple times by using the formula*

$$x \leftarrow (1 + \frac{1}{T})x + \frac{1}{(e_T(\alpha) - 1)T}.$$

Then x will be greater than or equal to 1 after at most αT such updates.

Algorithm 1 The primal-dual online algorithm

$x \leftarrow 0, \quad y_i \leftarrow 0, \quad \hat{y}_i \leftarrow 0, \quad z_{ij} \leftarrow 0 \quad \forall i, j$
if schedule is empty **then**
  $\mathbf{Z} \leftarrow Ax + B\sum_{i=1}^{n} y_i + \sum_{(i,j)\in D} z_{ij}$
  return x, y_i, z_{ij} as the fractional solution.
end if
Reveal the next rainy day $(i, j) \in D$.
while $x + y_i < 1$ **do** (denote the following procedure as "updating x, y_i, z_{ij}")
  $x \leftarrow (1 + \frac{1}{A})x + \frac{1}{(e_A(1)-1)A}$
  $y_i \leftarrow (1 + \frac{1}{B})y_i + \frac{1}{(e_B(1)-1)B}$
  $z_{ij} \leftarrow \max\{0, 1 - x - y_i\}$
  $\xi_{ij} \leftarrow 1$
end while

Proof. After αT updates

$$x = \frac{1}{(e_T(\alpha)-1)T} + \frac{1}{(e_T(\alpha)-1)T}(1+\frac{1}{T}) + \cdots + \frac{1}{(e_T(\alpha)-1)T}(1+\frac{1}{T})^{\alpha T-1}. \tag{11}$$

Using the formula for the sum of a geometric series and the definition of $e_T(\alpha)$ we get

$$x = \frac{1}{(e_T(\alpha)-1)T}\sum_{k=0}^{\alpha T-1}\left(1+\frac{1}{T}\right)^k = \frac{1}{(e_T(\alpha)-1)T}\frac{\left(1+\frac{1}{T}\right)^{\alpha T}-1}{\left(1+\frac{1}{T}\right)-1} = 1. \tag{12}$$

□

Theorem 1. *Algorithm 1 is* $1 + \frac{1}{(e_A(1)-1)} + \frac{1}{(e_B(1)-1)}$*-competitive.*

Proof. On each update the primal solution is feasible by definition of z_{ij}. Suppose the dual solution is not feasible. We will show that the constraints (8), (9) are satisfied. Suppose that

$$\sum_{(i,j)\in D} \xi_{ij} > A.$$

It is only possible after at least $A+1$ updates of x. Note that after A updates $x = 1$ by lemma 2. Thus, update number $A+1$ is not performed, as x is not less than 1, which leads to a contradiction. Similarly, for any $i = 1, \ldots, n$,

$$\sum_{j|(i,j)\in D} \xi_{ij} \le B.$$

Thus, the dual solution is feasible.

Denote the value of goal function for the primal problem as $\mathbf{Z}$, for the dual problem – as $\mathbf{D}$ and let $R = \mathbf{Z}/\mathbf{D}$. Since the cost of the dual solution is a lower bound on $OPT(I)$, R is the competitive ratio of Algorithm 1. At the start of the algorithm both of these values are zero. Suppose the algorithm has just processed an arbitrary rainy day. Let $\mathbf{\Delta Z}, \mathbf{\Delta D}$ be the increments of goal function for primal and dual problem, respectively. We will find a bound on the ratio $\frac{\mathbf{\Delta Z}}{\mathbf{\Delta D}}$.

$$\frac{\mathbf{\Delta Z}}{\mathbf{\Delta D}} \leq \mathbf{\Delta Z} = A\left(\frac{x}{A} + \frac{1}{(e_A(1)-1)A}\right) + B\left(\frac{y_i}{B} + \frac{1}{(e_B(1)-1)B}\right) + z_{ij} \leq$$
$$\leq x + \frac{1}{e_A(1)-1} + y_i + \frac{1}{e_B(1)-1} + z_{ij} \leq 1 + \frac{1}{e_A(1)-1} + \frac{1}{e_B(1)-1} \quad (13)$$

The last inequality follows from the fact that the update of x, y_i, and z_{ij} occurs when $x + y_i < 1$.

Since the cost of a feasible solution of the dual problem is a lower bound for the cost of the optimal solution of the primal problem, we get $R \leq 1 + \frac{1}{(e_A(1)-1)} + \frac{1}{(e_B(1)-1)}$. □

Remark 1. *Note that for $A \geq 4$ and $B \geq 2$ we get $R < 2.5$, and by increasing A and B the value of R tends to $\frac{e+1}{e-1} \approx 2.16$. If $A \leq 4$ and $B \leq 2$ a deterministic 2-competitive algorithm can be derived easily.*

Theorem 1 shows that the competitive ratio of Algorithm 1 is significantly less than the competitive ratio of any detirministic algorithm. Recall that no deterministic algorithm can achieve a competitiveness ratio less than 3 [15]. In the next section, we present a randomized algorithm that finds a solution based on a weather forecast for a selected period and a given confidence level. As will be seen from the subsequent analysis, a confidence level can be selected such that, in the worst-case scenario, we obtain a solution close in quality to the solution of the best deterministic algorithm, and in the case of an accurate forecast, we obtain a solution close to the optimal one.

3 Prediction-Augmented Primal-Dual Algorithm

In this section, we suppose that predictions are provided and design a prediction-augmented algorithm. For convenience, we will use binary parameters to indicate which forecast for the current week was obtained. For each i-th week we set $p_i = 1$ if a rainy week is expected and $p_i = 0$ otherwise.

