Optimal Online Algorithms for the Multi-Objective Time Series Search Problem

Shun Hasegawa  
hasegawa.s.aj@m.titech.ac.jp  
Department of Computer Science  
Tokyo Institute of Technology

Toshiya Itoh  
titoh@ip.titech.ac.jp  
Department of Information Processing  
Tokyo Institute of Technology

Abstract: Tiedemann, et al. [Proc. of WALCOM, LNCS 8973, 2015, pp.210-221] defined multi-objective online problems and the competitive analysis for multi-objective online problems, and showed best possible online algorithms with respect to several measures of the competitive analysis. In this paper, we first point out that the definitions and frameworks of the competitive analysis due to Tiedemann, et al. do not necessarily capture the efficiency of online algorithms for multi-objective online problems and provide modified definitions of the competitive analysis for multi-objective online problems. Under the modified framework, we present a simple online algorithm Balanced Price Policy ($\text{BPP}_k$) for the multi-objective ($k$-objective) time series search problem, and show that the algorithm $\text{BPP}_k$ is best possible with respect to any measure of the competitive analysis (defined by a monotone function $f$). For the modified framework, we also derive best possible values of the competitive ratio for the multi-objective time series search problem with respect to several representative measures of the competitive analysis.

Key Words: Multi-Objective Online Algorithms, Worst Component Competitive Ratio, Arithmetic Mean Component Competitive Ratio, Geometric Mean Component Competitive Ratio, Best Component Competitive Ratio.

1 Introduction

Single-objective online optimization problems are fundamental in computing, communicating, and other practical systems. To measure the efficiency of online algorithms for single-objective online optimization problems, a notion of competitive analysis was introduced by Sleator and Tarjan [7], and since then extensive research has been made for diverse areas, e.g., paging and caching (see [9] for a survey), metric task systems (see [5] for a survey), asset conversion problems (see [6] for a survey), buffer management of network switches (see [4] for a survey), etc. All of these are single-objective online problems. In practice, there are many online problems of multi-objective nature, but we have no general framework of competitive analysis and no definition of competitive ratio for multi-objective online problems. Tiedemann, et al. [8] first introduced a framework of multi-objective online problems as the online version of multi-objective optimization problems [2] and formulated a notion of the competitive ratio for multi-objective online problems by extending the competitive ratio for single-objective online problems. To define the competitive ratio for multi-objective ($k$-objective) online problems, Tiedemann, et al. [8] regarded multi-objective online problems as a family of (possibly dependent) single-objective online problems and applied a monotone function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ to the family of the single-objective online problems. Given an algorithm $\text{ALG}$ for a multi-objective ($k$-objective) online problem, we regard $\text{ALG}$ as a family of algorithms $\text{ALG}_i$ for the $i$th objective of the input sequence and let $c_i$ be the competitive ratio of the algorithm $\text{ALG}_i$. For the set $\{c_1, \ldots, c_k\}$ of $k$ competitive ratios, the algorithm $\text{ALG}$ is $f(c_1, \ldots, c_k)$-competitive with respect to a monotone function $f : \mathbb{R}^k \rightarrow \mathbb{R}$. In fact, Tiedemann, et al. [8] defined the worst component competitive ratio by a function $f_1(c_1, \ldots, c_k) = \max(c_1, \ldots, c_k)$, the arithmetic mean component competitive ratio by...
a function $f_2(c_1, \ldots, c_k) = (c_1 + \cdots + c_k)/k$, and the geometric mean component competitive ratio by a function $f_3(c_1, \ldots, c_k) = (c_1 \times \cdots \times c_k)^{1/k}$. Note that all of the functions $f_1$, $f_2$, and $f_3$ are continuous on $\mathbb{R}^k$ and monotone.

### 1.1 Previous Work

El-Yaniv, et al. [3] initially investigated the single-objective time series search problem. For the single-objective time series search problem, prices are revealed time by time and the goal of the algorithm is to select one of them as with high price as possible. Assume that $m > 0$ and $M > m$ are the minimum and maximum values of possible prices, respectively, and let $\phi = M/m$ be the fluctuation ratio of possible prices. Under the assumption that $M > m > 0$ are known to online algorithms, El-Yaniv, et al. [3] presented a deterministic algorithm reservation price policy RPP, which is shown to be $\sqrt{\phi}$-competitive and best possible, and a randomized algorithm exponential threshold EXPO, which is shown to be $O(\log \phi)$-competitive.

In a straightforward manner, Tiedemann, et al. [8] generalized the single-objective time series search problem and defined the multi-objective time series search problem. For the multi-objective (k-objective) time series search problem, a vector $\vec{p} = (p_1, \ldots, p_k)$ of $k$ (possibly dependent) prices are revealed time by time and the goal of the algorithm is to select one of the price vectors as with low competitive ratio as possible with respect to the monotone function $f : \mathbb{R}^k \to \mathbb{R}$. For each $1 \leq i \leq k$, assume that $m_i > 0$ and $M_i > m_i$ are the minimum and maximum values of possible prices for the $i$th objective, respectively, and $m_i, M_i$ are known to online algorithms. For each $i \in [1, k]$, we use $\text{ITV}_i = [m_i, M_i]$ to denote an interval of the prices for the $i$th objective. For the case that all of $\text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k]$ are real intervals, Tiedemann, et al. [8] presented best possible online algorithms for the multi-objective time series search problem with respect to the monotone functions $f_1$, $f_2$, and $f_3$, i.e., a best possible online algorithm for the multi-objective (k-objective) time series search problem with respect to the monotone function $f_1$ [8, Theorems 1 and 2], a best possible online algorithm for the bi-objective time series search problem with respect to the monotone function $f_2$ [8, Theorems 3 and 4] and a best possible online algorithm for the bi-objective time series search problem with respect to the monotone function $f_3$ [8 §3.2]. Note that the proofs of these results are correct under the assumption that all of $\text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k]$ are real intervals.

### 1.2 Our Contribution

We first observe that the definition and framework of competitive analysis given by Tiedemann, et al. [8, Definitions 1, 2, and 3] do not necessarily capture the efficiency of algorithms for multi-objective online problems. Then we introduce modified definition and framework of competitive analysis for multi-objective online problems.

As mentioned in Subsection 1.1, Tiedemann, et al. [8] showed best possible online algorithms for the multi-objective time series search problem with respect to the monotone continuous functions $f_1$, $f_2$ and $f_3$ under the assumption that all of $\text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k]$ are real intervals, however, the optimality for the algorithm with respect to each of the monotone continuous functions $f_1$, $f_2$ and $f_3$ is discussed separately and independently. In this paper, we present a simple online algorithm Balanced Price Policy (BPP$_k$) for the multi-objective time series search problem with respect to any monotone function $f : \mathbb{R}^k \to \mathbb{R}$ and then show that under the modified framework of competitive analysis, the algorithm BPP$_k$ is best possible for any monotone (not necessarily continuous) function $f : \mathbb{R}^k \to \mathbb{R}$ even if all of $\text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k]$ are not necessarily real intervals (in Theorem 1.1). In the case...
that all of $\text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k]$ are real intervals, we exactly formulate the competitive ratio of the algorithm $\text{BPP}_k$ for any monotone function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ (in Theorems 4.2 and 4.3). With respect to the existing monotone continuous functions $f_1, f_2,$ and $f_3,$ we derive the best possible values of the competitive ratio for the multi-objective time series search problem under the modified framework of competitive analysis in Theorems 5.1, 5.2, and 5.3 respectively. With respect to a new monotone function $f_4(c_1, \ldots, c_k) = \min(c_1, \ldots, c_k),$ we also derive the best possible value of the competitive ratio for the multi-objective time series search problem under the modified framework of competitive analysis in Theorem 5.4.

From Theorems 4.2 and 4.3 we note that (1) Theorem 5.1 gives another proof for the result that the algorithm in [8] Theorem 1] is best possible for the multi-objective time series search problem with respect to $f_1,$ (2) Theorem 5.2 disproves the result that the algorithm in [8, Theorem 3] is best possible for the bi-objective time series search problem with respect to $f_2,$ and (3) Theorem 5.3 gives a best possible online algorithm for the multi-objective time series search problem with respect to $f_3,$ which is an extension of the result that the algorithm in [8, Theorem 3] is best possible for the bi-objective time series search problem with respect to $f_3.$

2 Preliminaries

For the subsequent discussions, we present some notations and terminologies. For any pair of integers $a \leq b,$ we use $[a, b]$ to denote a set $\{a, \ldots, b\}$ and for any pair of vectors $\vec{x} = (x_1, \ldots, x_k) \in \mathbb{R}^k$ and $\vec{y} = (y_1, \ldots, y_k) \in \mathbb{R}^k,$ we use $\vec{x} \preceq \vec{y}$ to denote a componentwise order, i.e., $x_i \leq y_i$ for each $i \in [1, k].$ It is immediate that $\preceq$ is a partial order on $\mathbb{R}^k.$ A function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ is said to be monotone if $f(\vec{x}) \leq f(\vec{y})$ for any pair of vectors $\vec{x} \in \mathbb{R}^k$ and $\vec{y} \in \mathbb{R}^k$ such that $\vec{x} \preceq \vec{y}.$

2.1 Multi-Objective Online Problems

Tiedemann, et al. [8] formulated a framework of multi-objective online problems by using that of multi-objective optimization problems [2]. In this subsection, we present multi-objective maximization problems (multi-objective minimization problems can be defined analogously).

