

Competitive Algorithms for Online Conversion Problem with Interrelated Prices

Javeria Iqbal*¹

Department of Computer Science,
International Islamic University, Malaysia

*Corresponding Author

Asadullah Shah³

Department of Information Systems,
International Islamic University, Malaysia

Iftikhar Ahmad²

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

Abstract—The classical uni-directional conversion algorithms are based on the assumption that prices are arbitrarily chosen from the fixed price interval $[m, M]$ where m and M represent the estimated lower and upper bounds of possible prices $0 < m \leq M$. The estimated interval is erroneous and no attempts are made by the algorithms to update the erroneous estimates. We consider a real world setting where prices are interrelated, i.e., each price depends on its preceding price. Under this assumption, we derive a lower bound on the competitive ratio of randomized non-preemptive algorithms. Motivated by the fixed and erroneous price bounds, we present an update model that progressively improves the bounds. Based on the update model, we propose a non-preemptive reservation price algorithm RP^* and analyze it under competitive analysis. Finally, we report the findings of an experimental study that is conducted over the real world stock index data. We observe that RP^* consistently outperforms the classical algorithm.

Keywords—Time series search; one-way trading; online algorithms; update model

I. INTRODUCTION

In an online problem setting, decisions are made with no or partial information about the future. A classic example is the *uni-directional conversion problem*, where a player wants to convert an asset in hand to a desired asset, say dollars to yens. The objective is to obtain the maximum amount of the desired asset (yens) at the end of a fixed length time horizon. The caveat is the unavailability of future conversion prices and the inability to accept a price from the past. The uni-directional conversion problem is an extension of time series search problem, which is a well-studied problem in computer science and operations research with applications in many financial domains, e.g., robust option pricing, secretary selection, job employee searches, and lowest price of goods [1].

Formally, we define online conversion problem as follows. Let m and M be the estimated lower and upper bounds of all price offers. Assume (w.l.o.g) that at the start of the investment horizon, the online player has $D_0 = 1$ units of an asset and $Y_0 = 0$ units of the desired asset. At each time point $t \in [1, T]$, the online player is offered a price $q_t \in [m, M]$ to convert D to Y . She has to make an irrevocable conversion decision, with the objective to maximize the amount of Y at the end of the investment horizon. If the online player accepts q_t , she converts D (whole or a portion) to Y . The game ends

when the player has converted all of D into Y or the last price q_T is revealed, which must be accepted. The player may convert all her wealth at one point of time (*non-preemptive conversion*) or may divide her wealth and spend little by little (*preemptive conversion*). In this work, we restrict to non-preemptive conversion.

Competitive analysis is the prevalent method for the design and analysis of online conversion algorithms. Competitive analysis measures the performance of an online algorithm against an optimum offline algorithm. Let \mathcal{ON} be an online algorithm for a profit maximization problem \mathcal{P} , \mathcal{I} be the set of all inputs instances, and \mathcal{OPT} be an optimum offline algorithm for the same problem \mathcal{P} . Given any $I \in \mathcal{I}$, $\mathcal{ON}(I)$ and $\mathcal{OPT}(I)$ denote the performance of \mathcal{ON} and \mathcal{OPT} on input $I \in \mathcal{I}$ respectively. \mathcal{ON} is called c -competitive if $\forall I \in \mathcal{I}$,

$$\mathcal{ON}(I) \geq \frac{1}{c} \cdot \mathcal{OPT}(I) \quad (1)$$

A considerable body of literature is devoted to online conversion problems [2], [3], [1], [4], [5]. To design online algorithms for uni-directional conversion problems with a bounded competitive ratio, it is imperative to assume a-priori information about the future [3]. For instance, the classical reservation price policy (RP) of El-Yaniv [3] assumes a-priori knowledge of m and M , and calculates a reservation price $q^* = \sqrt{Mm}$. Any offered price which is at least q^* is accepted, if no such price is observed then the last price offer is accepted. RP guarantees a worst case competitive ratio $\sqrt{M/m}$ [3]. The assumption of RP and other algorithms about a priori information is impractical when the real world applicability of online algorithms is considered.

In the real world; *i*) such information is either not available or is prone to errors. Algorithms presented in the literature (see Mohr et al. [6]) keep the estimated information fixed for the complete duration of the investment horizon and make no attempt to improve the quality of the assumed parameters. *ii*) The price fluctuation is bounded and is enforced by the circuit breaker rules (Chen et al.,[2]; Hu et al. [4]). The rules fix a minimum and a maximum permissible price movement and restrict the drastic change in stock prices. Most of the algorithms including RP consider the arbitrary price movement in the range $[m, M]$, which permits the inter-day price fluctuation to

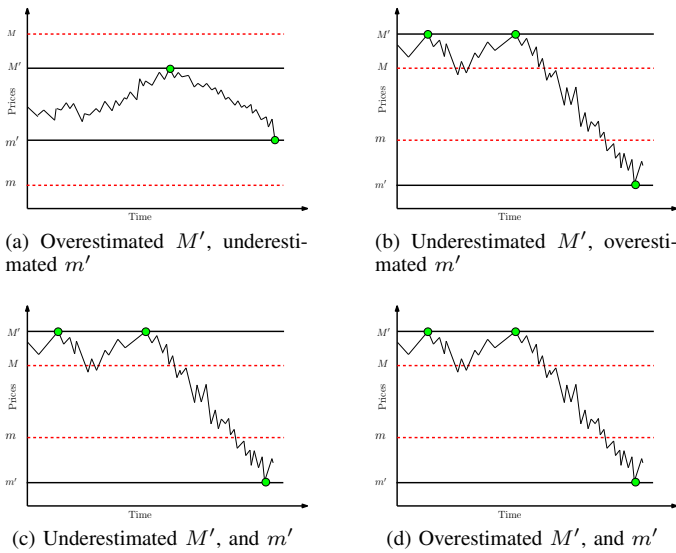


Fig. 1. Estimating the price interval $[m, M]$ can lead to four types of errors; the green dots represent the true (observed) lower and upper bounds, whereas the red dotted line represents the estimated bounds

be M/m , i.e., M can be followed by m and vice versa. These facts motivate us to study the problem of uni-directional non-preemptive conversion under the assumptions that the lower and upper price bounds are erroneous, fixed and arbitrarily drawn from the interval $[m, M]$. It must also be noted that source for acquiring the estimated values of m and M is not relevant as the values can be obtained by using a variety of forecasting techniques [7].

