Journal of the Operations Research Society of Japan
Vol. 63, No. 1, January 2020, pp. 1–30

ONE-WAY TRADING PROBLEMS VIA LINEAR OPTIMIZATION

Hiroshi Fujiwara *Shinshu University*
Naohiro Araki *Microtech Corp.*
Hiroaki Yamamoto *Shinshu University*

(Received September 25, 2018; Revised August 30, 2019)

Abstract In the one-way trading problem, we are asked to convert dollars into yen only by unidirectional conversions, while watching the exchange rate that fluctuates along time. The goal is to maximize the amount of yen we finally get, under the assumption that we are not informed of when the game ends. For this problem, an optimal algorithm was proposed by El-Yaniv et al. In this paper we formulate this problem into a linear optimization problem (linear program) and reduce derivation of an optimal algorithm to solving the linear optimization problem. This reveals that the optimality of the algorithm follows from the duality theorem. Our analysis demonstrates how infinite-dimensional linear optimization helps to design algorithms.

Keywords: Linear programming, algorithm, optimization, finance, economics

1. Introduction

Suppose that now you have 10,000 dollars. By converting them into yen referring to the exchange rate every day, how many yen can you get at last? Set a rule that you are not allowed to convert yen back into dollars. It is obvious that your algorithm of conversion affects the final result. Let us assume in addition that the yen/dollar rate takes values either 1, 1.25, 1.5, 1.75, or 2.0. The result of Section 3 leads us to an optimal algorithm as follows: Wait for the yen/dollar exchange rate to reach 1.25. When the rate reaches 1.25, convert 1,096 dollars. When the rate reaches 1.5, convert 4,109 dollars. When the rate reaches 1.75, convert 2,740 dollars. And, when the rate reaches 2.0, convert the remaining 2,055 dollars.

The reader will wonder what the optimality criterion here is. In such a situation, it would be usual that, assuming some probability distribution for the fluctuation of the exchange rate, one tries to maximize the expectation of the final amount of yen. On the contrary, our optimization framework, called *online optimization*, is done without using any probability distribution.

The optimality of the above algorithm was already proved by El-Yaniv et al [6]. What we do in this paper is just to reconsider this *one-way trading problem* as a *linear optimization problem* (linear program). The purpose of this paper consists of two aims. One is to reveal that the optimality of an algorithm is endorsed by an optimal solution to a linear optimization problem, which is not clear from the original problem-specific proof. The other is to demonstrate a good application of infinite-dimensional linear optimization where functional optimization works well as a powerful tool.

1.1. Our contribution

Before introducing our work, we state the one-way trading problem a bit more in details. At the beginning a player is given some dollars. On each occasion of conversion, the player is informed of the yen/dollar exchange rate. The player can convert arbitrary amount of dollars into yen but is not allowed to convert yen back into dollars. It is assumed that the value of the exchange rate is always between a lower and an upper bound. The game ends suddenly without any notice in advance. Then, the player is obliged to convert the remaining dollars into yen at the possible minimum rate. The goal is to maximize the amount of yen that the player finally owns. Our results are summarized as follows:

(I) In Section 3 we investigate the case where the value of the exchange rate takes any of the values that are distributed at even intervals along the range from the lower bound to the upper bound. (The example at the outset applies to this.) We formulate this setting into a finite-dimensional linear optimization problem and obtain an analytical optimal solution that characterizes an optimal algorithm. The optimality comes from the weak duality theorem. More specifically, the proof is done by constructing a primal and a dual feasible solution such that their objective function values coincide. Although we restrict the candidate of algorithms for formulating the problem, the resulting algorithm is guaranteed to be optimal over general algorithms; this additional proof is done separately outside the formulation.

(II) In Section 4 we consider the case where the value of the exchange rate can take arbitrary values between the lower bound to the upper bound, which is referred to as *Variant 2 of the one-way trading game* in the paper [6]. We formulate the problem into an infinite-dimensional linear optimization problem and get an analytical optimal solution that yields an optimal algorithm. The optimality comes from the weak duality theorem as well. However, the dual problem we formulate does not have an optimal solution. Instead, we establish the optimality of a primal solution by offering a family of dual feasible solutions whose objective function values are arbitrarily close to the primal optimal objective function value. We show the resulting algorithm is optimal over general algorithms by giving another proof outside the formulation, similarly to the finite-dimensional case.

The key to formulating linear optimization problems for (I) and (II) is quite simple: Just set the inverse of the *competitive ratio*, which is a performance measure of an algorithm, as a variable. Then, all the rules are described in a linear form. Once a problem is formulated, a standard method leads us to an optimal solution. It may seem redundant to analyze (II) after (I) is done. We insist on this way since an optimal solution for the finite-dimensional case helps to guess one for the infinite-dimensional case. Another reason is that we want to demonstrate that formulation of an infinite-dimensional linear optimization problem can be done quite analogously to that of a finite-dimensional linear optimization problem; the former is not so familiar in the literature.

From the viewpoint of optimization, what we achieve in this paper is merely to solve a special class of instances of linear optimization. However, it should be noted that we present analytic solutions to (I) and (II) in a closed form, not numeric ones computed by a solver. Although a “magic number” $\bar{\alpha}$ which cannot be presented explicitly is needed to describe the optimal solution to (I), that number is also uniquely determined according to the bounds of the exchange rate.

1.2. Related work

The one-way trading problem is a primitive problem closely related to operations research, computational finance, and decision-making theory. El-Yaniv et al. [6] gave a comprehensive investigation of that problem, proposing optimal algorithms for some cases. They classified the problem by (a) whether the player knows the total number of occasions of currency conversion and (b) whether the player knows both the lower and upper bounds on the exchange rate, or only the ratio between them. In Section 4 of this paper, we provide another proof of optimality of an online algorithm for the case where the player does not know the total number of occasions of conversion and he/she knows both the lower and upper bounds on the exchange rate, which is called *Variant 2* in [6]. In Section 3 we study this *Variant 2* with an additional restriction on the exchange rate value. Recently, Wang et al. [9] analyzed the problem by using a competitive difference as the optimality criterion. Chin et al. studied in [2] the case where the exchange rate is unbounded, which is out of the classification above.

In the *two-way trading problem*, the player can both get dollars by selling yen and get yen by selling dollars. Other settings are the same as the one-way trading problem. El-Yaniv et al. [6] gave also an algorithm for the two-way trading problem. Dannoura and Sakurai [4] designed an improved algorithm for the problem. Refer to the book by Dochow [5] for a survey on a more general framework, called *portfolio selection problems.*

2. The One-Way Trading Problem

2.1. Exchange rate sequence and algorithm

In the *one-way trading problem* that we discuss in this paper, a player has one dollar at the beginning. This normalization does not lose generality; we admit a trade of any small fraction of the initial asset. On day k ($k = 1, 2, \ldots$), the player is informed of the yen/dollar exchange rate r_k. The fluctuation is bounded by $m \leq r_k \leq M$ for each k. We assume that the player knows the bounds m and M in advance. The goal for the player is to earn as many yen as possible under the rules below.

On each day k, the player can convert arbitrary amount of dollars into yen, without paying any charge for exchange. At this moment, the player has known only the *exchange rate sequence* so far $(r_1, r_2, \ldots, r_k)$. The task on day k is to determine how many dollars the player exchanges based on that sequence, with no information about the future. It is not allowed to exchange the yen that the player has got back to dollar.

The game ends suddenly, without any previous notice. In other words, any day may be the last chance for exchange. The player cannot know that today is the last day before exchanging dollars on that day. After the game is over, the remaining dollars of the player are compulsorily converted into yen at the possible minimum rate m. Thus, the difficulty of the problem is that the player has to do trades without knowing whether that day is the last chance or not. The setting we mentioned so far is equivalent to *Variant 2 of the one-way trading game* in the paper by El-Yaniv et al [6]. (Although the rule of compulsory conversion at the end of the game is additional, this setting is equivalent to *Variant 2* as far as algorithms are evaluated by the competitive ratio, defined soon.)

We introduce the notion of a deterministic algorithm that determines how many dollars to exchange. We mentioned that the amount of dollars to exchange on day k is determined by the player referring to the exchange rate sequence so far $(r_1, r_2, \ldots, r_k)$. Thus, we identify a deterministic algorithm with a sequence of functions $(f_k)_{k=1}^{\infty}$ with $f_k : [m, M]^k \to [0, 1]$ in such a way that $f_k((r_1, r_2, \ldots, r_k))$ stands for the exchange amount on day k. Note that any adaptive algorithm can be represented in this way. The domain of f_k is $[m, M]^k$, which means that each exchange amount cannot depend on coming rates. We have to assume an infinite sequence of functions since there is no restriction on how long the game lasts.

Let the exchange rate sequence be

$$\sigma = (r_1, r_2, \ldots, r_n).$$

We can look over the whole sequence and know from it that day n is the last chance. However, as we already mentioned, the player notices this fact after the trade on day n. For this exchange rate sequence, consider an algorithm that converts dollars into yen according to a sequence of functions $(f_k)_{k=1}^{\infty}$. Let $ALG(\sigma)$ denote the resulting amount of yen. From the rules above, we have

$$ALG(\sigma) = \sum_{k=1}^{n} r_k \cdot f_k((r_1, r_2, \ldots, r_k)) + m \left(1 - \sum_{k=1}^{n} f_k((r_1, r_2, \ldots, r_k))\right).$$

The former sum is the total amount of yen obtained by conversion on each day. The latter sum is the amount of yen obtained by exchanging the remaining dollars at rate m when the game is over. (Formally, it may be better to use a notation explicitly associated with $(f_k)_{k=1}^{\infty}$, like $ALG_{(f_k)_{k=1}^{\infty}}(\sigma)$. But, throughout this paper, there will be no confusion.)

2.2. Competitive ratio

In online optimization, the performance of an algorithm is usually analyzed without a probabilistic assumption on the future input. The *competitive ratio* is often used for a performance measure of an algorithm for the one-way trading problem (for example, see [1]). Before defining the competitive ratio for the problem, we need to assume an *optimal offline algorithm* that, given the full information on the future, tells an optimal choice.

If the player knew the whole exchange rate sequence σ in advance, he/she would convert exactly one dollar on the day when the rate is the highest. Let $OPT(\sigma)$ denote the amount of yen for such an ideal case. We have

$$OPT(\sigma) = \max\{r_1, r_2, \ldots, r_n\}.$$

For the numerical example in Section 2.1, it holds that $OPT(\sigma_1) = 1.75$. Recalling $ALG(\sigma_1) = 1.275$, we see that the player obtains around 73% of the ideal case.

We are ready to define the competitive ratio. We call

$$\min\{c \mid (\forall n \in \mathbb{N})(\forall \sigma \in [m, M]^n)\ c \cdot ALG(\sigma) \geq OPT(\sigma)\}$$

the *competitive ratio* of the deterministic algorithm represented by $(f_k)_{k=1}^{\infty}$. It is easy to see that $\frac{M}{m}$ is an upper bound and 1 is a lower bound.

We say again that the player got around 73% of the optimal offline profit in the numerical example in Section 2.1. Roughly speaking, the competitive ratio is the maximum value of the inverse of the percentage over all possible exchange rate sequences. A smaller competitive ratio means a better performance of an algorithm. Throughout this paper our aim is to find an algorithm that achieves a small competitive ratio.

3. Solving as a Finite-Dimensional Linear Optimization

In this section we study the one-way trading problem under the setting that the set of values taken by the exchange rate is restricted to a finite subset of $[m, M]$. We focus on a narrowed set of algorithms and formulate a finite-dimensional linear optimization problem. We give an optimal solution to the problem, that is to say, an optimal algorithm. Finally, the algorithm is proved to be optimal also among general algorithms.

El-Yaniv et al [6] solved the original problem, that is, a problem without any restriction on the exchange rate values, except that each rate should be in the interval $[m, M]$. No restriction like ours is discussed in their paper. To investigate a restricted case provides us with plenty of helpful hints for considering the original problem as an infinite-dimensional linear optimization problem in Section 4.

In the original problem the exchange rate can take an arbitrary value in the interval $[m, M]$. In this section we assume that for a given integer $N \geq 2$, the exchange rate is any of

$$m, m + \frac{M-m}{N}, m + \frac{M-m}{N} \cdot 2, \ldots, m + \frac{M-m}{N} \cdot (N-1), \text{ and } M.$$

We write the j-th smallest value with $0 \leq j \leq N$ as

$$p(j) = m + \frac{M-m}{N} \cdot j.$$

Numerical examples that appeared so far all satisfy this assumption. Furthermore, this assumption may be acceptable to almost all application. Any rate value that we encounter in the real world is rational. Choosing a sufficiently large N we restrict the set of possible values as above.

