Optimal Buy-and-Hold Strategies for Financial Markets with Bounded Daily Returns

Gen-Huey Chen* Ming-Yang Kao† Yuh-Dauh Lyuu*‡ Hsing-Kuo Wong*‡

Abstract

A general solution is presented for any finite request-answer game to derive its optimal competitive ratio and optimal randomized on-line algorithm against the oblivious adversary. The solution is based on game theory. We then apply the framework to the practical buy-and-hold trading problem and find the exact optimal competitive ratio and an optimal randomized on-line algorithm. We also prove the uniqueness of the solution.

1 Introduction

Ben-David *et al.* [BDBK+94] formulated on-line problems as *request-answer games*. In the request-answer game, the *on-line algorithm* acts on each request before it serves the next one. Each such action generates a certain gain. In *competitive analysis*, an algorithm's performance is defined to be the ratio of the total gain of the best off-line algorithm and that of the on-line algorithm that services the same sequence of requests over the worst-case inputs. See [BEY98, Hoc97, MR95] for surveys. This ratio is called the *competitive ratio*, which the on-line algorithm seeks to minimize.

We present a general solution to any finite request-answer game by identifying the problem of solving the game with its corresponding linear programming problem. The optimal competitive ratio emerges as the reciprocal of the value of the game, and the optimal randomized on-line algorithm emerges from the optimal feasible solution. By this correspondence, we can solve for the optimal randomized on-line algorithm against the oblivious adversary. An adversary is *oblivious* if he does not have access to the on-line algorithm's random bits.

Previously, the use of game theory had been focused on deriving lower bounds, the so-called Yao Principle [Yao77] being a well-known result. It seems no one has recognized that, in fact, solving a game actually gives us the optimal competitive ratio and optimal randomized on-line algorithms for finite request-answer games. The discovery of this general methodology is a main contribution of this paper.

As an application of this general paradigm, we consider the buy-and-hold trading problem. This is a problem that faces millions of investors who save for retirement purposes on a long term basis. We find an optimal strategy that can be executed even by small investors who are not mathematically sophisticated. The *buy-and-hold trading problem* is an on-line financial problem that can be described as follows. An on-line investor proposes an n-day investment plan. The investor starts with some capital and plans to trade it for a certain security. The investor executes one transaction per day and may trade only partial capital. However, all the capital must be traded by the end of the investment horizon, and converting security back to capital is inhibited. For ease of expression, we will assume the capital is dollars and the security is yen throughout the paper. In this case, the relative price between the capital and the security used in determining the number of units purchased becomes the exchange rate. Note that *any* commodities in which *relative prices* are obtainable fit the above setup.

We adopt the *bounded daily return model.* This model assumes the next exchange rate e' depends on the current exchange rate e in a geometric manner, i.e., $e/\theta \leq e' \leq e\theta$ for some $\theta > 1$. We call such θ the *daily fluctuation ratio.* Stock markets often enforce such ratios through *circuit breakers.* In this paper, we assume that the problem horizon n and the bound of the daily fluctuation ratio θ are prior knowledge.

This financial model is mentioned in [BEY98, CCEY+95, EY98] and is related to the geometric Brownian motion model used extensively in the finance community [Hul97, Lyu99]. Under this model, we derive an optimal static buy-and-hold trading strategy called

*Dept. of Computer Science and Information Engineering, National Taiwan University.

†Dept. of Computer Science, Yale University.

‡Partially supported by NSC Grant 87-2416-H-002-031.

the *balanced strategy*, derive its competitive ratio, and prove the uniqueness of the optimal strategy. Specifically, the optimal competitive ratio is $\frac{n\theta-(n-2)}{\theta+1}$. The optimal strategy works as follows, starting with one dollar initially: It invests $\frac{\theta}{n\theta-(n-2)}$ dollars on the first and last days and $\frac{\theta-1}{n\theta-(n-2)}$ dollars on other days. We also design a dynamic strategy that improves the performance of optimal strategies on non-worst-case inputs.

There are few papers on on-line computation concerning the systematic solution to deriving the lower bound of optimal competitive ratio. The Yao Principle is the first application of the minimax theorem to derive the lower bound of optimal performance for randomized algorithms [Yao77]. Borodin *et al.* [BEY97] used the Yao Principle to summarize lower bounds for request-answer games with finite or infinite time horizon. The use of these formulas must efficiently specify a mixed strategy for the adversary to raise the lower bound. For that purpose, the uniformly mixed strategy is a common heuristic [Yao77]. However, §3.5 will present a case for which uniformly mixed strategy is not optimal. Specifically, that practice implies a lower bound $\frac{n(1-\theta^{-1})}{(1-\theta^{-\lceil n/2\rceil})+(\theta^{-1}-\theta^{-\lfloor n/2\rfloor-1})}$, which is strictly lower than our optimal competitive ratio.

Our paper is organized as follows. In §2 we present a general solution for any finite request-answer game to derive its optimal randomized strategy against the oblivious adversary. In §3 we employ the general solution to derive an optimal buy-and-hold trading strategy called the balanced strategy. Furthermore, we derive the optimal competitive ratio exactly and prove the uniqueness of the optimal strategy. In §4 we dicuss whether real-time information can help static strategies and derive some theoretical properties. Section 5 concludes the paper and points to future directions.

2 Optimal Solution to Finite Request-Answer Games

There are few papers on on-line computation concerning about the systematic approach to derive the lower bound of optimal competitive ratio. The Yao Principle is the first application of the minimax theorem in this regard for randomized algorithms [Yao77].

We fix some game-theoretical notations used for the rest of the paper. Let

$$Z_k \equiv \{1, 2, \dots, k\}.$$

Denote by $\Gamma_H(m,n)$ a finite zero-sum two-person game, where the set of pure strategies for maximizing and minimizing players are indexed by Z_m and Z_n, respectively. The payoff of players using strategies $i \in Z_m$ and $j \in Z_n$ is denoted by $H(i,j)$. Let $\Im(Z_k)$ be the set of all probability density functions defined on Z_k. Each $f \in \Im(Z_k)$ determines a mixed strategy that applies pure strategy $i \in Z_k$ with probability $f(i)$.

2.1 Definitions and Terminology

Ben-David *et al.* [BDBK+94] formulated on-line problems as request-answer games. This paper considers finite request-answer games with finite problem horizon. By finiteness we mean the number of possible request sequences and that of deterministic on-line algorithms are both finite. Deterministic on-line algorithms and request sequences can be viewed as pure strategies, while randomized on-line algorithms of the on-line player can be viewed as mixed strategies. In particular, both the on-line player and the adversary have a finite number of pure strategies.

For profit maximization problems, Z_m and Z_n denote the set of pure strategies for the on-line player and the adversary, respectively. Let $\mathrm{A}_i, i \in Z_m$, and $\mathbf{e}_j, j \in Z_n$, denote the deterministic on-line algorithm and the request sequence, respectively. Finally, let $H(i,j)$ be the ratio of the profit of A_i on $\mathbf{e}_j$ to the profit of the optimal off-line algorithm OPT on $\mathbf{e}_j$ as

$$H(i,j) \equiv \frac{\mathrm{A}_i(\mathbf{e}_j)}{\mathrm{OPT}(\mathbf{e}_j)}, \quad i \in Z_m, j \in Z_n. \tag{1}$$

Cost minimization problems can be formulated similarly. For convenience, the finite request-answer game is always assumed to be of the profit maximization kind throughout this paper.