Let $\mathcal{P}_k = (\mathcal{I}, \mathcal{X}, h)$ be a multi-objective optimization (maximization) problem, where $\mathcal{I}$ is a set of inputs, $\mathcal{X}(I) \subseteq \mathbb{R}^k$ is a set of feasible solutions for each input $I \in \mathcal{I},$ and $h : \mathcal{I} \times \mathcal{X} \rightarrow \mathbb{R}^k$ is a function such that $h(I, \vec{x}) \in \mathbb{R}^k$ represents the objective of each solution $\vec{x} \in \mathcal{X}(I).$ For an input $I \in \mathcal{I},$ an algorithm $\text{ALG}_k$ for $\mathcal{P}_k$ computes a feasible solution $\text{ALG}_k[I] \in \mathcal{X}(I).$ For an input $I \in \mathcal{I}$ and each feasible solution $\text{ALG}_k[I] \in \mathcal{X}(I),$ let $\text{ALG}_k(I) = h(I, \text{ALG}_k[I]) \in \mathbb{R}^k$ be the objective associate with $\text{ALG}_k[I].$ We say that a feasible solution $\vec{x}_{\text{max}} \in \mathcal{X}(I)$ is maximal if there exists no feasible solution $\vec{x} \in \mathcal{X}(I) \setminus \{\vec{x}_{\text{max}}\}$ such that $h(I, \vec{x}) \preceq h(I, \vec{x}_{\text{max}})$ and say that an algorithm $\text{OPT}_k$ for $\mathcal{P}_k$ is optimal if for any input $I \in \mathcal{I},$ $\text{OPT}_k[I] \subseteq \mathbb{R}^k$ is the set of maximal solutions for the input $I \in \mathcal{I},$ i.e., $\text{OPT}_k[I] = \{\vec{x} \in \mathcal{X}(I) : \vec{x} \text{ is a maximal solution for } I \in \mathcal{I}\}.$ We use $\text{OPT}_k(\vec{x}) \in \mathbb{R}^k$ to denote the objective associated with a solution $\vec{x} \in \text{OPT}_k[I].$

A multi-objective online problem can be defined in a way similar to a single-objective online problem [1]. We regard a multi-objective online problem as a multi-objective optimization problem in which the input is revealed bit by bit and an output must be produced in an online manner, i.e., after each new part of input is revealed, a decision affecting the output must be made.

2.2 Competitive Analysis for Multi-Objective Online Problems

Tiedemann, et al. [8] defined a notion of competitive analysis for multi-objective online problems. In this subsection, we introduce the notion of competitive analysis for multi-objective on-
line problems with respect to maximization problems (it is straightforward that the corresponding minimization problem can be defined analogously).

**Definition 2.1** [5]: Let $\mathcal{P}_k = (\mathcal{I}, \mathcal{X}, h)$ be a multi-objective optimization problem. For a vector $\vec{c} = (c_1, \ldots, c_k) \in \mathbb{R}^k$, we say that a multi-objective online algorithm $\text{ALG}_k$ for $\mathcal{P}_k$ is $\vec{c}$-competitive if for every input sequence $I \in \mathcal{I}$, there exists a maximal solution $\bar{x} \in \text{OPT}_k[I]$ such that

$$\bigwedge_{i \in [1,k]} [\text{OPT}_k(\bar{x})_i \leq c_i \cdot \text{ALG}_k(I)_i + \alpha_i],$$

where $\alpha = (\alpha_1, \ldots, \alpha_k) \in \mathbb{R}^k$ is a constant vector independent of input sequences $I \in \mathcal{I}$.

It should be noted that for multi-objective online algorithms, the notion of $\vec{c}$-competitive is defined by a vector $\vec{c} = (c_1, \ldots, c_k) \in \mathbb{R}^k$, while for single-objective online algorithms, the notion of $c$-competitive is defined by a single scalar $c \geq 1$.

**Definition 2.2** [5]: Let $\mathcal{P}_k = (\mathcal{I}, \mathcal{X}, h)$ be a multi-objective optimization problem. For a vector $\vec{c} = (c_1, \ldots, c_k) \in \mathbb{R}^k$, we say that a multi-objective online algorithm $\text{ALG}_k$ for $\mathcal{P}_k$ is strongly $\vec{c}$-competitive if for every input sequence $I \in \mathcal{I}$ and every maximal solution $\bar{x} \in \text{OPT}_k[I]$, the competitive ratio, the mean component, the best component, and the geometric mean component competitive ratio, the arithmetic mean component competitive ratio, the geometric mean component competitive ratio, and the best component competitive ratio, respectively. Note that all of the monotone functions $f_1$, $f_2$, $f_3$, and $f_4$ are continuous on $\mathbb{R}^k$ for any $k \geq 1$. 

Let $f : \mathbb{R}^k \to \mathbb{R}$ be a monotone function. For a multi-objective online algorithm $\text{ALG}_k$ for $\mathcal{P}_k$, the competitive ratio of $\text{ALG}_k$ with respect to $f$ is the infimum of $f(\vec{c})$ over all possible vectors $\vec{c} = (c_1, \ldots, c_k) \in \mathbb{R}^k$ such that $\text{ALG}_k$ is $\vec{c}$-competitive. Let $C[\text{ALG}_k]$ be the set of all possible vectors $\vec{c} = (c_1, \ldots, c_k) \in \mathbb{R}^k$ such that $\text{ALG}_k$ is $\vec{c}$-competitive and $C_s[\text{ALG}_k]$ be the set of all possible vectors $\vec{c} = (c_1, \ldots, c_k) \in \mathbb{R}^k$ such that $\text{ALG}_k$ is strongly $\vec{c}$-competitive, i.e.,

$$C[\text{ALG}_k] = \{ \vec{c} \in \mathbb{R}^k : \text{ALG}_k \text{ is } \vec{c}-\text{competitive} \};$$

$$C_s[\text{ALG}_k] = \{ \vec{c} \in \mathbb{R}^k : \text{ALG}_k \text{ is strongly } \vec{c}-\text{competitive} \}.$$
2.3 Multi-Objective Time Series Search Problem

A single-objective time series search problem is initially investigated by El-Yaniv, et al. [3] and it is defined as follows: An online player ALG is searching for the maximum price in a sequence of prices. At the beginning of each time period $t \in [1, T]$, a price $p_t$ is revealed to the online player ALG and it must decide whether to accept or reject the price $p_t$. If the online player ALG accepts the price $p_t$, then the game ends and the return for ALG is $p_t$. We assume that prices are chosen from the interval $\text{ITV} = [m, M]$, where $0 < m \leq M$, and that $m$ and $M$ are known to the online player ALG. If the online player ALG rejects the price $p_t$ for every $t \in [1, T]$, then the return for ALG is defined to be $m$. A multi-objective time series search problem [8] can be defined by a natural extension of the single-objective time series search problem.

In a multi-objective time series search problem, a price vector $\vec{p}_t = (p^1_t, \ldots, p^k_t) \in \mathbb{R}^k$ is revealed to the online player $\text{ALG}_k$ at the beginning of each time period $t \in [1, T]$, and the online player $\text{ALG}_k$ must decide whether to accept or reject the price vector $\vec{p}_t$. If the online player $\text{ALG}_k$ accepts the price vector $\vec{p}_t$, then the game ends and the return for $\text{ALG}_k$ is $\vec{p}_t$. As in the case of a single-objective time series search problem, assume that prices $p^i_t$ are chosen from the interval $\text{ITV}_i = [m_i, M_i]$ with $0 < m_i \leq M_i$ for each $i \in [1, k]$, and that the online player $\text{ALG}_k$ knows $m_i$ and $M_i$ for each $i \in [1, k]$. If the online player $\text{ALG}_k$ rejects the price vector $\vec{p}_t$ for every $t \in [1, T]$, then the return for of the online player $\text{ALG}_k$ is defined to be the minimum price vector $\vec{p}_{\text{min}} = (m_1, \ldots, m_k)$. Without loss of generality, we assume that $M_1/m_1 \geq \cdots \geq M_k/m_k$.

3 Observations on the Competitive Analysis

For the multi-objective ($k$-objective) time series search problem, it is natural to regard that $m_i$ and $M_i$ are part of the problem (not part of input sequences) for each $i \in [1, k]$. By setting $\alpha_i = M_i$ (as a constant independent of input sequences) for each $i \in [1, k]$, we can take $c_1 = \cdots = c_k = 0$ in Definitions 2.1 and 2.2. This implies that any algorithm ALG for the multi-objective ($k$-objective) time series search problem is $(0, \ldots, 0)$-competitive, i.e., for any monotone function $f : \mathbb{R}^k \to \mathbb{R}$, the competitive ratio of the algorithm ALG is $f(0, \ldots, 0)$. Thus in Definitions 2.1 and 2.2 we fix $\alpha_i = 0$ for each $i \in [1, k]$.

For simplicity, assume that $k = 2$ and $I_1 = I_2 = [m, M]$, where $0 < m < M$. Consider a simple algorithm $\text{ALG}_2$ that accepts the first price vector for any input sequence and observe how the competitive analysis for the algorithm $\text{ALG}_2$ works in the following examples:

Example 3.1: Let $I_1 = \{s_1, s_2\}$ be the set of input sequences. In the input sequence $s_1$, price vectors $\vec{p}_1 = (m, M)$, $\vec{p}_2 = (M, m)$, and $\vec{p}_3 = (m, m)$ are revealed to the algorithm $\text{ALG}_2$ at $t = 1$, $t = 2$, and $t = 3$, respectively, and in the input sequence $s_2$, price vectors $\vec{q}_1 = (M, m)$, $\vec{q}_2 = (m, m)$, and $\vec{q}_3 = (m, M)$ are revealed to the algorithm $\text{ALG}_2$ at $t = 1$, $t = 2$, and $t = 3$, respectively. For the input sequence $s_1$, the algorithm $\text{ALG}_2$ accepts $\vec{p}_1 = (m, M)$ which is maximal in $s_1$ and for the input sequence $s_2$, the algorithm $\text{ALG}_2$ accepts $\vec{p}_2 = (M, m)$ which is also maximal in $s_2$. From Definition 2.2, we have that the algorithm $\text{ALG}_2$ is strongly $(\frac{M}{m}, \frac{M}{m})$-competitive.

Example 3.2: Let $I_2 = \{\sigma\}$ be the set of input sequences. In the input sequence $\sigma$, price vectors $\vec{r}_1 = (m, M)$, $\vec{r}_2 = (m, M)$, and $\vec{r}_3 = (M, m)$ are revealed at $t = 1$, $t = 2$, and $t = 3$ to the algorithm $\text{ALG}_2$, respectively. The algorithm $\text{ALG}_2$ accepts $\vec{r}_1 = (m, m)$ which is not maximal in $\sigma$. From Definition 2.2 we have that the algorithm $\text{ALG}_2$ is strongly $(\frac{M}{m}, \frac{M}{m})$-competitive.