Let the true (exact) lower and upper bounds in a time series be represented by m' and M' respectively. The estimation errors in the values of m and M can be classified into four categories (see Fig. 1).

- i. m' is underestimated and M' is overestimated, i.e., $m < m'$ and $M > M'$ (Fig. 1a).
- ii. m' is overestimated and M' is underestimated, i.e., $m > m'$ and $M < M'$ (Fig. 1b).
- iii. Both m' and M' are underestimated, i.e., $m < m'$ and $M < M'$ (Fig. 1c)
- iv. Both m' and M' are overestimated, i.e., $m > m'$ and $M > M'$ (Fig. 1d)

This work is restricted to the type of error where $M \geq M'$ and $m \leq m'$. Other cases are not considered as it is not possible to design an online algorithm with a bounded competitive ratio when $M' > M$ (i.e., M' can be infinity) and/or $m' < m$ (i.e., m' can be zero) [3], [6].

This work contributes towards the applicability of online conversion algorithms in the real world by considering the *bounded daily return model* [4] where the current price q_t depends on the previous price q_{t-1} as follows:

$$(1 - \gamma)q_{t-1} \leq q_t \leq (1 + \gamma)q_{t-1} \quad (2)$$

Note that $\gamma \in]0, 1[$.

Rest of the paper is organized as follows: Section II briefly presents the relevant literature review. Next, based on the assumption that prices are interrelated, a lower bound on the competitive ratio for any randomized non-preemptive reservation price algorithm is derived in Section III. In Section IV, an update model is introduced that dynamically improves the erroneous bounds (m and M) based on the inter-day price fluctuation. As an application, the update model is applied to the classical algorithm RP resulting in the modified policy RP^* . RP^* policy is analyzed under competitive analysis and minimum improvement in the competitive ratio is derived in relation to RP . Further, the extension of the update model and algorithm RP^* to the general price function of Chen et al. [2] is also discussed. In Section V, RP and RP^* are simulated over the real world stock index and the relative performance of each algorithm with respect to the optimum offline algorithm is reported. Section VI concludes the work and provides directions for future research.

II. RELATED WORK

Besides the work of El-Yaniv [3], and El-Yaniv et al. [1], other major works includes that of Chen et al. [2], Hu et al. [4], and Schroeder et al. [5]. Chen et al. [2] observed that in the real world, the prices movement is not arbitrary but is restricted and governed by a set of rules. For example, in Bangkok stock market the current day price can fluctuate at maximum of 10% of the yesterday's price. Chen et al. [2] claimed, that the current day price $q_t \in [\frac{q_{t-1}}{\alpha}, \beta q_{t-1}]$ with $\alpha, \beta \geq 1$. Hu et al. [4] simplified the Chen et al. [2] model by presenting a *bounded daily return* model of stock prices where each price offer depends on the yesterday's price with $(1 - \gamma)q_{t-1} \leq q_t \leq (1 + \gamma)q_{t-1}$ for some fixed $0 < \gamma < 1$.

Iqbal and Ahmad [8] considered k -min search problem and proposed an optimum algorithm. The algorithm is based on reservation price and threat based algorithms of El-Yaniv et al. [1]. The authors validated the performance of their proposed algorithm against that of Lorenz et al. [9] by conducting experiments on the real world data. Wand and Xu [10] focused on a special case of the secretary problem where the number of applicants are not large, and there is a parameter to measure the ability of the applicant. The authors proved that their policy is better than that of Lorenz et al. [9]. Hasegawa and Itoh [11] considered multi-objective time series search problem (*MOTSS*) and put forth a modified version of competitive analysis for *MOTSS*. The authors presented an online algorithm called *Balanced Price Policy* for *MOTSS* and discussed that the presented policy is the best under the competitive analysis paradigm. Fung [12] considered two way trading problem, knowledge of m and M to derive a lower bound of $\phi = M/m$ for a single trade problem. Further, the author generalized the work to k -trade problem and shown the competitive ratio to be $\phi^{(2k+1)/3}$.

Schroeder et al. [5] considered the time series search problem with inter-related prices and a profit function. The authors presented algorithms with bounded competitive ratio, and proved the optimality of the algorithms as well. For a detailed survey on online algorithm for conversion problems, the reader is referred to Mohr et al. [6].

To the best of our knowledge, there is no conclusive evidence of any study which examines the online algorithms

under the estimated values of parameters and suggests measures to improve the competitive ratio by improving the quality of the estimated parameters. Mahdian et al. [13] analyzed different online problems under estimated values of input parameters. One such problem is allocation of online advertisement space. In an online allocation of advertisement space, a search engine allocates an advertisement slot to a bidder based on the query term of the user. In order to optimally allocate the advertisement space, the estimated frequency of each search term is assumed to be known and the problem is solved using linear programming. However, the estimation can turned out to be completely wrong and the solution is no more optimal and competitive. Mahdian et al.[13] analyzed the problem under incorrect estimates but did not discuss any mechanism to improve the erroneous information in order to improve the performance of the algorithm.

III. LOWER BOUND FOR RANDOMIZED ALGORITHM WITH INTER-DAY PRICE FLUCTUATION FUNCTION

Yao's minmax principle is used to obtain a lower bound on the competitive ratio of randomized algorithms [14]. In simple words, Yao's principle states that in order to obtain a lower bound on the competitive ratio of the best randomized algorithm, it is sufficient to calculate the performance of the best deterministic algorithm for a chosen probability distribution of input instances (see Borodin and El-Yaniv[14]; Lorenz et al. [9] for details). The same technique is employed to derive the lower bound of best uni-directional non-preemptive randomized algorithm when the prices are governed by the price function of Hu et al. [4](cf. Eq. (2)).

