

ANALYSIS OF THREAT BASED ALGORITHM USING DIFFERENT PERFORMANCE MEASURES

IFTIKHAR AHMAD^{1,*}, MARCUS PIRRON² AND GÜNTER SCHMIDT^{3,4}

Abstract. Since its introduction in 1985, competitive analysis is a widely used tool for the performance measurement of online algorithms. Despite its simplicity and popularity, competitive analysis has its own set of drawbacks which lead to the development of other performance measures. However, these measures were seldom applied to problems in other domains. Recently Boyar *et al.* (*Theor. Comput. Sci.* **532** (2014) 2–13) studied the online search problem using various performance analysis measures for non-preemptive algorithms. We extend the work by considering preemptive *threat-based* algorithms and evaluate it using competitive analysis, bijective analysis, average case and relative interval analysis. For competitive analysis, and average case analysis, our findings are in contrast with that of Boyar *et al.*, whereas for bijective and relative interval analysis our findings complement that of Boyar *et al.*

Mathematics Subject Classification. 90C27.

Received October 4, 2016. Accepted August 7, 2020.

1. INTRODUCTION

Online algorithms have a variety of applications in various domains such as conversion problems, cache replacement, and job scheduling [1, 2, 12]. One of the most widely used tools for the design and performance analysis of online algorithms is competitive analysis, introduced by Sleator and Tarjan [13]. Competitive analysis measures the performance of an online algorithm against the performance of an optimal offline algorithm. This inherits some problems, most notably the need of knowing how the optimal solution looks like? Also, the algorithm is tested in a worst-case scenario, which can make it impossible to compare the efficiency of two algorithms of quite different natures. Considering the paging problem, the strategies Flush When Full (FWF), First In First Out (FIFO) and Least Recently Used (LRU) all yield the same competitive ratio equal to the cache size, although LRU and variants of it are, in practice, the preferred strategies [14].

Therefore, for a variety of problems competitive analysis is not an appropriate evaluation measure and the need to evaluate online algorithms on other performance metrics arises. Alternative approaches exist, some of which do not rely on computing the optimal offline solution, like bijective analysis, average analysis or relative

Keywords. Online conversion problem, online algorithms, competitive analysis.

¹ Department of Computer Science and Information Technology, University of Engineering and Technology, Peshawar, Pakistan.

² Saarland University, Saarbrücken, Germany.

³ Saarland Informatics Campus, Saarbrücken, Germany.

⁴ School of Data Science and Computational Thinking, Stellenbosch University, Stellenbosch, South Africa.

*Corresponding author: ia@uetpeshawar.edu.pk

interval analysis. These approaches stem from specific fields of the problem, and some of the shortcomings of competitive analysis are fixed. However, like competitive analysis, these measures do not provide much insight when algorithms based on different design mechanism are compared. This motivates the need for a systematic study where online algorithms are compared on various performance measures to obtain more useful insight into the performance when the underlying assumptions of an algorithm's design are no more valid. For instance, an algorithm designed under worst-case competitive analysis tends to safeguard itself against the worst possible situation and treats every input like a worst case. By analyzing an algorithm on different performance measures, we can obtain a useful insight into its performance even when the underlying assumptions are modified (as this is normally the case in a real-world situation).

Motivated by the need of algorithms' analysis on various performance measures, we analyze two *threat-based* algorithms (\mathcal{T}_p and \mathcal{T}_q , see Sect. 2) for online uni-directional conversion problem under a spectrum of performance evaluation measures, namely competitive analysis, bijective analysis, average analysis, and relative interval analysis. Note that \mathcal{T}_p was proposed by El-Yaniv *et al.* [8], whereas \mathcal{T}_q is introduced in this work to make the comparison meaningful. The focus of the work is on the comparative study of the two *threat-based* algorithms using different performance measures. We aim to highlight the discrepancy in the reported ranking of algorithms based on the assumed performance measure. It is pertinent to mention that our work is different from that of Boyar *et al.* [5]. Boyar *et al.* [5] considered non-preemptive algorithms for conversion problem. Non-preemptive algorithm are relatively straightforward for analysis as they convert nothing or all when offered a price q_t . Threat based algorithms are more sophisticated, and decide the amount converted on day t based on the offered price q_t . These algorithm can convert any amount between 0 and 1 (assuming the total wealth is normalized to 1). Therefore, analysis of threat based algorithm is more challenging and sophisticated.