The problem of minimizing the competitive ratio is formulated as follows:

Problem $(\mathcal{OWT}_N)$
Input: Real values $m(>0)$ and $M(>m)$.
Output: A sequence of functions $(f_k)_{k=1}^{\infty}$ with $f_k : [m, M]^k \to [0, 1]$ for each k.
Objective: Minimize the competitive ratio

$$\min\{c \mid (\forall n \in \mathbb{N})(\forall \sigma \in \{p(j) \mid 0 \leq j \leq N\}^n) \; c \cdot ALG(\sigma) \geq OPT(\sigma)\},$$

where for $\sigma = (r_1, r_2, \ldots, r_n)$,

$$ALG(\sigma) = \sum_{k=1}^{n} r_k \cdot f_k((r_1, r_2, \ldots, r_k)) + m \left(1 - \sum_{k=1}^{n} f_k((r_1, r_2, \ldots, r_k))\right)$$

and

$$OPT(\sigma) = \max\{r_1, r_2, \ldots, r_n\}.$$

As we saw in Section 2.1, a sequence of functions $(f_k)_{k=1}^{\infty}$ represents a deterministic algorithm. It seems difficult to find an optimizer directly from general sequences of functions. So, we are going to restrict the solution space to a narrowed class. We will be back to general sequences of functions in Section 3.5.

3.1. Narrowed set of algorithms

From now until Section 3.4, we concentrate on algorithms that convert dollars only when the exchange rate is the *highest* seen so far. (El-Yaniv et al. [6] called this rule *Rule 1 of the threat-based policy.*) For each k, let

$$\hat{r}_k = \max\{r_1, r_2, \ldots, r_k\},$$

that is, the highest exchange rate until day k. A sequence of functions $(f_k)_{k=1}^{\infty}$ that represents such an algorithm must satisfy the following condition: For any exchange rate sequence $(r_1, r_2, \ldots, r_k)$, if $r_k > \hat{r}_{k-1}$ then

$$f_k((r_1, r_2, \ldots, r_k)) > 0,$$

otherwise

$$f_k((r_1, r_2, \ldots, r_k)) = 0.$$

In the paper by El-Yaniv et al. [6], they considered also the other rule of the threat-based policy, *Rule 2:* Exchange the exact amount of dollars such that a target competitive ratio is achieved if the game ends right now. Algorithms that obey both Rules 1 and 2 are designed there. In contrast, we apply only Rule 1 to our formulation.

We now describe our narrowed set of algorithms more specifically. We characterize an algorithm by a vector

$$(x_0, x_1, \ldots, x_N) \in [0,1]^{N+1}.$$

Then, the task of finding an optimal algorithm is done by finding an optimal vector. Given a vector as above, we set the exchange amount on day 1, when the rate is r_1,

$$f_1((r_1)) = \sum_{j=0}^{j_1} x_j, \tag{3.1}$$

where j_1 is an integer such that $p(j_1) = r_1$ holds. For $k = 2, 3, \ldots$, we set

$$f_k((r_1, r_2, \ldots, r_k)) = \begin{cases} \sum_{j=j_{k-1}+1}^{j_k} x_j, & \text{if } r_k > \hat{r}_{k-1}; \\ 0, & \text{if } r_k \le \hat{r}_{k-1}, \end{cases} \tag{3.2}$$

where j_{k-1} and j_k are such that $p(j_{k-1}) = \hat{r}_{k-1}$ and $p(j_k) = r_k$. Intuitively, x_j is the amount of dollars the player should convert when a rate of $p(j)$ or higher is witnessed for the first time. Since the player cannot convert more than the initial asset,

$$\sum_{j=0}^{N} x_j \le 1$$

must hold.

In this way, a vector $(x_0, x_1, \ldots, x_N)$ yields a solution to problem $(\mathcal{OWT}_N)$, which further leads us to an algorithm for the one-way trading problem with the restricted set of exchange rate values. The notation of $ALG(\cdot)$ is also defined accordingly. In the following arguments we refer to the resulting algorithm simply as the algorithm determined by $(x_0, x_1, \ldots, x_N)$.

3.2. Formulation as a linear optimization problem

We formulate a linear optimization problem (linear program) for deriving an optimal vector $(\bar{x}_0, \bar{x}_1, \ldots, \bar{x}_N)$. Some lemmas are needed to be proved.

Lemma 3.1. *The following (A) and (B) hold true for the algorithm determined by* $(x_0, x_1, \ldots, x_N) \in [0,1]^{N+1}$. *(A) For any exchange rate sequence σ whose highest rate is $p(i)$ with $0 \le i \le N$, it holds that*

$$ALG(\sigma) \ge \sum_{j=1}^{i} p(j) \cdot x_j + m\left(1 - \sum_{j=1}^{i} x_j\right).$$

(B) For any i with $0 \le i \le N$, there exists an exchange rate sequence σ such that its highest rate is $p(i)$ and

$$ALG(\sigma) = \sum_{j=1}^{i} p(j) \cdot x_j + m\left(1 - \sum_{j=1}^{i} x_j\right).$$

Proof. (A) Let $k_1 = 1$. For exchange rate sequence $\sigma = (r_1, r_2, \ldots, r_n)$, let $k_2, k_3, \ldots, k_b$ be $k(\geq 2)$ such that $r_k > \hat{r}_{k-1}$, indexed in an increasing order. Using this, we define $j_1, j_2, \ldots, j_b$ as $p(j_1) = r_{k_1}, p(j_2) = r_{k_2}, \ldots, p(j_b) = r_{k_b}$. From the hypothesis that the highest rate in σ is $p(i)$, we know $r_{k_b} = p(i)$ and therefore $j_b = i$.

For σ, the algorithm determined by $(x_0, x_1, \ldots, x_N)$ converts

$$\sum_{j=0}^{j_1} x_j + \sum_{j=j_1+1}^{j_2} x_j + \cdots + \sum_{j=j_{b-1}+1}^{j_b} x_j = \sum_{j=0}^{i} x_j$$

dollars. The amount of yen that the algorithm gains is evaluated as

$$\begin{aligned} p(j_1) \cdot \sum_{j=0}^{j_1} x_j + p(j_2) \cdot \sum_{j=j_1+1}^{j_2} x_j + \cdots + p(j_b) \cdot \sum_{j=j_{b-1}+1}^{j_b} x_j \\ \geq \sum_{j=0}^{j_1} p(j) \cdot x_j + \sum_{j=j_1+1}^{j_2} p(j) \cdot x_j + \cdots + \sum_{j=j_{b-1}+1}^{j_b} p(j) \cdot x_j = \sum_{j=0}^{i} p(j) \cdot x_j. \end{aligned}$$

Hence, together with the yen obtained by the exchange at the end of the game, we have

$$\begin{aligned} ALG(\sigma) \geq \sum_{j=0}^{i} p(j) \cdot x_j + m \left(1 - \sum_{j=0}^{i} x_j\right) &= \sum_{j=1}^{i} p(j) \cdot x_j + m \left(1 - \sum_{j=1}^{i} x_j\right) + p(0) \cdot x_0 - m \cdot x_0 \\ &= \sum_{j=1}^{i} p(j) \cdot x_j + m \left(1 - \sum_{j=1}^{i} x_j\right). \end{aligned}$$

(B) We show that for example, $\sigma = (p(0), p(1), \ldots, p(i))$ is such a sequence with $i + 1$ entries. The algorithm determined by $(x_0, x_1, \ldots, x_N)$ for this sequence converts $\sum_{j=0}^{i} x_j$ dollars. The resulting amount of yen is

$$p(0) \cdot x_0 + p(1) \cdot x_1 + \cdots + p(i) \cdot x_i = \sum_{j=0}^{i} p(j) \cdot x_j.$$

Therefore, eliminating x_0 as we did in the last part of (A), we have

$$ALG(\sigma) = \sum_{j=1}^{i} p(j) \cdot x_j + m \left(1 - \sum_{j=1}^{i} x_j\right).$$

□

Lemma 3.1 says that among exchange rate sequences with the highest rate being $p(i)$, $\sigma = (p(0), p(1), \ldots, p(i))$ is one of those that minimize $ALG(\sigma)$. Besides, please note that x_0 does not appear in the statement of Lemma 3.1. If we want to maximize the resulting yen, we can fix $x_0 = 0$. The intuition is that *it is worthless to exchange dollars at the minimum rate.* In the rest of this section we write a vector that determines an algorithm as $(x_1, \ldots, x_N)$, dropping the entry x_0.

Lemma 3.2. *A necessary and sufficient condition for the competitive ratio of the algorithm determined by* $(x_1, \ldots, x_N)$ *to be at most* c *is that: for all* i *with* $1 \leq i \leq N$*,*

$$\sum_{j=1}^{i} p(j) \cdot x_j + m\left(1 - \sum_{j=1}^{i} x_j\right) - \frac{1}{c} \cdot p(i) \geq 0 \tag{3.3}$$

holds.

Proof. (Sufficiency) Let σ be any exchange rate sequence with the highest rate being $p(i)$. We know that $OPT(\sigma) = p(i)$, since an optimal offline algorithm will convert the whole one dollar at $p(i)$. Applying Lemma 3.1 (A) to the condition, we obtain

$$ALG(\sigma) \geq \sum_{j=1}^{i} p(j) \cdot x_j + m\left(1 - \sum_{j=1}^{i} x_j\right) \geq \frac{1}{c} \cdot OPT(\sigma).$$

Then, the definition of the competitive ratio tells us that the competitive ratio of the algorithm is equal to or smaller than c.

(Necessity) Assume that for some c and i with $1 \leq i \leq N$, the condition is not satisfied. Lemma 3.1 (B) states the existence of σ' such that for the fixed i, the highest rate in σ' is $p(i)$ and

$$ALG(\sigma') = \sum_{j=1}^{i} p(j) \cdot x_j + m\left(1 - \sum_{j=1}^{i} x_j\right).$$

From the fact that $OPT(\sigma') = p(i)$ and the assumption, we derive

$$ALG(\sigma') - \frac{1}{c} \cdot OPT(\sigma') = \sum_{j=1}^{i} p(j) \cdot x_j + m\left(1 - \sum_{j=1}^{i} x_j\right) - \frac{1}{c} \cdot p(i) < 0,$$

which implies that the assumed c cannot be a candidate for the competitive ratio. Thus, the necessity is proved. □

We are now ready to formulate a linear optimization problem. The key is to handle the inverse of the competitive ratio as a variable y_1. By doing so, all the constraints are expressed in a linear form. The reader may rather welcome this trick; the inverse of the competitive ratio directly indicates the fraction compared to the optimal offline profit.

Applying Lemma 3.2 with the equality

$$\sum_{j=1}^{i} p(j) \cdot x_j + m\left(1 - \sum_{j=1}^{i} x_j\right) - p(i) \cdot y_1 = \sum_{j=1}^{i} (p(j) - m)x_j - p(i) \cdot y_1 + m,$$

we have a linear optimization problem.

$$\begin{array}{lll} (\mathcal{P}_N) \quad \text{maximize} & y_1 & \\ \quad\quad \text{subject to} & -\sum_{j=1}^{i} (p(j) - m)x_j + p(i) \cdot y_1 \leq m, & 1 \leq i \leq N \quad (3.4) \\ & \sum_{j=1}^{N} x_j \leq 1 & \quad (3.5) \\ & x_j \geq 0, & 1 \leq j \leq N \\ & y_1 \geq 0 & \end{array}$$

3.3. Dual problem

Before finding a solution to the primal problem $(\mathcal{P}_N)$, we formulate the dual problem of $(\mathcal{P}_N)$. The formulation of the dual problem can be done by a standard routine instructed in many books, [3] for example. We here describe the details as a preparation to deal with an infinite-dimensional linear optimization problem in Section 4. We believe that it is worth doing; the infinite-dimensional version of the formulation of the dual problem may not be as familiar as the finite-dimensional one.

The basic idea of the formulation of the dual problem is to obtain an upper bound on the optimal value of the objective function. We construct a linear combination of the constraints. The constraint (3.4) of $(\mathcal{P}_N)$ consists of N inequalities, each indexed by i with $1 \leq i \leq N$. Of course, one should not forget about (3.5).