Suppose that the algorithm receives binary values P, p_i and rational-valued parameters $\lambda, \mu \in (0,1)$. To simplify the presentation we assume that λA, μB, $\frac{A}{\lambda}$ and $\frac{B}{\mu}$ are integer numbers. Denote $\delta(T,\alpha) = \min\{T, \frac{1}{e_T(\alpha)-1}\}$. We note that $\delta(T,\alpha) = \frac{1}{e_T(\alpha)-1}$ if $\alpha \geq \frac{1}{T}$ and $\delta(T,\alpha) = T$ otherwise.

Same as Algorithm 1, the prediction-augmented algorithms start with a zero solution to the primal problem. Each new rainy day $(i,j) \in D$ adds a new constraint $x + y_i + z_{ij} \geq 1$. To satisfy this constraint, the algorithm updates the primal and dual variables. The dual variables are updated to maximize the dual objective function without violating constraints (8) and (9).

Depending on the annual prediction, we consider two algorithms: RAINY YEAR and CLEAR YEAR. The first algorithm is used if the year is predicted to be rainy, the other is used if the prediction doesn't recommend buying year-long permit. Algorithm RAINY YEAR increments x faster and Algorithm CLEAR YEAR increments x slower. The increase rates of weekly permit variables y_i similarly depend on the prediction.

Since the procedures are applied independently and the choice of a specific procedure depends on the initial forecast for the year, we will divide the description of the procedures and the analysis of their quality into two subsections.

3.1. Rainy Year. Let a rainy year be predicted. Denote $\bar{\mu} = \max\{\mu, \frac{B}{A\lambda}\}$. In this subsection we present Algorithm RAINY YEAR and establish its robustness and consistency. The algorithm starts with a zero solution to the primal and dual problem. Each new rainy day $(i, j) \in D$ adds a new constraint $x + y_i + z_{ij} \geq 1$. To satisfy this constraint, the algorithm updates the primal and dual variables. The values of the primal and dual variables do not decrease during the update. The algorithm outputs a fractional solution, which, according to Lemma 1, can be transformed into an online randomized algorithm with expected cost no greater than the cost of the fractional solution. See Algorithm 2 for pseudocode of the algorithm.

We first show that Algorithm RAINY YEAR construct feasible solutions of the primal and dual problems.

Lemma 3. *Algorithm RAINY YEAR produces a feasible solution of the primal and dual program.*

Proof. The primal solution is feasible by definition of z_{ij}.

We will show that the dual solution is also feasible. If $\delta(A, \lambda) = A$, the procedure makes exactly one update and the dual solution is clearly feasible. Let $\delta(A, \lambda) = \frac{1}{(e_A(\lambda)-1)}$. By lemma 2 after λA updates x is equal to 1, the condition in line 6 of the procedure RAINY YEAR is not satisfied and there are no more variable updates. Each update of x increments the sum $\sum\limits_{(i,j)\in D} \xi_{ij}$ at most by 1. Thus, $\sum_{(i,j)\in D} \xi_{ij} \leq A$.

Let us show that the constraint (9) holds for all i, i.e.,

$$\sum_{j|(i,j)\in D} \xi_{ij} \leq B.$$

Suppose that the week i is rainy. If $\delta(B, \mu) = B$, the procedure makes exactly one update of y_i and the dual solution is feasible. Otherwise $\delta(B, \mu) = \frac{1}{(e_B(\mu)-1)}$. Again, by lemma 2 after μB updates y_i reaches 1 and $\sum_{j|(i,j)\in D} \xi_{ij} \leq B$. Let the week i be clear. As mentioned previously, all variables are updated at most λA times. Thus, if $\bar{\mu} = \frac{B}{A\lambda}$ then $\sum_{j|(i,j)\in D} \xi_{ij} \leq B$.

Algorithm 2 RAINY YEAR

1: $x \leftarrow 0, \quad y_i \leftarrow 0, \quad z_{ij} \leftarrow 0, \quad \xi_{ij} \leftarrow 0, \quad \forall i, j$
2: **if** schedule is empty **then**
3:     $\mathbf{Z} \leftarrow Ax + B \sum_{i=1}^{n} y_i + \sum_{(i,j) \in D} z_{ij}$
4:     return x, y_i, z_{ij} as fractional solution.
5: **end if**
6: Reveal new rainy day $(i, j) \in D$. ▷ Denote the following steps as ”variables update”
7: **if** $x + y_i < 1$ **then**
8:     $x \leftarrow (1 + \frac{1}{A})x + \delta(A, \lambda)A^{-1}$
9:     **if** $p_i = 1$ **then**
10:         $\xi_{ij} \leftarrow 1$
11:         **if** $x + (1 + \frac{1}{B})y_i + \delta(B, \mu)B^{-1} < 1$ **then**
12:             $y_i \leftarrow (1 + \frac{1}{B})y_i + \delta(B, \mu)B^{-1}$
13:         **else**
14:             $y_i \leftarrow \max\{y_i, 1 - x\}$
15:         **end if**
16:     **end if**
17:     **if** $p_i = 0$ **then**
18:         $\xi_{ij} \leftarrow \bar{\mu}$
19:         **if** $x + (1 + \frac{1}{B})y_i + \delta(B, \mu^{-1})B^{-1} < 1$ **then**
20:             $y_i \leftarrow (1 + \frac{1}{B})y_i + \delta(B, \mu^{-1})B^{-1}$
21:         **else**
22:             $y_i \leftarrow \max\{y_i, 1 - x\}$
23:         **end if**
24:     **end if**
25:     $z_{ij} \leftarrow \max\{0, 1 - x - y_i\}$
26: **end if**
27: Go to step 2.