---

1 It is possible to show that if only the fluctuation ratio $\phi = M/m$ is known (but not $m$ or $M$) to the online player ALG, then no better competitive ratio than the trivial one of $\phi$ is achievable.
In Example 3.1, the algorithm \( \text{ALG}_2 \) accepts price vectors which are maximal in the input sequences \( s_1 \) and \( s_2 \), however, in Example 3.2, the algorithm \( \text{ALG}_2 \) accepts a price vector which is not maximal in the input sequence \( \sigma \). Thus it follows that for any monotone function \( f : \mathbb{R}^2 \to \mathbb{R} \), the strong competitive ratio of the algorithm \( \text{ALG}_2 \) is \( f(M/m, M/m) \) for both Examples 3.1 and 3.2 which does not necessarily capture the efficiency of online algorithms. To derive a more realistic framework, we need to modify the definition of competitive ratio.

Let \( \text{ALG}_k \) be an online algorithm for a multi-objective optimization (maximization) problem \( \mathcal{P}_k \). We use \( CR_f(\text{ALG}_k; I) \) to denote the competitive ratio of the algorithm \( \text{ALG}_k \) for an input sequence \( I \in I \) with respect to a monotone function \( f : \mathbb{R}^k \to \mathbb{R} \), i.e.,

\[
CR_f(\text{ALG}_k; I) = \sup_{\vec{x} \in \text{OPT}[I]} \frac{f(\text{OPT}_1(\vec{x}), \ldots, \text{OPT}_k(\vec{x}))}{f(\text{ALG}_1(I), \ldots, \text{ALG}_k(I))}.
\]

**Definition 3.1:** Let \( \text{ALG}_k \) be a multi-objective online algorithm for \( \mathcal{P}_k \). The competitive ratio of the algorithm \( \text{ALG}_k \) with respect to a monotone function \( f : \mathbb{R}^k \to \mathbb{R} \) is

\[
CR_f(\text{ALG}_k) = \sup_{I \in I} CR_f(\text{ALG}_k; I).
\]

It is easy to see that for the case that all of \( \text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k] \) are real intervals, all of the analyses on the competitive ratio by Tiedemann, et al. [8] hold under Definition 3.1. In the rest of the paper, we analyze the algorithms under Definition 3.1.

### 4 Online Algorithm: Balanced Price Policy

As mentioned in Section 1, Tiedemann, et al. [8] presented some online algorithms for the multi-objective \((k\text{-objective})\) time series search problem and analyzed the competitive ratio of those algorithms with respect to the monotone functions \( f_1, f_2, \) and \( f_3 \). The competitive analysis given in [8] heavily depends on the fact that the monotone functions \( f_1, f_2, \) and \( f_3 \) are continuous and the assumption that all of \( \text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k] \) are real intervals.

In this section, we present a simple online algorithm Balanced Price Policy \( \text{BPP}_k \) (in Figure 1) for the multi-objective \((k\text{-objective})\) time series search problem with respect to an arbitrary monotone function \( f : \mathbb{R}^k \to \mathbb{R} \).

![Figure 1: Balanced Price Policy \( \text{BPP}_k \)](image)

#### 4.1 General Case

In this subsection, we do not assume that all of \( \text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k] \) are real intervals (in fact, \( \text{ITV}_1 = [m_1, M_1] \) is allowed to be an integral interval) and we deal with any monotone (not necessarily continuous) function \( f : \mathbb{R}^k \to \mathbb{R} \).
Theorem 4.1: Let \( \text{ALG}_k \) be an arbitrary online algorithm for the multi-objective \((k\text{-objective})\) time series search problem. Then \( \mathcal{CR}^f(\text{BPP}_k) \leq \mathcal{CR}^f(\text{ALG}_k) \) for any monotone (not necessarily continuous) function \( f : \mathbb{R}^k \to \mathbb{R} \) and any integer \( k \geq 1 \).

Proof: We use \( I = (\vec{p}_1, \ldots, \vec{p}_T) \) to denote an arbitrary input sequence, where \( \vec{p}_t = (p^1_t, \ldots, p^k_t) \in \text{ITV}_1 \times \cdots \times \text{ITV}_k \) for each \( t \in [1, T] \). Let \( \mathcal{I} \) be the set of input sequences. Define \( \mathcal{I}_{\text{acc}} \subseteq \mathcal{I} \) to be the set of input sequences accepted by the algorithm \( \text{BPP}_k \) and \( \mathcal{I}_{\text{rej}} \subseteq \mathcal{I} \) to be the set of input sequences rejected by the algorithm \( \text{BPP}_k \), i.e.,

\[
\mathcal{I}_{\text{acc}} = \left\{ (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I} : \bigvee_{t \in [1, T]} \left[ f \left( \frac{M_1}{p^1_t}, \ldots, \frac{M_k}{p^k_t} \right) \leq f \left( \frac{p^1_t}{m_1}, \ldots, \frac{p^k_t}{m_k} \right) \right] \right\} ;
\]

\[
\mathcal{I}_{\text{rej}} = \left\{ (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I} : \bigwedge_{t \in [1, T]} \left[ f \left( \frac{M_1}{p^1_t}, \ldots, \frac{M_k}{p^k_t} \right) > f \left( \frac{p^1_t}{m_1}, \ldots, \frac{p^k_t}{m_k} \right) \right] \right\} .
\]

Let \( \text{ALG}_k \) be an arbitrary online algorithm for the multi-objective time series search problem. For each \( I = (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I}_{\text{acc}} \), the algorithm \( \text{BPP}_k \) halts at the earliest time \( t[I] \in [1, T] \) to accept a price vector \( \vec{p}_{t[I]} = (p^1_{t[I]}, \ldots, p^k_{t[I]}) \) such that

\[
f \left( \frac{M_1}{p^1_{t[I]}}, \ldots, \frac{M_k}{p^k_{t[I]}} \right) \leq f \left( \frac{p^1_{t[I]}}{m_1}, \ldots, \frac{p^k_{t[I]}}{m_k} \right),
\]

and let \( I^* = (\vec{p}_1, \ldots, \vec{p}_T) \), where \( \vec{p}_{t[I]} \) is the set of input sequences. Define \( \mathcal{I}_{*} \subseteq \mathcal{I}_{\text{acc}} \). For each \( I = (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I}_{\text{acc}} \), it is immediate that \( I^* = (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I}_{\text{acc}} \) and

\[
\mathcal{CR}^f(\text{BPP}_k; I) = \max_{x \in \text{OPT}_k[I]} f \left( \frac{\text{OPT}_k(x)_1}{\text{BPP}_k(I)_1}, \ldots, \frac{\text{OPT}_k(x)_k}{\text{BPP}_k(I)_k} \right)
\]

\[
= \max_{x \in \text{OPT}_k[I]} f \left( \frac{\text{OPT}_k(x)_1}{p^1_{t[I]}}, \ldots, \frac{\text{OPT}_k(x)_k}{p^k_{t[I]}} \right)
\]

\[
\leq f \left( \frac{M_1}{p^1_{t[I]}}, \ldots, \frac{M_k}{p^k_{t[I]}} \right) = \mathcal{CR}^f(\text{BPP}_k; I^*),
\]

(1)

where the inequality follows from the assumption that \( f : \mathbb{R}^k \to \mathbb{R} \) is monotone. Let \( \mathcal{I}^*_{\text{acc}} = \{ I^* = (\vec{p}_1, \ldots, \vec{p}_T) : I \in \mathcal{I}_{\text{acc}} \} \). Note that \( \mathcal{I}^*_{\text{acc}} \subseteq \mathcal{I}_{\text{acc}} \). For each \( J^* = (\vec{p}, \vec{p}_{\text{max}}) \in \mathcal{I}^*_{\text{acc}} \), define \( J^* \) according to how the algorithm \( \text{ALG}_k \) works on receiving the price vector \( \vec{p} = (p^1, \ldots, p^k) \). For the case that the algorithm \( \text{ALG}_k \) accepts the price vector \( \vec{p} \), let \( J^* = (\vec{p}, \vec{p}_{\text{max}}) \) and we have that

\[
\mathcal{CR}^f(\text{BPP}_k; J^*) = f \left( \frac{M_1}{p^1}, \ldots, \frac{M_k}{p^k} \right) = \mathcal{CR}^f(\text{ALG}_k; J^*).
\]

For the case that the algorithm \( \text{ALG}_k \) rejects the price vector \( \vec{p} \), let \( J^* = (\vec{p}) \) and we have that

\[
\mathcal{CR}^f(\text{BPP}_k; J^*) = f \left( \frac{M_1}{p^1}, \ldots, \frac{M_k}{p^k} \right) \leq f \left( \frac{p^1}{m_1}, \ldots, \frac{p^k}{m_k} \right) = \mathcal{CR}^f(\text{ALG}_k; J^*),
\]

where the inequality is due to the assumption that \( J^* = (\vec{p}, \vec{p}_{\text{max}}) \in \mathcal{I}_{\text{acc}} \), i.e., the algorithm \( \text{BPP}_k \) accepts \( \vec{p} = (p^1, \ldots, p^k) \) by the condition that \( f(M_1/p^1, \ldots, M_k/p^k) \leq f(p^1/m_1, \ldots, p^k/m_k) \). Thus for each \( I \in \mathcal{I}_{\text{acc}} \), there exists a price vector \( I^* \in \mathcal{I} \) such that

\[
\mathcal{CR}^f(\text{BPP}_k; I) \leq \mathcal{CR}^f(\text{ALG}_k; I^*).
\]