Suppose that $(x_1, \ldots, x_N, y_1)$ is a feasible solution to $(\mathcal{P}_N)$. We introduce coefficients of the linear combination, which will be seen as dual variables: $\lambda_1, \ldots, \lambda_N$ corresponding to the constraint (3.4) for each i with $1 \leq i \leq N$, and μ_1 corresponding to the constraint (3.5). If $\lambda_i \geq 0$ for all $i = 1, \ldots, N$, and $\mu_1 \geq 0$, then

$$\sum_{i=1}^{N} \lambda_i \cdot \left\{ -\sum_{j=1}^{i} (p(j) - m)x_j + p(i) \cdot y_1 \right\} + \mu_1 \sum_{j=1}^{N} x_j \leq m \sum_{i=1}^{N} \lambda_i + \mu_1 \tag{3.6}$$

holds. Changing the order of summation in the left-hand side of (3.6), we have

$$\begin{aligned}
&\sum_{i=1}^{N} \sum_{j=1}^{i} \lambda_i \cdot \{-(p(j) - m)x_j\} + \mu_1 \sum_{j=1}^{N} x_j + \left(\sum_{i=1}^{N} p(i) \cdot \lambda_i \right) \cdot y_1 \\
&\quad = \sum_{j=1}^{N} \sum_{i=j}^{N} \lambda_i \cdot \{-(p(j) - m)x_j\} + \mu_1 \sum_{j=1}^{N} x_j + \left(\sum_{i=1}^{N} p(i) \cdot \lambda_i \right) \cdot y_1 \\
&\quad = \sum_{j=1}^{N} \left\{ -(p(j) - m) \sum_{i=j}^{N} \lambda_i + \mu_1 \right\} \cdot x_j + \left(\sum_{i=1}^{N} p(i) \cdot \lambda_i \right) \cdot y_1. \qquad (3.7)
\end{aligned}$$

Regard the right-hand side of (3.7) as a linear combination of $x_1, \ldots, x_N$, and y_1. When the coefficients of $x_1, \ldots, x_N$ are all non-negative and the coefficient of y_1 is equal to or greater than one, the value of this linear combination is at least y_1, which is the objective function value for solution $(x_1, \ldots, x_N, y_1)$. Then, the right-hand side of (3.6) is an upper bound on the objective function value of $(\mathcal{P}_N)$. The dual problem is a problem of finding values for the dual variables such that the upper bound is minimized.

$$\begin{array}{lll}
(\mathcal{D}_N) \quad \text{minimize} & m \displaystyle\sum_{i=1}^{N} \lambda_i + \mu_1 & \\
\quad\quad\quad \text{subject to} & -(p(j) - m) \displaystyle\sum_{i=j}^{N} \lambda_i + \mu_1 \geq 0, \quad 1 \leq j \leq N & (3.8) \\
& \displaystyle\sum_{i=1}^{N} p(i) \cdot \lambda_i \geq 1 & (3.9) \\
& \lambda_i \geq 0, \quad 1 \leq i \leq N & \\
& \mu_1 \geq 0 &
\end{array}$$

The formulation of the dual problem $(\mathcal{D}_N)$ immediately leads us to the following famous property.

Lemma 3.3 (the weak duality theorem). *Let $(x_1, \ldots, x_N, y_1)$ and $(\lambda_1, \ldots, \lambda_N, \mu_1)$ be feasible solutions to $(\mathcal{P}_N)$ and $(\mathcal{D}_N)$, respectively. Then, the objective function value of $(\mathcal{P}_N)$ at $(x_1, \ldots, x_N, y_1)$ is equal to or less than the objective function value of $(\mathcal{D}_N)$ at $(\lambda_1, \ldots, \lambda_N, \mu_1)$. That is to say, it holds that*

$$y_1 \le m \sum_{i=1}^{N} \lambda_i + \mu_1.$$

3.4. Optimal solution

In this section we give a primal and dual optimal solution to $(\mathcal{P}_N)$ and $(\mathcal{D}_N)$, respectively. The optimality is established by the feasibility of the solutions and the coincidence of the objective function values. We obtained the optimal solutions as follows: first, guess which constraints are active from some computational experiments for numerical examples. Then, solve the potentially active constraints as a system of equations with the condition that the primal and dual objective function values are equal.

We introduce an important constant which appears in the primal and the dual optimal solution. Let $\bar{\alpha}$ be the minimum non-negative integer α such that

$$\sum_{j=\alpha+1}^{N} \frac{1}{j} \le \frac{p(\alpha)}{m}. \tag{3.10}$$

(Let the left-hand side be zero when $\alpha = N$.) It is going to turn out that: $\bar{\alpha}$ is such a constant that the algorithm determined by the optimal solution converts dollars only when the exchange rate is at least $p(\bar{\alpha})$. In other words, the algorithm never converts at a rate below $p(\bar{\alpha})$. (This fact seems almost impossible to be understood directly from the definition above.) Lemma 3.4 tells the range of $\bar{\alpha}$.

Lemma 3.4. *It follows that $1 \le \bar{\alpha} \le N$.*

Proof. For $\alpha = 0$, the difference between the left-hand side and the right-hand side of (3.10) is, noting that $p(0) = m$ and $N \ge 2$,

$$\sum_{j=1}^{N} \frac{1}{j} - \frac{m}{m} \ge \sum_{j=1}^{2} \frac{1}{j} - 1 = \frac{1}{2} > 0.$$

Therefore, $1 \le \bar{\alpha}$ must hold. On the other hand, the difference for $\alpha = N$ is $-\frac{M}{m} < 0$. Thus, $\bar{\alpha} \le N$. □

A primal and dual optimal solution is provided by the following Theorems 3.1 and 3.2, which follow from Lemmas 3.3, 3.6, 3.8, and 3.9.

Theorem 3.1. *The following $(\bar{x}_1, \ldots, \bar{x}_N, \bar{y}_1)$ is optimal to $(\mathcal{P}_N)$:*

$$\bar{x}_j = \begin{cases} 0, & 1 \le j \le \bar{\alpha} - 1; \\ \frac{p(\bar{\alpha}) \cdot \bar{y}_1 - m}{p(\bar{\alpha}) - m}, & j = \bar{\alpha}; \\ \frac{\bar{y}_1}{j}, & \bar{\alpha} + 1 \le j \le N \end{cases}$$

and

$$\bar{y}_1 = \left(1 + \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha})} \cdot \sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j}\right)^{-1}. \tag{3.11}$$

Theorem 3.2. *The following* $(\bar{\lambda}_1, \ldots, \bar{\lambda}_N, \bar{\mu}_1)$ *is optimal to* $(\mathcal{D}_N)$*:*

$$\bar{\lambda}_i = \begin{cases} 0, & 1 \le i \le \bar{\alpha} - 1; \\ \bar{\mu}_1 \cdot \frac{N}{M-m} \cdot \frac{1}{i(i+1)}, & \bar{\alpha} \le i \le N-1; \\ \bar{\mu}_1 \cdot \frac{1}{M-m}, & i = N \end{cases}$$

and

$$\bar{\mu}_1 = \left(\frac{p(\bar{\alpha})}{p(\bar{\alpha}) - m} + \sum_{i=\bar{\alpha}+1}^{N} \frac{1}{i}\right)^{-1} = \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha})} \cdot \bar{y}_1.$$

Lemma 3.5 is a helper lemma for showing the feasibility of the primal optimal solution. It means that the competitive ratio $\frac{1}{\bar{y}_1}$ approximately equals to $\frac{p(\bar{\alpha})}{m}$.

Lemma 3.5. *The following inequality holds true:*

$$\frac{m}{p(\bar{\alpha})} \le \bar{y}_1 < \frac{m}{p(\bar{\alpha} - 1)}.$$

Proof. Since $\bar{\alpha}$ is the minimum value that satisfies

$$\sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j} \le \frac{p(\bar{\alpha})}{m} \tag{3.12}$$

and $\bar{\alpha} \ge 1$ by Lemma 3.4, it holds that $\sum_{j=\bar{\alpha}}^{N} \frac{1}{j} > \frac{p(\bar{\alpha}-1)}{m}$. Subtracting $\frac{1}{\bar{\alpha}}$ from both sides, we get

$$\sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j} > \frac{p(\bar{\alpha} - 1)}{m} - \frac{1}{\bar{\alpha}}. \tag{3.13}$$

Using (3.12), we derive

$$\bar{y}_1 = \left(1 + \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha})} \cdot \sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j}\right)^{-1} \ge \left(1 + \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha})} \cdot \frac{p(\bar{\alpha})}{m}\right)^{-1} = \left(1 + \frac{p(\bar{\alpha}) - m}{m}\right)^{-1} = \frac{m}{p(\bar{\alpha})}.$$

Similarly, from (3.13), we get

$$\bar{y}_1 = \left(1 + \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha})} \cdot \sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j}\right)^{-1} < \left(1 + \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha})} \cdot \left(\frac{p(\bar{\alpha} - 1)}{m} - \frac{1}{\bar{\alpha}}\right)\right)^{-1} = \frac{m}{p(\bar{\alpha} - 1)}.$$

□

Lemma 3.6. $(\bar{x}_1, \ldots, \bar{x}_N, \bar{y}_1)$ *is feasible for* $(\mathcal{P}_N)$*.*

Proof. Lemma 3.5 implies that

$$\bar{x}_{\bar{\alpha}} = \frac{p(\bar{\alpha}) \cdot \bar{y}_1 - m}{p(\bar{\alpha}) - m} \geq 0.$$

It is clear that the other entries of $(\bar{x}_1, \ldots, \bar{x}_N, \bar{y}_1)$ are all non-negative. Thus, values of the variables are all non-negative. We next confirm the constraint (3.4). For each i with $1 \leq i \leq \bar{\alpha} - 1$, we have from Lemma 3.5 that

$$-\sum_{j=1}^{i} (p(j) - m)\bar{x}_j + p(i) \cdot \bar{y}_1 = p(i) \cdot \bar{y}_1 < p(\bar{\alpha} - 1) \cdot \frac{m}{p(\bar{\alpha} - 1)} = m.$$

For $i = \bar{\alpha}$, it simply holds that

$$-\sum_{j=1}^{i} (p(j) - m)\bar{x}_j + p(i) \cdot \bar{y}_1 = -(p(\bar{\alpha}) - m) \cdot \frac{p(\bar{\alpha}) \cdot \bar{y}_1 - m}{p(\bar{\alpha}) - m} + p(\bar{\alpha}) \cdot \bar{y}_1 = m.$$

Employing this equality, we have for $\bar{\alpha} + 1 \leq i \leq N$ that

$$\begin{aligned} -\sum_{j=1}^{i} (p(j) - m)\bar{x}_j + p(i) \cdot \bar{y}_1 &= m - \sum_{j=\bar{\alpha}+1}^{i} (p(j) - m) \cdot \frac{\bar{y}_1}{j} + (p(i) - p(\bar{\alpha})) \cdot \bar{y}_1 \\ &= m - \sum_{j=\bar{\alpha}+1}^{i} \frac{M - m}{N} \cdot j \cdot \frac{\bar{y}_1}{j} + (p(i) - p(\bar{\alpha})) \cdot \bar{y}_1 = m. \end{aligned}$$

Finally, applying (3.11), we see that the constraint (3.5) is fulfilled as

$$\begin{aligned} \sum_{j=1}^{N} \bar{x}_j = \frac{p(\bar{\alpha}) \cdot \bar{y}_1 - m}{p(\bar{\alpha}) - m} + \bar{y}_1 \cdot \sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j} &= \frac{p(\bar{\alpha})}{p(\bar{\alpha}) - m} \cdot \bar{y}_1 - \frac{m}{p(\bar{\alpha}) - m} + \bar{y}_1 \cdot \sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j} \\ &= \frac{p(\bar{\alpha})}{p(\bar{\alpha}) - m} \cdot \left(1 + \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha})} \cdot \sum_{j=\bar{\alpha}+1}^{N} \frac{1}{j}\right) \cdot \bar{y}_1 - \frac{m}{p(\bar{\alpha}) - m} = \frac{p(\bar{\alpha}) - m}{p(\bar{\alpha}) - m} = 1. \end{aligned}$$

□

We state a family of equalities derived in the proof of Lemma 3.6 as a lemma, which is referred from the proof of optimality over general algorithms.

Lemma 3.7. *For every i with $\bar{\alpha} \leq i \leq N$, it holds true that*

$$\sum_{j=\bar{\alpha}}^{i} p(j) \cdot \bar{x}_j + m \left(1 - \sum_{j=\bar{\alpha}}^{i} \bar{x}_j\right) = \bar{y}_1 \cdot p(i).$$

Lemma 3.8. *$(\bar{\lambda}_1, \ldots, \bar{\lambda}_N, \bar{\mu}_1)$ is feasible for $(\mathcal{D}_N)$.*

Proof. It is easy to see that the entries of $(\bar{\lambda}_1, \ldots, \bar{\lambda}_N, \bar{\mu}_1)$ are all non-negative. In the following argument we sometimes employ the identity

$$\frac{1}{i(i+1)} = \frac{1}{i} - \frac{1}{i+1}$$

for calculating the sum of a telescopic series. We examine the constraint (3.8). For each j with $1 \leq j \leq \bar{\alpha} - 1$,

$$\begin{aligned} -(p(j)-m)\sum_{i=j}^{N}\bar{\lambda}_i + \bar{\mu}_1 &= -(p(j)-m)\cdot\bar{\mu}_1\cdot\left\{\frac{N}{M-m}\cdot\sum_{i=\bar{\alpha}}^{N-1}\frac{1}{i(i+1)} + \frac{1}{M-m}\right\} + \bar{\mu}_1 \\ &= -(p(j)-m)\cdot\bar{\mu}_1\cdot\frac{N}{M-m}\cdot\left(\frac{1}{\bar{\alpha}} - \frac{1}{N} + \frac{1}{N}\right) + \bar{\mu}_1 = \frac{\bar{\mu}_1}{\bar{\alpha}}\cdot(-j+\bar{\alpha}) > 0. \end{aligned}$$