Theorem 1. *Assuming that prices follow the inter-day price fluctuation function as shown in Eq (2). Let fluctuation ratio $\phi > 1$, for online uni-directional non-preemptive conversion problem, no randomized algorithm can achieve a better competitive ratio than c where;*

$$c = \log \phi \frac{\gamma}{2(1+\gamma)\log(1+\gamma)}. \quad (3)$$

Proof: Let \mathcal{I} be the set of all input instances, we consider $\mathcal{I}^\gamma \in \mathcal{I}$ such that all $I \in \mathcal{I}^\gamma$ follows the price function of Hu et al. [4]. Let $RAND$ be the best randomized algorithm for online uni-directional non-preemptive conversion problem. We need to show that competitive ratio achieved by $RAND$, $c(RAND) \geq c$. Using the Yao's principle it will be sufficient to show that on the worst case probability distribution $y(I)$, the ratio of average optimum offline return to average return of any deterministic algorithm $ON \in \mathcal{ON}$ is bounded below from c [15], [9], i.e.,

$$\left(\max_{ON \in \mathcal{ON}} E \left[\frac{ON(I)}{OPT(I)} \right] \right)^{-1} \geq c \quad (4)$$

In order to prove Eq. (4), we consider the worst case sequences $I_j \in S^\gamma$ such that $S^\gamma \subset \mathcal{I}^\gamma$ denotes the worst case price sequences. We define our worst case input to be of the following form; ($0 \leq j \leq N-1$)

$$I_j = m(1+\gamma)^0, m(1+\gamma)^1, \dots, m(1+\gamma)^j, \\ m(1+\gamma)^j(1-\gamma), m(1+\gamma)^j(1-\gamma)^2, \dots, m$$

It shows that the worst case sequence I_j is increasing with a factor $(1+\gamma)$ till time point $j+1$, after which it decreases

with a factor $(1-\gamma)$. I_j represents a worst case instance because if the only player accepts an offered price (too early error), adversary has the option to keep on increasing the price sequence to achieve a maximum profit. Similarly, if the online player keeps on rejecting offered price (too late error), the adversary can drop the offered price gradually to m and force the online player to accept the last offered price m .

Our distribution of sequences follows price function and we are only interested in price sequence which is in the form of I_j . We define the probability distribution y on \mathcal{I}^γ by

$$y(I) = \begin{cases} \frac{1}{N} & \text{if } I \in S^\gamma \\ 0 & \text{otherwise} \end{cases}$$

Remark 1. *The maximum fluctuation ϕ of any price sequence in S^γ is bounded by $(1+\gamma)^N$.*

The above remark can easily be proved by considering the price sequence I_{N-1} ;

$$I_{N-1} = m(1+\gamma)^0, \dots, m(1+\gamma)^{N-1}, m(1+\gamma)^{N-1}(1-\gamma), \\ m(1+\gamma)^{N-1}(1-\gamma)^2, \dots, m$$

The maximum price observed is $m(1+\gamma)^{N-1}$ and the minimum observed price is m . Therefore, the fluctuation ratio $\phi = (1+\gamma)^{N-1} < (1+\gamma)^N$.

Now we compute $E \left[\frac{ON(I)}{OPT(I)} \right]$ as under; Consider the last price sequence I_{N-1} ;

$$I_{N-1} = m(1+\gamma)^0, \dots, m(1+\gamma)^{N-1}, m(1+\gamma)^{N-1}(1-\gamma), \\ m(1+\gamma)^{N-1}(1-\gamma)^2, \dots, m$$

Assume (w.l.o.g) that, ON accepts $m(1+\gamma)^p$ on I_{N-1} , it means that on all sequences where $m(1+\gamma)^p$ appears, ON will accept $m(1+\gamma)^p$. This case corresponds to *too early error*. On all other sequences, ON will convert at the last offered price m which corresponds to *too late error*. As all worst case sequences are increasing up to time point $j+1$, $m(1+\gamma)^p$ appears in I_j where $j = p$ to $N-1$. Therefore, $E \left[\frac{ON(I)}{OPT(I)} \right]$ will be;

$$E \left[\frac{ON(I)}{OPT(I)} \right] = \frac{1}{N} \left(\sum_{0 \leq j \leq p-1} \frac{m}{m(1+\gamma)^j} + \sum_{p \leq j < N} \frac{m(1+\gamma)^p}{m(1+\gamma)^j} \right), \\ = \frac{1}{N} \left(\frac{(1+\gamma) - (1+\gamma)^{1-p}}{\gamma} + \frac{(1+\gamma) - (1+\gamma)^{p+1-N}}{\gamma} \right) \\ = \frac{(1+\gamma)(2 - (1+\gamma)^{p-N} - (1+\gamma)^{-p})}{N\gamma}. \quad (5)$$

To find the value of p , where maximum value of $E \left[\frac{ON(I)}{OPT(I)} \right]$ is observed, we take the derivative of Eq (5) with respect to p ;

$$\frac{d}{dp} \left(E \left[\frac{ON(I)}{OPT(I)} \right] \right) = 0$$

$$\frac{d}{dp} \left(\frac{(1+\gamma)(2 - (1+\gamma)^{p-N} - (1+\gamma)^{-p})}{N\gamma} \right) = 0$$

We obtain, $p = N/2$. Replacing the value of $p = N/2$ and $N = \log_{(1+\gamma)}\phi$, we obtain;

$$\max E \left[\frac{ON(I)}{OPT(I)} \right] \leq \frac{1}{\log\phi} \left[\frac{2(1+\gamma)\log(1+\gamma)}{\gamma} \right]$$

Applying the Yao's principle;

$$c(RAND) \geq \left(\max_{ON \in \mathcal{ON}E} E \left[\frac{ON(I)}{OPT(I)} \right] \right)^{-1} \geq \log\phi \frac{\gamma}{2(1+\gamma)\log(1+\gamma)} \quad (6)$$

The proof follows. ■

Remember that ϕ cannot be ∞ as otherwise it will not be possible to design an online algorithm with bounded competitive ratio [3].