Clearly each finite request-answer game is a finite zero-sum two-person game $\Gamma_H(m,n)$, and each payoff value is positive, $H(i,j) > 0$. The optimal competitive ratio r^* of randomized on-line algorithms against the oblivious adversary is defined as

$$r^* \equiv \inf_{f \in \Im(Z_m)} \max_{j \in Z_n} \frac{\mathrm{OPT}(\mathbf{e}_j)}{\mathbf{E}_{f(i)} \mathrm{A}_i(\mathbf{e}_j)}. \tag{2}$$

A randomized on-line algorithm will be called *optimal* if it achieves the above competitive ratio.

2.2 Minimax Theorem and Its Applications

Since $\Gamma_H(m,n)$ is a finite zero-sum two-person game, the minimax theorem of von Neumann holds [PZ96, Theorem 1.6.1]. Thus,

$$\begin{aligned} v^* &= \max_{f \in \Im(Z_m)} \min_{g \in \Im(Z_n)} \mathbf{E}_{f(i)} \mathbf{E}_{g(j)} H(i,j) \\ &= \min_{g \in \Im(Z_n)} \max_{f \in \Im(Z_m)} \mathbf{E}_{f(i)} \mathbf{E}_{g(j)} H(i,j), \end{aligned}$$

where v^* is called the *value* of the game. With (1), (2) and [PZ96, Theorem 1.7.3], we also obtain the following relationship for the optimal competitive ratio,

$$\frac{1}{r^*} = \max_{f \in \Im(Z_m)} \min_{j \in Z_n} \mathbf{E}_{f(i)} H(i,j) = \min_{g \in \Im(Z_n)} \max_{i \in Z_m} \mathbf{E}_{g(j)} H(i,j) = v^*. \quad (3)$$

We conclude that the optimal mixed strategy of the on-line player is the optimal randomized on-line algorithm against the oblivious adversary and $r^* = 1/v^*$. One of our main purposes in this paper is to derive the exact value of r^* and the optimal randomized on-line algorithm.

Note that $1/r^* \leq \max_{i \in Z_m} \mathbf{E}_{g(j)} H(i,j)$ for any mixed strategy g of the adversary. Thus the value of $\min_{i \in Z_m} \left(\mathbf{E}_{g(j)} H(i,j)\right)^{-1}$ is a lower bound of r^*. This is called the Yao principle [BEY97].

Note also that if the payoff function defined in (1) is defined by $H'(i,j) \equiv H(i,j)^{-1}$, and the on-line player and the adversary are the minimizing and maximizing player respectively, then the value of $\min_{i \in Z_m} \mathbf{E}_{g(j)} H'(i,j)$ is not guaranteed to be a lower bound for r^*. The fallacy of believing otherwise comes from the dubious identity marked with a question mark below

$$\begin{aligned} r^* &= \inf_{f \in \Im(Z_m)} \max_{j \in Z_n} \frac{1}{\mathbf{E}_{f(i)} H(i,j)} \\ &\stackrel{?}{=} \inf_{f \in \Im(Z_m)} \max_{j \in Z_n} \mathbf{E}_{f(i)} \frac{1}{H(i,j)} \\ &= \inf_{f \in \Im(Z_m)} \max_{j \in Z_n} \mathbf{E}_{f(i)} H'(i,j) \\ &= \sup_{g \in \Im(Z_n)} \min_{i \in Z_m} \mathbf{E}_{g(i)} H'(i,j) \\ &\geq \min_{i \in Z_m} \mathbf{E}_{g(j)} H'(i,j). \end{aligned}$$

This wrong use of the Yao Principle was pointed out by Borodin *et al.* [BEY97]. Based on the Yao Principle, they summarized lower bound formulas for request-answer games with finite or infinite time horizon. However, in order to obtain a tight lower bound with these formulas, one must specify a sufficiently bad mixed strategy for the the adversary to maximize $\min_{i \in Z_m} \left(\mathbf{E}_{g(j)} H(i,j)\right)^{-1}$. A common heuristic is to employ the uniformly mixed strategy [Yao77]. Section 3.5 will present a case in which the uniformly mixed strategy is not optimal.

2.3 Request-Answer Games and the Primal and Dual Problems

For convenience, We view each mixed strategy as a point in the Euclidean space and represent the payoff function by a matrix. Let $\mathbf{H}$ be the payoff matrix of the finite request-answer game $\Gamma_H(m,n)$. We adapt a theorem from [PZ96, Theorem 1.6.2] as follows.

Lemma 1 *For each finite request-answer game $\Gamma_H(m,n)$, define its corresponding primal and dual problems as follows,*

Primal:		*Dual:*	
minimize	$\mathbf{x}^T \mathbf{u}_m$	*maximize*	$\mathbf{y}^T \mathbf{u}_n$
subject to	$\mathbf{x}^T \mathbf{H} \geq \mathbf{u}_n^T$	*subject to*	$\mathbf{H}\,\mathbf{y} \leq \mathbf{u}_m$
	$\mathbf{x} \geq 0$		$\mathbf{y} \geq 0$

where $\mathbf{u}_i$ is the vector consisting of i 1's. Let $\bar{X}$ and $\bar{Y}$ denote the set of optimal feasible solutions to the primal and dual problems, respectively and let X^* and Y^* denote the set of optimal mixed strategies of the on-line player and the adversary, respectively. Then the following properties hold:

1. *$\bar{X} \neq \emptyset$, $\bar{Y} \neq \emptyset$, and $1/v^* = \min_{\mathbf{x}} \mathbf{x}^T \mathbf{u}_m = \max_{\mathbf{y}} \mathbf{y}^T \mathbf{u}_n$, where $\mathbf{x}$ and $\mathbf{y}$ range over feasible solutions.*

2. *$X^* = v^* \bar{X}$ and $Y^* = v^* \bar{Y}$.*

Solving any finite request-answer game is therefore equivalent to the corresponding primal and dual problems. Lemma 1 will help us to find optimal randomized strategy and its competitive ratio for finite request-answer games.

By Equation (3), we have the following corollary.

Corollary 2 *For each finite request-answer game $\Gamma_H(m,n)$, let $\mathbf{x}$ and $\mathbf{y}$ denote the feasible solutions to its corresponding primal and dual problems, respectively. The following properties hold:*

1. *$r^* = \min_{\mathbf{x}} \mathbf{x}^T \mathbf{u}_m = \max_{\mathbf{y}} \mathbf{y}^T \mathbf{u}_n$.*

2. *$\mathbf{y}^T \mathbf{u}_n \leq r^* \leq \mathbf{x}^T \mathbf{u}_m$.*

3. *$\mathbf{x}' \equiv \frac{\mathbf{x}}{\mathbf{x}^T \mathbf{u}_m}$ and $\mathbf{y}' \equiv \frac{\mathbf{y}}{\mathbf{y}^T \mathbf{u}_n}$, where $\mathbf{y} \neq \mathbf{0}$, are mixed strategies for the on-line player and the adversary.*

4. *The pure strategy j of adversary that $\mathbf{x}^T \mathbf{H}^j = \min_{l \in Z_n} \mathbf{x}^T \mathbf{H}^l$ is a worst case for the on-line player's mixed strategy $\mathbf{x}'$.*

Observe that for any feasible solution $\mathbf{y}$ to the dual problem, the value $\mathbf{y}^T \mathbf{u}_n$ is a lower bound for the optimal competitive ratio of randomized on-line algorithms against the oblivious adversary.