For $\bar{\alpha} \leq j \leq N-1$,

$$\begin{aligned} -(p(j)-m)\sum_{i=j}^{N}\bar{\lambda}_i + \bar{\mu}_1 &= -(p(j)-m)\cdot\bar{\mu}_1\cdot\left\{\frac{N}{M-m}\cdot\sum_{i=j}^{N-1}\frac{1}{i(i+1)} + \frac{1}{M-m}\right\} + \bar{\mu}_1 \\ = -(p(j)-m)\cdot\bar{\mu}_1\cdot\frac{N}{M-m}\cdot\left(\frac{1}{j} - \frac{1}{N} + \frac{1}{N}\right) + \bar{\mu}_1 &= -\frac{M-m}{N}\cdot j\cdot\bar{\mu}_1\cdot\frac{N}{M-m}\cdot\frac{1}{j} + \bar{\mu}_1 = 0. \end{aligned}$$

The constraint (3.9) is shown to be satisfied by

$$\begin{aligned} &\sum_{i=1}^{N}p(i)\cdot\lambda_i = \frac{\bar{\mu}_1}{M-m}\cdot\left\{N\cdot\sum_{i=\bar{\alpha}}^{N-1}\left(m + \frac{M-m}{N}\cdot i\right)\cdot\frac{1}{i(i+1)} + M\right\} \\ &= \frac{\bar{\mu}_1}{M-m}\cdot\left\{Nm\cdot\left(\frac{1}{\bar{\alpha}} - \frac{1}{N}\right) + (M-m)\cdot\sum_{i=\bar{\alpha}+1}^{N}\frac{1}{i} + M\right\} = \bar{\mu}_1\cdot\left\{\frac{p(\bar{\alpha})}{p(\bar{\alpha})-m} + \sum_{i=\bar{\alpha}+1}^{N}\frac{1}{i}\right\} = 1. \end{aligned}$$

□

Lemma 3.9. *The objective function value of* $(\mathcal{P}_N)$ *at* $(\bar{x}_1, \ldots, \bar{x}_N, \bar{y}_1)$ *equals to the objective function value of* $(\mathcal{D}_N)$ *at* $(\bar{\lambda}_1, \ldots, \bar{\lambda}_N, \bar{\mu}_1)$*. That is to say, it holds that*

$$\bar{y}_1 = m\sum_{i=1}^{N}\bar{\lambda}_i + \bar{\mu}_1.$$

Proof. Applying $\frac{1}{i(i+1)} = \frac{1}{i} - \frac{1}{i+1}$, we have

$$\begin{aligned} m\sum_{i=1}^{N}\bar{\lambda}_i &= m\cdot\bar{\mu}_1\cdot\left\{\frac{N}{M-m}\cdot\sum_{i=\bar{\alpha}}^{N-1}\frac{1}{i(i+1)} + \frac{1}{M-m}\right\} \\ &= m\cdot\bar{\mu}_1\cdot\frac{N}{M-m}\cdot\left(\frac{1}{\bar{\alpha}} - \frac{1}{N} + \frac{1}{N}\right) = m\cdot\bar{\mu}_1\cdot\frac{N}{M-m}\cdot\frac{1}{\bar{\alpha}} = \frac{m}{p(\bar{\alpha})-m}\cdot\bar{\mu}_1. \end{aligned}$$

Recalling $\bar{\mu}_1 = \frac{p(\bar{\alpha})-m}{p(\bar{\alpha})}\cdot\bar{y}_1$, we conclude

$$m\sum_{i=1}^{N}\bar{\lambda}_i + \bar{\mu}_1 = \left(\frac{m}{p(\bar{\alpha})-m} + 1\right)\cdot\bar{\mu}_1 = \frac{p(\bar{\alpha})}{p(\bar{\alpha})-m}\cdot\bar{\mu}_1 = \frac{p(\bar{\alpha})}{p(\bar{\alpha})-m}\cdot\frac{p(\bar{\alpha})-m}{p(\bar{\alpha})}\cdot\bar{y}_1 = \bar{y}_1.$$

□

3.5. Optimality over general algorithms

One should note that what we have discussed so far is the optimality over the narrowed set of algorithms. We now establish the optimality over general algorithms. The proof is done outside the scope of linear optimization. However, the lemma that tells the active constraints in $(\mathcal{P}_N)$, Lemma 3.7, plays a significant role.

The idea of the proof of Theorem 3.3 is due to El-Yaniv et al [6], where an *adversary* generates an exchange rate sequence, depending on the behavior of the player, so that the player's performance worsens. We believe that with assuming an adversary, one can more intuitively understand the analysis on an exchange rate sequence that fools the player's algorithm. Refer to the book [1] for more on competitive analyses with an adversary.

Theorem 3.3. *The algorithm determined by* $(\bar{x}_1, \ldots, \bar{x}_N)$*, that is, the sequence of functions* $(f_k)_{k=1}^{\infty}$ *derived from (3.1) and (3.2) with* $(\bar{x}_1, \ldots, \bar{x}_N)$*, is optimal to* $(\mathcal{OWT}_N)$*. Its competitive ratio is* $\frac{1}{\bar{y}_1}$*.*

Proof. Fix a sequence of functions $(f_k)_{k=1}^{\infty}$, that is, an algorithm, arbitrarily. We are going to prove that the competitive ratio of this algorithm is at least $\frac{1}{\bar{y}_1}$. According to definition of the competitive ratio, it is enough to claim the existence of σ such that

$$ALG(\sigma) \leq \bar{y}_1 \cdot OPT(\sigma).$$

In the following we show that some prefix of $(p(\bar{\alpha}), p(\bar{\alpha}+1), \ldots, p(N))$ satisfies our purpose.

We write for $\bar{\alpha} \leq j \leq N$,

$$s_j = f_j((p(\bar{\alpha}), p(\bar{\alpha}+1), \ldots, p(j))),$$

which is the amount of dollars which the algorithm exchanges when the rate is $p(j)$. Let i_0 be the minimum integer i such that $\bar{\alpha} \leq i \leq N$ and

$$\sum_{j=\bar{\alpha}}^{i} s_j \leq \sum_{j=\bar{\alpha}}^{i} \bar{x}_j. \tag{3.14}$$

Then, for all i with $\bar{\alpha} \leq i \leq i_0 - 1$, $\sum_{j=\bar{\alpha}}^{i} s_j > \sum_{j=\bar{\alpha}}^{i} \bar{x}_j$ holds. Suppose that the adversary chooses a prefix of length $i_0 - \bar{\alpha} + 1$:

$$\sigma_{i_0} = (p(\bar{\alpha}), p(\bar{\alpha}+1), \ldots, p(i_0))$$

and give it to the algorithms. (One can think that the adversary watches the behavior of the algorithm and stops the game when the condition (3.14) is satisfied for the first time.) We analyze the performance of the algorithm for each value of i_0.

(i) Case $i_0 > \bar{\alpha}$: The algorithm gets

$$ALG(\sigma_{i_0}) = \sum_{j=\bar{\alpha}}^{i_0} p(j) \cdot s_j + m \left(1 - \sum_{j=\bar{\alpha}}^{i_0} s_j\right)$$

yen for exchange rate sequence σ_{i_0}. By modifying the algorithm, this amount can be increased, which will be proved below. Specifically, we construct a new sequence of functions $(f'_k)_{k=1}^{\infty}$: For each j with $\bar{\alpha} \leq j \leq i_0 - 1$,

$$f'_j((p(\bar{\alpha}), \ldots, p(j))) = \bar{x}_j,$$

and

$$f'_{i_0}((p(\bar{\alpha}),\dots,p(i_0))) = s_{i_0} + \sum_{j=\bar{\alpha}}^{i_0-1}(s_j - \bar{x}_j). \tag{3.15}$$

The value of (3.15) must be between zero and $\bar{x}_{i_0}$. The upper bound comes from

$$s_{i_0} + \sum_{j=\bar{\alpha}}^{i_0-1}(s_j - \bar{x}_j) = \sum_{j=\bar{\alpha}}^{i_0} s_j - \sum_{j=\bar{\alpha}}^{i_0-1} \bar{x}_j \le \sum_{j=\bar{\alpha}}^{i_0} \bar{x}_j - \sum_{j=\bar{\alpha}}^{i_0-1} \bar{x}_j = \bar{x}_{i_0}.$$

Note that this modification keeps the total amount of dollars that are exchanged.

We show the following inequality as a preparation:

$$\begin{aligned} p(i_0)\cdot\sum_{j=\bar{\alpha}}^{i_0-1}(s_j - \bar{x}_j) - \sum_{j=\bar{\alpha}}^{i_0-1} p(j)\cdot(s_j - \bar{x}_j) &= \sum_{j=\bar{\alpha}}^{i_0-1}(p(i_0) - p(j))(s_j - \bar{x}_j) \\ = \frac{M-m}{N}\cdot\sum_{j=\bar{\alpha}}^{i_0-1}\sum_{l=j}^{i_0-1}(s_j - \bar{x}_j) &= \frac{M-m}{N}\cdot\sum_{l=\bar{\alpha}}^{i_0-1}\sum_{j=\bar{\alpha}}^{l}(s_j - \bar{x}_j) > 0. \end{aligned}$$

The following inequality shows that the amount of yen obtained by $(f'_k)_{k=1}^{\infty}$, denoted by $ALG'(\sigma_{i_0})$, is indeed more than that obtained by $(f_k)_{k=1}^{\infty}$.

$$\begin{aligned} ALG'(\sigma_{i_0}) &= \sum_{j=\bar{\alpha}}^{i_0-1} p(j)\cdot\bar{x}_j + p(i_0)\cdot\left\{s_{i_0} + \sum_{j=\bar{\alpha}}^{i_0-1}(s_j - \bar{x}_j)\right\} + m\left(1 - \sum_{j=\bar{\alpha}}^{i_0} s_j\right) \\ &> \sum_{j=\bar{\alpha}}^{i_0-1} p(j)\cdot\bar{x}_j + p(i_0)\cdot s_{i_0} + \sum_{j=\bar{\alpha}}^{i_0-1} p(j)\cdot(s_j - \bar{x}_j) + m\left(1 - \sum_{j=\bar{\alpha}}^{i_0} s_j\right) = ALG(\sigma_{i_0}). \end{aligned}$$

The proof for this case is concluded by bounding $ALG'(\sigma_{i_0})$ from above. We have

$$\begin{aligned} ALG'(\sigma_{i_0}) &= \sum_{j=\bar{\alpha}}^{i_0-1} p(j)\cdot\bar{x}_j + p(i_0)\cdot\left\{s_{i_0} + \sum_{j=\bar{\alpha}}^{i_0-1}(s_j - \bar{x}_j)\right\} + m\left(1 - \sum_{j=\bar{\alpha}}^{i_0} s_j\right) \\ &= \sum_{j=\bar{\alpha}}^{i_0-1} p(j)\cdot\bar{x}_j + m\left(1 - \sum_{j=\bar{\alpha}}^{i_0-1} \bar{x}_j\right) + (p(i_0) - m)\cdot\left\{s_{i_0} + \sum_{j=\bar{\alpha}}^{i_0-1}(s_j - \bar{x}_j)\right\} \\ &\le \sum_{j=\bar{\alpha}}^{i_0-1} p(j)\cdot\bar{x}_j + m\left(1 - \sum_{j=\bar{\alpha}}^{i_0-1} \bar{x}_j\right) + (p(i_0) - m)\cdot\bar{x}_{i_0} \\ &= \sum_{j=\bar{\alpha}}^{i_0} p(j)\cdot\bar{x}_j + m\left(1 - \sum_{j=\bar{\alpha}}^{i_0} \bar{x}_j\right) = \bar{y}_1\cdot p(i_0) = \bar{y}_1\cdot OPT(\sigma_{i_0}) \end{aligned}$$

The second last equal sign is due to Lemma 3.7. Consequently, we have $ALG(\sigma_{i_0}) < \bar{y}_1\cdot OPT(\sigma_{i_0})$.

(ii) Case $i_0 = \bar{\alpha}$: Since $s_{\bar{\alpha}} \le \bar{x}_{\bar{\alpha}}$ holds, we obtain

$$\begin{aligned} ALG(\sigma_{i_0}) &= p(\bar{\alpha})\cdot s_{\bar{\alpha}} + m\,(1 - s_{\bar{\alpha}}) = (p(\bar{\alpha}) - m)\cdot s_{\bar{\alpha}} + m \\ &\le (p(\bar{\alpha}) - m)\cdot\bar{x}_{\bar{\alpha}} + m = p(\bar{\alpha})\cdot\bar{x}_{\bar{\alpha}} + m\,(1 - \bar{x}_{\bar{\alpha}}) = \bar{y}_1\cdot p(i_0) = \bar{y}_1\cdot OPT(\sigma_{i_0}). \end{aligned}$$

The second last equal sign is again implied by Lemma 3.7. We thus have $ALG(\sigma_{i_0}) \le \bar{y}_1\cdot OPT(\sigma_{i_0})$. □

4. Solving as an Infinite-Dimensional Linear Optimization

In this section we study the one-way trading problem under the setting that the exchange rate can take arbitrary values in $[m, M]$. This case was already researched by El-Yaniv et al. [6] as *Variant 2 of the one-way trading game.* Their results are based on a problem-specific analysis. We here show that in contrast, an optimal algorithm is derived from the weak duality theorem on linear optimization in the same way as in Section 3.