Lorenz et al. [9] showed that for online uni-directional non-preemptive conversion problem, no randomized algorithm can achieve a competitive ratio better than $\bar{c} \geq \frac{\ln(\phi)}{2}$. The authors assumed that the prices are arbitrarily drawn from the known interval $[m, M]$ and did not enforce an inter-day price function. A $\tilde{c} \geq \log\phi \frac{\gamma}{2(1+\gamma)\log(1+\gamma)}$ lower bound of is derived assuming that the inter-day price movement is bounded by the price function of Hu et al. [4]. Fig. 2 shows the ratio \bar{c}/\tilde{c} based on the increasing value of $\gamma \in]0, 1[$. The results conform to the real world observation.

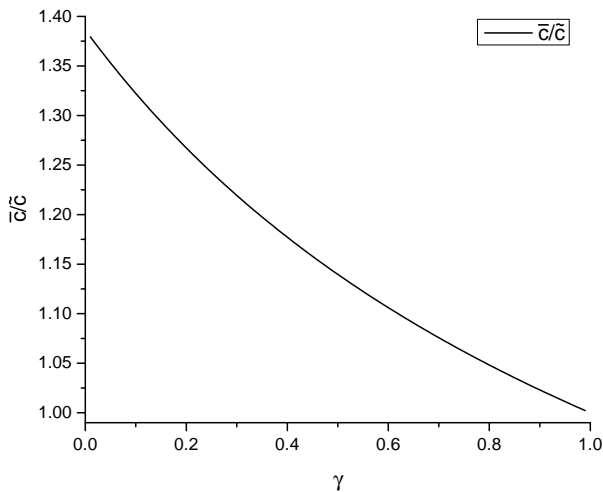


Fig. 2. Ratio of \bar{c} and \tilde{c}

Remark 2. Intuitively speaking, the lower the inter-day price fluctuation, the lower the price movement and thus a better performance can be achieved. Our results confirms the same, for values of γ closer to 0, \tilde{c} is considerably lower than \bar{c} . As γ approaches 1, \tilde{c} approaches \bar{c} . However, our lower bound is lower than the lower bound shown by Lorenz et al. [9] for all values of $\gamma \in]0, 1[$.

IV. PROPOSED MODEL AND ALGORITHM

In the following, the update model and the deterministic non-preemptive reservation price policy RP^* is presented and analyzed. The performance improvement of RP^* in comparison to RP is derived and the applicability of results for a general price fluctuation function is discussed.

A. Update Model

The basic purpose of the *update model* is to keep the initial estimated upper and lower bounds realistic based on the observed price, the inter-day price fluctuation and the remaining number of days. Further, the need of using the price function of Hu et al. [4] is discussed and the *update model* is presented.

A challenging aspect of any update mechanism is to model the stock prices. The behavior of stock prices is highly volatile and a plethora of work is dedicated to obtain an accurate view of stock price movement [16], [17]. Geometric Brownian Motion (*GBM*) is one of several attempts to model the random stock price movement in Black-Scholes model [18], [19]. However, it is hard to obtain an accurate GBM and the approximation of GBM is utilized [2].

In this work, the *bounded daily return* model of stock market introduced by Hu et al. [4] is considered. Recall that the price function restricts the price movement as under;

$$(1-\gamma)q_{t-1} \leq q_t \leq (1+\gamma)q_{t-1} \text{ with } 0 < \gamma < 1 \quad (7)$$

Eq.(7) is considered as an approximation to *GBM* [4] which can be used to model the stock price behavior.

El-Yaniv [3] assumed a fixed price interval in which all the prices must lie whereas Hu et al. [4] considered the bounds for inter-day price fluctuation. In this work, a dual approach is employed and the initial estimated bounds $[m, M]$ as well as the inter-day price fluctuation factor γ are considered.

The update model computes the minimum and the maximum price bounds that can be achieved in the remaining number of days based on the current day price q_t and γ , as shown in Fig. 3. On each day t a price offer q_t is observed and compute:

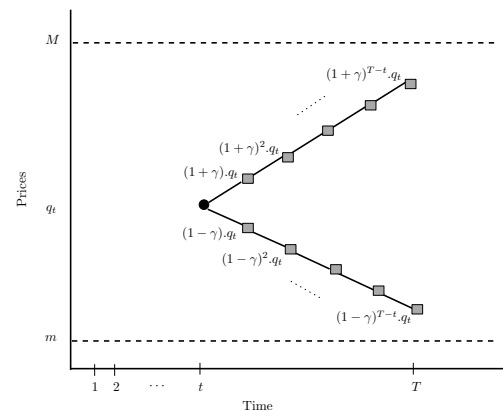


Fig. 3. Computing the achievable lower and upper bounds on day t

- i. The minimum price (lower bound) that can be reached if prices keep on decreasing with a factor γ of which is $q_t(1-\gamma)^{T-t}$.
- ii. The maximum price (upper bound) that can be reached if prices keep on increasing with a factor γ of which is $q_t(1+\gamma)^{T-t}$.

Considering the type 1 error (Fig. 1a) i.e., $M \geq M'$ and $m \leq m'$, the update model should improve the upper bound in a non-increasing manner whereas the lower bound in a non-decreasing way. At time period $t = 0$, the initial estimated lower and upper bounds are $m_0 = m$ and $M_0 = M$ respectively such that $m_0 \leq q_1 \leq M_0$. For each price offer q_t where $t \in [1, T - 1]$, the price interval $[m_t, M_t]$ is updated as follows:

$$m_t = \max\{m_{t-1}, q_t(1 - \gamma)^{T-t}\} \quad (8)$$

$$M_t = \min\{M_{t-1}, q_t(1 + \gamma)^{T-t}\} \quad (9)$$

It can be seen that M_t is non-increasing whereas m_t is non-decreasing with respect to time i.e., $M_t \leq M_{t-1}$ and $m_t \geq m_{t-1}$. The updated price bounds $[m_t, M_t]$ are used for re-computing the reservation price for the proposed algorithm RP^* .