In Lemma 1, if $\mathbf{H}$ is symmetric and square, and $\mathbf{x}$ is a feasible solution to the primal and dual problems, then $\mathbf{x}$ is an optimal feasible solution to the primal and dual problems, and $r^* = \mathbf{x}^T \mathbf{u}_n$ [CZ96, Theorem 17.1]. This useful observation is summarized below.

Lemma 3 *For each finite request-answer game $\Gamma_H(m,n)$, if $m = n$, $\mathbf{H}^T = \mathbf{H}$ and there exists an $\mathbf{x} \geq 0$ such that $\mathbf{Hx} = \mathbf{u}_n$, then $\mathbf{x}$ is an optimal feasible solution to the primal and dual problems.*

3 Applications to Optimal Static Buy-and-Hold Strategies

In this section, we will apply the general paradigm in §2 to solve a practical and complicated buy-and-hold trading problem. In particular, we derive explicitly both the optimal competitive ratio and the algorithm to achieve it.

3.1 Problem Definition and Notations

Start with θ, the maximum fluctuation ratio of two adjacent daily exchange rates, and n, the problem horizon (the number of trading days), such that $1 < \theta$ and $n \geq 2$. The numbers θ and n are known a priori to the on-line investor. We consider exchange rate sequences $\mathbf{e} = \langle e_1, e_2, \ldots, e_n \rangle$ satisfying the bounded fluctuation ratio, i.e., $e_i \in [e_{i-1}\theta^{-1}, e_{i-1}\theta]$. We normalize e_0 to 1 to simplify the presentation. The exchange rate sequence $\mathbf{e}$ will be revealed sequentially to the on-line investor. Upon each revelation, the investor must decide on the fraction of dollars to be traded for yen without any information regarding future exchange rates. All dollars must be traded for yen in n transactions and converting yen back to dollars is not allowed. Without loss of generality, the investor starts with one dollar.

Let $\mathcal{E} \equiv \{\mathbf{e} : \mathbf{e} = \langle e_1, e_2, \ldots, e_n \rangle$ and $e_i \in [e_{i-1}\theta^{-1}, e_{i-1}\theta]$, for $i \in Z_n\}$, denote the set of all *admissible* exchange rate sequences. OPT will denote the optimal off-line trading algorithm. The return of OPT on an admissible exchange rate sequence $\mathbf{e} = \langle e_1, e_2, \ldots, e_n \rangle$ is clearly $\mathrm{OPT}(\mathbf{e}) = \max_{1 \leq i \leq n} e_i$.

For any deterministic (randomized, resp.) on-line trading algorithm S, define its return (expected return, resp.) on the exchange rate sequence $\mathbf{e}$ by $\mathrm{S}(\mathbf{e}) \equiv \sum_{i=1}^{n} a_i e_i$, where a_i is the amount (expected amount, resp.) of dollars invested by S on the ith day. (The a_i's may be dependent on the current and past exchange rates, $e_1, e_2, \ldots, e_i$.) Its competitive ratio against the oblivious adversary is defined as

$$\Upsilon_{\mathrm{S}} \equiv \sup_{\mathbf{e} \in \mathcal{E}} \frac{\mathrm{OPT}(\mathbf{e})}{\mathrm{S}(\mathbf{e})},$$

where the oblivious adversary does not have access to S's future coin flips.

We say an on-line trading algorithm is *static* if the (expected) amount of dollars invested by the algorithm on the ith day is fixed for all exchange rate sequences. An algorithm is *dynamic* otherwise.

For the on-line trading algorithm S_i which trades all the dollars on the ith day, we call it a *trade-once algorithm* on the ith day. Clearly this algorithm is static and its return on $\mathbf{e}$ is $\mathrm{S}_i(\mathbf{e}) = e_i$.

The optimal competitive ratio for static trading algorithms is defined as

$$r_{\mathrm{s}}^* \equiv \inf_{\mathrm{S}} \sup_{\mathbf{e} \in \mathcal{E}} \frac{\mathrm{OPT}(\mathbf{e})}{\mathrm{S}(\mathbf{e})}, \tag{4}$$

where S ranges over all static trading algorithms. For each static trading algorithm S, let s_i be the (expected) amount of dollars invested by S on the ith day. Then the (expected) return of S on $\mathbf{e}$ is $\mathrm{S}(\mathbf{e}) = \sum_{i=1}^{n} s_i e_i$. Since $s_i \geq 0$ for all i and $\sum_{i=1}^{n} s_i = 1$, these s_i's define a probability density function $f \in \Im(Z_n)$ in which $f(i) = s_i$ is the probability of applying the trade-once algorithm S_i. In other words,

$$\mathrm{S}(\mathbf{e}) = \sum_{i=1}^{n} f(i)\, \mathrm{S}_i(\mathbf{e}) = \mathbf{E}_{f(i)} \mathrm{S}_i(\mathbf{e}).$$

Thus $\Im(Z_n)$ can represent the set of all static trading algorithms.

The above and (4) combined, the optimal competitive ratio of static trading algorithms can be restated as follows,

$$r_{\mathrm{s}}^* = \inf_{f \in \Im(Z_n)} \sup_{\mathbf{e} \in \mathcal{E}} \frac{\mathrm{OPT}(\mathbf{e})}{\mathbf{E}_{f(i)} \mathrm{S}_i(\mathbf{e})}. \tag{5}$$

(By "optimal" we mean, throughout the rest of the paper, "optimal in terms of competitive ratio.")

We can view the buy-and-hold trading problem as an infinite request-answer game. The on-line player has only n pure strategies, but the adversary has infinite pure strategies. In order to use our general solution in §2, we will reduce this infinite request-answer game to be finite. We need to eliminate the *dominated* pure strategies (non-worst-case exchange rate sequences) of adversary in the next subsection.

3.2 Elimination of Dominated Pure Strategies of Adversary

An important step in finding a problem's worst case for competitive analysis is in reducing the infinite number of possibilities to a finite number. We will carry out that step here.

Let S be a static trading algorithm and $f \in \Im(Z_n)$ be its corresponding probability density function. Consider an exchange rate sequence $\mathbf{e} = \langle e_1, e_2, \ldots, e_n \rangle \in \mathcal{E}$. We define its corresponding fluctuation ratio sequence by $\mathbf{d} = \langle d_1, d_2, \ldots, d_n \rangle$, where $d_i \equiv e_i / e_{i-1}$. Recall that $e_0 = 1$. Let j be such that $e_j = \mathrm{OPT}(\mathbf{e})$.