Intuitively, under competitive analysis \mathcal{T}_p should always perform better than \mathcal{T}_q as \mathcal{T}_p is specifically designed under the competitive analysis, however, we show that this is not the case and the answer depends on the first offered price. Further, we found that under bijective analysis and average analysis \mathcal{T}_p performs better than \mathcal{T}_q , whereas under finite relative interval analysis \mathcal{T}_q performance is superior to that of \mathcal{T}_p .

Rest of the paper is organized in the following manner. Section 2 presents the preliminaries required to understand the rest of the paper. In Section 3, the algorithms are evaluated using competitive analysis paradigm. Section 4 presents analysis of the algorithms under under bijective analysis. Average case analysis is performed in Section 5, and relative interval analysis is reported in Section 6. Concluding remarks are presented in Section 7.

2. PRELIMINARIES

In this paper, we study the uni-directional conversion problem in an online setting. In the problem settings, a player wants to convert a given amount of dollars D into yens Y with the objective to maximize the amount of Y after a fixed time interval T . At the start of the game, it is assumed that $D_0 = 1$ and $Y_0 = 0$. At each time point $t = \{1, 2, \dots, T\}$, the online player is presented with a new exchange rate q_t ($1 \leq t \leq T$), at which she can exchange an amount $s_t \in [0, 1]$. The online nature of the problem means that at each time point t the online player does not have the knowledge of the future prices $t + 1, t + 2, \dots, T$. Likewise, the decision taken at each time point t is irrevocable. The only additional information available to the online player is about the lower (m) and upper (M) bounds of future prices, *i.e.*, $q_t \in [m, M]$. However, it is not mandatory that m and/or M is observed. The knowledge of m and M is necessary as otherwise, it will not be possible to design an online algorithm with bounded competitive ratio [10, 12]. Online uni-directional conversion problem can be categorized in two different classes as follows;

- (1) Uni-directional non-preemptive conversion problem: The online algorithm is restricted to convert all the wealth (say dollars to yens) at one time point, *i.e.*, $s_t \in \{0, 1\}$. Such algorithms are also called “reservation price algorithms” [12].
- (2) Uni-directional preemptive conversion problem: The online algorithm is allowed to convert little by little as it sees fit, *i.e.*, $s_t \in [0, 1]$.

One of the seminal work in the domain of online conversion problem is that of El-Yaniv *et al.* [7]. El-Yaniv *et al.* [7, 8] presented online non-preemptive and preemptive algorithms for conversion problem. The non-preemptive algorithm assumes a-priori information about the estimated lower (m) and upper (M) bounds of prices, and calculates a reservation price $q^* = \sqrt{Mm}$. The first offered price which is at least q^* is accepted, and whole of the wealth is converted.

El Yaniv *et al.* [8] also proposed a preemptive algorithm (commonly known as *threat based solution*) for online uni-directional conversion problem. The basic rules of the algorithm are as following;

- (1) Consider conversion from D to Y only when the offered price is the highest seen so far.
- (2) Whenever converting D to Y , convert just enough to ensure that a worst-case competitive ratio c (see Eq. (2.1)) is achieved if an adversary sets all the future prices to m .

The algorithm is called *threat-based* (referred to as \mathcal{T} in this text) as the online player faces a potential threat that an adversary can drop the future prices to m without any indication and keep it there for the rest of the investment horizon. El-Yaniv *et al.* [8] calculated c to be solution of equation (2.1);

$$c = \ln \frac{M - m}{m(c - 1)}. \quad (2.1)$$

Further it was shown that the first price accepted for conversion must be greater than cm , *i.e.*, in order to perform the first conversion the offered price must be greater than cm [8]. We call this the *reservation price*. For subsequent conversions, the price must fulfill the criterion of highest so far.

We consider two threat-based algorithms, namely \mathcal{T}_p and \mathcal{T}_q . \mathcal{T}_p refers to the threat-based algorithm proposed by El-Yaniv *et al.* [8] which has a reservation price $p = cm$. \mathcal{T}_q is another threat-based algorithm (the working principle of \mathcal{T}_q remains the same as that of \mathcal{T}_p) but has a higher reservation price $q = cm + \Delta$. Note that \mathcal{T}_q is introduced only to make the comparison more meaningful.

We assume \mathcal{T}_p and \mathcal{T}_q to be real-valued algorithms when using competitive analysis, average analysis, and real interval analysis. For bijective analysis, we assume \mathcal{T}_p and \mathcal{T}_q to be integer-valued, so that a bijection can be constructed. This should prove to be no difference, as real values can be scaled arbitrarily.