Let $\bar{\mu} = \mu$. By lemma 2 y_i reaches 1 after $B\mu^{-1}$ updates. Each update increases $\sum_{j|(i,j) \in D} \xi_{ij}$ by at most μ. Thus, $\sum_{j|(i,j) \in D} \xi_{ij} \leq B$ and the constraint (9) of the dual problem is satisfied. □

First, we analyze the robustness of the algorithm. Let $R(\lambda, \mu)$ denote the competitive ratio of the algorithm for different values of these parameters. We start with extreme cases where Algorithm RAINY YEAR makes a small number of updates.

Lemma 4. *Let a rainy year be predicted. If $\lambda A \leq 1$, then Algorithm RAINY YEAR makes a single update of x, $\mathbf{Z} = A$ and $R(\lambda, \mu) \leq A$ for an arbitrary μ.*

Proof. Let the algorithm get the first rainy day, say $(i, j) \in D$. The current values of the variables x and y_i are 0. When $\lambda A \leq 1$, we have $\delta(A, \lambda) = A$ and $\delta(A, \lambda)A^{-1} = 1$. It follows that x will be assigned the value 1 on line 8. After this step, the algorithm does not make any new updates to the variables. So, we get $\mathbf{Z} = A$ and $R(\lambda, \mu) \leq A$. □

Lemma 5. *Let a rainy year be predicted. If* $1 < \lambda A \leq B$, *then Procedure RAINY YEAR makes at most* B *updates of* x *and*

$$R(\lambda,\mu) \leq \max \left\{ \begin{array}{l} \delta(A,\lambda)\left(1-\frac{1}{A}\right)+\delta(B,\mu)\left(1-\frac{1}{B}\right)+1, \\ B+\delta(A,\lambda)\left(1-\frac{B}{A}\right), \\ \frac{1}{\lambda}+B \end{array} \right\}.$$

Proof. From the proof of lemma 3 it follows that Procedure RAINY YEAR makes at most $\lambda A \leq B$ updates. Let the number of updates be r.

Suppose that the algorithm made exactly one update, i.e., $r = 1$. After the first update we have $x = \delta(A,\lambda)A^{-1} < 1$. The variable y is updated as much as possible while maintaining the constraint $x + y \leq 1$. Therefore two cases may occur.

Case 1. $\delta(B,\mu)B^{-1} \leq 1 - x$. Then $y = \delta(B,\mu)B^{-1}$, and $z = 1 - x - y$. Substituting the values of x, y, and z into the objective, we obtain

$$Z = \delta(A,\lambda)\left(1-\frac{1}{A}\right)+\delta(B,\mu)\left(1-\frac{1}{B}\right)+1.$$

Case 2. $\delta(B,\mu)B^{-1} > 1 - x$. In this case the constraint becomes tight before y reaches its target value, hence $y = 1 - x$, and consequently $z = 0$. Thus

$$Z = Ax + B(1-x).$$

Substituting $x = \delta(A,\lambda)A^{-1}$, we get

$$Z = \delta(A,\lambda) + B\left(1-\frac{\delta(A,\lambda)}{A}\right) = B+\delta(A,\lambda)\left(1-\frac{B}{A}\right).$$

Since for $r = 1$ we have $OPT(I) \geq 1$, the competitive ratio is

$$R(\lambda,\mu) \leq \max \left\{ \begin{array}{l} \delta(A,\lambda)\left(1-\frac{1}{A}\right)+\delta(B,\mu)\left(1-\frac{1}{B}\right)+1, \\ B+\delta(A,\lambda)\left(1-\frac{B}{A}\right) \end{array} \right\}.$$

Let $1 < r \leq \lambda A$. Since $\lambda A \leq B$, no more than B rainy days have passed and $OPT(I) \geq r$. Then

$$x = \frac{1}{(e_A(\lambda)-1)A}\sum_{k=0}^{r-1}\left(1+\frac{1}{A}\right)^k = \frac{(1+\frac{1}{A})^r-1}{e_A(\lambda)-1}.$$

With each update, the sum of the variable values increases by no more than 1. Thus,

$$R(\lambda,\mu) \leq \frac{A((1+\frac{1}{A})^r-1)}{r(e_A(\lambda)-1)} + B \leq \leq \frac{A((1+\frac{1}{A})^{\lambda A}-1)}{\lambda A(e_A(\lambda)-1)} + B = \frac{1}{\lambda}+B. \quad (14)$$

□

Lemma 6. *Let a rainy year be predicted. If* $\frac{B}{A} < \lambda$, *then*

$$R(\lambda, \mu) \leq \max\{1 + \delta(A, \lambda) + \delta(B, \mu), \bar{\mu}^{-1} \cdot [1 + \delta(A, \lambda) + \delta(B, \mu^{-1})]\}.$$

Proof. Let $\mathbf{Z}$ and $\mathbf{D}$ be the values of the goal function for the primal and dual programs, respectively. At the start of Algorithm RAINY YEAR both of these values are equal to zero. Suppose that the algorithm has just processed an arbitrary rainy day. Let $\Delta\mathbf{Z}, \Delta\mathbf{D}$ be the increments of the goal functions. We will find a bound on the ratio $\frac{\Delta\mathbf{Z}}{\Delta\mathbf{D}}$.