Since the competitive ratio of S is

$$\Upsilon_{S} \equiv \sup_{\mathbf{e}\in\mathcal{E}} \frac{\mathrm{OPT}(\mathbf{e})}{\sum_{i=1}^{n} f(i)\,\mathrm{S}_i(\mathbf{e})},$$

the adversary must make the remaining rates decrease monotonically with the maximum ratio θ^{-1}; i.e., $d_i = \theta^{-1}$ for $i = j+1, \ldots, n$. Fix an arbitrary $l \le j$. Define $\lambda \ge 1$ such that $d_l' = \lambda d_l = \theta$ and let $\mathbf{d}' = \langle d_1', d_2', \ldots, d_n' \rangle$ where $d_i' = d_i$ for $i \ne l$. Then for $i < l$,

$$\frac{\mathrm{S}_i(\mathbf{e}')}{\mathrm{OPT}(\mathbf{e}')} = \frac{d_1 \cdots d_i}{\lambda d_1 \cdots d_j} \le \frac{d_1 \cdots d_i}{d_1 \cdots d_j} = \frac{\mathrm{S}_i(\mathbf{e})}{\mathrm{OPT}(\mathbf{e})},$$

while for $i \ge l$,

$$\frac{\mathrm{S}_i(\mathbf{e}')}{\mathrm{OPT}(\mathbf{e}')} = \frac{\lambda d_1 \cdots d_i}{\lambda d_1 \cdots d_j} = \frac{d_1 \cdots d_i}{d_1 \cdots d_j} = \frac{\mathrm{S}_i(\mathbf{e})}{\mathrm{OPT}(\mathbf{e})}.$$

Thus $\mathbf{e}$ is dominated by $\mathbf{e}'$. So

$$\frac{\mathrm{S}(\mathbf{e}')}{\mathrm{OPT}(\mathbf{e}')} \le \frac{\mathrm{S}(\mathbf{e})}{\mathrm{OPT}(\mathbf{e})},$$

for any static trading algorithm S. Therefore,

Lemma 4 *Let $\mathbf{e}_j = \langle e_1, e_2, \ldots, e_n \rangle \in \mathcal{E}$, for $j \in Z_n$, be the exchange rate sequence defined by*

$$e_i = \begin{cases} \theta^i & \text{if } i \le j \\ \theta^{2j-i} & \text{if } i > j. \end{cases}$$

Then each worst-case sequence for any static algorithms is dominated by $\mathbf{e}_j$ for some $j \in Z_n$.

Therefore, we need only consider the n exchange rate sequences: $\mathbf{e}_j$, for $j \in Z_n$. We call each of these a *downturn*. Figure 1 illustrates the relationships among the $\mathbf{e}_j$'s.

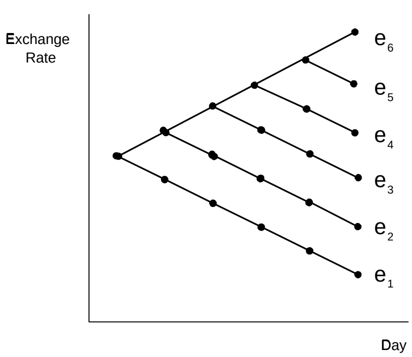

Figure 1: The patterns of downturns.

Lemma 4 and (5) show the optimal competitive ratio of static trading algorithms to be

$$r_{\mathrm{s}}^* = \inf_{f\in\Im(Z_n)} \max_{1\le j\le n} \frac{\mathrm{OPT}(\mathbf{e}_j)}{\mathbf{E}_{f(i)}\mathrm{S}_i(\mathbf{e}_j)}. \tag{6}$$

3.3 Reduction to Finite Request-Answer Games

Our problem can now be reduced to a finite request-answer game. By pure strategy i (for the on-line player) we mean the trade-once algorithm S_i for $i \in Z_n$, and by pure strategy j (for the adversary) we mean the downturn $\mathbf{e}_j$ for $j \in Z_n$. Denote the payoff function if players use pure strategies i and j by $K(i,j) \equiv \frac{\mathrm{S}_i(\mathbf{e}_j)}{\mathrm{OPT}(\mathbf{e}_j)}$. As $\mathrm{OPT}(\mathbf{e}_j) = \theta^j$, clearly

$$K(i,j) = \theta^{-|i-j|}, \quad \text{for } i,j \in Z_n. \tag{7}$$

The objective of the on-line player is to maximize this payoff function, while that of the adversary the opposite. Our finding is summarized below.

Lemma 5 *Static buy-and-hold trading problem can be formulated as a finite request-answer game $\Gamma_K(n,n)$. Furthermore, if the on-line player adopts pure strategy $i \in Z_n$ and the adversary adopts pure strategy $j \in Z_n$, then their payoff function is as defined in* (7).

The optimal mixed strategy of the on-line player is hence an optimal static trading algorithm. It only remains to solve this game.

3.4 Deriving Optimal Competitive Ratio and Randomized On-Line Algorithm

Since the payoff matrix $\mathbf{K}$ of $\Gamma_K(n,n)$ is symmetric and square, if we can find an n-dimensional vector $\bar{\mathbf{b}} \ge \mathbf{0}$ such that $\mathbf{K}\bar{\mathbf{b}} = \mathbf{u}_n$, then by Lemma 3 and Corollary 2, we have found the optimal competitive ratio $r_{\mathrm{s}}^* = \bar{\mathbf{b}}^T\mathbf{u}_n$. Let $\mathbf{B}^*$ be the corresponding randomized on-line algorithm of $\mathbf{b}^* \equiv \bar{\mathbf{b}}/r_{\mathrm{s}}^*$. By Lemma 1, $\mathbf{b}^*$ is the optimal mixed strategy of the on-line player. Clearly $\mathbf{b}^*$ must satisfy

$$\mathbf{K}\,\mathbf{b}^* = \frac{1}{r_{\mathrm{s}}^*}\mathbf{u}_n. \tag{8}$$

Thus, $\mathbf{b}^* = (1/r_{\mathrm{s}}^*)\,\mathbf{K}^{-1}\,\mathbf{u}_n$ if $\det(\mathbf{K}) \ne 0$.

Equation (8) implies $\mathbf{E}_{\mathbf{b}^*(i)}K(i,j) = 1/r_{\mathrm{s}}^*$ for $j \in Z_n$. We therefore call $\mathbf{B}^*$ the *balanced trading algorithm* or *balanced strategy* in short. The intuition behind the algorithm is that it balances the performance of the on-line player on all downturns. We now proceed to show the existence of the balanced algorithm and derive explicitly its competitive ratio and strategy.

Recall from (7) that $K(i,j) = \theta^{-|i-j|}$. We can express the matrix form of $\mathbf{K}$ below,

$$\mathbf{K} = \begin{pmatrix} \theta^0 & \theta^{-1} & \theta^{-2} & \cdots & \theta^{1-n} \\ \theta^{-1} & \theta^0 & \theta^{-1} & \cdots & \theta^{2-n} \\ \theta^{-2} & \theta^{-1} & \theta^0 & \cdots & \theta^{3-n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \theta^{1-n} & \theta^{2-n} & \theta^{3-n} & \cdots & \theta^0 \end{pmatrix}.$$

The following lemma is easy to verify.