B. Proposed Algorithm

The formal description of RP^* is given by Algorithm 1. It must be noted that the lower and upper bounds are updated

Algorithm 1 RP^*

Require: m, M, γ, T

- 1: Set $m_0 = m, M_0 = M$
 - 2: **for** $t=1$ to $T-1$ **do**
 - 3: A new price q_t is observed.
 - 4: Compute achievable lower bound m_t :

$$m_t = \max\{m_{t-1}, q_t(1 - \gamma)^{T-t}\}$$
 - 5: Compute achievable upper bound M_t :

$$M_t = \min\{M_{t-1}, q_t(1 + \gamma)^{T-t}\}$$
 - 6: Calculate new reservation price $q_t^* = \sqrt{m_t M_t}$
 - 7: **if** $q_t \geq q_t^*$ **then**
 - 8: Accept the offered price q_t
 - 9: **end if**
 - 10: **end for**
 - 11: If no price offer is accepted till the last day T , accept the last price q_T which may be the minimum.
-

till $T - 1$ as if the online player has not accepted a price till the last day, then she must accept the last offered price q_T . Therefore, updating the bounds on the last day is irrelevant.

C. Competitive Analysis of RP^*

Theorem 2. Algorithm RP^* is $c^w(RP^*)$ such that;

$$c^w(RP^*) = \sqrt{\frac{q_j(1 + \gamma)^{T-j}}{q_i(1 - \gamma)^{T-i}}} \quad (10)$$

where i and j are the time points where the last update occurs in the value of m and M , respectively.

Proof: Let q^* be the reservation price (threshold) for accepting a price. The player waits for an offer such that $q_t \geq q^*$. Two cases are possible.

Too Early Error:: If the reservation price computed by RP^* is too low, the adversary offers q^* early in the price sequence and then raises the offered price to the maximum $q_{max} = q_j(1 + \gamma)^{T-j}$. Thus, RP^* converts at q^* , and OPT converts at q_{max} resulting in a competitive ratio as follows:

$$\begin{aligned} c_1^* &= \frac{OPT}{RP^*} \\ &= \frac{q_j(1 + \gamma)^{T-j}}{q^*} \\ &= \frac{q_j(1 + \gamma)^{T-j}}{\sqrt{q_j(1 + \gamma)^{T-j} * q_i(1 - \gamma)^{T-i}}} \\ &= \sqrt{\frac{q_j(1 + \gamma)^{T-j}}{q_i(1 - \gamma)^{T-i}}} \end{aligned} \quad (11)$$

Too Late Error:: If the reservation price calculated by RP^* is too high, the adversary offers the prices such that $q_{max} < q^*$. The player waits q^* for which never occurs in the price sequence. Thus RP^* converts at $q_{min} = q_i(1 - \gamma)^{T-i}$, and OPT converts at $q^* - \epsilon$ where $0 < \epsilon \ll 1$. Thus, the competitive ratio achieved by RP^* is as follows:

$$\begin{aligned} c_2^* &= \frac{OPT}{RP^*} \\ &= \frac{q^* - \epsilon}{q_i(1 - \gamma)^{T-i}} \\ &= \frac{\sqrt{q_j(1 - \gamma)^{T-j} * q_i(1 - \gamma)^{T-i}} - \epsilon}{q_i(1 - \gamma)^{T-i}} \\ &\leq \sqrt{\frac{q_j(1 + \gamma)^{T-j}}{q_i(1 - \gamma)^{T-i}}} \end{aligned} \quad (12)$$

In both too low and too high scenarios, the competitive ratio is no worse than $c^w(RP^*)$. The theorem follows. ■

D. Improvement in Competitive Ratio

In this section, the minimum improvement in competitive ratio of RP^* in relation to RP is derived. It is pertinent to mention that improvement can only be observed when there is an update in the values of m and/or M , otherwise it is nonsensical to discuss improvement. The argument is inline with that presented in the literature related to online search problem such as that of Al-Binali [20]. Al-Binali [20] claimed that improvement in the competitive ratio can only be observed when the forecast is true. In the underlying case, the improvement in competitive ratio is restricted to cases, when an update in the bounds of m and/or M occurs, if no update occurs, it means that the estimated values of m and M were accurate enough and the update model is not required. Let, c^r be the restricted competitive ratio when an update in the m and/or M occurs.

Lemma 1. The minimum improvement observed in the competitive ratio of RP^* in relation to RP is as follows:

$$\Delta c = \min \left\{ \sqrt{\frac{q_1(1 - \gamma)^{T-1}}{m}}, \sqrt{\frac{M}{q_1(1 + \gamma)^{T-1}}}, \sqrt{\left(\frac{1 - \gamma}{1 + \gamma}\right)^{T-1}} \sqrt{\frac{M}{m}} \right\} \quad (13)$$

Proof: Depending on the modified value of m and/or M , there are three possible scenarios;

- i. *Update in the value of m only:* As there is an update in the value of m , the upper bound M remains unchanged. The restricted competitive ratio $c_1^r(RP^*)$ is as follows:

$$c_1^r(RP^*) = \sqrt{\frac{M}{q_i(1-\gamma)^{T-i}}}$$

As M is constant, the maximum competitive ratio $c_1^r(RP^*)$ is observed when the update occurs at day 1, i.e., $i = 1$. The restricted competitive ratio observed is as follows:

$$c_1^r(RP^*) \leq \sqrt{\frac{M}{q_1(1-\gamma)^{T-1}}}$$

The minimum improvement in the competitive ratio of RP^* in comparison to $c^w(RP)$ is as follows:

$$\begin{aligned} \frac{c^w(RP)}{c_1^r(RP^*)} &\geq \frac{\sqrt{M/m}}{\sqrt{\frac{M}{q_1(1-\gamma)^{T-1}}}} \\ &\geq \sqrt{\frac{q_1(1-\gamma)^{T-1}}{m}} \end{aligned} \quad (14)$$