There are two possible combinations of increments.

(1) $p_i = 1$:

$$\frac{\Delta\mathbf{Z}}{\Delta\mathbf{D}} \leq \Delta\mathbf{Z} \leq A\left(\frac{x}{A} + \frac{\delta(A, \lambda)}{A}\right) + B\left(\frac{y_i}{B} + \frac{\delta(B, \mu)}{B}\right) + z_{ij} \leq \\ \leq 1 + \delta(A, \lambda) + \delta(B, \mu). \quad (15)$$

(2) $p_i = 0$:

$$\frac{\Delta\mathbf{Z}}{\Delta\mathbf{D}} \leq \frac{\Delta\mathbf{Z}}{\bar{\mu}} \leq \\ \leq \frac{1}{\bar{\mu}} \cdot \left[A\left(\frac{x}{A} + \frac{\delta(A, \lambda)}{A}\right) + B\left(\frac{y_i}{B} + \frac{\delta(B, \mu^{-1})}{B}\right) + z_{ij}\right] \leq \\ \leq \frac{1}{\bar{\mu}} \cdot \left[1 + \delta(A, \lambda) + \delta(B, \mu^{-1})\right]. \quad (16)$$

Combining these two cases we obtain the statement of Lemma 6. □

The above lemmas give the following theorem.

Let

$$\beta_1(\lambda, \mu) = \begin{cases} A & \text{if } \lambda \leq \frac{1}{A} \\ \max\begin{cases} \delta(A, \lambda)\left(1 - \frac{1}{A}\right) + \delta(B, \mu)\left(1 - \frac{1}{B}\right) + 1, \\ B + \delta(A, \lambda)\left(1 - \frac{B}{A}\right) \\ \frac{1}{\lambda} + B \end{cases} & \text{if } \frac{1}{A} < \lambda \leq \frac{B}{A} \\ \max\begin{cases} 1 + \delta(A, \lambda) + \delta(B, \mu) \\ \bar{\mu}^{-1} \cdot [1 + \delta(A, \lambda) + \delta(B, \mu^{-1})] \end{cases} & \text{if } \frac{B}{A} < \lambda \end{cases}$$

Theorem 2. *Algorithm RAINY YEAR obtains a fractional solution with a cost not greater than* $\beta_1(\lambda, \mu) \cdot OPT(I)$ *if a rainy year was predicted, where* $OPT(I)$ *denotes the cost of optimal solution for the given problem instance.*

The Table 1 provides an example of robustness bound as a function of λ and μ. For this example arbitrary values of $A = 3000$ and $B = 30$ were chosen.

TABLE 1. Sample β_1 values for given λ, μ (lower is better)

$\lambda \backslash \mu$	0.01	0.1	0.25	0.5	0.75	0.9	0.99
0.01	130.000	130.000	130.000	130.000	130.000	130.000	130.000
0.1	105.100	105.101	42.120	21.345	14.505	12.238	11.210
0.25	113.037	45.215	18.166	9.368	6.520	5.584	5.161
0.5	127.091	25.419	10.247	5.409	3.881	3.385	3.161
0.75	142.160	18.955	7.662	4.116	3.019	2.666	2.508
0.9	151.676	16.853	6.821	3.696	2.739	2.433	2.296
0.99	157.552	15.915	6.446	3.508	2.614	2.329	2.201

Next, we analyze the consistency of Algorithm RAINY YEAR. Let

$$\gamma_1(\lambda,\mu) = \begin{cases} 1 & \text{if} \quad \lambda \leq \frac{1}{A} \\ \lambda\delta(A,\lambda) + \left(\lambda - \frac{1}{A}\right)(1+\delta(B,\mu)) & \text{if} \quad \frac{1}{A} < \lambda \leq 1 \end{cases}$$

Theorem 3. *Algorithm RAINY YEAR obtains a fractional solution with a cost not greater than $\gamma_1(\lambda,\mu)\cdot c(\Pi)$ if a rainy year was predicted, where $c(\Pi)$ denotes the cost of the predicted solution.*

Proof. We suppose that the prediction Π is correct. Then the cost of the optimal offline solution is A. We consider two cases.

Case 1: $\lambda \leq \frac{1}{A}$. In this case we have $\delta(A,\lambda) = A$. Let $(i,j) \in D$ be the first rainy day. We have $\delta(A,\lambda)A^{-1} = 1$. It follows that x will be assigned the value 1 and we get $\gamma_1(\lambda,\mu) = 1$.

Case 2: $\lambda > \frac{1}{A}$. In this case we have $\delta(A,\lambda) = \frac{1}{e_A(\lambda)-1}$. Suppose the algorithm receives a new rainy day $(i,j) \in D$ and $x+y_i < 1$. From inequality (15) and (16) we have $\Delta\mathbf{Z} \leq 1 + \delta(A,\lambda) + \max\{\delta(B,\mu), \delta(B,\mu^{-1})\}$. Since $\delta(B,\mu^{-1}) = \frac{1}{e_B(\mu^{-1})-1} \leq \frac{1}{e_B(\mu)-1} = \delta(B,\mu)$ we obtain

$$\Delta\mathbf{Z} \leq 1 + \delta(A,\lambda) + \delta(B,\mu).$$

Moreover, in the last iteration of updating x the variables y_i and z_{ij} are not updated. Since x reaches 1 after λA updates we obtain

$$\mathbf{Z} = \lambda A\delta(A,\lambda) + (\lambda A - 1)(1+\delta(B,\mu)).$$

Since $Cost(I,\Pi) = A$, we have

$$\gamma_1(\lambda,\mu) = \lambda\delta(A,\lambda) + \left(\lambda - \frac{1}{A}\right)(1+\delta(B,\mu)).$$

□

The Table 2 provides an example of consistency bounds as a function of λ and μ with the same arbitrary values of $A = 3000$ and $B = 30$.