Lemma 6 *The determinant of* $\mathbf{K}$ *is positive and is equal to* $\det(\mathbf{K}) = (1-\theta^{-2})^{n-1} > 0$ *for* $n \geq 2$.

Now that we have $\det(\mathbf{K}) > 0$, the existence of the balanced trading algorithm is thus established. Clearly $\mathbf{b}^* = (1/r_s^*)\,\mathbf{K}^{-1}\,\mathbf{u}_n$. We finally solve for $\mathbf{b}^*$ and r_s^* in the following theorem.

Theorem 7 *Let* $\mathbf{B}^*$ *be a static trading algorithm that invests* b_i^* $(i = 1, 2, \ldots, n)$ *dollars on the ith day where*

$$b_i^* = \begin{cases} \frac{\theta}{n\theta-(n-2)} & i = 1 \text{ or } n \\ \frac{\theta-1}{n\theta-(n-2)} & i = 2, \ldots, n-1 \text{ and } n > 2. \end{cases}$$

Then $\mathbf{B}^*$ *is the unique balanced trading algorithm and is optimal for the class of static trading algorithms; in other words,*

$$\frac{\mathrm{OPT}(\mathbf{e}_j)}{\mathbf{B}^*(\mathbf{e}_j)} = r_s^*, \quad \text{for } j \in Z_n.$$

Furthermore, the optimal competitive ratio for static trading algorithms equals

$$r_s^* = \frac{n\theta-(n-2)}{\theta+1}.$$

Proof. Since these b_i^*'s specify a probability density function, $\mathbf{B}^*$ is a static trading algorithm. To prove that $\mathbf{B}^*$ is balanced, it suffices to show that $\sum_{i=1}^n b_i^* K(i,j) = \frac{\theta+1}{n\theta-(n-2)}$ for $j \in Z_n$.

- For $j = 1$ or n:

$$\begin{aligned}&(n\theta-(n-2))\sum_{i=1}^n b_i^* K(i,j)\\ &= \theta\theta^0 + (\theta-1)\theta^{-1} + (\theta-1)\theta^{-2} + \\ &\qquad \cdots + (\theta-1)\theta^{2-n} + \theta\theta^{1-n}\\ &= \theta+1.\end{aligned}$$

- For $2 \leq j \leq n-1$:

$$\begin{aligned}&(n\theta-(n-2))\sum_{i=1}^n b_i^* K(i,j)\\ &= \theta\theta^{1-j} + \sum_{2\leq i\leq n-1} (\theta-1)\theta^{-|i-j|} + \theta\theta^{j-n}\\ &= \theta+1.\end{aligned}$$

Thus, for all $j \in Z_n$, the values of $\sum_{i=1}^n b_i^* K(i,j)$ are identical and equal $\frac{\theta+1}{n\theta-(n-2)}$, i.e., (8) is satisfied. Therefore, algorithm $\mathbf{B}^*$ is a balanced trading algorithm and is optimal.

The uniqueness of the balanced trading algorithm is obvious because the solution $\mathbf{B}^*$ in (8) is unique. □

3.5 Comparison with the Dollar-Averaging Strategy

We analyze the popular *dollar-averaging strategy* (DA). DA is a uniformly mixed strategy for the on-line player. It invests equal amounts of dollars on each trading day with a return on the exchange rate sequence $\mathbf{e} = \langle e_1, e_2, \ldots, e_n\rangle$ equal to $\mathrm{DA}(\mathbf{e}) = (\sum_{i=1}^n e_i)/n$.

By Lemma 4, the competitive ratio of DA is

$$\begin{aligned}\Upsilon_{\mathrm{DA}} &= \max_{1\leq j\leq n} \frac{n}{\sum_{i=1}^n \theta^{-|i-j|}}\\ &= \max_{1\leq j\leq n} \frac{n\,(1-\theta^{-1})}{(1+\theta^{-1}) - (\theta^{-j}+\theta^{j-n-1})} \qquad (9)\\ &= \frac{n\,(1-\theta^{-1})}{1-\theta^{-n}}.\end{aligned}$$

By Equation (3), we apply the uniformly mixed strategy (DA) to the Yao Principle, and obtain a lower bound for r_s^*.

$$\begin{aligned}r_s^* &\geq \min_{i\in Z_m} \left(\mathbf{E}_{\mathrm{DA}(j)} K(i,j)\right)^{-1}\\ &= \min_{1\leq j\leq n} \frac{n}{\sum_{i=1}^n \theta^{-|i-j|}}\\ &= \min_{1\leq j\leq n} \frac{n\,(1-\theta^{-1})}{(1-\theta^{-j}) + (\theta^{-1}-\theta^{j-n-1})}\\ &= \frac{n\,(1-\theta^{-1})}{(1-\theta^{-\lceil n/2\rceil}) + (\theta^{-1}-\theta^{-\lfloor n/2\rfloor-1})}\\ &\equiv r_u. \qquad (10)\end{aligned}$$

Then by (9) and (10), we have

$$r_u < r_s^* < \Upsilon_{\mathrm{DA}}.$$

Since the finite request-answer game $\Gamma_K(n,n)$ in this case is symmetric, each mixed strategy of the on-line player can be viewed as a mixed strategy of the adversary. As the balanced strategy is the optimal strategy for the adversary by Theorem 7, the uniformly mixed strategy (DA) is sub-optimal. The sub-optimality of DA shows the common heuristic of the Yao Principle mentioned in §2.2 may not lead to tight lower bounds in general. Figure 2 illustrates the ratio r_s^*/r_u with $1 < \theta \leq 2$ and $2 \leq n \leq 100$. Figure 3 shows the relationships among Υ_{DA}, r_s^* and r_u in financial market with $\theta = 1.07$ and $2 \leq n \leq 100$.

3.6 Uniqueness of the Optimal Strategy

Based on the correspondence between the finite request-answer game and the primal and dual problems, we can prove the uniqueness of optimal strategy. (The uniqueness of optimal strategies does not always hold for all finite zero-sum two-person games.) To prove this assertion, we need a few lemmas.

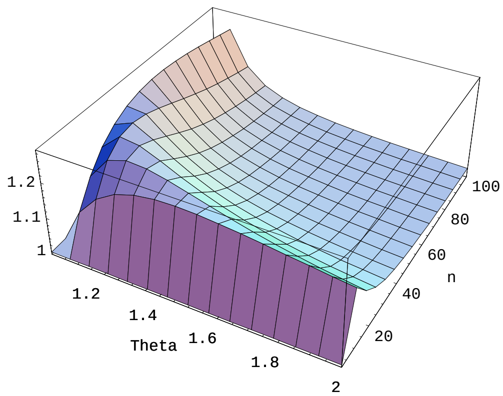

Figure 2: Plot of r_s^*/r_u.

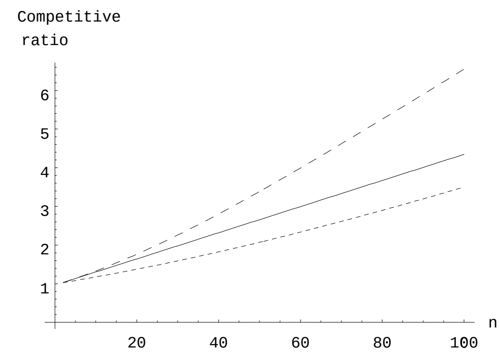

Figure 3: The dashed, solid and dotted lines denote the values of Υ_{DA}, r_s^*, and r_u, respectively.