Definition 2.1. $\mathcal{A} >_e \mathcal{B}$, if using a performance evaluation criterion e , the performance of algorithm \mathcal{A} is found to be superior than that of algorithm \mathcal{B} . If both algorithms perform the same according to some criterion e , denote this by $\mathcal{A} \equiv_e \mathcal{B}$.

Boyar *et al.* [5] considered the non-preemptive algorithms for analysis on various performance measures. Beside considering the non-preemptive algorithm of El-Yaniv *et al.* [8] with reservation price \sqrt{Mm} , authors also introduced two fictitious algorithms with reservation prices $\frac{M+m}{2}$, and M respectively. The algorithms are compared using different performance measures.

3. COMPETITIVE ANALYSIS

One of the most widely used tools for evaluating the performance of an online algorithm is competitive analysis, introduced by Sleator and Tarjan [13]. In the competitive analysis, the performance of an online algorithm is compared against the performance of an optimal offline algorithm, which has complete knowledge of the input [11].

Let \mathcal{A} be an online algorithm for a profit maximization problem \mathcal{P} . The performance of \mathcal{A} on a sequence $I \in \mathcal{I}$ is denoted by $\text{perf}_{\mathcal{A}}(I)$. Higher performances are preferred. Let OPT be the optimal offline algorithm for \mathcal{P} . Performing on an input sequence I , the performance of OPT is given by $\text{perf}(I)$. Online algorithm \mathcal{A} is $c_{\mathcal{A}}$ competitive iff, for all $I \in \mathcal{I}$;

$$c_{\mathcal{A}} \leq \frac{\text{perf}(I)}{\text{perf}_{\mathcal{A}}(I)}.$$

The competitive ratio is always equal to or greater than 1. The closer it is to 1, the better, as the performance of the online algorithm matches the performance of the optimum offline algorithm. Therefore, we wish to find an online algorithm which minimize the optimal competitive ratio c^* :

$$c^* = \inf c_{\mathcal{A}} = \inf \sup \frac{\text{perf}(I)}{\text{perf}_{\mathcal{A}}(I)}.$$

From *rule 2* of threat-based algorithm follows that a competitive ratio of c (for \mathcal{T}_p) is always attainable if the maximum rate is cm , even if all exchanges occurs at a rate m [8].

Depending on the initial rate q_1 , the algorithm behaves differently; El-Yaniv *et al.* [7, 8] distinguished two possible cases: first, the initial rate q_1 is less than or equal to cm , and second, the initial rate q_1 is higher than cm . The achievable competitive ratio depends on q_1 .

Theorem 3.1. *If $q_1 \in (cm, cm + \Delta]$ then $\mathcal{T}_p >_c \mathcal{T}_q$, otherwise $\mathcal{T}_p \equiv_c \mathcal{T}_q$.*

Proof. According to [7], the competitive ratio achieved by the threat based algorithm depends on the first offered price q_1 . Two cases are discussed. In *case 1*, $q_1 \leq cm$ and in *case 2* $q_1 > cm$. We discuss both of the cases as following;

Case 1: $q_1 \leq cm$.

A worst-case sequence can be of the following form;

$$cm, \dots, \dots, m.$$

The first offered price is cm and the subsequent prices are all set to m . \mathcal{T}_p and \mathcal{T}_q both converts on the last offered price m , thereby achieving the same competitive ratio, *i.e.*, $\mathcal{T}_p \equiv_c \mathcal{T}_q$.

Case 2: $q_1 > cm$.

We can distinguish between two sub-cases based on q_1 .

Case 2.1: $q_1 \in (cm, cm + \Delta]$.

Consider an input sequence of the following form;

$$q_1, m, \dots, \dots, m.$$

First offered price is $q_1 \in (cm, cm + \Delta]$ and all the subsequent prices are set to m . Recall $s_t \in [0, 1]$ represents the amount of wealth invested at offered price q_t by a threat-based algorithm. As $q_1 \in (cm, cm + \Delta]$, *i.e.*, $cm < q_1 \leq cm + \Delta$, \mathcal{T}_p invests $s_1 \in (0, 1)$ on q_1 . As rest of the prices are all set to m , \mathcal{T}_p does not invest on any other day except on the last day T , *i.e.*, $s_T = 1 - s_1$. The total return of \mathcal{T}_p is $s_1 q_1 + (1 - s_1)m$. Optimum offline algorithm converts all the wealth at the highest price *i.e.*, at $q_1 \in (cm, cm + \Delta]$. The resultant competitive ratio of \mathcal{T}_p is;