We formulate an infinite-dimensional linear optimization problem, while we formulated a finite-dimensional one in Section 3. It may seem redundant to describe both versions individually. In fact, as we will see in Section 4.6, the finite-dimensional and the infinite-dimensional problems lead to essentially the same optimal solution. A remarkable difference is that the dual of our infinite-dimensional linear optimization problem does not have an optimal solution in spite of its boundedness. The way of managing this difficulty will be helpful for further application, which is one reason why we present both the discussions.

The outline of the analysis as an infinite-dimensional linear optimization problem is the same as Section 3. Thus, we briefly mention only the difference that deserves attention. For the convenience of comparison, we use the same letters to denote corresponding variables. We begin with stating the problem to be solved. The difference from $(\mathcal{OWT}_N)$ is only in the set of exchange rate sequences in the definition of the competitive ratio.

Problem $(\mathcal{OWT}_\infty)$
Input: Real values $m(> 0)$ and $M(> m)$.
Output: A sequence of functions $(f_k)_{k=1}^{\infty}$ with $f_k : [m, M]^k \to [0, 1]$ for each k.
Objective: Minimize the competitive ratio

$$\min\{c \mid (\forall n \in \mathbb{N})(\forall \sigma \in [m, M]^n)\ c \cdot ALG(\sigma) \geq OPT(\sigma)\},$$

where for $\sigma = (r_1, r_2, \ldots, r_n)$,

$$ALG(\sigma) = \sum_{k=1}^{n} r_k \cdot f_k((r_1, r_2, \ldots, r_k)) + m\left(1 - \sum_{k=1}^{n} f_k((r_1, r_2, \ldots, r_k))\right)$$

and

$$OPT(\sigma) = \max\{r_1, r_2, \ldots, r_n\}.$$

4.1. Narrowed set of algorithms

From now until Section 4.4, we consider algorithms that convert dollars only when the exchange rate is the *highest* seen so far. Such an algorithm can be represented by a function defined on interval $[m, M]$. (Recall that in Section 3.1, an algorithm was characterized by a vector $(x_0, x_1, \ldots, x_N)$.) We formulate a linear optimization problem by regarding the function as an element of an infinite-dimensional vector space.

We characterize an algorithm by a function $x : [m, M] \to \mathbb{R}_{\geq 0}$. Given a function x, we set the exchange amount on day 1, when the rate is r_1,

$$f_1((r_1)) = \int_m^{r_1} x(q)dq. \tag{4.1}$$

For $k = 2, 3, \ldots$, we set

$$f_k((r_1, r_2, \ldots, r_k)) = \begin{cases} \int_{\hat{r}_{k-1}}^{r_k} x(q)dq, & \text{if } r_k > \hat{r}_{k-1}; \\ 0, & \text{if } r_k \leq \hat{r}_{k-1}. \end{cases} \tag{4.2}$$

(Recall the notation $\hat{r}_k = \max\{r_1, r_2, \ldots, r_k\}$.) Since the player cannot convert more than the initial asset,

$$\int_m^M x(q)dq \leq 1$$

must hold.

We characterize a solution to problem ($\mathcal{OWT}_\infty$) by a function x in this way. We will refer to the algorithm simply as the algorithm determined by x. The notation of $ALG(\cdot)$ is also defined accordingly.

4.2. Formulation as a linear optimization problem

We show lemmas used to formulate an infinite-dimensional linear optimization problem. Although Lemma 4.1 (B) looks different from the counterpart of the finite-dimensional analysis, Lemma 3.1 (B), it plays a similar role.

Lemma 4.1. *The following (A) and (B) hold true for the algorithm determined by $x : [m, M] \to \mathbb{R}_{\geq 0}$. (A) For any exchange rate sequence σ whose highest rate is p with $m \leq p \leq M$, it holds that*

$$ALG(\sigma) \geq \int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right).$$

(B) For any $\Delta > 0$ and p with $m \leq p \leq M$, there exists an exchange rate sequence σ such that its highest rate is p and

$$ALG(\sigma) < \int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right) + \Delta.$$

Proof. (A) Let $k_1 = 1$. For exchange rate sequence $\sigma = (r_1, r_2, \ldots, r_n)$, let $k_2, k_3, \ldots, k_b$ be $k(\geq 2)$ such that $r_k > \hat{r}_{k-1}$, indexed in an increasing order. From the hypothesis that the highest rate in σ is p, we know $r_{k_b} = p$.

For σ, the algorithm determined by x converts

$$\int_m^{r_{k_1}} x(q)dq + \int_{r_{k_1}}^{r_{k_2}} x(q)dq + \int_{r_{k_2}}^{r_{k_3}} x(q)dq + \cdots + \int_{r_{k_{b-1}}}^{r_{k_b}} x(q)dq = \int_m^p x(q)dq$$

dollars. The amount of yen that the algorithm gains is evaluated as

$$\begin{aligned} & r_{k_1} \cdot \int_m^{r_{k_1}} x(q)dq + r_{k_2} \cdot \int_{r_{k_1}}^{r_{k_2}} x(q)dq + r_{k_3} \cdot \int_{r_{k_2}}^{r_{k_3}} x(q)dq + \cdots + r_{k_b} \cdot \int_{r_{k_{b-1}}}^{r_{k_b}} x(q)dq \\ \geq & \int_m^{r_{k_1}} qx(q)dq + \int_{r_{k_1}}^{r_{k_2}} qx(q)dq + \int_{r_{k_2}}^{r_{k_3}} qx(q)dq + \cdots + \int_{r_{k_{b-1}}}^{r_{k_b}} qx(q)dq = \int_m^p qx(q)dq. \end{aligned}$$

Hence, together with the yen obtained by the exchange at the end of the game, we have

$$ALG(\sigma) \geq \int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right).$$

(B) For arbitrarily fixed $\Delta > 0$ and p, we construct a sequence of $\sigma = (r_1, r_2, \ldots, r_{n-1}, r_n)$ in such a way that: $r_1 = m$, $r_n = p$, the sequence $r_1, r_2, \ldots, r_n$ is monotonically increasing, and each increment $r_j - r_{j-1}$ $(2 \leq j \leq n)$ is less than Δ. That is to say, the length n is determined depending on Δ and p.

For this σ, the algorithm determined by x converts $\int_m^p x(q)dq$ dollars. We below use the inequality $r_j < q + \Delta$ for any q with $r_{j-1} < q < r_j$. The amount of yen that the algorithm gains is

$$\begin{aligned} & r_1 \cdot \int_m^{r_1} x(q)dq + r_2 \cdot \int_{r_1}^{r_2} x(q)dq + \cdots + r_n \cdot \int_{r_{n-1}}^{r_n} x(q)dq \\ & \quad < \int_m^{r_1} (q+\Delta)x(q)dq + \int_{r_1}^{r_2} (q+\Delta)x(q)dq + \cdots + \int_{r_{n-1}}^{r_n} (q+\Delta)x(q)dq \\ & \quad = \int_m^p (q+\Delta)x(q)dq = \int_m^p qx(q)dq + \Delta \cdot \int_m^p x(q)dq \le \int_m^p qx(q)dq + \Delta. \end{aligned}$$

The last inequality sign holds since $\int_m^p x(q)dq \le \int_m^M x(q)dq \le 1$. Consequently, we have

$$ALG(\sigma) < \int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right) + \Delta.$$

□

Lemma 4.2. *A necessary and sufficient condition for the competitive ratio of the algorithm determined by $x : [m, M] \to \mathbb{R}_{\ge 0}$ to be at most c is that: For all p with $m \le p \le M$,*

$$\int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right) - \frac{1}{c} \cdot p \ge 0 \tag{4.3}$$

holds.

Proof. (Sufficiency) Let σ be any exchange rate sequence with the highest rate being p. We know that $OPT(\sigma) = p$, since an optimal offline algorithm will convert the whole one dollar at p. Applying Lemma 4.1 (A) to the condition, we obtain

$$ALG(\sigma) \ge \int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right) \ge \frac{1}{c} \cdot OPT(\sigma).$$

Then, the definition of the competitive ratio tells us that the competitive ratio of the algorithm is equal to or smaller than c.

(Necessity) Assume that for some c and p with $m \le p \le M$, the condition is not satisfied. Choose Δ so as to satisfy

$$0 < \Delta < -\left\{\int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right) - \frac{1}{c} \cdot p\right\}.$$

Lemma 4.1 (B) states the existence of σ' such that the highest rate in σ' is p and

$$ALG(\sigma') - \frac{1}{c} \cdot OPT(\sigma') < \int_m^p qx(q)dq + m\left(1 - \int_m^p x(q)dq\right) - \frac{1}{c} \cdot OPT(\sigma') + \Delta. \tag{4.4}$$

We know that $OPT(\sigma') = p$. Then, it is seen that due to the range of Δ, the right-hand side of (4.4) is negative, which implies that the assumed c cannot be a candidate for the competitive ratio. Thus, the necessity is proved. □

We formulate an infinite-dimensional linear optimization problem by setting the inverse of the competitive ratio as a variable y.

$$
\begin{array}{llll}
(\mathcal{P}_\infty) & \text{maximize} & y & \\
& \text{subject to} & -\displaystyle\int_m^p (q-m)x(q)dq + py \le m, & m \le p \le M \quad (4.5) \\
& & \displaystyle\int_m^M x(q)dq \le 1 & \quad (4.6) \\
& & x(q) \ge 0, & m \le q \le M \\
& & y \ge 0 &
\end{array}
$$

4.3. Dual problem

The formulation of a dual problem we did in Section 3.3 is summarized as follows: A linear combination of the constraints of the primal problem is taken for evaluating an upper bound on the primal objective function value. The coefficients are dual variables. The upper bound is the objective function of the dual problem. The constraints of the dual problem are obtained as a condition that the upper bound is valid. We here do the same over an infinite-dimensional vector space.

The constraint (4.5) means that there are an infinite number of inequalities, each for p such that $m \le p \le M$. We construct a linear combination of these inequalities and (4.6).

Suppose that (x, y) is a feasible solution to $(\mathcal{P}_\infty)$. We introduce dual variables: $\lambda : [m, M] \to \mathbb{R}$ for the constraint (4.5), each $\lambda(p)$ corresponding to the inequality for p, and $\mu \in \mathbb{R}$ for the constraint (4.6). (The validity will be discussed in Section 4.7.) We take a linear combination as: If $\lambda(p) \ge 0$ for all $m \le p \le M$ and $\mu \ge 0$,

$$\int_m^M \lambda(p) \cdot \left\{ -\int_m^p (q-m)x(q)dq + py \right\} dp + \mu \int_m^M x(q)dq \le m \int_m^M \lambda(p)dp + \mu \tag{4.7}$$

holds. Changing the order of integrals in the left-hand side of (4.7), we have

$$
\begin{aligned}
&\int_m^M \left[\int_m^p \lambda(p) \cdot \{-(q-m)x(q)\}\, dq \right] dp + \mu \int_m^M x(q)dq + \left(\int_m^M p\lambda(p)dp \right) \cdot y \\
&\quad = \int_m^M \left[\int_q^M \lambda(p) \cdot \{-(q-m)x(q)\}\, dp \right] dq + \mu \int_m^M x(q)dq + \left(\int_m^M p\lambda(p)dp \right) \cdot y \\
&\qquad = \int_m^M \left\{ -(q-m) \int_q^M \lambda(p)dp + \mu \right\} \cdot x(q)dq + \left(\int_m^M p\lambda(p)dp \right) \cdot y
\end{aligned} \tag{4.8}
$$

Regard the right-hand side of (4.8) as a linear combination of x and y. When the coefficient of $x(q)$ each for $m \le q \le M$, which is the expression surrounded by curly brackets, is non-negative and the coefficient of y is equal to or greater than one, the value of this linear combination is at least y, which is the objective function value for solution (x, y). Then, the right-hand side of (4.7) is an upper bound on the objective function value of $(\mathcal{P}_\infty)$. The dual problem is a problem of finding values for the dual variables such that the upper bound is minimized.

$$
\begin{aligned}
(\mathcal{D}_\infty) \qquad \text{minimize} \qquad & m\int_m^M \lambda(p)dp + \mu \\
\text{subject to } -(q-m)\int_q^M & \lambda(p)dp + \mu \geq 0, \qquad m \leq q \leq M \qquad (4.9) \\
& \int_m^M p\lambda(p)dp \geq 1 \qquad (4.10) \\
& \lambda(p) \geq 0, \qquad m \leq p \leq M \\
& \mu \geq 0
\end{aligned}
$$

The weak duality theorem follows straightforwardly from the derivation of $(\mathcal{D}_\infty)$.

Lemma 4.3 (the weak duality theorem)**.** *Let (x, y) and (λ, μ) be feasible solutions to $(\mathcal{P}_\infty)$ and $(\mathcal{D}_\infty)$, respectively. Then, the objective function value of $(\mathcal{P}_\infty)$ at (x, y) is equal to or less than the objective function value of $(\mathcal{D}_\infty)$ at (λ, μ). That is to say, it holds that*

$$y \leq m\int_m^M \lambda(p)dp + \mu.$$

4.4. Optimal solution

We here give a primal optimal solution to $(\mathcal{P}_\infty)$ and a dual asymptotic optimal solution to $(\mathcal{D}_\infty)$. Note that $(\mathcal{D}_\infty)$ does not have an optimal solution. (For more on this, see Section 4.7.) As will be confirmed in Section 4.6, the primal optimal solution is essentially the same as the finite-dimensional one. The optimality is established by the feasibility of the solutions and the convergence of the objective function values.