- ii. *Update in the value of M only:* When an update in the value of only M is observed, $c_2^r(RP^*)$ achieved is:

$$c_2^r(RP^*) = \sqrt{\frac{q_j(1+\gamma)^{T-j}}{m}}$$

To maximize $c_2^r(RP^*)$, $q_j(1+\gamma)^{T-j}$ must be maximized (m being constant). $q_j(1+\gamma)^{T-j}$ is maximized at $j = 1$, and thus the competitive ratio $c_2^r(RP^*)$ is achieved as follows:

$$c_2^r(RP^*) \leq \sqrt{\frac{q_1(1+\gamma)^{T-1}}{m}}$$

The improvement in the competitive ratio is:

$$\begin{aligned} \frac{c^w(RP)}{c_2^r(RP^*)} &\geq \frac{\sqrt{M/m}}{\sqrt{\frac{q_1(1+\gamma)^{T-1}}{m}}} \\ &\geq \sqrt{\frac{M}{q_1(1+\gamma)^{T-1}}} \end{aligned} \quad (15)$$

- iii. *Update in the value of m and M :* When an update in the value of m and M is observed, $c_3^r(RP^*)$ is achieved such that

$$c_3^r(RP^*) = \sqrt{\frac{q_j(1+\gamma)^{T-j}}{q_i(1-\gamma)^{T-i}}}$$

To maximize $c_3^r(RP^*)$, $q_j(1+\gamma)^{T-j}$ must be maximized, whereas $q_i(1-\gamma)^{T-i}$ must be minimized. The maximum value of $q_j(1+\gamma)^{T-j}$ is observed for $j = 1$, whereas the minimum value of $q_i(1-\gamma)^{T-i}$ is observed for $i = 1$. The restricted competitive ratio $c_3^r(RP^*)$ is as follows:

$$\begin{aligned} c_3^r(RP^*) &\leq \sqrt{\frac{q_1(1+\gamma)^{T-1}}{q_1(1-\gamma)^{T-1}}} \\ &\leq \sqrt{\left(\frac{1+\gamma}{1-\gamma}\right)^{T-1}} \end{aligned}$$

The improvement in competitive ratio is as follows:

$$\begin{aligned} \frac{c^w(RP)}{c_3^r(RP^*)} &\geq \frac{\sqrt{M/m}}{\sqrt{\left(\frac{1+\gamma}{1-\gamma}\right)^{T-1}}} \\ &\geq \sqrt{\left(\frac{1-\gamma}{1+\gamma}\right)^{T-1}} \sqrt{\frac{M}{m}} \end{aligned} \quad (16)$$

The proof follows from Eqs. (14,15, and 16). ■

E. Extending Results to a General Price Function

In this work, the price function of Hu et al. [4] is used, which is based on a constant price fluctuation factor γ . The results can be extended to the general price fluctuation function of Chen et al.[2] where the inter-day price fluctuation is bounded by two constants α and β , i.e., q_t can either decrease by a factor α of q_{t-1} or increase by β times q_{t-1} . In the following, the competitive ratio achieved by RP^* using the price fluctuation function of Chen et al. [2] is discussed.

Lemma 2. $c^w(RP^*)$ is valid for the general price function of Chen et al. [2].

Proof: Recall the price fluctuation function of Hu et al. [4] is given as follows:

$$(1-\gamma)q_{t-1} \leq q_t \leq (1+\gamma)q_{t-1}, \quad \text{with } 0 < \gamma < 1. \quad (17)$$

The price fluctuation function of Chen et al. [2] is given as follows;

$$\frac{q_{t-1}}{\alpha} \leq q_t \leq \beta q_{t-1}, \quad \text{with } \alpha, \beta \geq 1. \quad (18)$$

Comparing both Eq. (17) and (18) follows;

$$\begin{aligned} \alpha &= \frac{1}{1-\gamma} \\ \beta &= 1+\gamma. \end{aligned}$$

From Eq. (10), the competitive ratio $c^w(RP^*)$ achieved by RP^* is as follows:

$$c^w(RP^*) = \sqrt{\frac{q_j(1+\gamma)^{T-j}}{q_i(1-\gamma)^{T-i}}}$$

Replacing $(1+\gamma)$ by β and $(1-\gamma)$ by $1/\alpha$;

$$c^w(RP^*) = \sqrt{\frac{q_j(\beta)^{T-j}}{q_i(1/\alpha)^{T-i}}}$$

The proof follows. ■

V. EXPERIMENTAL SIMULATION

The online algorithms for conversion problems are experimentally evaluated using real world data. For details, the reader is referred to Chen et al. [2], Hu et al. [4], Iqbal et al. [21] and Mohr et al. [6] (Section 5). In this section, OPT , RP , and RP^* are simulated over the daily closing prices of *Deutscher Aktien Index* (DAX30) for ten years (01.Jan.2008 to 31.Dec.2017). Recall that OPT is the optimum offline algorithm that always converts on the maximum observed price.

A. Methodology

As OPT , RP , and RP^* perform only a single transaction, therefore, no transaction fee is considered. It is also assumed that the algorithms have the required a-priori information (γ and T). However, in a real world setting, the true (exact) lower and upper bounds (m' and M') of prices are unknown. Therefore, it is assumed that there is error in estimation in the values of m' and M' . If true bounds are known, the reservation price calculated by RP is $q^* = \sqrt{Mm}$. The algorithms rely on the estimated bounds (m , M). Therefore, the reservation price q^* can either decrease or increase depending on the estimation error in bounds, i.e., $q^* = \delta\sqrt{Mm}$ where $0 < \delta < \infty$.

The estimation error σ in the lower and upper bounds are introduced such that the reservation price q^* (of RP) is either decreased ($\delta = 0.85$) or increased ($\delta = 1.15$) by 15% of its exact value. The findings are based on the monthly as well as the yearly performance of algorithms. The dataset is partitioned into subsets where each subset represents an investment horizon of one month, thus resulting in 120 subsets for 10 years. The price fluctuation factor γ is extracted from each subset using Eq. (2).

On a given input σ , let, $OPT(\sigma)$, $RP(\sigma)$ and $RP^*(\sigma)$ be the performance of OPT , RP , and RP^* respectively. The performance of each algorithm (OPT , RP , and RP^*) is observed by executing them over 120 subsets of data. For ease of comparison, the performance ratios $c^e(RP) = \frac{OPT(\sigma)}{RP(\sigma)}$ and $c^e(RP^*) = \frac{OPT(\sigma)}{RP^*(\sigma)}$ is computed. As no algorithm can perform better than OPT , the performance ratios $c^e(RP)$, and $c^e(RP^*)$ are at least one. The lower value of performance ratio (close to 1) indicates a better outcome. Further, the performance improvement of RP^* over RP by the improvement factor $\frac{c^e(RP)}{c^e(RP^*)}$, based on the varying values of the estimation error $\sigma = [0.75, 1.25]$ is shown.