$$c_p = \frac{q_1}{s_1 q_1 + (1 - s_1)m}. \quad (3.1)$$

On the other hand, \mathcal{T}_q does not invest on the first offered price q_1 as $q_1 > cm + \Delta$, *i.e.*, $s_1 = 0$. \mathcal{T}_q invests all the wealth on the last offered price m resulting in a competitive ratio c_q such that;

$$c_q = \frac{q_1}{m}. \quad (3.2)$$

As $q_1 > m$ and $s_1 > 0 \implies s_1 q_1 + (1 - s_1)m > m \implies c_p < c_q$, *i.e.*, $\mathcal{T}_p >_c \mathcal{T}_q$.

Case 2.2 $q_1 > cm + \Delta > cm$.

Consider an input sequence of the following form;

$$q_1, \dots, \dots, m.$$

Both \mathcal{T}_p and \mathcal{T}_q converts on the first offered price, resulting in the same competitive ratio for both algorithms, i.e., $\mathcal{T}_p \equiv_c \mathcal{T}_q$.

Summarizing, if $cm < q_1 \leq cm + \Delta$ then $\mathcal{T}_p >_c \mathcal{T}_q$, otherwise $\mathcal{T}_p \equiv_c \mathcal{T}_q$. \square

4. BIJECTIVE ANALYSIS

Bijective analysis was first introduced by Angelopoulos *et al.* [4] in 2007. In bijective analysis, to compare an algorithm \mathcal{A} to an algorithm \mathcal{B} , a bijection on the set of all possible sequences is constructed. Contrary to competitive analysis, bijective analysis does not focus on a single worst case sequence, but on the performance of all possible sequences. Additionally, there is no need to know an optimal offline algorithm as a baseline, instead, algorithms can be compared directly.

Boyar *et al.* [5] defined bijective analysis with regards to the online conversion problem as follows:

Definition 4.1. An online conversion algorithm \mathcal{A} is no better than an online conversion algorithm \mathcal{B} according to bijective analysis if there exists an integer $n_0 \geq 1$ such that for each $n \geq n_0$, there is bijection $b : \mathcal{I}_n \leftrightarrow \mathcal{I}_n$ satisfying $\mathcal{A}(\mathcal{I}) \leq \mathcal{B}(b(\mathcal{I}))$.

For this performance measure, we assume that \mathcal{T}_p and \mathcal{T}_q both use integer values. Through scaling real values can be converted into integer values without any loss of information, therefore, the result will not be distorted. It is important to mention that the assumption of integer values holds for prices only and online algorithms can invest little by little, i.e., $s_t \in [0, 1]$.

Theorem 4.2. A bijection between \mathcal{T}_p and \mathcal{T}_q exists if $p = m$ and $m < q \leq M$, else \mathcal{T}_p and \mathcal{T}_q are incomparable.

Proof. First, we calculate the number of possible permutations which yield Y yens when given an exchange rate p at which the first trade should occur. In the second step, we show that a bijection between \mathcal{T}_p and \mathcal{T}_q is possible. Denote the length of the trading sequences by n (In analogy to T in the continuous case).

Depending on the reservation price p , we divide the number of possible permutations which yield Y yens into two cases. In *case 1*, we consider all sequences which yield between m and p yens, and in *case 2*, we consider all sequences which yield between $p + 1 = q$ and M yens.

For *case 1*, consider the sequence in Table 1. In this sequence, $p'_i := c \cdot m + \epsilon_i$, $\epsilon_i < \epsilon_{i+1}$, and for the ease of notation assume that p'_0 is evanescent close to $p = cm$. In other words, on the first and second day the exchange rate q_i is less than the reservation price p , so that no trades are executed. This is true for $p - m$ prices, and there are $i - 1$ possible positions. On the third day, the exchange rate is greater than p , so a trade occurs. As algorithm \mathcal{T} is preemptive, subsequent trades will follow on all days, on which the exchange rate reaches a new maximum and $D \neq 0$. This is the case for $M - p'_0, M - p'_1, \dots$, for a total of $\binom{M-m+1}{t}$ possibilities, where t is additional number of trades after the initial one, which is caused by the preemptive nature of \mathcal{T} . If $D = 0$, no trade is executed on the remaining days. Note that q_j and q_k can be arbitrarily close to M .