The derivation of the solutions can be done as follows: Referring to the optimal solution to $(\mathcal{P}_N)$, guess that the inequality in the constraints is tight on a certain interval, and get an integral equation. Note that the property that the optimal solution to $(\mathcal{P}_N)$ is zero by a certain index and positive elsewhere is helpful here. Then, solve the integral equation together with the condition that the primal and dual objective function values converge. The resulting algorithm coincides with that of El-Yaniv et al [6].

As a preparation, we define a constant $\bar{c}$ as the unique root of the equation

$$c = \ln\frac{M-m}{mc-m}.$$

Lemma 4.4. *The root of the equation $c = \ln\frac{M-m}{mc-m}$ uniquely exists in the interval $(1, \frac{M}{m})$.*

Proof. Consider the function $h : c \mapsto \ln\frac{M-m}{mc-m} - c$. Obviously, h is continuous over $(1, \frac{M}{m})$. Moreover, h is monotonically decreasing since both the functions $c \mapsto \ln\frac{M-m}{mc-m}$ and $c \mapsto -c$ are monotonically decreasing. The lemma holds by $\lim_{c\to 1+0} h(c) = +\infty$ and $h(\frac{M}{m}) = -\frac{M}{m} < 0$. □

Theorem 4.1 provides a primal optimal solution, which follows from Lemmas 4.5, 4.7, and 4.8.

Theorem 4.1. *The following* $(\bar{x}, \bar{y})$ *is optimal to* $(\mathcal{P}_\infty)$*:*

$$\bar{x}(q) = \begin{cases} 0, & m \le q < m\bar{c}; \\ \frac{1}{\bar{c}(q-m)}, & m\bar{c} \le q \le M \end{cases}$$

and

$$\bar{y} = \frac{1}{\bar{c}}.$$

Lemma 4.5. $(\bar{x}, \bar{y})$ *is feasible for* $(\mathcal{P}_\infty)$*.*

Proof. The feasibility is ensured simply by plugging $(\bar{x}, \bar{y})$ into the constraints as below. We first see $1 < \bar{c} < \frac{M}{m}$ from Lemma 4.4. The constraint (4.5) holds true for each p with $m \le p < m\bar{c}$ as

$$-\int_m^p (q-m)\bar{x}(q)dq + p\bar{y} = p \cdot \frac{1}{\bar{c}} < m\bar{c} \cdot \frac{1}{\bar{c}} = m.$$

For $m\bar{c} \le p \le M$,

$$-\int_m^p (q-m)\bar{x}(q)dq + p\bar{y} = -\frac{1}{\bar{c}}\int_{m\bar{c}}^p dq + p \cdot \frac{1}{\bar{c}} = \frac{1}{\bar{c}} \cdot (-p + m\bar{c} + p) = m.$$

The constraint (4.6) is confirmed as

$$\int_m^M \bar{x}(q)dq = \int_{m\bar{c}}^M \frac{dq}{\bar{c}(q-m)} = \frac{1}{\bar{c}} \cdot \ln\frac{M-m}{m\bar{c}-m} = 1.$$

We have for $m\bar{c} \le q \le M$,

$$\bar{x}(q) = \frac{1}{\bar{c}(q-m)} \ge \frac{1}{\bar{c}(M-m)} > 0.$$

It is clear that $\bar{y}$ is positive. □

We pick up a family of equalities derived from the proof of Lemma 4.5 and state it as a lemma for the later analysis.

Lemma 4.6. *For every* p *with* $m\bar{c} \le p \le M$*, it holds true that*

$$\int_{m\bar{c}}^p q\bar{x}(q)dq + m\left(1 - \int_{m\bar{c}}^p \bar{x}(q)dq\right) = p\bar{y}.$$

To prove the optimality of $(\bar{x}, \bar{y})$, we employ a family of dual feasible solutions below, parameterized by ε.

Lemma 4.7. *For any* $0 < \varepsilon < M - m\bar{c}$*, the following* $(\bar{\lambda}, \bar{\mu})$ *is feasible for* $(\mathcal{D}_\infty)$*:*

$$\bar{\lambda}(p) = \begin{cases} 0, & m \le p < m\bar{c}; \\ \frac{\bar{\mu}}{(p-m)^2}, & m\bar{c} \le p < M - \varepsilon; \\ \frac{\bar{\mu}}{M-m} \cdot \frac{1}{\varepsilon}, & M - \varepsilon \le p \le M \end{cases}$$

and

$$\bar{\mu} = \left(-\frac{m}{M-\varepsilon-m} + \frac{1}{\bar{c}-1} + \ln\frac{M-\varepsilon-m}{m\bar{c}-m} + \frac{M}{M-m} - \frac{1}{M-m} \cdot \frac{\varepsilon}{2}\right)^{-1}.$$

Proof. We first show that $\bar{\mu}$ is positive. Applying $\varepsilon < M - m\bar{c}$, we derive

$$\begin{aligned}\frac{1}{\bar{\mu}} &= -\frac{m}{M-\varepsilon-m} + \frac{1}{\bar{c}-1} + \ln\frac{M-\varepsilon-m}{m\bar{c}-m} + \frac{M}{M-m} - \frac{1}{M-m}\cdot\frac{\varepsilon}{2} \\ &> -\frac{m}{m\bar{c}-m} + \frac{1}{\bar{c}-1} + \frac{M}{M-m} - \frac{M-m\bar{c}}{2(M-m)} = \frac{M+m\bar{c}}{2(M-m)} > 0.\end{aligned}$$

It is then immediate that $\bar{\lambda}$ is non-negative on $[m, M]$. The rest of the proof is done by confirming the constraints (4.9) and (4.10). The constraint (4.9) needs to be checked with respect to three intervals. We have for $m \le q < m\bar{c}$,

$$\begin{aligned}-(q-m)\int_q^M \bar{\lambda}(p)dp+\bar{\mu} &= -(q-m)\left(\int_q^{m\bar{c}} 0\,dp + \int_{m\bar{c}}^{M-\varepsilon}\frac{\bar{\mu}}{(p-m)^2}dp + \int_{M-\varepsilon}^{M}\frac{\bar{\mu}}{M-m}\cdot\frac{1}{\varepsilon}dp\right)+\bar{\mu} \\ &= \bar{\mu}\cdot\left\{-(q-m)\left(-\frac{1}{M-\varepsilon-m}+\frac{1}{m\bar{c}-m}+\frac{1}{M-m}\right)+1\right\} \\ &> \bar{\mu}\cdot\left\{-(m\bar{c}-m)\left(-\frac{1}{M-\varepsilon-m}+\frac{1}{m\bar{c}-m}+\frac{1}{M-m}\right)+1\right\} \\ &= \bar{\mu}\cdot(m\bar{c}-m)\left(\frac{1}{M-\varepsilon-m}-\frac{1}{M-m}\right) > 0.\end{aligned}$$

For $m\bar{c} \le q < M-\varepsilon$,

$$\begin{aligned}-(q-m)\int_q^M \bar{\lambda}(p)dp+\bar{\mu} &= -(q-m)\left(\int_q^{M-\varepsilon}\frac{\bar{\mu}}{(p-m)^2}dp + \int_{M-\varepsilon}^{M}\frac{\bar{\mu}}{M-m}\cdot\frac{1}{\varepsilon}dp\right)+\bar{\mu} \\ &= \bar{\mu}\cdot\left\{-(q-m)\left(-\frac{1}{M-\varepsilon-m}+\frac{1}{q-m}+\frac{1}{M-m}\right)+1\right\} \\ &= \bar{\mu}\cdot(q-m)\left(\frac{1}{M-\varepsilon-m}-\frac{1}{M-m}\right) > 0.\end{aligned}$$

And for $M-\varepsilon \le q \le M$,

$$\begin{aligned}-(q-m)\int_q^M \bar{\lambda}(p)dp+\bar{\mu} &= -(q-m)\left(\int_q^{M}\frac{\bar{\mu}}{M-m}\cdot\frac{1}{\varepsilon}dp\right)+\bar{\mu} \\ &= \bar{\mu}\cdot\left\{-(q-m)\cdot\frac{1}{M-m}\cdot\frac{1}{\varepsilon}\cdot(M-q)+1\right\} \\ &> \bar{\mu}\cdot\left\{-(M-m)\cdot\frac{1}{M-m}\cdot\frac{1}{\varepsilon}\cdot\varepsilon+1\right\} = 0.\end{aligned}$$

The constraint (4.10) holds true since we can calculate as

$$\begin{aligned}&\int_m^M p\bar{\lambda}(p)dp = \bar{\mu}\cdot\left\{\int_{m\bar{c}}^{M-\varepsilon}\frac{p}{(p-m)^2}dp + \frac{1}{M-m}\cdot\frac{1}{\varepsilon}\cdot\int_{M-\varepsilon}^{M}pdp\right\} \\ &= \bar{\mu}\cdot\left\{-\frac{m}{M-\varepsilon-m}+\frac{1}{\bar{c}-1}+\ln\frac{M-\varepsilon-m}{m\bar{c}-m}+\frac{1}{M-m}\cdot\frac{1}{\varepsilon}\cdot\left(\frac{1}{2}M^2-\frac{1}{2}(M-\varepsilon)^2\right)\right\} \\ &\quad = \bar{\mu}\cdot\left(-\frac{m}{M-\varepsilon-m}+\frac{1}{\bar{c}-1}+\ln\frac{M-\varepsilon-m}{m\bar{c}-m}+\frac{M}{M-m}-\frac{1}{M-m}\cdot\frac{\varepsilon}{2}\right) = 1.\end{aligned}$$

□

Lemma 4.8 states that the dual objective function value at $(\bar{\lambda}, \bar{\mu})$ can be arbitrarily close to the primal objective function value at $(\bar{x}, \bar{y})$. Note that ε is a parameter that determines $(\bar{\lambda}, \bar{\mu})$.

Lemma 4.8. *For all $\delta > 0$, there exists ε with $0 < \varepsilon < M - m\bar{c}$ such that*

$$\bar{y} \le m \int_m^M \bar{\lambda}(p)dp + \bar{\mu} < \bar{y} + \delta.$$

Proof. The part $\bar{y} \le m \int_m^M \bar{\lambda}(p)dp + \bar{\mu}$ is an immediate consequence of Lemma 4.3. Now we prove the other inequality. By substituting, we have

$$m \int_m^M \bar{\lambda}(p)dp + \bar{\mu} = \frac{-\frac{m}{M-\varepsilon-m} + \frac{1}{\bar{c}-1} + \frac{M}{M-m}}{-\frac{m}{M-\varepsilon-m} + \frac{1}{\bar{c}-1} + \ln \frac{M-\varepsilon-m}{m\bar{c}-m} + \frac{M}{M-m} - \frac{1}{M-m} \cdot \frac{\varepsilon}{2}}. \tag{4.11}$$

View the right-hand side of (4.11) as a function with respect to ε. It is obvious that the function is continuous on interval $[0, M - m\bar{c})$. (Note that the solution $(\bar{\lambda}, \bar{\mu})$ is not defined for $\varepsilon = 0$.) If we substitute zero for ε, we get

$$\frac{-\frac{m}{M-m} + \frac{1}{\bar{c}-1} + \frac{M}{M-m}}{-\frac{m}{M-m} + \frac{1}{\bar{c}-1} + \ln \frac{M-m}{m\bar{c}-m} + \frac{M}{M-m}} = \frac{1 + \frac{1}{\bar{c}-1}}{1 + \frac{1}{\bar{c}-1} + \ln \frac{M-m}{m\bar{c}-m}} = \frac{1 + \frac{1}{\bar{c}-1}}{1 + \frac{1}{\bar{c}-1} + \bar{c}} = \frac{1}{\bar{c}} = \bar{y}.$$

The continuity of the function implies that for all $\delta > 0$, there exists ε with $0 < \varepsilon < M - m\bar{c}$ such that

$$\left| m \int_m^M \bar{\lambda}(p)dp + \bar{\mu} - \bar{y} \right| < \delta.$$

The inequality of the lemma is thus obtained. □

4.5. Optimality over general algorithms

We show the optimality of the obtained algorithm over general algorithms, similarly to Section 3.5. A difference is that here one should care about an "error" that arises in finite-time conversions. Lemma 4.9 helps to deal with the error.

Lemma 4.9. *Fix $y_0 > 0$ and a sequence of functions $(f_k)_{k=1}^{\infty}$, and consider the algorithm that is determined by $(f_k)_{k=1}^{\infty}$. Suppose that for all $\Delta > 0$, there exists σ such that*

$$ALG(\sigma) < y_0 \cdot OPT(\sigma) + \Delta.$$

Then, the competitive ratio of the algorithm is at least $\frac{1}{y_0}$.