(2)
For each \( I = (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I}_{\text{rej}} \), the algorithm BPP\(_k\) rejects a price vector \( \vec{p}_t \) for every \( t \in [1, T] \), i.e., \( f(M_1/p^t_1, \ldots, M_k/p^t_k) > f(p^1_1/m_1, \ldots, p^k_k/m_k) \) for every \( t \in [1, T] \), and settles in the minimum price vector \( \vec{p}_{\text{min}} = (m_1, \ldots, m_k) \). At time \( \tau[I] \in [1, T] \), however, the optimal offline algorithm OPT\(_k\) can accept a price vector \( \vec{p}_{\tau[I]} = (p^1_{\tau[I]}, \ldots, p^k_{\tau[I]}) \) such that

\[
f \left( \frac{p^1_{\tau[I]}}{m_1}, \ldots, \frac{p^k_{\tau[I]}}{m_k} \right) > f \left( \frac{p^1_1}{m_1}, \ldots, \frac{p^k_k}{m_k} \right),
\]

and let \( I^* = (\vec{p}_{\tau[I]}) \). For each \( I = (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I}_{\text{rej}} \), it is immediate that \( I^* = (\vec{p}_{\tau[I]}) \in \mathcal{I}_{\text{rej}} \) and

\[
\mathcal{CR}^I(\text{BPP}_k; I) = \max_{\vec{x} \in \text{OPT}_k[I]} f \left( \frac{\text{OPT}_k(\vec{x})_1}{\text{BPP}_k(I)_1}, \ldots, \frac{\text{OPT}_k(\vec{x})_k}{\text{BPP}_k(I)_k} \right)
\]

\[
= \max_{\vec{x} \in \text{OPT}_k[I]} f \left( \frac{\text{OPT}_k(\vec{x})_1}{m_1}, \ldots, \frac{\text{OPT}_k(\vec{x})_k}{m_k} \right)
\]

\[
= f \left( \frac{\vec{p}_{\tau[I]}^1}{m_1}, \ldots, \frac{\vec{p}_{\tau[I]}^k}{m_k} \right) = \mathcal{CR}^I(\text{BPP}_k; I^*). \tag{3}
\]

Let \( \mathcal{I}_{\text{rej}}^* = \{I^* = (\vec{p}_{\tau[I]}) : I \in \mathcal{I}_{\text{rej}}\} \). Note that \( \mathcal{I}_{\text{rej}}^* \subseteq \mathcal{I}_{\text{rej}} \). For each \( J^* = (\vec{p}) \in \mathcal{I}_{\text{rej}}^* \), define \( J^* \) according to how the algorithm ALG\(_k\) works on receiving the price vector \( \vec{p} = (p^1, \ldots, p^k) \). For the case that the algorithm ALG\(_k\) accepts the price vector \( \vec{p} \), let \( J^* = (\vec{p}, \vec{p}_{\text{max}}) \) and we have that

\[
\mathcal{CR}^J(\text{BPP}_k; J^*) = f \left( \frac{p^1}{m_1}, \ldots, \frac{p^k}{m_k} \right) < f \left( \frac{M_1}{p^1}, \ldots, \frac{M_k}{p^k} \right) = \mathcal{CR}^J(\text{ALG}_k; J^*),
\]

where the inequality is due to the assumption that \( J^* = (\vec{p}) \in \mathcal{I}_{\text{rej}} \), i.e., the algorithm BPP\(_k\) rejects \( \vec{p} = (p^1, \ldots, p^k) \) by the condition that \( f(M_1/p^1, \ldots, M_k/p^k) > f(p^1/m_1, \ldots, p^k/m_k) \).

Thus for each \( I \in \mathcal{I}_{\text{rej}} \), there exists a price vector \( I' \in \mathcal{I} \) such that

\[
\mathcal{CR}^I(\text{BPP}_k; I) \leq \mathcal{CR}^I(\text{ALG}_k; I'). \tag{4}
\]

Then from Definition \( 3.4 \) it follows that

\[
\mathcal{CR}^I(\text{BPP}_k) = \sup_{I \in \mathcal{I}} \mathcal{CR}^I(\text{BPP}_k; I)
\]

\[
= \max \left\{ \sup_{I \in \mathcal{I}_{\text{acc}}} \mathcal{CR}^I(\text{BPP}_k; I), \sup_{I \in \mathcal{I}_{\text{rej}}} \mathcal{CR}^I(\text{BPP}_k; I) \right\}
\]

\[
\leq \max \left\{ \sup_{I' \in \mathcal{I}} \mathcal{CR}^I(\text{ALG}_k; I'), \sup_{I' \in \mathcal{I}} \mathcal{CR}^I(\text{ALG}_k; I') \right\} = \max_{I' \in \mathcal{I}} \mathcal{CR}^I(\text{ALG}_k; I') = \mathcal{CR}^I(\text{ALG}_k),
\]

where the inequality follows from Equations \( 2 \) and \( 4 \).
2 Special Case: Monotone Continuous Functions

In this subsection, we assume that all of \( \mathit{ITV}_1 = [m_1, M_1], \ldots, \mathit{ITV}_k = [m_k, M_k] \) are real intervals and deal with only monotone continuous functions \( f : \mathbb{R}^k \to \mathbb{R} \).

Let \( z_j^k = \sup_{(x_1, \ldots, x_k) \in \mathbb{R}^k} f(M_1/x_1, \ldots, M_k/x_k) \), where

\[
\mathcal{S}_j^k = \left\{(x_1, \ldots, x_k) \in \mathit{ITV}_1 \times \cdots \times \mathit{ITV}_k : f \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = f \left( \frac{x_1}{m_1}, \ldots, \frac{x_k}{m_k} \right) \right\}.
\]

By setting \( x_i = \sqrt{m_i M_i} \in I_i = [m_i, M_i] \) for each \( i \in [1, k] \), we have that

\[
f \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = f \left( \frac{M_1}{\sqrt{m_1 M_1}}, \ldots, \frac{M_k}{\sqrt{m_k M_k}} \right) = f \left( \frac{\sqrt{m_1 M_1}}{m_1}, \ldots, \frac{\sqrt{m_k M_k}}{m_k} \right),
\]

\[
f \left( \frac{x_1}{m_1}, \ldots, \frac{x_k}{m_k} \right) = f \left( \frac{\sqrt{m_1 M_1}}{m_1}, \ldots, \frac{\sqrt{m_k M_k}}{m_k} \right) = f \left( \frac{\sqrt{m_1 M_1}}{m_1}, \ldots, \frac{\sqrt{m_k M_k}}{m_k} \right).
\]

Thus, for any monotone continuous function \( f \), it follows that \( (\sqrt{m_1 M_1}, \ldots, \sqrt{m_k M_k}) \in \mathcal{S}_j^k \). So we have that \( \mathcal{S}_j^k \neq \emptyset \) and \( z_j^k = \sup_{(x_1, \ldots, x_k) \in \mathcal{S}_j^k} f(M_1/x_1, \ldots, M_k/x_k) \) is well-defined.

In this subsection, we show that the exact value of the competitive ratio of the algorithm \( \text{BPP}_k \) is \( z_j^k \) for any monotone continuous function \( f : \mathbb{R}^k \to \mathbb{R} \) and any integer \( k \geq 1 \) (Corollary 4.1). More precisely, we show that \( \text{CR}^f(\text{BPP}_k) \leq z_j^k \) (Theorem 4.2) and that \( \text{CR}^f(\text{ALG}_k) \geq z_j^k \) for any algorithm \( \text{ALG}_k \) (Theorem 4.3). From Theorem 4.1 and Corollary 4.1 it follows that \( z_j^k \) is the best possible value of the competitive ratio for the multi-objective time series search problem.

**Theorem 4.2:** If all of \( \mathit{ITV}_1 = [m_1, M_1], \ldots, \mathit{ITV}_k = [m_k, M_k] \) are real intervals, then for any monotone continuous function \( f : \mathbb{R}^k \to \mathbb{R} \) and any integer \( k \geq 1 \), \( \text{CR}^f(\text{BPP}_k) \leq z_j^k \).

**Proof:** Let \( I = (\vec{p}_1, \ldots, \vec{p}_T) \) be an arbitrary input sequence, where \( \vec{p}_t = (p_{1t}, \ldots, p_{kt}) \in \mathbb{R}^k \) for each \( t \in [1, T] \), and \( \mathcal{I} \) be the set of input sequences. As in the proof of Theorem 4.1, we consider the set \( \mathcal{I}_{\text{acc}} \subseteq \mathcal{I} \) of input sequences accepted by the algorithm \( \text{BPP}_k \) and the set \( \mathcal{I}_{\text{rej}} \subseteq \mathcal{I} \) of input sequences rejected by the algorithm \( \text{BPP}_k \).

For each \( I = (\vec{p}_1, \ldots, \vec{p}_T) \in \mathcal{I}_{\text{acc}} \), the algorithm \( \text{BPP}_k \) halts at the earliest time \( t[I] \in [1, T] \) to accept \( \vec{p}_{t[I]} = (p_{1t[I]}, \ldots, p_{kt[I]}) \) such that \( f(M_1/p_{1t[I]}, \ldots, M_k/p_{kt[I]}) \leq f(p_{1t[I]}/m_1, \ldots, p_{kt[I]}/m_k) \). Thus from Equation (11), we have that

\[
\text{CR}^f(\text{BPP}_k; I) \leq f \left( \frac{M_1}{p_{1t[I]}}, \ldots, \frac{M_k}{p_{kt[I]}} \right).
\]

To show that \( f(M_1/p_{1t[I]}, \ldots, M_k/p_{kt[I]}) \leq z_j^k \), we consider the following cases:

1. \( f(M_1/p_{1t[I]}, \ldots, M_k/p_{kt[I]}) = f(p_{1t[I]}/m_1, \ldots, p_{kt[I]}/m_k) \);
2. \( f(M_1/p_{1t[I]}, \ldots, M_k/p_{kt[I]}) < f(p_{1t[I]}/m_1, \ldots, p_{kt[I]}/m_k) \).