For *case 2*, consider all sequences with $m \leq Y < p$. m will be chosen as output only if it is the last exchange rate in the sequence, and all other q_i are less than p . There are $p - m$ possible prices, which are smaller than p , and $n - 1$ prices which are subject to change in each sequence. This results in a total of $(p - m)^{n-1}$ possible sequences. Note that while \mathcal{T} is a preemptive algorithm, and as such is likely to trade over a period of days, this cannot happen if it trades on the last day.

This results in the following equation which, given a reservation price p , yield an output of Y :

$$N_{p,Y} = \begin{cases} (p - m)^{n-1} & m \leq Y \leq p, t = 0 \\ \sum_{i=1}^n ((p - m)^{i-1} \binom{M-m+1}{t} N^{n-(t+i)}) & p \leq Y \leq M, t \leq n - i. \end{cases} \quad (4.1)$$

TABLE 1. Exchange sequence for $N_{p,Y}$, where $p'_0 < p'_1 < p'_2$.

d	1	2	3	4	5	6	7
q_i	$< p'_0$	$< p'_0$	p'_0	p'_1	p'_2	q_j	q_k
$\#V$	$p - m$	$p - m$	1	$M - p'_0$	$M - p'_1$	N	N

For $m < p$, no bijection between two algorithms \mathcal{T}_p and \mathcal{T}_q is possible. The number of sequences which results in an output of $m \leq Y \leq p$ for \mathcal{T}_q is greater than that of \mathcal{T}_p , since $(q-m)^{n-1} > (p-m)^{n-1}$ as per the requirement, i.e., $p < q$. For the number of sequences with output $Y \in [p, M]$, \mathcal{T}_q accumulates higher number of yens than \mathcal{T}_p (with the same reasoning). As $p < q$, the sum of sequences which result in an output $p \leq Y \leq M$ for \mathcal{T}_q , is greater than the ones for \mathcal{T}_p .

For $m = p$, a bijection $b : I_n \leftrightarrow I_n$ can be constructed. Algorithm \mathcal{T}_p accepts the first price. Each possible price will be the output of

$$(p-m)^{1-1} \binom{M-m+1}{t} N^{n-(t+1)} = \binom{M-m+1}{t} N^{n-(t+1)}$$

sequences.

For \mathcal{T}_q , the number of sequences can be derived as followed: Each price in range $[m, q-1] = [m, p]$ is the result of $(q-m)^{n-1}$ sequences, each price in $[q, M] = [p+1, M]$ is result of

$$\sum_{i=1}^n ((p-m)^{i-1} \binom{M-m+1}{t} N^{n-(t+i)})$$

sequences. For constructing a bijective mapping $b : \mathcal{I}_n \leftrightarrow \mathcal{I}_n$, each sequence with output $k < q$ of algorithm \mathcal{T}_p is mapped to sequences with the same output from algorithm \mathcal{T}_q . The number of sequences, which cannot be mapped in this way, are denoted by E_M . The same holds for all sequences with output $k \geq q$, and the number of sequences from \mathcal{T}_p , which cannot be mapped in this way to sequences with the same output from \mathcal{T}_q are denoted by E_q . As [5] points out, $E_M = E_q$, which also holds in this model. Thus, a bijective mapping b can be constructed such that $\mathcal{T}_p(I) > \mathcal{T}_q(b(I)) \forall I \in \mathcal{I}$.

In other words, for $m = p$, \mathcal{T}_p and \mathcal{T}_q are comparable according to bijective analysis, and $\mathcal{T}_p >_b \mathcal{T}_q$. \square

Corollary 4.3. *According to bijective Analysis, \mathcal{T}_p is better than \mathcal{T}_q ; $\mathcal{T}_p >_b \mathcal{T}_q$.*

5. AVERAGE ANALYSIS

With bijective analysis, comparing two algorithms can prove difficult as this measure creates a strong relationship between two algorithms which may not exist [3]. Instead, average analysis compares the average performance of two algorithms, which can give a tendency towards which algorithm will perform – on average – better.

Definition 5.1. An online conversion algorithm \mathcal{A} is no better than an online conversion algorithm \mathcal{B} according to average analysis, if $\frac{\int_{I_n} \mathcal{B}}{\int_{I_n} \mathcal{A}} \geq 1 \ \forall n \geq n_o, n_o \geq 1$.