Proof. Suppose that the hypothesis is true. Fix $y > y_0$ arbitrarily. Set $\Delta = (y - y_0) \cdot m$. Then, we have $ALG(\sigma) < y_0 \cdot OPT(\sigma) + (y - y_0) \cdot m$. Since $OPT(\sigma) \ge m$ holds for any σ, we derive

$$ALG(\sigma) < y_0 \cdot OPT(\sigma) + (y - y_0) \cdot OPT(\sigma) = y \cdot OPT(\sigma). \tag{4.12}$$

By definition of the competitive ratio, (4.12) implies that the competitive ratio of the algorithm is greater than $\frac{1}{y}$. □

We prove Theorem 4.2 in almost the same way Theorem 3.3. Again, the idea of the proof is due to El-Yaniv et al [6].

Theorem 4.2. *The algorithm determined by $\bar{x}$, that is, the sequence of functions $(f_k)_{k=1}^{\infty}$ derived from (4.1) and (4.2) with $\bar{x}$, is optimal to $(\mathcal{OWT}_\infty)$. Its competitive ratio is $\frac{1}{\bar{y}}$.*

Proof. Fix a sequence of functions $(f_k)_{k=1}^{\infty}$, that is, an algorithm, arbitrarily. We are going to prove that the competitive ratio of this algorithm is at least $\frac{1}{\bar{y}}$, using Lemma 4.9. That is to say, our goal is to show that for all $\Delta > 0$, there exists σ such that $ALG(\sigma) < \bar{y} \cdot OPT(\sigma) + \Delta$.

Choose an integer $N \geq 2$. Set $\hat{\Delta} = \frac{M - m\bar{c}}{N}$, which can be arbitrarily small according to the choice of N. Let

$$p_j = m\bar{c} + j \cdot \hat{\Delta}$$

for $0 \leq j \leq N$. See that $p_0 = m\bar{c}$ and $p_N = m\bar{c} + N \cdot \hat{\Delta} = M$. (Note that p_j is different from $p(j)$ in Section 3.) In the following we show that some prefix of $(p_1, p_2, \ldots, p_N)$ satisfies our purpose.

For simplicity of notation, we write

$$\hat{x}_j = \int_{p_{j-1}}^{p_j} \bar{x}(q)dq$$

for $1 \leq j \leq N$. We also write for $1 \leq j \leq N$, $s_j = f_j((p_1, p_2, \ldots, p_j))$, which is the amount of dollars which the algorithm exchanges when the rate is p_j. Let i_0 be the minimum integer i such that $1 \leq i \leq N$ and

$$\sum_{j=1}^{i} s_j \leq \sum_{j=1}^{i} \hat{x}_j. \tag{4.13}$$

Note that if such i_0 does not exist, then $\sum_{j=1}^{N} s_j > \sum_{j=1}^{N} \hat{x}_j = \int_{m\bar{c}}^{M} \bar{x}(q)dq = 1$, which is a contradiction. Then, for all i with $1 \leq i \leq i_0 - 1$, $\sum_{j=1}^{i} s_j > \sum_{j=1}^{i} \hat{x}_j$ holds. Suppose that the adversary chooses a prefix of length i_0:

$$\sigma_{i_0} = (p_1, p_2, \ldots, p_{i_0})$$

and give it to the algorithms. We analyze the performance of the algorithm for each value of i_0.

(i) Case $i_0 \geq 2$: The algorithm gains $ALG(\sigma_{i_0}) = \sum_{j=1}^{i_0} p_j \cdot s_j + m\left(1 - \sum_{j=1}^{i_0} s_j\right)$ yen for exchange rate sequence σ_{i_0}. We consider a modified algorithm which can earn more. Specifically, we construct a new sequence of functions $(f'_k)_{k=1}^{\infty}$: For each j with $1 \leq j \leq i_0 - 1$,

$$f'_j((p_1, \ldots, p_j)) = \hat{x}_j,$$

and

$$f'_{i_0}((p_1, \ldots, p_{i_0})) = s_{i_0} + \sum_{j=1}^{i_0-1}(s_j - \hat{x}_j). \tag{4.14}$$

The value of (4.14) must be between zero and $\hat{x}_{i_0}$. The upper bound comes from

$$s_{i_0} + \sum_{j=1}^{i_0-1}(s_j - \hat{x}_j) = \sum_{j=1}^{i_0} s_j - \sum_{j=1}^{i_0-1} \hat{x}_j \leq \sum_{j=1}^{i_0} \hat{x}_j - \sum_{j=1}^{i_0-1} \hat{x}_j = \hat{x}_{i_0}.$$

Note that this modification keeps the total amount of dollars that are exchanged.

We show an inequality which will be needed soon.

$$p_{i_0} \cdot \sum_{j=1}^{i_0-1}(s_j - \hat{x}_j) - \sum_{j=1}^{i_0-1} p_j \cdot (s_j - \hat{x}_j) = \sum_{j=1}^{i_0-1}(p_{i_0} - p_j)(s_j - \hat{x}_j)$$
$$= \frac{M - m\bar{c}}{N} \cdot \sum_{j=1}^{i_0-1}\sum_{l=j}^{i_0-1}(s_j - \hat{x}_j) = \frac{M - m\bar{c}}{N} \cdot \sum_{l=1}^{i_0-1}\sum_{j=1}^{l}(s_j - \hat{x}_j) > 0. \quad (4.15)$$

The following inequality shows that the amount of yen obtained by $(f'_k)_{k=1}^{\infty}$, denoted by $ALG'(\sigma_{i_0})$, is more than that obtained by $(f_k)_{k=1}^{\infty}$.

$$ALG'(\sigma_{i_0}) = \sum_{j=1}^{i_0-1} p_j \cdot \hat{x}_j + p_{i_0} \cdot \left\{ s_{i_0} + \sum_{j=1}^{i_0-1}(s_j - \hat{x}_j) \right\} + m\left(1 - \sum_{j=1}^{i_0} s_j\right)$$
$$> \sum_{j=1}^{i_0-1} p_j \cdot \hat{x}_j + p_{i_0} \cdot s_{i_0} + \sum_{j=1}^{i_0-1} p_j \cdot (s_j - \hat{x}_j) + m\left(1 - \sum_{j=1}^{i_0} s_j\right) = ALG(\sigma_{i_0}).$$

We bound $ALG'(\sigma_{i_0})$ from above as

$$ALG'(\sigma_{i_0}) = \sum_{j=1}^{i_0-1} p_j \cdot \hat{x}_j + p_{i_0} \cdot \left\{ s_{i_0} + \sum_{j=1}^{i_0-1}(s_j - \hat{x}_j) \right\} + m\left(1 - \sum_{j=1}^{i_0} s_j\right)$$
$$= \sum_{j=1}^{i_0-1} p_j \cdot \hat{x}_j + m\left(1 - \sum_{j=1}^{i_0-1} \hat{x}_j\right) + (p_{i_0} - m) \cdot \left\{ s_{i_0} + \sum_{j=1}^{i_0-1}(s_j - \hat{x}_j) \right\}$$
$$\leq \sum_{j=1}^{i_0-1} p_j \cdot \hat{x}_j + m\left(1 - \sum_{j=1}^{i_0-1} \hat{x}_j\right) + (p_{i_0} - m) \cdot \hat{x}_{i_0}$$
$$= \sum_{j=1}^{i_0} p_j \cdot \hat{x}_j + m\left(1 - \sum_{j=1}^{i_0} \hat{x}_j\right)$$
$$= \sum_{j=1}^{i_0} p_j \cdot \int_{p_{j-1}}^{p_j} \bar{x}(q)dq + m\left(1 - \int_{m\bar{c}}^{p_{i_0}} \bar{x}(q)dq\right).$$

We see that the inequality $p_j < q + \hat{\Delta}$ holds for any q with $p_{j-1} < q < p_j$, since $p_j - p_{j-1} = \hat{\Delta}$. Similarly to the proof of Lemma 4.1 (B), we have

$$\sum_{j=1}^{i_0} p_j \cdot \int_{p_{j-1}}^{p_j} \bar{x}(q)dq < \sum_{j=1}^{i_0} \int_{p_{j-1}}^{p_j} (q + \hat{\Delta})\bar{x}(q)dq = \int_{m\bar{c}}^{p_{i_0}} (q + \hat{\Delta})\bar{x}(q)dq \leq \int_{m\bar{c}}^{p_{i_0}} q\bar{x}(q)dq + \hat{\Delta}.$$

By Lemma 4.6, it is derived that

$$ALG(\sigma_{i_0}) < ALG'(\sigma_{i_0}) < \int_{m\bar{c}}^{p_{i_0}} q\bar{x}(q)dq + m\left(1 - \int_{m\bar{c}}^{p_{i_0}} \bar{x}(q)dq\right) + \hat{\Delta}$$
$$= p_{i_0}\bar{y} + \hat{\Delta} = \bar{y} \cdot OPT(\sigma_{i_0}) + \hat{\Delta}.$$

Applying Lemma 4.9, we know that the competitive ratio of the algorithm determined by $(f_k)_{k=1}^{\infty}$ is at least $\frac{1}{\bar{y}}$.

(ii) Case $i_0 = 1$: From the fact that $s_1 \leq \hat{x}_1$, we obtain

$$ALG(\sigma_{i_0}) = (p_1 - m) \cdot s_1 + m \leq (p_1 - m) \cdot \hat{x}_1 + m = p_1 \cdot \int_{p_0}^{p_1} \bar{x}(q)dq + m\left(1 - \int_{p_0}^{p_1} \bar{x}(q)dq\right).$$

Similarly to the proof of Lemma 4.1 (B), using the inequality $p_1 < q + \hat{\Delta}$ for any q with $p_0 < q < p_1$, we have

$$p_1 \cdot \int_{p_0}^{p_1} \bar{x}(q)dq < \int_{p_0}^{p_1} (q + \hat{\Delta})\bar{x}(q)dq = \int_{p_0}^{p_1} q\bar{x}(q)dq + \hat{\Delta} \cdot \int_{p_0}^{p_1} \bar{x}(q)dq \leq \int_{p_0}^{p_1} q\bar{x}(q)dq + \hat{\Delta}.$$

By Lemma 4.6, it holds that

$$ALG(\sigma_{i_0}) < \int_{p_0}^{p_1} q\bar{x}(q)dq + m\left(1 - \int_{p_0}^{p_1} \bar{x}(q)dq\right) + \hat{\Delta} = p_1\bar{y} + \hat{\Delta} = \bar{y} \cdot OPT(\sigma_{i_0}) + \hat{\Delta}$$

From Lemma 4.9, we know that the competitive ratio of the algorithm determined by $(f_k)_{k=1}^{\infty}$ is at least $\frac{1}{\bar{y}}$. □

4.6. Comparison of finite and infinite-dimensional problems

We compare the optimal algorithms for $(\mathcal{OWT}_N)$ and $(\mathcal{OWT}_\infty)$. The optimal algorithm for $(\mathcal{OWT}_N)$ starts converting when the exchange rate reaches $p(\bar{\alpha})$. On the other hand, the optimal algorithm for $(\mathcal{OWT}_\infty)$ starts converting when the exchange rate reaches $m\bar{c}$. There is no essential difference between these two problems. So, one may wonder if $p(\bar{\alpha}) = m + \frac{M-m}{N} \cdot \bar{\alpha} \approx m\bar{c}$, that is to say, $\bar{\alpha} \approx \frac{m\bar{c}-m}{M-m} \cdot N$. Lemma 4.10 shows that this intuition is true.

Lemma 4.10. *It holds that*

$$\left\lfloor \frac{m\bar{c} - m}{M - m} \cdot N \right\rfloor \leq \bar{\alpha} \leq \left\lceil \frac{m\bar{c} - m}{M - m} \cdot N \right\rceil.$$

Proof. Consider the region S that is bounded by $q \mapsto \frac{1}{q-m}$, $q = m\bar{c}$, $q = M$, and the q-axis. Its area is

$$\int_{m\bar{c}}^{M} \frac{dq}{q - m} = \ln \frac{M - m}{m\bar{c} - m} = \bar{c}.$$

The second equality follows from definition of $\bar{c}$. The lower (upper) bound on $\bar{\alpha}$ is established by calculating the area of a collection of rectangles that circumscribe (inscribe, respectively) the region S.