For the case (1), it is immediate that \( \vec{p}_{t[I]} \in \mathcal{S}_j^k \) and \( f(M_1/p_{1t[I]}, \ldots, M_k/p_{kt[I]}) \leq z_j^k \) by definition. For the case (2), let \( J = \{ j \in [1, k] : M_j/p_{jt[I]} \leq p_{jt[I]}/m_j \} \). We claim that \( J \neq \emptyset \). Assume for

---

2 By contradiction. If \( J = \emptyset \), then \( M_i/p_{it[I]} > p_{it[I]}/m_i \) for each \( i \in [1, k] \). Since the function \( f : \mathbb{R}^k \to \mathbb{R} \) is monotone, we have that \( f(M_1/p_{1t[I]}, \ldots, M_k/p_{kt[I]}) \geq f(p_{1t[I]}/m_1, \ldots, p_{kt[I]}/m_k) \), which contradicts the assumption that \( f(M_1/p_{1t[I]}, \ldots, M_k/p_{kt[I]}) < f(p_{1t[I]}/m_1, \ldots, p_{kt[I]}/m_k) \).
simplicity that $J = \{1, \ldots, u\}$ for $u \geq 1$. By setting $p^j_{t[I]} = m_j$ for each $j \in J$, we have that

$$f \left( \frac{M_1}{m_1}, \ldots, \frac{M_u}{m_u}, \frac{M_{u+1}}{p^k_{t[I]}}, \ldots, \frac{M_k}{p^k_{t[I]}} \right) \geq f \left( 1, \ldots, 1, \frac{p^k_{u+1}}{m_{u+1}}, \ldots, \frac{p^k_{t[I]}}{m_k} \right).$$

Since $f$ is monotone and continuous, there exist $q^j_{t[I]} \in [m_1, p^j_{t[I]}]$, $q^j_{t[I]} \in [m_k, p^k_{t[I]}]$ such that

$$f \left( \frac{M_1}{q^1_{t[I]}}, \ldots, \frac{M_u}{q^u_{t[I]}}, \frac{M_{u+1}}{q^u_{t[I]}}, \ldots, \frac{M_k}{q^k_{t[I]}} \right) \leq f \left( \frac{1}{m_1}, \ldots, \frac{1}{m_k} \right).$$

Then it turns out that $(q^1_{t[I]}, \ldots, q^u_{t[I]}, p^u_{t[I]}, \ldots, p^k_{t[I]}) \in S^k_f$ and it follows that

$$f \left( \frac{M_1}{q^1_{t[I]}}, \ldots, \frac{M_k}{q^k_{t[I]}} \right) \leq f \left( \frac{1}{m_1}, \ldots, \frac{1}{m_k} \right) \leq f \left( \frac{p^k_{t[I]}/m_1, \ldots, p^k_{t[I]}/m_k}{m_1}, \ldots, \frac{p^k_{t[I]}_k/m_k}{m_k} \right) \leq z^k_f.$$

For each $I = (\bar{p}_1, \ldots, \bar{p}_T) \in I_{rel}$, the algorithm BPP$_k$ rejects a price vector $\bar{p}_t$ for every $t \in [1, T]$, and settles in the minimum price vector $\bar{p}_{min} = (m_1, \ldots, m_k)$, but at time $\tau[I] \in [1, T]$, the optimal offline algorithm OPT$_k$ can accept a price vector $\bar{p}_t[I] = (p^1_t[I], \ldots, p^k_t[I])$ satisfying that $f(p^1_t[I]/m_1, \ldots, p^k_t[I]/m_k) = \max_{t \in [1, T]} f(p^1_t/I, \ldots, p^k_t/I)$. So from Equation (3), we have that

$$CR^f(BPP_k; I) = f \left( \frac{p^1_{t[I]}/m_1, \ldots, p^k_{t[I]}/m_k}{m_1}, \ldots, \frac{p^k_{t[I]}_k/m_k}{m_k} \right).$$

We show that $f(p^1_t[I]/m_1, \ldots, p^k_t[I]/m_k) \leq z^f$. Since the algorithm BPP$_k$ rejects a price vector $\bar{p}_t$ for every $t \in [1, T]$, it is immediate that $f(M_1/p^1_{t[I]}, \ldots, M_k/p^k_{t[I]}) \leq f(p^1_{t[I]}/m_1, \ldots, p^k_{t[I]}/m_k)$ by definition. Let $H = \{h \in [1, k] : M_h/p^k_{t[I]} \geq p^k_{t[I]}/m_h \}$. We claim that $H \neq \emptyset$. For simplicity, we assume that $H = \{1, \ldots, v\}$ for $v \geq 1$. By setting $p^1_{t[I]} = M_h$ for each $h \in H$, we have that

$$f \left( 1, \ldots, 1, \frac{M_{u+1}}{p^1_{t[I]}}, \ldots, \frac{M_k}{p^k_{t[I]}} \right) \leq f \left( \frac{M_1}{m_1}, \ldots, \frac{M_v}{m_v}, \frac{p^1_{t[I]}/m_1, \ldots, p^k_{t[I]}/m_k}{m_1}, \ldots, \frac{p^k_{t[I]}_k/m_k}{m_k} \right).$$

Since $f$ is monotone and continuous, there exist $q^1_{t[I]} \in [p^1_{t[I]}, M_1], \ldots, q^v_{t[I]} \in [p^v_{t[I]}, M_v]$ such that

$$f \left( \frac{p^1_{t[I]}/m_1, \ldots, p^k_{t[I]}/m_k}{m_1}, \ldots, \frac{p^k_{t[I]}_k/m_k}{m_k} \right) \leq f \left( \frac{q^1_{t[I]}/m_1, \ldots, q^v_{t[I]}/m_v, p^1_{t[I]}/m_1, \ldots, p^k_{t[I]}_k/m_k}{m_1}, \ldots, \frac{p^k_{t[I]}_k/m_k}{m_k} \right) \leq f \left( \frac{M_1}{m_1}, \ldots, \frac{M_k}{p^k_{t[I]}_k} \right).$$

Then it turns out that $(q^1_{t[I]}, \ldots, q^v_{t[I]}, p^v_{t[I]}, \ldots, p^k_{t[I]}) \in S^k_f$ and it follows that

$$f \left( \frac{p^1_{t[I]}/m_1, \ldots, p^k_{t[I]}/m_k}{m_1}, \ldots, \frac{p^k_{t[I]}_k/m_k}{m_k} \right) \leq f \left( \frac{q^1_{t[I]}/m_1, \ldots, q^v_{t[I]}/m_v, p^v_{t[I]}/m_1, \ldots, p^k_{t[I]}_k/m_k}{m_1}, \ldots, \frac{p^k_{t[I]}_k/m_k}{m_k} \right) \leq z^f.$$
Note that $I_{\text{acc}} \cap I_{\text{rej}} = \emptyset$ and $I_{\text{acc}} \cup I_{\text{rej}} = I$. Thus for any $I \in \mathcal{I}$, we have that $\mathcal{C}R^I(\text{BPP}_k; I) \leq z^k_I$ and we can conclude that $\mathcal{C}R^I(\text{BPP}_k) = \sup_{I \in \mathcal{I}} \mathcal{C}R^I(\text{BPP}_k; I) \leq z^k_I$. \hfill $\blacksquare$

**Theorem 4.3:** Let $\text{ALG}_k$ be an arbitrary online algorithm for the multi-objective ($k$-objective) time series search problem. If all of $\text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k]$ are real intervals, then for any monotone continuous function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ and any integer $k \geq 1$, $\mathcal{C}R^I(\text{ALG}_k) \geq z^k_I$.

**Proof:** Let $\text{ALG}_k$ be an arbitrarily online algorithm and $(x^*_1, \ldots, x^*_k) \in \mathcal{S}_k^k$ be a price vector such that $z^k_I = f(M_1/x^*_1, \ldots, M_k/x^*_k)$. The adversary reveals a price vector $\bar{p} = (x^*_1, \ldots, x^*_k)$. If the algorithm $\text{ALG}_k$ accepts $\bar{p}$, then the adversary reveals another price vector $\bar{p}_{\text{max}} = (M_1, \ldots, M_k)$ and accepts $\bar{p}_{\text{max}}$. Let $I = (\bar{p}, \bar{p}_{\text{max}})$ be an input sequence. Then we have that

$$\mathcal{C}R^I(\text{ALG}_k; I) = f \left( \frac{M_1}{x^*_1}, \ldots, \frac{M_k}{x^*_k} \right) = z^k_I.$$  

If the algorithm $\text{ALG}_k$ rejects $\bar{p}$, then the adversary accepts $\bar{p}$ but reveals no further price vectors until the algorithm $\text{ALG}_k$ settles in the minimum price vector $\bar{p}_{\text{min}} = (m_1, \ldots, m_k)$. Let $J = (\bar{p})$ be an input sequence. Note that $z^k_J = f(x^*_1/m_1, \ldots, x^*_k/m_k)$. Then we also have that

$$\mathcal{C}R^J(\text{ALG}_k; J) = f \left( \frac{x^*_1}{m_1}, \ldots, \frac{x^*_k}{m_k} \right) = z^k_J.$$  

Thus for any online algorithm $\text{ALG}_k$, it follows that $\mathcal{C}R^I(\text{ALG}_k) = \sup_{I \in \mathcal{I}} \mathcal{C}R^I(\text{ALG}_k; I) \geq z^k_I$. \hfill $\blacksquare$

From Theorems 4.2 and 4.3, we immediately have the following result.

**Corollary 4.1:** If all of $\text{ITV}_1 = [m_1, M_1], \ldots, \text{ITV}_k = [m_k, M_k]$ are real intervals, then for any monotone continuous function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ and any integer $k \geq 1$, $\mathcal{C}R^I(\text{BPP}_k) = z^k_I$.

### 4.3 Discussions

As mentioned in Subsection 1.1, El-Yaniv, et al. [3] presented the algorithm RPP (reservation price policy) for the single-objective time series search problem (see Figure 2). We refer to $p^*$ as the reservation price, where $p^*$ is the solution of $M/p = p/m$.

```
for t = 1, 2, \ldots, T do
    Accept $p_t$ if $p_t \geq p^* = \sqrt{Mm}$.
end
```

**Figure 2:** Reservation Price Policy: RPP

For the monotone continuous functions $f_1$, $f_2$, and $f_3$, we have that $f_1(x) = f_2(x) = f_3(x) = x$ if $k = 1$, and the algorithm BPP coincides with the algorithm RPP with respect to the functions $f_1$, $f_2$, and $f_3$, however, this is not necessarily the case for any nondecreasing continuous functions $f : \mathbb{R} \rightarrow \mathbb{R}$. Let us consider the following nondecreasing continuous function $g : \mathbb{R} \rightarrow \mathbb{R}$.

