The Statistical Adversary Allows Optimal Money-Making Trading Strategies

EXTENDED ABSTRACT

Andrew Chou* Jeremy Cooperstock† Ran El–Yaniv‡ Michael Klugerman§ Tom Leighton¶

November, 1993

Abstract

The *distributional approach* and *competitive analysis* have traditionally been used for the design and analysis of on-line algorithms. The former assumes a specific distribution on inputs, while the latter assumes inputs are chosen by an unrestricted adversary. This paper employs the *statistical adversary* (recently proposed by Raghavan) to analyze and design on-line algorithms for *two-way currency trading*. The statistical adversary approach may be viewed as a hybrid of the distributional approach and competitive analysis. By statistical adversary, we mean an adversary that generates input sequences, where each sequence must satisfy certain general statistical properties. The on-line algorithms presented in this paper have some very attractive properties. For instance, the algorithms are *money-making*; they are guaranteed to be profitable when the optimal off-line algorithm is profitable. Previous on-line algorithms although "competitive", can lose money, even though the *optimal off-line* algorithm makes money. Against a weak statistical adversary, our methods yield an algorithm that outperforms the optimal off-line "buy-and-hold" strategy. Furthermore, it is guaranteed to make a substantial profit when it is known that the market is *active* and *stable* (i.e. there are fluctuations but the upward and downward fluctuations tend to balance each other). In fact, our algorithm even makes money when the market exhibits a slightly unfavorable trend.

*MIT, Supported by NSF Graduate Fellowship. achou@theory.lcs.mit.edu

†University of Toronto, jer@dgp.toronto.edu

‡University of Toronto, ran@theory.lcs.mit.edu

§MIT, This research was supported by the Defense Advanced Research Pro jects Agency under Contract N00014-91-J-1698. klugermn@theory.lcs.mit.edu

¶MIT, This research was supported by the Defense Advanced Research Projects Agency under Contracts N00014-91-J-1698 and N00014-92-J-1799, and the Air Force under Contract AFOSR-49620-92-J-0125. ftl@theory.lcs.mit.edu

1 Introduction

In many situations, we are forced to choose between different alternatives without knowledge of each alternative's future worth. Our choices must be made in an *on-line* manner. However, we often have partial information about the future. For example, we may know that with high probability, certain statistical properties are satisfied. When making choices to maximize our future gain, it makes sense to incorporate this information into these decisions. Similarly, on-line algorithms that solve these types of problems can gain significantly by including some knowledge of the future. In the areas of finance, economics, and operations research, we find examples of this kind of decision process[8, 16, 21]. In this paper, we examine the two-way currency trading problem against a *statistical adversary*[17].

1.1 Techniques for analyzing on-line algorithms

The analysis of on-line algorithms has typically involved either *distributional analysis* or *competitive analysis*. In the former approach, the input is assumed to conform to a "natural" or "typical" probability distribution. Based upon this distribution, one seeks strategies with good average case performance. In the latter approach, the input is generated by an adversary. In this case, one seeks to design on-line algorithms which compare favorably against the optimal off-line algorithm.

In practice, the distributional approach often does not reflect the nature of the input. In addition, even if the input in question follows a particular fixed (or stable) probability distribution, it is often difficult, to identify or construct a distributional model that accurately reflects the true distributions. For instance, a great deal of effort has been invested in an attempt to identify probability distributions of currency exchange rates. However, there is still no evidence that such a distribution exists. A wide variety of different opinions about the existence and/or nature of such distributions can be found in (e.g. [8, 9, 20, 15, 16, 18]).

The approach of competitive analysis first appeared in works on bin packing in the 1970's [2, 11, 12, 22], and then it was explicitly formulated in the 1980's [5, 7, 13, 19]. The idea of competitive analysis is to consider input sequences that are generated by an adversary and to measure performance with respect to the *optimal off-line* algorithm using the same input sequence. Under this model, one avoids making assumptions required by the distributional approach. Instead, the assumption is made that the input will be strictly adversarial. Specifically, the input sequence is chosen to minimize the algorithm's overall performance relative to the optimal off-line algorithm. Few restrictions are typically placed on the adversary other than possible limitations on the adversary's knowledge of the on-line algorithm's actions (e.g. important for randomized algorithms). Such a powerful adversary often does not reflect the nature of the input to many practical problems.

Because the input to most problems lies somewhere between the pessimistic approach of competitive analysis and the more optimistic distributional approach, a number of other approaches were introducted [4, 6, 14, 17, 23]. In this paper we focus on Raghavan's *statistical adversary* approach[17]. Here, the underlying idea is to limit the power of the adversary in some way dependent on the particular problem. Namely, the adversary is required to generate input sequences satisfying certain (statistical) properties. For example, the adversary may be required to maintain certain bounds on the number of requests of a certain type, or to produce input sequences of which certain subsequences must satisfy particular constraints. The premise of this approach is that input sequences arising in reality exhibit and conform to certain long-term statistics. The idea behind this approach is to eliminate the possibility of extremely bad input sequences, which do not occur frequently in reality. The hope is to show that one can place limited and realistic restrictions on an adversary which allows the on-line algorithm to perform well.