Lemma 8 *For the finite request-answer game $\Gamma_K(n,n)$, denote each mixed strategy as a point in the Euclidean space R^n. Let X^* and Y^* denote the set of optimal mixed strategies of the on-line player and the adversary, respectively. Then the sets X^* and Y^* are convex polyhedron. Moreover, the balanced strategy is an extreme point of Y^*.*

Proof. That X^* and Y^* are convex polyhedron is guaranteed by [PZ96, Theorem 1.7.4]. We will prove the second assertion with linear programming.

Since $\mathbf{K}$ is symmetric, by Lemma 1, the problem of solving the finite request-answer game $\Gamma_K(m,n)$ in Lemma 5 is equivalent to the following primal and dual problems

Primal:		Dual:	
minimize	$\mathbf{u}_n^T \mathbf{x}$	maximize	$\mathbf{y}^T \mathbf{u}_n$
subject to	$\mathbf{K}\,\mathbf{x} \geq \mathbf{u}_n$	subject to	$\mathbf{y}^T\,\mathbf{K} \leq \mathbf{u}_n^T$
	$\mathbf{x} \geq 0$		$\mathbf{y} \geq 0$

Recall that $v^* = 1/r_s^*$ is the value of the game $\Gamma_K(n,n)$. Let $\bar{X}$ and $\bar{Y}$ denote the set of optimal feasible solutions to the primal and dual problems, respectively. We have $X^* = v^*\,\bar{X}$ and $Y^* = v^*\,\bar{Y}$ by Lemma 1.

Let $\mathbf{b}^* \equiv [\,b_1^*, b_2^*, \ldots, b_n^*\,]^T$ and $\bar{\mathbf{b}} \equiv r_s^*\,\mathbf{b}^*$, where b_i^* is defined in Theorem 7. Since the payoff matrix $\mathbf{K}$ is symmetric, the balanced strategy $\mathbf{b}^*$ is also an optimal strategy of the adversary. Moreover $\bar{\mathbf{b}} = \mathbf{K}^{-1}\,\mathbf{u}_n$. Thus $\bar{\mathbf{b}}$ is an extreme point of $\bar{Y}$, and $\mathbf{b}^*$, the balanced strategy, is hence also an extreme point of Y^*. □

For our finite request-answer game $\Gamma_K(n,n)$, we need a useful theorem for the optimal strategy as an extreme point due to Shapley and Snow [Rag94, p. 742].

Lemma 9 *The claim that x^* and y^* are extreme points of X^* and Y^*, respectively, holds if and only if there is a square submatrix $\mathbf{K}' = (a_{ij})_{i\in I, j\in J}$ of $\mathbf{K}$ such that*

1. *$\mathbf{K}'$ is nonsingular.*
2. *$\sum_{i\in I} a_{ij} x_i^* = v^*, \quad j \in J \subseteq Z_n$.*
3. *$\sum_{j\in J} a_{ij} y_j^* = v^*, \quad i \in I \subseteq Z_n$.*
4. *$x_i^* = 0$ if $i \notin I$.*
5. *$y_j^* = 0$ if $j \notin J$.*

Finally, we prove that the balanced algorithm is the unique optimal static strategy for the static buy-and-hold trading problem.

Theorem 10 *A static trading algorithm that is not balanced cannot be optimal for the class of static trading algorithms.*

Proof. We continue to use the notations defined in Lemma 8. Since the convex polyhedron X^* has only a finite number of extreme points, any optimal mixed strategy $x^* \in X^*$ must be a finite convex combination of these extreme points. Therefore it is sufficient to show that these extreme points of X^* are all equal to $\mathbf{b}^*$.

Let x^* be any extreme point of X^*. By Lemma 8, $\mathbf{b}^*$ is an extreme point of Y^*. Then by Lemma 9, there is a square submatrix $\mathbf{K}' = (a_{ij})_{i\in I, j\in J}$ of $\mathbf{K}$ such that the conditions of Lemma 9 hold with $y^* = \mathbf{b}^*$. Since each $b_j^* > 0$, for $j \in Z_n$, Condition 5 there implies $J = Z_n$. By Condition 2, we have $\sum_{i\in I} a_{ij} x_i^* = v^*$, for $j \in Z_n$. With Condition 4, we have

$$\sum_{i=1}^{n} a_{ij} x_i^* = v^* \quad \text{for } j \in Z_n. \tag{11}$$

Note that the system of equations in (11) is precisely (8) because $v^* = 1/r_s^*$. Therefore we have $\mathbf{K}\,\mathbf{x}^* = (1/r_s^*)\,\mathbf{u}_n$. Then $\mathbf{x}^* = (1/r_s^*)\,\mathbf{K}^{-1}\,\mathbf{u}_n = \mathbf{b}^*$. □

4 Does Real-Time Information Help Static Strategies?

In §3 we designed the balanced strategy and proved its optimality, uniqueness and balance property. The balanced strategy is a static strategy that ignores all real-time information. It is an important issue whether real-time information can be used to improve upon static strategies. In other words, we ask if it is better for the on-line player to sequentially optimalize the investment on each daily rate.

4.1 Sequentially Optimized Strategy

At the kth day after the exchange rate is revealed, the on-line player can formulate the remaining trading problem as a new finite request-answer game and solve for the optimal strategy by treating the remaining problem as a new problem of length $n-k+1$. The on-line player then invests according to this new strategy for this day. The above steps are repeated at each day. We call this strategy SOS (for *Sequentially Optimized Strategy*). Clearly SOS is a dynamic strategy.

We now describe SOS in more details. Consider the kth day, $1 \leq k < n$, with the fluctuation ratios up to then being $\mathbf{p} \equiv \langle d_1, d_2, \ldots, d_k \rangle$. Define $O_k \equiv \max\{ e_1, e_2, \ldots, e_k \}$, where $e_i \equiv \prod_{1 \leq l \leq i} d_l$, $i \in Z_k$, and Y_{k-1} and D_{k-1} to be the yen and dollar amounts at the beginning of the kth day, respectively. Note that $Y_0 = 0$ and $D_0 = 1$. These parameters completely specify the trading problem for the remaining m transactions, where $m \equiv n-k+1$.