\footnote{For $k = 1$, it is obvious that any monotone continuous function $f : \mathbb{R} \rightarrow \mathbb{R}$ is nondecreasing and continuous.}
From the assumption that $0 < m < M$, it follows that $M/m > 1$ and we can take any constant $c$ such that $1 < c < \sqrt{M/m}$. Then it is immediate that

\[
\begin{align*}
g(M/p) &> g(p/m) & \text{for } m \leq p < \sqrt{Mm}/c; \\
g(M/p) &= g(p/m) & \text{for } \sqrt{Mm}/c \leq p \leq c\sqrt{Mm}; \\
g(M/p) &< g(p/m) & \text{for } c\sqrt{Mm} < p \leq M.
\end{align*}
\]

Thus the algorithm $bpp$ does not coincide with the algorithm $rpp$ with respect to the nondecreasing continuous (equivalently monotone) function $g : \mathbb{R} \to \mathbb{R}$ in Figure 3.

5 Analysis for Competitive Ratio

For the case that all of $\text{itv}_1 = [m_1, M_1], \ldots, \text{itv}_k = [m_k, M_k]$ are real intervals, Corollary 4.1 gives the best possible value of the competitive ratio for the multi-objective time series search problem with respect to any monotone continuous function $f$. In this section, we assume that all of $\text{itv}_1 = [m_1, M_1], \ldots, \text{itv}_k = [m_k, M_k]$ are real intervals, and derive the best possible values of the competitive ratio for the multi-objective time series search problem with respect to the monotone functions $f_1, f_2, f_3,$ and $f_4$ in Subsections 5.1, 5.2, 5.3, and 5.4 respectively.

5.1 Worst Component Competitive Ratio

In this subsection, we show that $\mathcal{CR}^{f_1}(bpp_k) = z_{f_1}^k = \max\{\sqrt{M_1/m_1, M_2/m_2}\}$. This implies that the algorithm $rpp$ with respect to the function $f_1(c_1, \ldots, c_k) = \max(c_1, \ldots, c_k)$. For the function $f_1$, let

\[
\mathcal{S}_{f_1}^k = \left\{(x_1, \ldots, x_k) \in I_1 \times \cdots \times I_k : \max\left(\frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k}\right) = \max\left(\frac{x_1}{m_1}, \ldots, \frac{x_k}{m_k}\right)\right\}.
\]
Thus \( z^k_{f_1} = \sup_{(x_1, \ldots, x_k) \in S^k_{f_1}} \left[ \max \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) \right] \).

**Theorem 5.1** \( z^k_{f_1} = \max \{ \sqrt{M_1/m_1}, M_2/m_2 \} \) for any integer \( k \geq 2 \).

**Proof:** Consider the following two cases: (1) \( \sqrt{M_1/m_1} \geq M_2/m_2 \) and (2) \( \sqrt{M_1/m_1} < M_2/m_2 \).

For the case (1), we further consider the following three subcases: (1.1) \( x_1 > \sqrt{m_1M_1} \), (1.2) \( x_1 < \sqrt{m_1M_1} \), and (1.3) \( x_1 = \sqrt{m_1M_1} \). For the subcase (1.1), we have that

\[
\begin{align*}
\frac{M_1}{x_1} &< \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\frac{M_2}{x_2} &\leq \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\vdots \\
\frac{M_k}{x_k} &\leq \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\Rightarrow f_1 \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = \max \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) &\leq \frac{\sqrt{M_1}}{m_1}.
\end{align*}
\]

Thus \( f_1(M_1/x_1, \ldots, M_k/x_k) < f_1(x_1/m_1, \ldots, x_k/m_k) \). For the subcase (1.2), we have that

\[
\begin{align*}
\frac{M_1}{x_1} &> \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\frac{M_2}{x_2} &\leq \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\vdots \\
\frac{M_k}{x_k} &\leq \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\Rightarrow f_1 \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = \max \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) &> \frac{\sqrt{M_1}}{m_1}.
\end{align*}
\]

Thus \( f_1(M_1/x_1, \ldots, M_k/x_k) > f_1(x_1/m_1, \ldots, x_k/m_k) \). For the subcase (1.3), we have that

\[
\begin{align*}
\frac{M_1}{x_1} &= \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\frac{M_2}{x_2} &\leq \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\vdots \\
\frac{M_k}{x_k} &\leq \frac{\sqrt{M_1}}{\sqrt{m_1M_1}} = \frac{\sqrt{M_1}}{m_1} \\
\Rightarrow f_1 \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = \max \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) &= \frac{\sqrt{M_1}}{m_1}.
\end{align*}
\]

Then for the case (1), we have that \( z^k_{f_1} = \sqrt{M_1/m_1} \), which is achieved at any \( \vec{p} = (x_1, \ldots, x_k) \in [m_1, M_1] \times \cdots \times [m_k, M_k] \) such that \( x_1 = \sqrt{m_1M_1} \in [m_1, M_1] \).
For the case (2), we consider the following two subcases: (2.1) \( x_1 \leq M_1 m_2 / M_2 \) and (2.2) \( x_1 \geq M_1 m_2 / M_2 \). Note that \( m_1 \leq M_1 m_2 / M_2 \leq M_1 \). For the subcase (2.1), we have that

\[
\begin{align*}
\frac{M_1}{x_1} > \frac{M_1 M_2}{m_2} = \frac{M_2}{m_2} \\
\vdots \\
\frac{M_k}{x_k} \leq \frac{M_k}{m_k} \leq \frac{M_k}{m_2} \\
\Rightarrow f_1 \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = \max \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) > \frac{M_2}{m_2};
\end{align*}
\]

and

\[
\begin{align*}
\frac{M_1}{m_1} < \frac{M_1 m_2}{m_2} < \left( \frac{M_1 m_2}{M_2} \right)^2 = \frac{M_2}{m_2} \\
\Rightarrow f_1 \left( \frac{x_1}{m_1}, \ldots, \frac{x_k}{m_k} \right) = \max \left( \frac{x_1}{m_1}, \ldots, \frac{x_k}{m_k} \right) \leq \frac{M_2}{m_2}.
\end{align*}
\]

Thus \( f_1(M_1/x_1, \ldots, M_k/x_k) > f_1(x_1/m_1, \ldots, x_k/m_k) \). For the subcase (2.2), we have that

\[
\begin{align*}
\frac{M_1}{x_1} \leq \frac{M_1 M_2}{m_2} = \frac{M_2}{m_2} \\
\vdots \\
\frac{M_k}{x_k} \leq \frac{M_k}{m_k} \leq \frac{M_k}{m_2} \\
\Rightarrow f_1 \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = \max \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) \leq \frac{M_2}{m_2},
\end{align*}
\]

which implies that \( z_{f_1} = \sup_{(x_1, \ldots, x_k) \in S_1^t} f_1(M_1/x_1, \ldots, M_k/x_k) \leq M_2/m_2 \). For the subcase (2.2), we show that \( z_{f_2} = M_2/m_2 \). Let \( x_1' = M_1 m_2 / M_2 \). Since \( M_1 / m_1 \geq M_2 / m_2 \), we have that \( x_1' \in [m_1, M_1] \), and from the assumption that \( \sqrt{M_1 m_1} < M_2 / m_2 \), we have that \( x_1'/m_1 < M_2/m_2 \). So from the fact that \( M_2/m_2 \geq M_1 / x_i \geq 1 \) and \( M_2/m_2 \geq x_i/m_i \) for each \( i \in [3, k] \), it follows that for \( x_1' = M_1 m_2 / M_2, x_2' = M_2 \in [m_2, M_2] \), and any \( x_3 \in [m_3, M_3], \ldots, x_k \in [m_k, M_k] \),

\[
\begin{align*}
\frac{M_1}{x_1'} > \frac{M_1 M_2}{m_2} M_3, \ldots, \frac{M_k}{x_k} \\
\Rightarrow f_1 \left( \frac{M_1}{x_1'}, \frac{M_2}{x_2'}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right) = \max \left( \frac{M_1}{x_1'}, \frac{M_2}{x_2'}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right)
&= \max \left( \frac{M_1}{x_1'}, \frac{M_2}{m_2}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right) \\
&= \max \left( \frac{M_2}{m_2}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right) = \frac{M_2}{m_2};
\end{align*}
\]

Let \( x_1'' = m_1 M_2 / m_2 \). Since \( M_1/m_1 \geq M_2 / m_2 \), we have that \( x_1'' \in [m_1, M_1] \), and from the assumption that \( \sqrt{M_1 m_1} < M_2 / m_2 \), we also have that \( x_1'' \geq M_1 m_2 / M_2 \) and \( M_1 / x_1'' < M_2/m_2 \). So from the fact that \( M_2/m_2 \geq M_1 / x_1 \) and \( M_2/m_2 \geq x_i/m_i \) for each \( i \in [3, k] \), it follows that for \( x_1'' = m_1 M_2 / m_2, x_2'' = m_2 \in [m_2, M_2] \), and any \( x_3 \in [m_3, M_3], \ldots, x_k \in [m_k, M_k] \),

\[
\begin{align*}
\frac{M_1}{x_1''}, \frac{M_2}{x_2''}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \\
\Rightarrow f_1 \left( \frac{M_1}{x_1''}, \frac{M_2}{x_2''}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right) = \max \left( \frac{M_1}{x_1''}, \frac{M_2}{x_2''}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right)
&= \max \left( \frac{M_1}{x_1''}, \frac{M_2}{m_2}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right) \\
&= \max \left( \frac{M_1}{x_1''}, \frac{M_2}{m_2}, \frac{M_3}{x_3}, \ldots, \frac{M_k}{x_k} \right) = \frac{M_2}{m_2};
\end{align*}
\]
Then for the case (2), we have that $z_{f_1}^k = M_2/m_2$, which is achieved at any $\vec{p} = (x_1, \ldots, x_k) \in [m_1, M_1] \times \cdots \times [m_k, M_k]$ such that $x_1 = M_1 m_2 / M_2 \in [m_1, M_1]$ and $x_2 = M_2 \in [m_2, M_2]$ or $x_1 = m_1 M_2 / m_2 \in [m_1, M_1]$ and $x_2 = m_2 \in [m_2, M_2]$.