In Fujiwara *et al.* [9], the threat based algorithm is defined with help of Lebesgue-integrals. Assuming that the function, which maps the exchange rate q_t , is piecewise continuous, with time $t \geq 0$. Then $\int_t^{t+\Delta t} dS(q_t)$ dollars can be exchanged into $\int_t^{t+\Delta t} q_t dS(q_t)$ yens for arbitrarily small time intervals Δt . Denote by $S(q_t) : [0, \infty] \rightarrow [0, 1]$ the strategy of the player.

The strategy $S_q(q_t)$ for \mathcal{T}_p is given as

$$S_p(q_t) = \begin{cases} \frac{1}{c(q_t - m)}, & cm \leq q_t < M \\ 0, & m \leq q_t < cm \end{cases} \quad (5.1)$$

and the strategy $S_q(q_t)$ for \mathcal{T}_q is given as

$$S_q(q_t) = \begin{cases} \frac{1}{c(q_t - m)}, & cm + \Delta cm \leq q_t < M \\ 0, & m \leq q_t < cm + \Delta cm. \end{cases} \quad (5.2)$$

(see El-Yaniv *et al.* [7])

Depending on the initial rate q_0 , two cases can be distinguished:

Case 1. $cm \leq q_0$. In this case, no trade occurs, and $D(q_t) = 1$, $Y(q_t) = 0$.

Case 2. $cm > q_0$. In this case, the total profit $Y(q_t)$ is $m + \int_I (q_t - m) dS(q_t)$.

Theorem 5.2. *According to average analysis, $\mathcal{T}_p >_a \mathcal{T}_q$.*

Proof. Consider $\frac{\mathcal{T}_p}{\mathcal{T}_q}$, with $\mathcal{T}_p = m + \int_I (q_t - m) dS_p(q_t)$ and $\mathcal{T}_q = m + \int_I (q_t - m) dS_q(q_t)$.

We choose the interval borders $[m, cm]$ and $[cm, M]$.

Solving the Lebesgue-integrals yields

$$\begin{aligned} \frac{\mathcal{T}_p}{\mathcal{T}_q} &= \frac{m + \int_I (q_t - m) dS_p(q_t)}{m + \int_I (q_t - m) dS_q(q_t)} \\ &= \frac{m + \pi_{m, cm}^p + \pi_{cm, M}^p + (q_t - m) * (S_p(cm + \epsilon) - S_p(cm - \epsilon))}{m + \pi_{m, cm}^q + \pi_{cm, M}^q + (q_t - m) * (S_q(cm + \epsilon) - S_q(cm - \epsilon))} \\ &= \frac{m + \pi_{m, M}^p + (q_t - m)(S_p(cm + \epsilon) - S_p(cm - \epsilon))}{m + \pi_{m, M}^q + (q_t - m)(S_q(cm + \epsilon) - S_q(cm - \epsilon))} \end{aligned}$$

where $\pi_{a,b}^q = \int_a^b (q_t - m) \frac{dS_q(q_t)}{dq_t} dq_t$ with lower boundary a and upper boundary b , and, respectively, $\pi_{a,b}^p = \int_a^b (q_t - m) \frac{dS_p(q_t)}{dq_t} dq_t$.

Because of $\lim_{\epsilon \rightarrow 0} S_p(cm + \epsilon) - S_p(cm - \epsilon) = 0$ this simplifies to

$$\frac{m + \pi_{m, M}^p}{m + \pi_{m, M}^q}.$$

Solving this produces

$$\frac{m + 1/c(1 + m(\ln(M - m) - \ln(cm - m)) - M - cm)}{m + 1/c(1 + m(\ln(M - m) - \ln(cm + \Delta cm - m)) - M - cm - \Delta cm)}.$$

As $\ln(cm - m) < \ln(cm + \Delta cm - m)$ and $M + cm < M + cm + \Delta cm$ it follows that this fraction is always greater than 1 for all $\Delta cm > 0$, therefore, $\mathcal{T}_p >_a \mathcal{T}_q$. \square

6. RELATIVE INTERVAL ANALYSIS

Relative interval analysis was introduced by Dorrigiv *et al.* [6] and applied to the paging problem. It compares two algorithms directly across their entire performance spectrum, *i.e.*, not just on a worst case sequence, and not using the optimal offline algorithm as a reference.