B. Results

Fig. 4 summarizes the monthly competitive ratios by RP and RP^* for $\sigma = 0.85$. It is observed that on all 120 monthly data sets, $c^e(RP^*) \leq c^E(RP)$. On average, $c^e(RP^*)$ is 2.5% better than $c^e(RP)$. It is noticed that on few instance, the experimental competitive ratio of RP is worse (greater) than the guaranteed worst case competitive ratio, i.e., $c^e(RP) > c^w(RP)$. This behavior suggests that the estimation error can render the worst case guarantee useless. In contrast to RP , algorithm RP^* performs better by consistently achieving lower competitive ratio. On all 120 subsets, RP^* never performs worse than $\sqrt{M/m}$, thus ensuring that the worst case bounds are always respected. The worst (highest) value of $c^e(RP)$

and $c^e(RP^*)$ are 1.21 and 1.09 respectively. Similarly, for $\sigma = 1.15$, RP^* outperforms RP as depicted in Fig. 5. Interestingly, the maximum (worst) $c^e(RP)$ is 1.21 whereas the corresponding value of $c^e(RP^*)$ is 1.09, i.e., the same values of $c^e(RP)$ and $c^e(RP^*)$ is observed when σ changes from 0.85 to 1.15.. It must be noted that for $\sigma = 0.85$, the worst (maximum) value is observed in Oct 2008, whereas for $\sigma = 1.15$ the worst value is observed for Feb 2009.

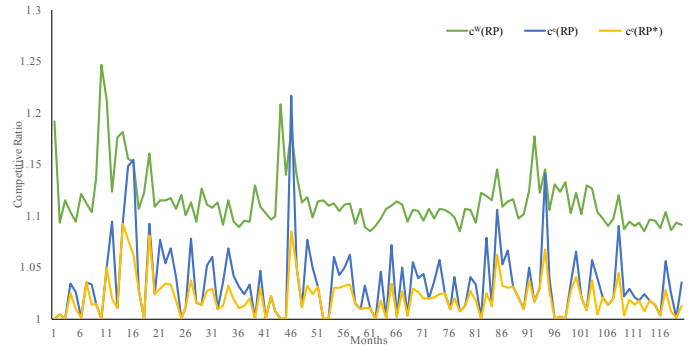


Fig. 4. Monthly Performance with Estimation Error $\delta = 0.85$

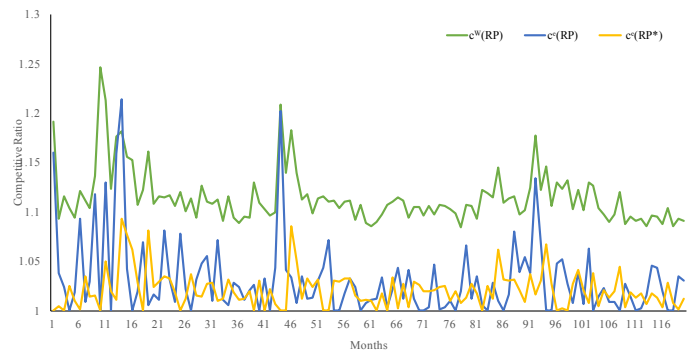


Fig. 5. Monthly Performance with Estimation Error $\delta = 1.15$

The competitive ratios of RP and RP^* over yearly data with $\sigma = 0.85$ are shown in Fig. 6. The worst $c^e(RP)$ is 1.06 observed for year 2009. In contrast, the worst $c^e(RP^*)$ over 10 years period is 1.04 also seen in year 2009. RP^* beats RP over all the years and never violates the worst case bound $c^w(RP^*)$. The performance of both algorithms with $\sigma = 1.15$ is shown in Fig. 7. The same trends is observed for $\sigma = 1.15$ as is seen for $\sigma = 0.85$. Again, RP^* outperforms RP .

In order to observe the improvement in experimentally achieved competitive ratio for varying values of σ , consider $\sigma = \{0.75, 0.76, \dots, 1, \dots, 1.24, 1.25\}$. For each value of σ , the average competitive ratio of RP and RP^* is computed over the 120 monthly subsets of data. Fig. 8 plots the improvement factor $c^e(RP)/c^e(RP^*)$. An improvement factor greater than 1 depicts that RP^* outperforms RP . For $\sigma < 0$, RP^* performs better than RP . A constant improvement of 1.014 is observed when $\sigma = [0.75, 0.9]$. For $\sigma = 1$, both algorithms achieve a same competitive ratio. Over a small range when $\sigma = [1.01, 1.04]$, the performance of RP is better than RP^* .