Let $\Sigma(m, \mathbf{p})$ be the trading problem after the on-line player has traded along $\mathbf{p}$ according to SOS for the first $k-1$ days, dynamically invests at the kth day, and is static for the rest. The on-line player formulates $\Sigma(m, \mathbf{p})$ as a finite request-answer game and solves for its optimal strategy as follows. Let $\mathbf{d}' = \langle d_{k+1}, \ldots, d_n \rangle$, in which $d_{k+l} \in \Delta$, where $\Delta \equiv [\theta^{-1}, \theta]$, for $l \in Z_{n-k}$, denotes the future fluctuation ratio sequence. Consider the static strategy initially with Y_{k-1} yen and D_{k-1} dollars. Thus the performance ratio of the trade-once algorithm S_{k+i}, $i = 0, 1, \ldots, n-k$, to OPT on $\mathbf{d}'$ is

$$\frac{\mathrm{S}_{k+i}(\mathbf{d}')}{\mathrm{OPT}(\mathbf{d}')} = \frac{Y_{k-1} + (D_{k-1} e_k) \prod_{1 \leq l \leq i} d_{k+l}}{\max\{O_k, e_k \max_{1 \leq j \leq n-k} \prod_{l=1}^{j} d_{k+l}\}}. \tag{12}$$

Notice that O_k, e_k, Y_{k-1}, and D_{k-1} in (12) are all known to the on-line player. With the same reason as in §3.1, each worst case for the considered strategy is dominated by one of the downturns $\mathbf{d}'_j = \langle d_{k+1}, \ldots, d_n \rangle$, $j = 0, 1, \ldots, n-k$, in which

$$d_{k+l} = \begin{cases} \theta & \text{if } l \leq j \\ \theta^{-1} & \text{if } l > j, \end{cases} \tag{13}$$

where $l \in Z_{n-k}$. Strategy $\mathrm{F}_{\mathbf{p}}$ for the on-line player is then picked from one of the optimal strategies given by the finite request-answer game $\Gamma_{K'}(m, m)$ with the payoff function $K'(i,j)$, $i, j \in Z_m$, defined by

$$K'(i,j) \equiv \frac{\mathrm{S}_{k+i-1}(\mathbf{d}'_{j-1})}{\mathrm{OPT}(\mathbf{d}'_{j-1})}. \tag{14}$$

By Lemma 1 and Corollary 2, the on-line player can obtain $\mathrm{F}_{\mathbf{p}}$ by solving its corresponding primal problem. The on-line player thus invests a certain amount of dollars at the kth day according to $\mathrm{F}_{\mathbf{p}}$. Let SOS_k denote the amount of dollars invested along $\mathbf{p}$ at the kth day according to $\mathrm{F}_{\mathbf{p}}$. Thus SOS_k is a function from Δ^k to $[0,1]$. Denote $\mathrm{SOS} \equiv \langle \mathrm{SOS}_1, \ldots, \mathrm{SOS}_n \rangle$. Thus $\sum_{k=1}^{n} \mathrm{SOS}_k = 1$. We summarize the above in the next algorithm.

Algorithm 11 SOS *is a dynamic strategy that sequentially optimizes its investment by solving the finite request-answer game as defined in (14) at each day.*

4.2 Properties of the Sequentially Optimized Strategy

Let $\mathcal{D}$ be the set of all admissible fluctuation ratio sequences, and $\mathcal{D}_{\mathbf{p}}$ be the set of all $\mathbf{d} \in \mathcal{D}$ having the prefix segment $\mathbf{p}$. Let $\mathcal{F}_k$, where $k \in Z_n$, denote the set of all strategies that are admissible in $\Sigma(m, \mathbf{p})$ for all $\mathbf{p} \in \Delta^k$, and let r_k be the optimal competitive ratio for the strategies in $\mathcal{F}_k$. Then we have

$$r_k \equiv \inf_{\mathrm{F} \in \mathcal{F}_k} \sup_{\mathbf{d} \in \mathcal{D}} \frac{\mathrm{OPT}(\mathbf{d})}{\mathrm{F}(\mathbf{d})}.$$

Lemma 12 *For $k \in Z_n$, we have*

$$r_k = \sup_{\mathbf{p} \in \Delta^k} \inf_{\mathrm{F} \in \mathcal{F}_k} \sup_{\mathbf{d} \in \mathcal{D}_{\mathbf{p}}} \frac{\mathrm{OPT}(\mathbf{d})}{\mathrm{F}(\mathbf{d})}, \tag{15}$$

and there exists an optimal strategy $\mathrm{F}_k \in \mathcal{F}_k$ such that F_k achieves the optimal competitive ratio r_k.

Proof. Let r'_k denote the right-hand side in (15). It is clear that $r_k \geq r'_k$.

Given $\mathbf{p} \in \Delta^k$. Thus each $\mathrm{F} \in \mathcal{F}_k$ is static on $\mathcal{D}_{\mathbf{p}}$. Let $\mathrm{F}_{\mathbf{p}}$ be the static strategy solved in §4.1. Thus $\mathrm{F}_{\mathbf{p}}$ achieves the infimum $\inf_{\mathrm{F} \in \mathcal{F}_k} \sup_{\mathbf{d} \in \mathcal{D}_{\mathbf{p}}} \frac{\mathrm{OPT}(\mathbf{d})}{\mathrm{F}(\mathbf{d})}$. Note that $\mathrm{F}_{\mathbf{p}}$ is defined only on $\mathbf{d} \in \mathcal{D}_{\mathbf{p}}$. Let F_k be the aggregate of those $\mathrm{F}_{\mathbf{p}}$ for all $\mathbf{p} \in \Delta^k$. Thus we have

$F_k(\mathbf{d}) = F_{\mathbf{p}}(\mathbf{d})$ for all $\mathbf{d} \in \mathcal{D}_{\mathbf{p}}$ and $\mathbf{p} \in \Delta^k$. Therefore, $F_k \in \mathcal{F}_k$. By the definition of r'_k and r_k, we have

$$r'_k = \sup_{\mathbf{p}\in\Delta^k} \sup_{\mathbf{d}\in\mathcal{D}_{\mathbf{p}}} \frac{\mathrm{OPT}(\mathbf{d})}{F_{\mathbf{p}}(\mathbf{d})} = \sup_{\mathbf{d}\in\mathcal{D}} \frac{\mathrm{OPT}(\mathbf{d})}{F_k(\mathbf{d})} \geq r_k.$$

Thus F_k is the required strategy. □

Note that SOS invests also according to F_k at the kth day. Thus SOS coincides with F_k for the first k days.

Theorem 13 $r_0 \geq r_1 \geq r_2 \cdots \geq r_{n-1} = r_n = \Upsilon_{\mathrm{SOS}}$, *where* r_0 *and* Υ_{SOS} *denote the competitive ratio of the balanced strategy and* SOS, *respectively.*

Proof. Given $k \in Z_n$. Let F_0 denote the balanced strategy. By the definition, F_{k-1} coincides with SOS for the first $k-1$ days and is static for the rest. Thus $F_{k-1} \in \mathcal{F}_k$. Since $r_{k-1} = \Upsilon_{F_{k-1}}$, we have $r_{k-1} \geq r_k$.

Notice that SOS coincides with F_{n-1} at the first $n-1$ days, and the on-line player has no choices at the last day; thus $F_{n-1} = F_n = \mathrm{SOS}$ and $r_{n-1} = r_n = \Upsilon_{\mathrm{SOS}}$. □

Given $k \in Z_{n-1}$ and $\mathbf{p} \in \Delta^k$, we say an admissible strategy $F_{\mathbf{p}}$ for $\Sigma(m, \mathbf{p})$ is *nice* if it possesses the optimality, uniqueness and balance properties (meaning $F_{\mathbf{p}}$ is the unique optimal strategy for the reduced finite request-answer game defined in (14) and $F_{\mathbf{p}}$ balances the downturns defined in (13)). Note that the balance condition is the same as the optimal solution to the corresponding primal problem defined in Lemma 1 equalizing those constraints.