Before we are able to define the relative interval fI of two algorithms \mathcal{A} and \mathcal{B} , we need to define some auxiliary functions as given in [5]. Denote by $\mathcal{A}(\sigma)$ the returns of \mathcal{A} on input σ (respectively $\mathcal{B}(\sigma)$ for \mathcal{B}).

$$\begin{aligned}\text{Min}_{\mathcal{A}, \mathcal{B}}(n) &= \min_{|\sigma|=n} \{\mathcal{A}(\sigma) - \mathcal{B}(\sigma)\} \\ \text{Max}_{\mathcal{A}, \mathcal{B}}(n) &= \max_{|\sigma|=n} \{\mathcal{A}(\sigma) - \mathcal{B}(\sigma)\}.\end{aligned}$$

Definition 6.1. The finite relative interval fI of two online algorithms \mathcal{A} and \mathcal{B} is defined as

$$fI(\mathcal{A}, \mathcal{B}) = [\text{Min}(\mathcal{A}, \mathcal{B}), \text{Max}(\mathcal{A}, \mathcal{B})]$$

where $\text{Min}(\mathcal{A}, \mathcal{B}) = \inf_{n \geq 2} \{\text{Min}_{\mathcal{A}, \mathcal{B}}(n)\}$ and $\text{Max}(\mathcal{A}, \mathcal{B}) = \sup_{n \geq 2} \{\text{Max}_{\mathcal{A}, \mathcal{B}}(n)\}$.

Definition 6.2. According to finite relative interval analysis, if $\text{Max}(\mathcal{A}, \mathcal{B}) > |\text{Min}(\mathcal{A}, \mathcal{B})|$, then \mathcal{A} is better than \mathcal{B} . Especially if $fI(\mathcal{A}, \mathcal{B}) = [0, \beta]$, $\beta > 0$, then \mathcal{A} dominates \mathcal{B} .

Theorem 6.3. According to finite interval analysis, $fI(\mathcal{T}_q, \mathcal{T}_p) = [-s_1(m - cm - \epsilon), s_1(M - cm - \epsilon)]$.

Proof. For $\text{Min}(\mathcal{A}, \mathcal{B})$, consider a sequence of two prices $cm + \epsilon, m$ such that $\epsilon < \Delta$. \mathcal{T}_p invests s_1 on $cm + \epsilon$ and the remaining $(1 - s_1)$ on m . The resultant return of \mathcal{T}_p is $s_1(cm + \epsilon) + (1 - s_1)m$. \mathcal{T}_q invests all the wealth at m . Therefore,

$$\begin{aligned}\text{Min}(\mathcal{A}, \mathcal{B}) &= m - [s_1(cm + \epsilon) + (1 - s_1)m] \\ &= -s_1(m - cm - \epsilon).\end{aligned}\tag{6.1}$$

For $\text{Max}(\mathcal{A}, \mathcal{B})$, consider a sequence of two prices $cm + \epsilon, M$ such that $\epsilon < \Delta$. \mathcal{T}_p invests s_1 on $cm + \epsilon$ and the remaining $(1 - s_1)$ on M . The resultant return of \mathcal{T}_p is $s_1(cm + \epsilon) + (1 - s_1)M$. \mathcal{T}_q invests all the wealth at M . Therefore,

$$\begin{aligned}\text{Max}(\mathcal{A}, \mathcal{B}) &= M - [s_1(cm + \epsilon) + (1 - s_1)M] \\ &= s_1(M - cm - \epsilon).\end{aligned}\tag{6.2}$$

From equations (6.1) and (6.2), $fI(\mathcal{T}_q, \mathcal{T}_p) = [-s_1(m - cm - \epsilon), s_1(M - cm - \epsilon)]$. \square

Corollary 6.4. According to finite interval analysis, $\mathcal{T}_q >_{fI} \mathcal{T}_p$.

Proof. To prove that $\mathcal{T}_q >_{fI} \mathcal{T}_p$, we need to show that $\text{Max}(\mathcal{T}_q, \mathcal{T}_p) > |\text{Min}(\mathcal{T}_q, \mathcal{T}_p)|$. From Theorem 6.3, we know that $\text{Max}(\mathcal{T}_q, \mathcal{T}_p) = s_1(M - cm - \epsilon)$ and $\text{Min}(\mathcal{T}_q, \mathcal{T}_p) = -s_1(m - cm - \epsilon)$. For all values of m and M such that $0 < m < M$, $\text{Max}(\mathcal{T}_q, \mathcal{T}_p) > |\text{Min}(\mathcal{T}_q, \mathcal{T}_p)| \implies \mathcal{T}_q >_{fI} \mathcal{T}_p$. \square