We can see that for the RAINY YEAR algorithm consistency can be traded for robustness by increasing the inverse trust parameter λ, with the robustness ratio approaching the competitive ratio of the Algorithm 1. This

TABLE 2. Sample γ_1 values for given λ, μ (lower is better)

$\lambda \backslash \mu$	0.01	0.1	0.25	0.5	0.75	0.9	0.99
0.01	1.295	1.098	1.040	1.020	1.014	1.012	1.011
0.1	4.041	2.015	1.408	1.208	1.142	1.121	1.111
0.25	8.620	3.545	2.026	1.523	1.359	1.305	1.282
0.5	16.261	6.104	3.063	2.057	1.728	1.622	1.574
0.75	23.911	8.674	4.110	2.601	2.108	1.948	1.876
0.9	28.506	10.220	4.743	2.932	2.341	2.148	2.062
0.99	31.265	11.149	5.125	3.133	2.482	2.270	2.176

is expected, since for high λ algorithm RAINY YEAR ignores the predicted information, effectively defaulting to the prediction-less version.

In case of low values of λ, the consistency ratio approaches 1, which means algorithm RAINY YEAR finds solutions close to the predicted one. Surprisingly, the consistency bound also decreases with the increase of μ. This is a side effect of the algorithm purchasing weekly permits aggressively when a rainy week is predicted, and in the case of a rainy year, lowering the trust in weekly predictions slows down the rate of these unnecessary purchases.

Consider for example $\lambda = 0.75, \mu = 0.99$. In this case, RAINY YEAR algorithm achieves consistency less than 1.9, which means that if the prediction was correct, the algorithm pays no more than 2 times the price of optimal solution. At the same time, even if the prediction was completely wrong, RAINY YEAR algorithm still has robustness ratio less than 2.7, which means that it still pays less than 3 times the price of optimal solution. As was discussed previously, no deterministic algorithm can guarantee a consistency ratio less than 3.

3.2. Clear Year. Let a clear year be predicted. In this subsection we present Algorithm CLEAR YEAR and establish its robustness and consistency. See Algorithm 3 for pseudo-code of the procedure.

Similarly to Lemma 3 we have the following result.

Lemma 7. *Algorithm CLEAR YEAR produces a feasible solution of the primal and dual program.*

We now provide guarantees for Algorithm CLEAR YEAR.

Let

$$\beta_2(\lambda, \mu) = \max \begin{cases} \lambda^{-1} \cdot [1 + \delta(A, \lambda^{-1}) + \delta(B, \mu)] \\ (\min\{\mu, \lambda\})^{-1} \cdot [1 + \delta(A, \lambda^{-1}) + \delta(B, \mu^{-1})] \end{cases}$$

Theorem 4. *Algorithm CLEAR YEAR obtains a fractional solution with cost not greater than* $\beta_2(\lambda, \mu) \cdot OPT(I)$*, where* $OPT(I)$ *denotes the cost of optimal solution for the given problem instance.*

Algorithm 3 CLEAR YEAR

1: $x \leftarrow 0, \quad y_i \leftarrow 0, \quad z_{ij} \leftarrow 0, \quad \xi_{ij} \leftarrow 0, \quad \forall i, j$
2: **if** schedule is empty **then**
3: $\quad$ $\mathbf{Z} \leftarrow Ax + B\sum_{i=1}^{n} y_i + \sum_{(i,j)\in D} z_{ij}$
4: $\quad$ return x, y_i, z_{ij} as fractional solution.
5: **end if**
6: Reveal new rainy day $(i, j) \in D$. $\quad$ ▷ Denote the following steps as "variables update"
7: **if** $x + y_i < 1$ **then**
8: $\quad$ $x \leftarrow (1 + \frac{1}{A})x + \delta(A, \lambda^{-1})A^{-1}$
9: $\quad$ **if** $p_i = 1$ **then**
10: $\quad\quad$ $\xi_{ij} \leftarrow \lambda$
11: $\quad\quad$ **if** $x + (1 + \frac{1}{B})y_i + \delta(B, \mu)B^{-1} < 1$ **then**
12: $\quad\quad\quad$ $y_i \leftarrow (1 + \frac{1}{B})y_i + \delta(B, \mu)B^{-1}$
13: $\quad\quad$ **else**
14: $\quad\quad\quad$ $y_i \leftarrow \max\{y_i, 1 - x\}$
15: $\quad\quad$ **end if**
16: $\quad$ **end if**
17: $\quad$ **if** $p_i = 0$ **then**
18: $\quad\quad$ $\xi_{ij} \leftarrow \min\{\lambda, \mu\}$
19: $\quad\quad$ **if** $x + (1 + \frac{1}{B})y_i + \delta(B, \mu^{-1})B^{-1} < 1$ **then**
20: $\quad\quad\quad$ $y_i \leftarrow (1 + \frac{1}{B})y_i + \delta(B, \mu^{-1})B^{-1}$
21: $\quad\quad$ **else**
22: $\quad\quad\quad$ $y_i \leftarrow \max\{y_i, 1 - x\}$
23: $\quad\quad$ **end if**
24: $\quad$ **end if**
25: $\quad$ $z_{ij} \leftarrow \max\{0, 1 - x - y_i\}$
26: **end if**
27: Go to step 2.