Let $\mathbf{d}_j$, for $j \in Z_n$, denote the fluctuation ratio sequence corresponding to the downturn $\mathbf{e}_j$ defined in Lemma 4. For the downturn $\mathbf{d}_n$, we have an interesting property in the following.

Property 14 *When the adversary sequentially reveals the exchange rates along the downturn* $\mathbf{d}_n$, *at each day* SOS *is nice and coincides with the balanced strategy.*

The proof of Property 14 is omitted.

Note that along the downturn $\mathbf{d}_n$ the balanced strategy sequentially balances its return ratio on downturns $\{\mathbf{d}_k, \mathbf{d}_{k+1}, \ldots, \mathbf{d}_n\}$. Since $\mathbf{d}_n$ is a worst case for the balanced strategy and it coincides with SOS on $\mathbf{d}_n$, we conclude $\Upsilon_{\mathrm{SOS}} \geq r_0$. Combine this with Theorem 13, we have the next theorem.

Theorem 15 *The competitive ratio of* SOS *equals that of the balanced strategy.*

Property 14 and Theorem 15 seem to suggest that real-time information does not help improve the static strategies on the worst cases. In the following, however, we will illustrate a non-worst-case scenario for which SOS is strictly better than the balanced strategy.

4.3 Performance Comparison Between the Balanced Strategy and the Sequentially Optimized Strategy on a non-worst-case scenario

We illustrate the sequential optimization process with an example. Assume $(n, k) = (6, 2)$ and the revealed daily fluctuation ratios are $\mathbf{p} = \langle \theta, 1 \rangle$. Thus $Y_1 = \frac{\theta^2}{n\theta-(n-2)}$ and $D_1 = 1 - \frac{\theta}{n\theta-(n-2)}$. We formulate the remaining trading problem as a request-answer game $\Gamma_{K'}(5,5)$, where the payoff function is defined as

$$K'(i,j) = \frac{Y_1}{\theta j} + D_1 \theta^{-|i-j|} \text{ for } i, j \in Z_5.$$

By Lemma 1 and Corollary 2, we solve $\Gamma_{K'}(5,5)$ and obtain the optimal competitive ratio $\Upsilon'_{\mathrm{SOS}} \equiv \frac{5\theta-3}{\theta+1}$ and the optimal strategy $\mathrm{SOS} \equiv [f_1, \ldots, f_5]^T$ for $\Sigma(5, \mathbf{p})$ where

$$f_i = \begin{cases} \frac{\theta(\theta-1)}{(5\theta-3)(5\theta-4)} & i = 1 \\ \frac{2(\theta-1)(3\theta-2)}{(5\theta-3)(5\theta-4)} & i = 2,3,4 \\ \frac{2\theta(3\theta-2)}{(5\theta-3)(5\theta-4)} & i = 5 \end{cases}$$

In comparison, the competitive ratio Υ'_{BAL} of the balanced strategy for $\Sigma(5, \mathbf{p})$ is

$$\begin{aligned} \Upsilon'_{\mathrm{BAL}} &= \max_{j\in Z_5} \left(\mathbf{b}'^T \mathbf{K}'^j\right)^{-1} \\ &= \frac{2\theta^4(3\theta-2)}{\theta^5+\theta^4+\theta-1}, \end{aligned}$$

where $\mathbf{b}' \equiv [b'_1, \ldots, b'_5]^T$ and

$$b'_i = \begin{cases} \frac{\theta-1}{5\theta-4} & i = 1,2,3,4 \\ \frac{\theta}{5\theta-4} & i = 5 \end{cases}$$

Clearly, the dynamic strategy SOS is better than the balanced strategy B*. Figure 4 illustrates the competitive ratios of the balanced strategy and SOS.

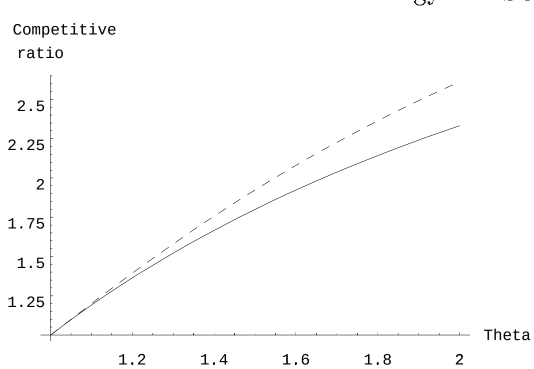

Figure 4: Performance improvement over the balanced strategy. The dashed and solid lines denote the values of Υ'_{BAL} and Υ'_{SOS}, respectively.

5 Conclusions and Future Work

In this paper, we presented a general solution for the problem of deriving optimal randomized on-line algorithms against the oblivious adversary. It is applicable to any finite request-answer games. We also successfully made use of this general strategy to derive an optimal static buy-and-hold strategy and its competitive ratio. Two future directions are possible from here. The first is to characterize the payoff matrix of finite request-answer game and systematically solve or approximate the optimal strategy and its competitive ratio. The second is to solve an infinite request-answer game with game theory.

References

[BDBK+94] S. Ben-David, A. Borodin, R. M. Karp, G. Tardos, and A. Wigderson. On the power of randomization in on-line algorithms. *Algorithmica*, 11:2–14, 1994.

[BEY97] A. Borodin and R. El-Yaniv. On randomization in online computations. In *Proc. of the Twelfth Annual IEEE Conference on Computational Complexity*, pages 226–238, 1997.

[BEY98] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis.* Cambridge: Cambridge University Press, 1998.

[CCEY+95] A. Chou, J. Cooperstock, R. El-Yaniv, M. Klugerman, and T. Leighton. The statistical adversary allows optimal money-making trading strategies. In *Proc. of the 6th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1995.

[CZ96] K. P. Edwin Chong and Stanislaw H. Zak. *An Introduction to Optimization.* New York: John Wiley & Sons, 1996.

[EY98] R. El-Yaniv. Competitive solutions for on-line financial problems. *ACM Computing Surveys*, 30:28–69, 1998.

[Hoc97] D. S. Hochbaum. *Approximation Algorithms for NP-Hard Problems.* Boston: PWS Publishing, 1997.

[Hul97] J. C. Hull. *Options, Futures, and Other Derivatives.* Prentice-Hall, 3rd edition, 1997.

[Lyu99] Y. D. Lyuu. *Financial Engineering and Computation: Principles, Mathematics, Algorithms.* To be published, 1999.

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge: Cambridge University Press, 1995.

[PZ96] L. A. Petrosjan and N. A. Zenkevich. *Game Theory.* Singapore: World Scientific, 1996.

[Rag94] T. Raghavan. Zero-sum two-person games. In R. Aumann and S. Hart, editors, *Handbook of Game Theory*, volume 2, pages 735–759. Elsevier Science Publishers B. V., 1994.

[Yao77] A. C. Yao. Probabilistic computations: Towards a unified measure of complexity. In *Proc. of the 18th Annual Symposium on Foundations of Computer Science*, pages 222–227, 1977.