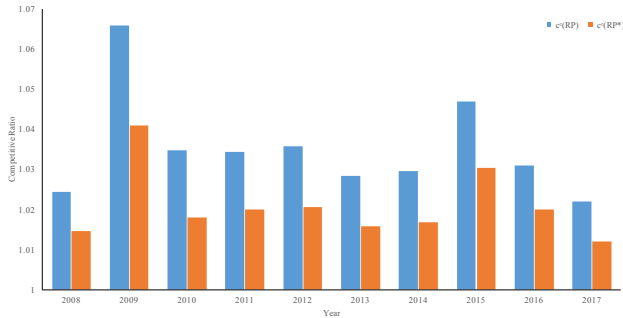


Fig. 6. Yearly Competitive Ratios with Estimation Error $\delta = 0.85$

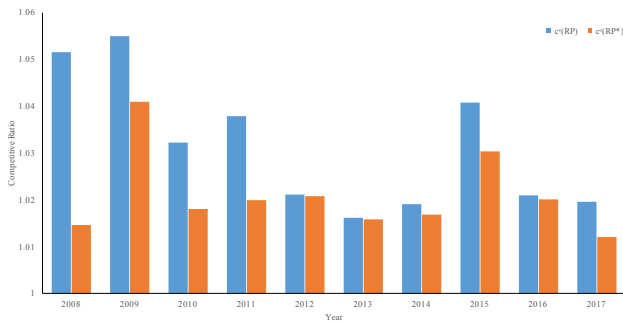


Fig. 7. Yearly Competitive Ratios with Estimation Error $\delta = 1.15$

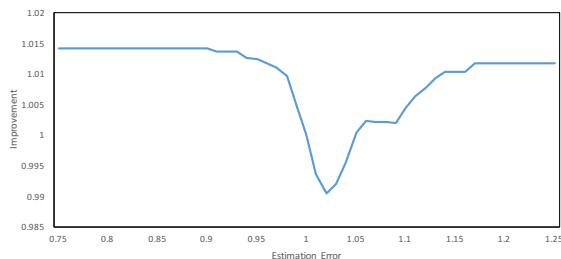


Fig. 8. Improvement Factor for $\delta \in [0.75, 1.25]$

However, a consistent improvement is reported in competitive ratio of RP^* to that of RP , for $\sigma > 1.04$.

VI. CONCLUSION

This work focused on bridging the gap between the theory and practice of online conversion problems by improving the quality of the estimated bounds using the update model. The model is a first step towards modifying the erroneous information for improving the performance of algorithms. A lower bound on the competitive ratio of randomized non-preemptive algorithms assuming the prices are interrelated is derived. Furthermore, algorithm is presented and evaluated using theoretical as well as experimental measures. The experimental simulation of algorithms over the real world data validated the approach of the update model. The work can be extended to investigate the preemptive algorithms under the update model. Further, the online algorithms under estimated values of the parameters can be studied in other areas such as job scheduling, cache algorithms, online advertisement space

allocation, etc.

REFERENCES

- [1] R. El-Yaniv, A. Fiat, R. M. Karp, and G. Turpin, "Optimal search and one-way trading online algorithms," *Algorithmica*, vol. 30, pp. 101–139, 2001.
- [2] G. Chen, M. Kao, Y. Lyuu, and H. Wong, "Optimal buy-and-hold strategies for financial markets with bounded daily returns," *SIAM Journal on Computing*, vol. 31, no. 2, pp. 447–459, 2001.
- [3] R. El-Yaniv, "Competitive solutions for online financial problems," *ACM Computing Surveys*, vol. 30, no. 1, pp. 28–69, 1998.
- [4] S. Hu, Q. Guo, and H. Li, "Competitive analysis of on-line securities investment," in *Algorithmic Applications in Management*, ser. Lecture Notes in Computer Science, N. Megiddo, Y. Xu, and B. Zhu, Eds. Springer, 2005, vol. 3521, pp. 224–232.
- [5] P. Schroeder, R. Dochow, and G. Schmidt, "Optimal solutions for the online time series search and one-way trading problem with interrelated prices and a profit function," *Computers & Industrial Engineering*, vol. 119, pp. 465 – 471, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S036083521830127X>
- [6] E. Mohr, I. Ahmad, and G. Schmidt, "Online algorithms for conversion problems: a survey," *Surveys in Operations Research and Management Science*, vol. 19, no. 2, pp. 87–104, 2014.
- [7] J. Arroyo, R. Espínola, and C. Maté, "Different approaches to forecast interval time series: A comparison in finance," *Computational Economics*, vol. 37, pp. 169–191, 2011. [Online]. Available: <http://dx.doi.org/10.1007/s10614-010-9230-2>
- [8] J. Iqbal and I. Ahmad, "Optimal online k -min search," *EURO Journal on Computational Optimization*, vol. 3, no. 2, pp. 147–160, 2015.
- [9] J. Lorenz, K. Panagiotou, and A. Steger, "Optimal algorithms for k -search with application in option pricing," *Algorithmica*, vol. 55, no. 2, pp. 311–328, 2009.
- [10] S. Wang and Y. Xu, "Online k -max search algorithms with applications to the secretary problem," in *International Conference on Algorithmic Applications in Management*. Springer, 2016, pp. 209–221.
- [11] S. Hasegawa and T. Itoh, "Optimal online algorithms for the multi-objective time series search problem," *Theoretical Computer Science*, vol. 718, pp. 58–66, 2018.
- [12] S. P. Fung, "Optimal online two-way trading with bounded number of transactions," in *International Computing and Combinatorics Conference*. Springer, 2017, pp. 212–223.
- [13] M. Mahdian, H. Nazerzadeh, and A. Saberi, "Online optimization with uncertain information," *ACM Trans. Algorithms*, vol. 8, no. 1, pp. 2:1–2:29, Jan. 2012. [Online]. Available: <http://doi.acm.org/10.1145/2071379.2071381>
- [14] A. Borodin and R. El-Yaniv, "On randomization in on-line computation," *Information and Computation*, vol. 150, no. 2, pp. 244–267, 1999.
- [15] —, *Online computation and competitive analysis*. New York, NY, USA: Cambridge University Press, 1998.
- [16] R. Norvaiša, "Modelling of stock price changes: A real analysis approach," *Finance and Stochastics*, vol. 4, pp. 343–369, 2000. [Online]. Available: <http://dx.doi.org/10.1007/s007800050077>
- [17] R. Sollis, P. Newbold, and S. Leybourne, "Stochastic unit roots modelling of stock price indices," *Applied Financial Economics*, vol. 10, no. 3, pp. 311–315, 2000.
- [18] D. Duffie, *Dynamic Asset Pricing Theory*. Princeton University Press, 2001.
- [19] J. Hull, *Options, Futures, and Other Derivative Securities*. Prentice Hall, 1993.
- [20] S. al-Binali, "A risk-reward framework for the competitive analysis of financial games," *Algorithmica*, vol. 25, pp. 99–115, 1999.
- [21] J. Iqbal, I. Ahmad, and G. Schmidt, "Can online trading algorithms beat the market? an experimental evaluation," in *Proceedings of 3rd Student Conference on Operational Research (SCOR)*, ser. OASICS, vol. 22. Schloss Dagstuhl - Leibniz Zentrum für Informatik, 2012, pp. 43–52.