Proof. Let $\mathbf{Z}$ and $\mathbf{D}$ be the value of goal function for the primal and dual program, respectively. At the start of the algorithm both of these values are equal to zero. Suppose the algorithm has just processed an arbitrary rainy day. Let $\Delta\mathbf{Z}, \Delta\mathbf{D}$ be the increments of the goal functions. We will find a bound on the ratio $\frac{\Delta\mathbf{Z}}{\Delta\mathbf{D}}$.

Consider possible increment combinations.

(1) $p_i = 1$:

$$\frac{\Delta\mathbf{Z}}{\Delta\mathbf{D}} \leq \frac{\Delta\mathbf{Z}}{\lambda} \leq$$
$$\leq \frac{1}{\lambda} \cdot \left[A\left(\frac{x}{A} + \frac{\delta(A, \lambda^{-1})}{A}\right) + B\left(\frac{y_i}{B} + \frac{\delta(B, \mu)}{B}\right) + z_{ij} \right] \leq$$
$$\leq \frac{1}{\lambda} \cdot \left[1 + \delta(A, \lambda^{-1}) + \delta(B, \mu)\right]. \quad (17)$$

(2) $p_i = 0$:

$$\frac{\Delta\mathbf{Z}}{\Delta\mathbf{D}} \le \frac{\Delta\mathbf{Z}}{\min\{\mu,\lambda\}} \le$$
$$\le \frac{1}{\min\{\mu,\lambda\}} \cdot \left[A\left(\frac{x}{A} + \frac{\delta(A,\lambda^{-1})}{A}\right) + B\left(\frac{y_i}{B} + \frac{\delta(B,\mu^{-1})}{B}\right) + z_{ij}\right] \le$$
$$\le \frac{1}{\min\{\mu,\lambda\}} \cdot \left[1 + \delta(A,\lambda^{-1}) + \delta(B,\mu^{-1})\right]. \quad (18)$$

Combining these two cases considered together we obtain the statement of the theorem.

The Table 3 provides an example of robustness bounds as a function of λ and μ with the same arbitrary values of $A = 3000$ and $B = 30$. ☐

TABLE 3. Sample β_2 bound values for given λ, μ (lower is better)

$\lambda\backslash\mu$	0.01	0.1	0.25	0.5	0.75	0.9	0.99
0.01	3100.000	1067.395	458.678	257.397	191.636	170.236	160.674
0.1	310.000	106.740	45.868	25.740	19.164	17.024	16.068
0.25	124.075	42.770	18.422	10.371	7.740	6.884	6.502
0.5	115.658	21.661	9.487	5.461	4.146	3.718	3.527
0.75	135.806	14.709	6.593	3.909	3.033	2.747	2.620
0.9	149.088	14.909	6.043	3.405	2.675	2.437	2.331
0.99	157.293	15.730	6.371	3.471	2.589	2.308	2.202

To present the consistency bound, we introduce new notations.
Let

$$\gamma_2(\lambda,\mu) = \begin{cases} 1 + \delta(A,\lambda^{-1}) + \delta(B,\mu^{-1}) & \text{if} \quad \mu \le \frac{1}{B} \\ \delta(A,\lambda^{-1}) + \mu(1 + \delta(B,\mu)) & \text{if} \quad \frac{1}{B} < \mu \le 1. \end{cases}$$

Theorem 5. *Algorithm CLEAR YEAR obtains a fractional solution with a cost not greater than* $\gamma_2(\lambda,\mu) \cdot c(\Pi)$ *otherwise, where* $c(\Pi)$ *denotes the cost of the predicted solution.*

Proof. Denote the number of rainy weeks as r_0, and the number of rainy days that are not included in any rainy week as d_0. We suppose that the prediction Π is correct. Then the cost of the optimal offline solution is $r_0 B + d_0$.

Each update of the d_0 updates performed outside rainy weeks, according to (18), adds to objective function the cost

$$1 + \delta(A,\lambda^{-1}) + \delta(B,\mu^{-1}). \quad (19)$$

Supposing that i is a rainy week, we consider two cases.

Case 1: $\mu \le \frac{1}{B}$. We have $\delta(B,\mu) = B$ and $x + (1+\frac{1}{B})y_i + \delta(B,\mu)B^{-1} \ge 1$. According to lines (11-14) of Procedure CLEAR YEAR, the variable y_i will receive the value $1-x$ and no other variable updates will be made this week. It follows that the total cost will increase by no more than

$$x + \delta(A, \lambda^{-1}) + By_i \le B + \delta(A, \lambda^{-1}).$$

Taking into account (19) we get

$$\mathbf{Z} \le d_0 \left(1 + \delta(A, \lambda^{-1}) + \delta(B, \mu^{-1})\right) + r_0 \left(B + \delta(A, \lambda^{-1})\right).$$

Since $OPT = r_0 B + d_0$ we have

$$\gamma_2 \le \frac{d_0 \left(1 + \delta(A, \lambda^{-1}) + \delta(B, \mu^{-1})\right) + r_0 \left(B + \delta(A, \lambda^{-1})\right)}{r_0 B + d_0} \le$$
$$\le 1 + \left(\frac{(r_0 + d_0)\delta(A, \lambda^{-1}) + d_0 \delta(B, \mu^{-1})}{r_0 B + d_0}\right)$$
$$\le 1 + \delta(A, \lambda^{-1}) + \delta(B, \mu^{-1}),$$

where the last inequality follows from $B > 1$ and $r_0 B + d_0 > 0$.