7. CONCLUSION

In this paper, we gave an example of how competitive, bijective, average and relative interval analysis can be applied to preemptive threat based online algorithms for conversion problem. Two algorithms \mathcal{T}_p and \mathcal{T}_q , with reservation prices $p < q$, were subjected to these measurements and compared. It turned out that finite interval analysis favored the algorithm with a higher reservation price *i.e.*, \mathcal{T}_q , while bijective analysis and average analysis preferred the algorithm with a lower reservation price, *i.e.*, \mathcal{T}_p . In comparison to the work of Boyar *et al.* [5], our findings are different in a number of ways. Boyar *et al.* [5] observed that the non-preemptive algorithm of El-Yaniv *et al.* [8] with a reservation price \sqrt{Mm} performs better under competitive analysis paradigm than algorithms with reservation prices $\frac{M+m}{2}$, and M . In contrast, we observed that for threat based algorithm, there is no clear winner, and the answer depends on the first offered price. Boyar *et al.* [5] concluded that under relative interval analysis, reservation price $\frac{M+m}{2}$ is a better choice than \sqrt{Mm} . Note that $\frac{M+m}{2} \geq \sqrt{Mm}$ for

$0 < m < M$. We came to the same conclusion, as \mathcal{T}_q is found superior to \mathcal{T}_p under finite interval analysis. Further, Boyar *et al.* [5] found that average case analysis favors algorithms with a higher reservation price. However, for non-preemptive algorithms, we found that this is not the case. Boyar *et al.* [5] further observed that the respective algorithms were not comparable under bijective analysis, where as our bijective analysis favored \mathcal{T}_p .

Acknowledgements. The authors would like to thank the editor-in-chief, and the anonymous referees for their valuable feedback and suggestions.

REFERENCES

- [1] I. Ahmad, M.I.K. Khalil and S.A.A. Shah, Optimization-based workload distribution in geographically distributed data centers: a survey. *Int. J. Commun. Syst.* **33** (2020) e4453.
- [2] S. Albers, Online algorithms: a survey. *Math. Prog.* **97** (2003) 3–26.
- [3] S. Angelopoulos and P. Schweitzer, Paging and list update under bijective analysis. *J. ACM* **60** (2013) 7:1–7:18.
- [4] S. Angelopoulos, R. Dorrigiv and A. López-Ortiz, On the separation and equivalence of paging strategies. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms. SODA '07. Society for Industrial and Applied Mathematics. Philadelphia, PA (2007) 229–237.
- [5] J. Boyar, K.S. Larsen and A. Maiti, A comparison of performance measures *via* online search. *Theor. Comput. Sci.* **532** (2014) 2–13.
- [6] R. Dorrigiv, A. López-Ortiz and J. Ian Munro, On the relative dominance of paging algorithms. *Theor. Comput. Sci.* **410** (2009) 3694–3701.
- [7] R. El-Yaniv, A. Fiat, R.M. Karp and G. Turpin, Competitive analysis of financial games. In: IEEE Symposium on Foundations of Computer Science, Manhattan, NY (1992) 327–333.
- [8] R. El-Yaniv, A. Fiat, R.M. Karp and G. Turpin, Optimal search and one-way trading online algorithms. *Algorithmica* **30** (2001) 101–139.
- [9] H. Fujiwara, K. Iwama and Y. Sekiguchi, Average-case competitive analyses for one-way trading. *J. Comb. Optim.* **21** (2011) 83–107.
- [10] J. Iqbal and I. Ahmad, Optimal online k -min search. *EURO J. Comput. Optim.* **3** (2015) 147–160.
- [11] J. Iqbal, I. Ahmad and A. Shah, Disparity between theory & practice: Beyond the worst-case competitive analysis. In: 2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS). IEEE, Manhattan, NY (2018) 1–6.
- [12] E. Mohr, I. Ahmad and G. Schmidt, Online algorithms for conversion problems: a survey. *Surv. Oper. Res. Manage. Sci.* **19** (2014) 87–104.
- [13] D. Sleator and R. Tarjan, Amortized efficiency of list update and paging rules. *Commun. ACM* **28** (1985) 202–208.
- [14] N. Young, The k -server dual and loose competitiveness for paging. *Algorithmica* **11** (1994) 525–541.