Since we have that $z_{f_1}^k = \sqrt{M_1/m_1}$ for the case (1) $\sqrt{M_1/m_1} \geq M_2/m_2$ and $z_{f_1}^k = M_2/m_2$ for the case (2) $\sqrt{M_1/m_1} < M_2/m_2$, we can conclude that $z_{f_1}^k = \max\{\sqrt{M_1/m_1}, M_2/m_2\}$. \[ \blacksquare \]

With respect to the function $f_1$, Tiedemann, et al. [8] presented the algorithm RPP-HIGH and showed that $CR^{f_1}(\text{RPP-HIGH}) = \max\{\sqrt{M_1/m_1}, M_2/m_2\}$ [8, Theorems 1 and 2]. By combining Corollary 4.1 and Theorem 5.1, we have that $CR^{f_1}(\text{BPP}_k) = z_{f_1}^k = \max\{\sqrt{M_1/m_2}, M_2/m_2\}$, and this is another proof for the optimality on the worst component competitive ratio.

### 5.2 Arithmetic Mean Component Competitive Ratio

For $c_1, \ldots, c_k \in \mathbb{R}$, let $f_2(c_1, \ldots, c_k) = (c_1 + \cdots + c_k) / k$. For the function $f_2 : \mathbb{R}^k \to \mathbb{R}$, let

$$S_{f_2}^k = \left\{ (x_1, \ldots, x_k) \in I_1 \times \cdots \times I_k : \frac{1}{k}\left(\frac{M_1}{x_1} + \cdots + \frac{M_k}{x_k}\right) = \frac{1}{k}\left(\frac{x_1}{m_1} + \cdots + \frac{x_k}{m_k}\right) \right\};$$

$$z_{f_2}^k = \sup_{(x_1, \ldots, x_k) \in S_{f_2}^k} \frac{1}{k}\left(\frac{M_1}{x_1} + \cdots + \frac{M_k}{x_k}\right) = \frac{1}{k}\sup_{(x_1, \ldots, x_k) \in S_{f_2}^k} \left(\frac{M_1}{x_1} + \cdots + \frac{M_k}{x_k}\right).$$

With respect to the function $f_2$, it follows from Corollary 4.1 that $\mathcal{R}_{f_2}^2(\text{BPP}_k) = z_{f_2}^k$. In general, it would be difficult to explicitly represent $z_{f_2}^k$ by $m_1, \ldots, m_k$ and $M_1, \ldots, M_k$. So we consider the case that $k = 2$ and we give an explicit form of $z_{f_2}^2$ by $m_1, m_2$ and $M_1, M_2$.

**Theorem 5.2:** With respect to the function $f_2$ for $k = 2$, the following holds:

$$z_{f_2}^2 = \frac{1}{2} \left[ \sqrt{\left(\frac{1}{2}\left(\frac{M_2}{m_2} - 1\right)\right)^2} + \frac{M_1}{m_1} + \frac{1}{2}\left(\frac{M_2}{m_2} + 1\right) \right].$$

**Proof:** Let $k = 2$. Then $S_{f_2}^2$ and $z_{f_2}^2$ are given by

$$S_{f_2}^2 = \left\{ (x_1, x_2) \in I_1 \times I_2 : \frac{1}{2}\left(\frac{M_1}{x_1} + \frac{M_2}{x_2}\right) = \frac{1}{2}\left(\frac{x_1}{m_1} + \frac{x_2}{m_2}\right) \right\};$$

$$z_{f_2}^2 = \sup_{(x_1, x_2) \in S_{f_2}^2} \frac{1}{2}\left(\frac{M_1}{x_1} + \frac{M_2}{x_2}\right) = \frac{1}{2}\sup_{(x_1, x_2) \in S_{f_2}^2} \left(\frac{M_1}{x_1} + \frac{M_2}{x_2}\right);$$

$$= \frac{1}{2} \sup_{(x_1, x_2) \in S_{f_2}^2} \left\{ \frac{1}{2}\left(\frac{M_1}{x_1} + \frac{M_2}{x_2}\right) + \frac{1}{2}\left(\frac{x_1}{m_1} + \frac{x_2}{m_2}\right) \right\}.$$
Let $g_1(x_1) = \frac{M_1}{x_1} - \frac{x_1}{m_1}$ and $g_2(x_2) = -(\frac{M_2}{x_2} - \frac{x_2}{m_2})$. Then $(p_1, p_2) \in S^2_f$ iff $g_1(p_1) = g_2(p_2)$. Notice that $g_1$ is monotonically decreasing on $[m_1, M_1]$ and $g_2$ is monotonically increasing on $[m_2, M_2]$. Then for any $x_1 \in [m_1, M_1]$, we have that

$$- \left( \frac{M_1}{m_1} - 1 \right) = g_1(M_1) \leq g_1(x_1) \leq g_1(m_1) = \frac{M_1}{m_1} - 1,$$

and for any $x_2 \in [m_2, M_2]$, we also have that

$$- \left( \frac{M_2}{m_2} - 1 \right) = g_2(m_2) \leq g_2(x_2) \leq g_2(M_2) = \frac{M_2}{m_2} - 1.$$

For any $(x_1, x_2) \in S^2_f$, we claim that $- \left( \frac{M_1}{m_1} - 1 \right) \leq g_1(x_1) \leq \frac{M_1}{m_1} - 1$. Let $L_1 \in [m_1, M_1]$ such that $g_1(L_1) = g_2(M_2) = \frac{M_1}{m_1} - 1$ and $R_1 \in [m_1, M_1]$ such that $g_1(R_1) = g_2(m_2) = - \left( \frac{M_2}{m_2} - 1 \right)$, i.e.,

$$L_1 = - \frac{m_1}{2} \left( \frac{M_2}{m_2} - 1 \right) + \sqrt{ \left( \frac{m_1}{2} \left( \frac{M_2}{m_2} - 1 \right) \right)^2 + m_1 M_1};$$

$$R_1 = \frac{m_1}{2} \left( \frac{M_2}{m_2} - 1 \right) + \sqrt{ \left( \frac{m_1}{2} \left( \frac{M_2}{m_2} - 1 \right) \right)^2 + m_1 M_1}. $$

It is immediate that $(L_1, M_2) \in S^2_f$ and $(R_1, m_2) \in S^2_f$.

Let $h_1(x_1) = \frac{\frac{M_1}{x_1} + \frac{x_1}{m_1}}{2}$ and $h_2(x_2) = \frac{\frac{M_2}{x_2} + \frac{x_2}{m_2}}{2}$. Since $h_1$ is convex on $[L_1, R_1] \subseteq [m_1, M_1]$ and $h_2$ is convex on $[m_2, M_2]$, we have that $\max_{x_1 \in [L_1, R_1]} h_1(x_1) = \max\{h_1(L_1), h_1(R_1)\},$ where

$$h_1(L_1) = h_1(R_1) = \sqrt{ \left( \frac{1}{2} \left( \frac{M_2}{m_2} - 1 \right) \right)^2 + \frac{M_1}{m_1}},$$

and $\max_{x_2 \in [m_2, M_2]} h_2(x_2) = \max\{h_2(m_2), h_2(M_2)\}$, where $h_2(m_2) = h_2(M_2) = \frac{1}{2} \left( \frac{M_2}{m_2} + 1 \right)$. Thus it follows that $z^2_f = \frac{1}{2} \{h_1(L_1) + h_2(M_2)\} = \frac{1}{2} \{h_1(R_1) + h_2(m_2)\}$. \hfill \(\blacksquare\)

With respect to the function $f_2$ for $k = 2$, Tiedemann, et al. \cite{8} presented the algorithm RPP-MULT and showed that $\mathcal{CR}^{f_2}(\text{RPP-MULT}) \leq \sqrt{(M_1 M_2)/(m_1 m_2)}$ \cite[Theorem 3]{8} (this is shown by Definition 2.3, but also can be shown by Definition 5.1). Note that $\sqrt{(M_1 M_2)/(m_1 m_2)} < z^2_f$. So from Theorems 4.3 and 5.2, we have that $\mathcal{CR}^{f_2}(\text{ALG}_2) \geq z^2_f$ for any algorithm ALG_2, which disproves the result \cite[Theorem 3]{8}. This is because in the proof of the result \cite[Theorem 3]{8}, the maximum in \cite[Equation (9)]{8} cannot be achieved at $\sqrt{M_1 z^*/M_2}$, where $z^* = \sqrt{m_1 M_2 m_2 M_1}$.

### 5.3 Geometric Mean Component Competitive Ratio

For $c_1, \ldots, c_k \in \mathbb{R}$, let $f_3(c_1, \ldots, c_k) = \sqrt[k]{\prod_{i=1}^k c_i}$. For the function $f_3 : \mathbb{R}^k \to \mathbb{R}$, let

$$S^k_{f_3} = \left\{(x_1, \ldots, x_k) \in I_1 \times \ldots \times I_k : \prod_{i=1}^k M_i \frac{x_i}{m_i} = \prod_{i=1}^k \frac{x_i}{m_i} \right\};$$

$$z^k_{f_3} = \sup_{(x_1, \ldots, x_k) \in S^k_{f_3}} \prod_{i=1}^k \frac{M_i \frac{x_i}{m_i}}{x_i}.$$

\footnote{Recall that $- (\frac{M_1}{m_1} - 1) \leq g_2(x_2) \leq \frac{M_1}{m_1} - 1$. If $- (\frac{M_2}{m_2} - 1) > g_1(x_1)$ or $\frac{M_2}{m_2} - 1 < g_1(x_1)$, then $(x_1, x_2) \not\in S_f$.}
With respect to the function \( f_3 \) for \( k = 2 \), it is easy to see that the algorithm BPP\(_2\) is identical to the algorithm RPP-MULT [8]. In fact, Tiedemann, et al. [8] showed that \( CR_j(f_3, \text{RPP-MULT}) = \sqrt{(M_1 M_2)/(m_1 m_2)} \) with respect to the function \( f_3 \) for \( k = 2 \), and this can be generalized to the result that \( CR_j(f_3, \text{BPP}_k) = z_{f_3}^k \) for any integer \( k \geq 2 \) (see Corollary 4.1 with respect to \( f_3 \)).