Case 2: $\mu > \frac{1}{B}$. We have $\delta(B, \mu) = \frac{1}{e_B(\mu) - 1}$. In this case, on each rainy week μB updates are performed. Each of these updates, according to (17), increases the total cost by no more than $1 + \delta(A, \lambda^{-1}) + \delta(B, \mu)$. Taking into account (19) we obtain the total cost of the fractional solution is

$$\mathbf{Z} \le \mu r_0 B \left(1 + \delta(A, \lambda^{-1}) + \delta(B, \mu)\right) + d_0 \left(1 + \delta(A, \lambda^{-1}) + \delta(B, \mu^{-1})\right) =$$
$$(\mu r_0 B + d_0)\delta(A, \lambda^{-1}) + \mu r_0 B(1 + \delta(B, \mu)) + d_0(1 + \delta(B, \mu^{-1})). \quad (20)$$

Let us estimate the expression in brackets in the last term in (20):

$$1 + \delta(B, \mu^{-1}) = 1 + \frac{1}{e_B(\mu^{-1}) - 1} = \frac{e_B(\mu^{-1})}{e_B(\mu^{-1}) - 1} = \frac{1}{1 - e_B(-\mu^{-1})} =$$
$$= \frac{1}{1 - (1 + 1/B)^{-B/\mu}} \le \frac{\mu}{1 - (1 + 1/B)^{-\mu B}} = \frac{\mu}{1 - e_B(-\mu)} =$$
$$= \frac{\mu e_B(\mu)}{e_B(\mu) - 1} = \mu + \frac{\mu}{e_B(\mu) - 1} = \mu(1 + \delta(B, \mu)),$$

where the inequality follows from Lemma 19(2) in [3].

Substituting the obtained expression into (20) we obtain

$$\mathbf{Z} \le (\mu r_0 B + d_0)\delta(A, \lambda^{-1}) + \mu r_0 B(1 + \delta(B, \mu)) + d_0 \mu(1 + \delta(B, \mu))$$

Since the cost of optimal solution is equal to $r_0 B + d_0$, we get

$$\gamma_2(\lambda, \mu) \le \delta(A, \lambda^{-1}) + \mu(1 + \delta(B, \mu)).$$

□

The Table 4 provide an example of consistency bounds as a function of λ and μ with the same arbitrary values of $A = 3000$ and $B = 30$.

Similar to the RAINY YEAR algorithm, CLEAR YEAR allows consistency to be traded for robustness by increasing the inverse trust parameters λ and μ. Low values lead to algorithm following the prediction accurately

Table 4. Sample γ_2 values for given λ, μ

$\lambda \backslash \mu$	0.01	0.1	0.25	0.5	0.75	0.9	0.99
0.01	1.000	1.067	1.147	1.287	1.437	1.532	1.591
0.1	1.000	1.067	1.147	1.287	1.437	1.532	1.591
0.25	1.019	1.086	1.165	1.306	1.456	1.551	1.609
0.5	1.157	1.224	1.303	1.444	1.594	1.689	1.747
0.75	1.358	1.425	1.505	1.645	1.795	1.890	1.949
0.9	1.491	1.558	1.638	1.778	1.928	2.023	2.082
0.99	1.573	1.640	1.720	1.860	2.010	2.105	2.164

at the cost of worst-case scenario robustness. As with algorithm RAINY YEAR, some combinations of λ and μ (e.g. $\lambda = 0.9, \mu = 0.75$) allow to both achieve a consistency less than 2 while still having better robustness than any deterministic algorithm. Based on the observed quality of predictions, other values can be chosen, if following the prediction more accurately is expected to give better results.

Conclusion

A parking problem with three permit types is considered. First, we present a randomized primal-dual algorithm. Then we consider a model in which the algorithm receives two types of predictions: long-term and short-term. We propose prediction-augmented algorithms that make decisions based on the predictions received and a given level of confidence in them. We prove consistency and robustness bounds for the algorithms which depend on confidence levels. In turn, calculations of bounds can influence the choice of confidence levels when using the algorithm. In particular, based on the calculations presented in Table 2, in a rainy year it is more profitable to focus on the annual permit and ignore the weekly permits.

In our case, the levels of confidence in the forecasts are assumed to be given and not changing during the algorithm's operation. The next step in studying this problem would be to consider a more flexible algorithm that responds to the accuracy of the forecasts obtained.

References

[1] Noga Alon, Baruch Awerbuch, and Yossi Azar. (2003). The online set cover problem. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing, STOC'03*, page 100-105, New York, NY, USA, ACM, ISBN 1581136749. doi:10.1145/780542.780558.

[2] Spyros Angelopoulos, Christoph Durr, Shendan Jin, Shahin Kamali, and Marc P. Renault. (2020). Online computation with untrusted advice. *Journal of Computer and System Sciences*, 144, 103545. https://doi.org/10.1016/j.jcss.2024.103545.

[3] Bamas, Etienne, Andreas Maggiori, and Ola Svensson. (2020). The primal-dual method for learning augmented algorithms. Advances in Neural Information Processing Systems **33**, 20083-20094.

[4] Bampis, E., Dogeas, K., Kononov, A. V., Lucarelli, G., and Pascual, F. (2022). Scheduling with untrusted predictions. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022*, Vienna, Austria: 23-29.