(i) The lower bound $\alpha_1 = \lfloor \frac{m\bar{c}-m}{M-m} \cdot N \rfloor$. By deforming $\alpha_1 \leq \frac{m\bar{c}-m}{M-m} \cdot N$, we get $p(\alpha_1) \leq m\bar{c}$. Consider an equal-interval subdivision $q = p(\alpha_1), p(\alpha_1 + 1), p(\alpha_1 + 2), \ldots, p(N)$ and a collection of rectangles on each interval that circumscribe the region S (see Figure 1). The total area is

$$\sum_{j=\alpha_1}^{N-1} \frac{M - m}{N} \cdot \frac{1}{p(j) - m} = \sum_{j=\alpha_1}^{N-1} \frac{M - m}{N} \cdot \frac{1}{\frac{M-m}{N} \cdot j} = \sum_{j=\alpha_1}^{N-1} \frac{1}{j},$$

which must be greater than $\bar{c}$. We have

$$\frac{p(\alpha_1 - 1)}{m} - \sum_{j=\alpha_1}^{N} \frac{1}{j} < \frac{p(\alpha_1 - 1)}{m} - \sum_{j=\alpha_1}^{N-1} \frac{1}{j} < \frac{p(\alpha_1)}{m} - \sum_{j=\alpha_1}^{N-1} \frac{1}{j} < \bar{c} - \bar{c} = 0.$$

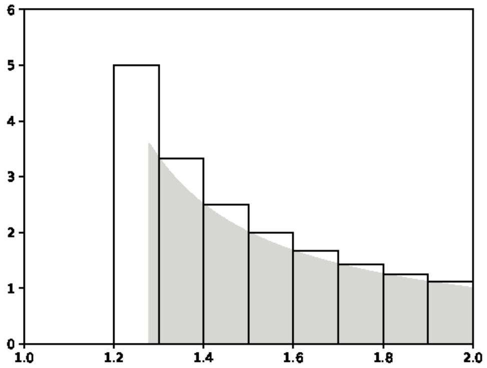

Figure 1: Proof of Lemma 4.10. Collection of rectangles that circumscribe the region S, the shaded area, for $m = 1$, $M = 2$, and $N = 10$. Then, $\bar{c} \approx 1.28$ and $\alpha_1 = 2$

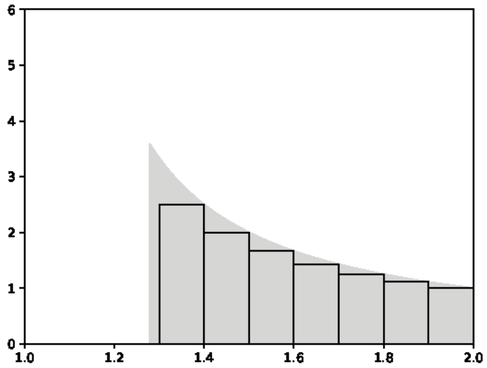

Figure 2: Proof of Lemma 4.10. Collection of rectangles that inscribe the region S, the shaded area, for $m = 1$, $M = 2$, and $N = 10$. Then, $\bar{c} \approx 1.28$ and $\alpha_2 = 3$

Recalling the definition of $\bar{\alpha}$, we know that $\alpha_1 - 1$ cannot be a candidate for $\bar{\alpha}$. Hence, $\bar{\alpha} \geq \alpha_1$.

(ii) The upper bound $\alpha_2 = \lceil \frac{m\bar{c}-m}{M-m} \cdot N \rceil$. From $\alpha_2 \geq \frac{m\bar{c}-m}{M-m} \cdot N$, we get $p(\alpha_2) \geq m\bar{c}$. The area of a collection of rectangles above $p(\alpha_2), p(\alpha_2+1), p(\alpha_2+2), \dots, p(N)$ that inscribe the region S is

$$\sum_{j=\alpha_2+1}^{N-1} \frac{M-m}{N} \cdot \frac{1}{p(j)-m} = \sum_{j=\alpha_2+1}^{N-1} \frac{1}{j} < \bar{c}$$

(see Figure 2). The inequality

$$\frac{p(\alpha_2)}{m} - \sum_{j=\alpha_2+1}^{N} \frac{1}{j} > \bar{c} - \bar{c} = 0$$

says that α_2 is a candidate for $\bar{\alpha}$. Thus, $\bar{\alpha} \leq \alpha_2$. □

One may realize that for the case where $\frac{m\bar{c}-m}{M-m} \cdot N$ is not an integer, the lower and upper bound do not coincide. Unfortunately, Lemma 4.10 is the tightest possible as long as we approximate $\bar{\alpha}$ by employing $\frac{m\bar{c}-m}{M-m} \cdot N$. Indeed, the lower bound and the upper bound both may become tight due to the parameters. We give an example of $m = 1$ and $M = 2$. For either $N = 8$ or 9, it holds that $2 < \frac{m\bar{c}-m}{M-m} \cdot N < 3$. For $N = 8$, we have $\bar{\alpha} = 2$, which matches the lower bound. For $N = 9$, we have $\bar{\alpha} = 3$, which matches the upper bound. It is open whether a more precise bound can be given, or $\bar{\alpha}$ can be written in a closed form with $\bar{c}$.

A natural intuition is that the competitive ratio of the optimal algorithm for $(\mathcal{OWT}_N)$ approaches that for $(\mathcal{OWT}_\infty)$ as N grows, which is proved by Theorem 4.3.

Theorem 4.3. *It holds that*

$$\lim_{N \to \infty} \bar{y}_1 = \bar{y}(= \frac{1}{\bar{c}}).$$

Table 1: The number of partitions N and the competitive ratio $\frac{1}{\bar{y}_1}$ for $m=1$ and $M=2$

N	2	3	4	5	6	7	8	$\cdots$	∞
$\frac{1}{\bar{y}_1}$	1.16667	1.20833	1.21667	1.22381	1.2375	1.24286	1.24357	$\cdots$	1.27846

Proof. We bound $\bar{y}_1$ from below and from above, using Lemmas 3.5 and 4.10:

$$\bar{y}_1 \geq \frac{m}{p(\bar{\alpha})} \geq \frac{m}{m+\frac{M-m}{N}\cdot\lceil\frac{m\bar{c}-m}{M-m}\cdot N\rceil} > \frac{m}{m+\frac{M-m}{N}\cdot\left(\frac{m\bar{c}-m}{M-m}\cdot N+1\right)} = \frac{1}{\bar{c}+\frac{M-m}{Nm}}$$

and

$$\bar{y}_1 \leq \frac{m}{p(\bar{\alpha}-1)} \leq \frac{m}{m+\frac{M-m}{N}\cdot\left(\lfloor\frac{m\bar{c}-m}{M-m}\cdot N\rfloor-1\right)} < \frac{m}{m+\frac{M-m}{N}\cdot\left(\frac{m\bar{c}-m}{M-m}\cdot N-2\right)} = \frac{1}{\bar{c}-2\cdot\frac{M-m}{Nm}}.$$

As N grows, the lower and upper bounds each approach $\frac{1}{\bar{c}}(=\bar{y})$. By the squeeze theorem, we conclude that $\bar{y}_1$ also converges to $\frac{1}{\bar{c}}$. □

Here we give some numerical values for $m=1$ and $M=2$. Table 1 presents the values of $\frac{1}{\bar{y}_1}$ for $N=2,3,\ldots,8$ and its limit $(=\bar{c})$ as $N\to\infty$. The result is interpreted as the followings. A small N means that the number of possible events with the exchange rate is few. Then, it does not affect the performance so much whether the future information is available or not. In contrast, when N is large, the merit of having the knowledge of the future becomes relatively significant.

4.7. On the gap between the primal and dual objective function values

The solution $(\bar{\lambda},\bar{\mu})$ to $(\mathcal{D}_\infty)$, parameterized by ε, is seen as an asymptotic optimal solution. The primal and dual objective function values can be arbitrarily close but never coincide. Note that $(\bar{\lambda},\bar{\mu})$ is not defined for $\varepsilon=0$; the value of $\bar{\lambda}(p)$ for $p=M$ diverges to infinity as $\varepsilon\to 0$.

There is a way to avoid this difficulty. See where this difficulty comes from. We have failed to construct an *arbitrary* linear combination of constraints of $(\mathcal{P}_\infty)$ in Section 4.3. In fact, λ introduced there does not represent an arbitrary linear combination. Viewed from a different perspective, $(\mathcal{D}_\infty)$ is a problem for which only the *weak* duality theorem holds true.

In this section we properly use the terminology *functional* rather than combination, since we discuss what maps a function to a real number. The representation of an arbitrary linear functional depends on the function space that the constraints belong to. (We have skipped such careful arguments so far.) Assume that x is a Lebesgue integrable function and integrals appearing in $(\mathcal{P}_\infty)$ are Lebesgue integrals. Then, it is known that the left-hand side of (4.5) is a continuous function of p. Theorem 4.4 tells us how we should describe an arbitrary linear functional. The right-hand side of (4.16) is a Lebesgue-Stieltjes integral.

Theorem 4.4 (the Riesz representation theorem (see [7] for example))**.** *Every linear functional J on the set of continuous functions from $[m,M]$ to $\mathbb{R}$ can be represented in the form*

$$J(g)=\int_m^M g(p)d\Lambda(p), \tag{4.16}$$

where g is a continuous function on $[m,M]$ and Λ is a bounded variation on $[m,M]$.

Applying Theorem 4.4, we have the following inequality, instead of (4.7),

$$\int_m^M \left\{ -\int_m^p (q-m)x(q)dq + py \right\} d\Lambda(p) + \mu \int_m^M x(q)dq \le m \int_m^M d\Lambda(p) + \mu.$$

Then we have the dual problem with variables (Λ, μ) and get an optimal Λ:

$$\bar{\Lambda}(p) = \begin{cases} 0, & m \le p < m\bar{c}; \\ -\frac{\bar{\mu}}{(p-m)} + \frac{\bar{\mu}}{(m\bar{c}-m)}, & m\bar{c} \le p < M; \\ \frac{\bar{\mu}}{M-m}, & p = M. \end{cases}$$

It is easily confirmed that the derivative of $\bar{\Lambda}$ equals to $\bar{\lambda}$ on intervals $(m, m\bar{c})$ and $(m\bar{c}, M-\varepsilon)$. Calculating the integrals, we have that there is no longer a gap between the primal and dual objective function values.

To conduct rigorous analysis for infinite-dimensional linear optimization, discussion on function spaces is required. See [8] for example. Before this section we have managed without such discussion. Our claim is not that we should always take good care about function spaces. We would like to say that it is sometimes the case that for practical examples of infinite-dimensional linear optimization, such as the one-way trading problem, an asymptotic optimal solution works enough.

Indeed, although λ introduced in Section 4.3 does not represent an arbitrary linear combination, the objective function of $(\mathcal{D}_\infty)$ is still valid as an upper bound on the objective function of $(\mathcal{P}_\infty)$. Hence, it is done if we find a pair of a primal and a dual solution such that their objective function values can be arbitrarily close.

Our analysis is valid under the assumption that integrals are all Riemann integrals. We have avoided introducing a Stieltjes integral by letting the value of $\bar{\lambda}$ on interval $[M-\varepsilon, M]$ be finite. All the facts we want can be proved as long as $\varepsilon > 0$ is chosen arbitrarily.

5. Concluding Remarks

As far as we know, there are only a few papers in the literature that solve a specific set of instances of infinite-dimensional linear optimization problems. In Section 4 we demonstrated that a problem in the real world can be formulated into an infinite-dimensional linear optimization problem. This paper will be successful if the reader sees that it is sometimes possible to find an analytical solution to such a linear optimization problem.

Most applications in the real world are a problem of a finite dimension. Even in such a case, it would be helpful to consider both finite-dimensional and infinite-dimensional optimization as follows. First, run some computational experiments on a finite-dimensional problem and check if there is any irregularity such as the jump in $(\bar{\lambda}_1, \ldots, \bar{\lambda}_N)$ in Theorem 3.2. Next, formulate an infinite-dimensional problem and find an optimal solution. Finally, obtain a solution to the finite-dimensional problem by some rounding.

Acknowledgments

We thank the editor and the anonymous reviewers for their valuable comments. We are very grateful to Keiji Hirao for correcting errors in expressions in the draft. We also thank participants of the Enumeration Algorithms Seminars, Ikaho, Japan, for their helpful discussions. This work was supported by KAKENHI (16K00033, 17K00013, and 17K00183).

References

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

[2] F.Y.L. Chin, B. Fu, J. Guo, S. Han, J. Hu, M. Jiang, G. Lin, H. Ting, L. Zhang, Y. Zhang, and D. Zhou: Competitive algorithms for unbounded one-way trading. *Theoretical Computer Science*, **607** (2015), 35–48.

[3] V. Chvátal: *Linear Programming* (Freeman, 1983).

[4] E. Dannoura and K. Sakurai: An improvement on El-Yaniv-Fiat-Karp-Turpin's money-making bi-directional trading strategy. *Information Processing Letters*, **66** (1998), 27–33.

[5] R. Dochow: *Online Algorithms for the Portfolio Selection Problem* (Springer Gabler, 2016).

[6] R. El-Yaniv, A. Fiat, R.M. Karp, and G. Turpin: Optimal search and one-way trading online algorithms. *Algorithmica*, **30** (2001), 101–139.

[7] A.N. Kolmogorov and S.V. Fomin: *Introductory Real Analysis* (Dover Publications, 1975).

[8] D.G. Luenberger: *Optimization by Vector Space Methods* (John Wiley & Sons Inc., New York, 1969).

[9] W. Wang, L.Y. Wang, Y.J. Lan, and J.X. Zhang: Competitive difference analysis of the one-way trading problem with limited information. *European Journal of Operational Research*, **252** (2016), 879–887.

Hiroshi Fujiwara
Department of Electrical and Computer Engineering
Shinshu University
4-17-1 Wakasato Nagano, 380-8553 Japan
E-mail: `fujiwara@cs.shinshu-u.ac.jp`