**Theorem 5.3:** \( z_{f_3}^k = \frac{\sqrt{\prod_{i=1}^k M_i}}{m_i} \) for any integer \( k \geq 2 \).

**Proof:** From the definition of \( S_{f_3}^k \), it follows that \( \sqrt{\prod_{i=1}^k M_i/x_i} = \sqrt{\prod_{i=1}^k x_i/m_i} \) for any integer \( k \geq 2 \) and any \((x_1, \ldots, x_k) \in S_{f_3}^k \). Then \( \prod_{i=1}^k x_i = \sqrt{\prod_{i=1}^k M_i M_i} \), and this implies that

\[
\prod_{i=1}^k \frac{M_i}{x_i} = \prod_{i=1}^k \frac{M_i}{\prod_{i=1}^k x_i} = \sqrt{\prod_{i=1}^k M_i m_i} = \prod_{i=1}^k \frac{M_i}{m_i}.
\]

Thus we can conclude that \( z_{f_3}^k = \sup_{(x_1, \ldots, x_k) \in S_{f_3}^k} \sqrt{\prod_{i=1}^k M_i/x_i} = \frac{2}{\sqrt{\prod_{i=1}^k M_i/m_i}}. \)

**5.4 Best Component Competitive Ratio**

In this subsection, we deal with a new and natural continuous monotone function \( f_4 : \mathbb{R}^k \to \mathbb{R} \). For \( c_1, \ldots, c_k \in \mathbb{R} \), let \( f_4(c_1, \ldots, c_k) = \min(c_1, \ldots, c_k) \). For the function \( f_4 : \mathbb{R}^k \to \mathbb{R} \), let

\[
S_{f_4}^k = \left\{(x_1, \ldots, x_k) \in I_1 \times \cdots \times I_k : \min \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = \min \left( \frac{x_1}{m_1}, \ldots, \frac{x_k}{m_k} \right) \right\};
\]

\[
z_{f_4}^k = \sup_{(x_1, \ldots, x_k) \in S_{f_4}^k} \left[ \min \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) \right].
\]

**Theorem 5.4:** \( z_{f_4}^k = \sqrt{M_k/m_k} \) for any integer \( k \geq 1 \).

**Proof:** We first show that \( z_{f_4}^k \leq \sqrt{M_k/m_k} \). Assume by contradiction that \( z_{f_4}^k > \sqrt{M_k/m_k} \) and let \( \bar{y} = (y_1, \ldots, y_k) \in S_{f_4}^k \) such that

\[
z_{f_4}^k = f_4 \left( \frac{M_1}{y_1}, \ldots, \frac{M_k}{y_k} \right) = f_4 \left( \frac{y_1}{m_1}, \ldots, \frac{y_k}{m_k} \right) > \sqrt{M_k/m_k}.
\]

Since \( f_4(c_1, \ldots, c_k) = \min(c_1, \ldots, c_k) \), we have that for each \( i \in [1, k] \),

\[
\frac{M_i}{y_i} \geq z_{f_4}^k > \sqrt{M_k/m_k},
\]

\[
\frac{y_i}{m_i} \geq z_{f_4}^k > \sqrt{M_k/m_k}.
\]

In particular, we have that \( M_k/y_k > \sqrt{M_k/m_k} \) and \( y_k/m_k > \sqrt{M_k/m_k} \). This implies that

\[
\frac{M_k}{m_k} = \frac{M_k}{y_k} \cdot \frac{y_k}{m_k} > \sqrt{M_k/m_k} \cdot \sqrt{M_k/m_k} = \frac{M_k}{m_k}.
\]
and this is a contradiction. So it follows that \( z^k_{f_4} \leq \sqrt{M_k/m_k} \). Next we show that there exists \( \vec{x}^* = (x^*_1, \ldots, x^*_k) \in S^k_{f_4} \) such that

\[
z^k_{f_4} = f_4 \left( \frac{M_1}{x^*_1}, \ldots, \frac{M_k}{x^*_k} \right) = f_4 \left( \frac{x^*_1}{m_1}, \ldots, \frac{x^*_k}{m_k} \right) = \sqrt{M_k/m_k}.
\]

For each \( i \in [1, k] \), let \( x^*_i = \sqrt{m_i M_i} \). Then it is immediate that

\[
f_4 \left( \frac{M_1}{x^*_1}, \ldots, \frac{M_k}{x^*_k} \right) = \min \left\{ \frac{M_1}{\sqrt{m_1 M_1}}, \ldots, \frac{M_k}{\sqrt{m_k M_k}} \right\} = \sqrt{M_k/m_k}
\]

\[
f_4 \left( \frac{x^*_1}{m_1}, \ldots, \frac{x^*_k}{m_k} \right) = \min \left\{ \sqrt{m_1 M_1}, \ldots, \sqrt{m_k M_k} \right\} = \sqrt{M_k/m_k}.
\]

Thus we have that \( z^k_{f_4} = \sqrt{M_k/m_k} \) for each integer \( k \geq 1 \).

## 6 Concluding Remarks

In this paper, we have proposed a simple online algorithm Balanced Price Policy (BPP_k) for the multi-objective (\( k \)-objective) time series search problem and have shown that BPP_k is best possible with respect to any monotone (not necessarily continuous) function \( f : \mathbb{R}^k \to \mathbb{R} \) even if all of \( \text{itv}_1 = [m_1, M_1], \ldots, \text{itv}_k = [m_k, M_k] \) are not necessarily real intervals (Theorem 4.1). In the case that all of \( \text{itv}_1 = [m_1, M_1], \ldots, \text{itv}_k = [m_k, M_k] \) are real intervals, we have formulated the best possible value of the competitive ratio exactly for any monotone continuous function \( f : \mathbb{R}^k \to \mathbb{R} \) (Theorems 4.2 and 4.3). We also have derived the best possible values of the competitive ratio for the multi-objective time series search problem with respect to several known measures of the competitive analysis, i.e., the best possible value of the competitive ratio for the multi-objective time series search problem with respect to the worst component competitive analysis (Theorem 5.1), the best possible value of the competitive ratio for the bi-objective time series search problem with respect to the arithmetic mean component competitive analysis (Theorem 5.2), and the best possible value of the competitive ratio for the multi-objective time series search problem with respect to the geometric mean component competitive analysis (Theorem 5.3). For a new measure of the competitive analysis, we derive the best possible value of the competitive ratio for the multi-objective time series search problem with respect to the best component competitive analysis (Theorem 5.4).

For each \( i \in [1, k] \), let \( I_i = [m_i, M_i] \) with \( 0 < m_i \leq M_i \). As we have shown in Theorem 5.2, the best possible value of the competitive ratio for the bi-objective time series search problem with respect to the arithmetic mean component competitive analysis is

\[
z^2_{f_2} = \frac{1}{2} \left[ \sqrt{\left( \frac{1}{2} \left( \frac{M_2}{m_2} - 1 \right) \right)^2 + \frac{M_1}{m_1} + \frac{1}{2} \left( \frac{M_2}{m_2} + 1 \right)} \right].
\]
In Corollary 4.1, we have given the best possible value $z^k_{f_2}$ of the competitive ratio for the multi-objective ($k$-objective) time series search problem with respect to the arithmetic mean component competitive analysis, where

$$z^k_{f_2} = \sup_{(x_1, \ldots, x_k) \in S^k_{f_2}} f_2 \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right),$$

$$S^k_{f_2} = \left\{ (x_1, \ldots, x_k) \in I_1 \times \cdots \times I_k : f_2 \left( \frac{M_1}{x_1}, \ldots, \frac{M_k}{x_k} \right) = f_2 \left( \frac{x_1}{m_1}, \ldots, \frac{x_k}{m_k} \right) \right\}.$$

So we have the following interesting open problem for the multi-objective time series search problem with respect to the arithmetic mean component competitive analysis.

1. For any integer $k \geq 3$, find an explicit representation of $z^k_{f_2}$ or natural conditions for $m_1, \ldots, m_k, M_1, \ldots, M_k$ to explicitly represent $z^k_{f_2}$.

In fact, we may have many practical multi-objective online problems other than the multi-objective time series search problem. Then we also have the following problem for future work.

2. For a practical multi-objective ($k$-objective) online problem $P_k$, design an efficient online algorithm $\text{ALG}_k$ with respect to a natural monotone function $f : \mathbb{R}^k \to \mathbb{R}$, and analyze the competitive ratio of the algorithm $\text{ALG}_k$ with respect to the monotone function $f$.

References

[1] A. Borodin and R. El-Yaniv. Online Computation and Competitive Analysis. Cambridge University Press (1998).

[2] M. Ehrgott. Multicriteria Optimization. Springer (2005).

[3] R. El-Yaniv, A. Fiat, R. M. Karp, and G. Turpin. Optimal Search and One-Way Trading Online Algorithms. Algorithmica, 30(1), pp.101-139 (2001).

[4] M. H. Goldwasser. A Survey of Buffer Management Policies for Packet Switches. ACM SIGACT New, 41(1), pp.100-128 (2010).

[5] E. Koutsoupias. The $k$-Server Conjecture. Computer Science Review, 3(2), pp.105-118 (2009).

[6] E. Mohr, I. Ahmad, and G. Schmidt. Online Algorithms for Conversion Problems: A Survey. Survey in Operations Research and Management Science, 19(2), pp.87-104 (2014).

[7] D. D. Sleator and R. Tarjan. Amortized Efficiency of List Update and Paging Rules. Communication of the ACM, 28(2), pp.202-208 (1985).

[8] M. Tiedemann, J. Ide, and A. Schöbel. Competitive Analysis for Multi-Objective Online Algorithms. in Proc. of the 9th International Workshop on Algorithms and Computation, WALCOM 2015, Lecture Notes in Computer Science 8973, pp.210-221 (2015).

[9] N. E. Young. Online Paging and Caching. Encyclopedia of Algorithms, Springer (2008).