[5] Evripidis Bampis, Alexander Kononov, Giorgio Lucarelli, and Fanny Pascual. (2023). Non-Clairvoyant Makespan Minimization Scheduling with Predictions. In *34th International Symposium on Algorithms and Computation (ISAAC 2023). Leibniz International Proceedings in Informatics (LIPIcs)*, Volume 283, pp. 9:1-9:15, Schloss Dagstuhl-Leibniz-Zentrum fur Informatik.

[6] Buchbinder, N., Jain, K., Naor, J. (2007). Online primal-dual algorithms for maximizing ad-auctions revenue. In *European Symposium on Algorithms (ESA 2007)*, 253-264. Berlin, Heidelberg: Springer.

[7] Niv Buchbinder and Joseph (Seffi) Naor. (2009). The Design of Competitive Online algorithms via a Prima-Dual Approach. *Found. Trends Theor. Comput. Sci.*, **3**(2-3), 93-263. doi:10.1561/0400000024.

[8] Fleicher, R. (2001). On the Bahncard problem. *Theoretical Computer Science*, 268(1), 161-174.

[9] Gollapudi, S., Panigrahi, P. (2019). Online algorithms for rent-or-buy with expert advice. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, Long Beach, California, USA*, 2319-2327.

[10] Im, S., Kumar, R., Montazer Qaem, M., and Purohit, M. (2021). Non-clairvoyant scheduling with predictions. In SPAA, pages 285-294. ACM.

[11] Karlin, A. R., Kenyon, C., Randall, D. (2001). Dynamic TCP acknowledgement and other stories about e/(e-1). In Proceedings of the thirty-third annual ACM symposium on Theory of Computing (STOC 2001): 502-509.

[12] Karlin, A. R., Manasse M.S., Rudolph L., and Sleator D.D. (1988). Competitive snoopy cashing. Algorithmica, **3**, 79-119.

[13] Karlin, A. R., Manasse M.S., McGeoch L.A., and Owicki S. (1994). Competitive randomized algorithms for nonuniform problems. Algorithmica, **11**(6), 542-571.

[14] Khanafer, A., Kodialam, M., and Puttaswamy, K. P. N. (2013) The constrained ski-rental problem and its application to on-line cloud cost optimization. In *Proceedings of the INFO-Com*, pp. 1492-1500.

[15] Kharchenko, Y., Kononov, A. (2024). A Learning-Augmented Algorithm for the Parking Permit Problem with Three Permit Types. In *Proceedings of the 23rd International Conference, MOTOR 2024*, LNCS 14766: 116-126. DOI: 10.1007/978-3-031-62792-7 8.

[16] Rohan Kodialam. (2019). Optimal algorithms for ski rental with soft machine-learned predictions. *CoRR, abs/1903.00092, URL http://arxiv.org/abs/1903.00092.*

[17] Kumar, R., Purohit, M., and Svitkina, Z. (2018). Improving online algorithms via ML predictions. *Advances in Neural Information Processing Systems* 31, 9661-9670.

[18] Lattanzi, S., Lavastida, T., Moseley, B., and Vassilvitskii, S. (2020). Online scheduling via learned weights. In *Proceedings of the Fourteens Annual ACM-SIAM Symposium on Discrete Algorithms, (SODA '20)* , 1859-1877.

[19] M.S. de Lima, San Felice M. C., Lee O. (2017) On Generalization of the Parking Permit Problem and Network Leasing Problem. *Electronic notes in discrete mathematics* 62, 225-230.

[20] Lindermayr, A., Megow, N. (2022). Permutation predictions for non-clairvoyant scheduling. In Proceedings of 34th ACM Symposium on Parallelism in Algorithms and Architectures, (SPAA'22) 357-368, ACM.

[21] Lotker, Z., Patt-Shamir, B., and Rawitz, D. (2008) Rent, lease or buy: Randomized algorithms for multislope ski rental. In *Proceedings of the 25th Annual Symposium on the Theoretical Aspects of Computer Science (STACS)*, pp 503-514.

[22] Lykouris, T., Vassilvitskii, S. (2021). Competitive cashing with machine learned advice. *Journal of the ACM*, 68(4), 1-25.

[23] Andres Mu?oz Medina, Sergei Vassilvitskii. (2017). Revenue optimization with approximate bid predictions. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017*, Long Beach, CA, USA, 1858-1866.

[24] Meyerson, A. (2005). The parking permit problem. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, 274-282. IEEE.

[25] Seiden Steven. (2000). A guessing game and randomized online algorithms, In Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, STOC'00, 592-601, New York, NY, USA, Association for Computing Machinery. ISBN 1581131844. doi: 10.1145/335305.335385.

[26] Shin, Y., Lee, C., Lee, G., and An, H.-C. (2023). Improved Learning-Augmented Algorithms for the Multi-Option Ski Rental Problem via Best-Possible Competitive Analysis, *arXiv preprint arXiv:2302.06832*.

[27] S. Wang, J. Li, and S. Wang. (2020) Online algorithms for multi-shop ski rental with machine learned advice. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 8150-8160.

[28] Wei, A., Zhang, F. (2020). Optimal robustness-consistency trade-offs for learning augmented online algorithms. In Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

Yaroslav Kharchenko
Department of Mathematics and Mechanics, Novosibirsk State University,
pr. Koptyuga, 2,
630090, Novosibirsk, Russia
Email address: kharchenko.yar@gmail.com

Alexander Kononov
Sobolev Institute of Mathematics,
pr. Koptyuga, 4,
630090, Novosibirsk, Russia
Email address: alvenko@math.nsc.ru