An important issue regarding the statistical adversary is that of whether the on-line player should be allowed to make use of the statistical parameters associated with the adversary. On one hand, it would be more elegant to seek on-line algorithms that perform optimally (or well) for all possible choices of the parameters and use these parameters only for the analysis (i.e. to express the performance in terms of these parameters). On the other hand, we may allow our algorithms to use these parameters for the purpose of obtaining better performance. We refer to algorithms of the former type as *universal* algorithms, and algorithms of the latter type as *non-universal* algorithms. Certainly, universal algorithms are more desirable because they do not assume knowledge of the parameters. However, in the universal approach, the performance of two different algorithms are likely to depend differently upon the parameters. In order to decide which algorithm to use, one needs to judge, perhaps by conducting statistical tests, which algorithm will perform better under the particular circumstances. By doing so, one is transforming the two universal algorithms into a single non-universal algorithm. In addition, since a non-universal algorithm is tuned to the parameters, it can be designed to perform significantly better than a corresponding universal algorithm.

In this paper we use the framework of the statistical adversary to analyze the *two-way trading* game that is discussed in [10]. Specifically, we consider a discrete variant of this problem in which the on-line player begins with some money, say dollars, and is given an opportunity to invest in another currency, say yen, for some period of time. The player would like to maximize his returns during this time by taking advantage of fluctuations in the exchange rates by converting back and forth between dollars and yen. The player assumes that a statistical adversary is controlling the change in exchange rates. We analyze this problem for adversaries with both weak and strong restrictions.

This problem was first analyzed using traditional competitive analysis in [10]. The authors present a competitive on-line strategy. However, though this strategy is competitive, it is possible to experience losses although the off-line strategy might be profitable.

1.2 Summary of Results

We begin by considering a statistical adversary that is forced to generate sequences of exchange rates of a known length, say n, such that the optimal off-line return on these sequences is larger than a known quantity, Π. Such a constraint, defined by the pair, (n, Π), is a valid statistical feature since it can be learned from historical sequences of exchange rates. Note that the adversary is completely unrestricted in his choice of the number and size of unfavorable transactions.

- We present a general scheme that identifies optimal on-line strategies against statistical adversaries constrained by such (and similar) features. Against each particular adversary, this scheme easily yields the optimal on-line strategy in the form of a dynamic program in terms of the statistical parameters. For usage, one can then efficiently pre-compute the on-line algorithm for arbitrary choices of these parameters.

- We identify the optimal on-line strategy relative to the above adversary and show that it is *money-making* as long as $\Pi > 1$. Unfortunately, it is shown that this strategy can only guarantee a very small fraction of the optimal off-line profit.

- Next we consider stronger adversaries that do not provide the player with either the value of n or the value of Π. Against these adversaries, we show that it is not possible to design a money-making strategy.

- In contrast, we then consider a weaker adversary in which the player has knowledge of the overall movement of the exchange rates during the n time periods and the factor $\alpha > 1$ by which the exchange rate changes (either up or down) during each time period. We identify and analyze the corresponding optimal on-line strategy. This strategy exhibits some striking properties.

  - Aside from being money-making, this strategy always outperforms the optimal off-line *buy-and-hold strategy* whereby during the initial transaction period, the *buy-and-hold strategy* either converts all dollars to yen or does not invest at all.
  - When the market is *stable* and *active* (i.e. there are fluctuations but the upward and downward fluctuations tend to balance each other with respect to the trading period), this strategy yields exponential profits (in n), even if the market has a slight *unfavorable* trend. This is somewhat surprising. Intuitively, it is reasonable to believe that one should avoid any financial transactions during stable periods (in comparison, the buy-and-hold strategy will not make any profit in this situation).

  Based on preliminary experimental results, it appears that this weak adversarial model may provide a practical approach to investing strategies.

The rest of this paper is organized as follows. In § 2 we formally define the problem. In § 3 we identify the optimal money-making strategy when the adversary supplies the player with (n, Π) and then derive some of its properties. In § 4 we show that by strengthening the adversary we eliminate the possibility of a money-making strategy. Conversely, in § 5 we identify the optimal strategy against a considerably weaker adversary. This strategy can obtain exponential return, even when the market exhibits a slightly unfavorable trend.

2 Definitions and notation

In this paper we follow the statistical adversary framework to analyze the *two-way trading* game that is also discussed in [10]. Specifically, we consider a discrete variant of this problem in which the on-line player is given D_0 dollars, and is given the job of maximizing his return over a time period of n days by exchanging the money back and forth to/from yen.With these two currencies, there is an associated exchange rate sequence $E = e_1, e_2, \ldots$, where e_i, the exchange rate for the ith day, equals the number of yen that can be purchased for one dollar on that day. The player is required to finish the game with all the money converted back to the initial currency. We assume that there are no transaction fees. Our new assumption is that the exchange rate sequences are generated by a statistical adversary.

For any strategy S and finite exchange rate sequence E, let $R_S(E)$ denote the *return* of S with the sequence E when it begins the game with D_0 dollars. Let OPT denote the optimal off-line two-way trading strategy. Notice that for a given exchange rate sequence, E, OPT will always convert all available dollars to yen at all exchange rates which are local maxima in E, and all available yen back to dollars at all exchange rates which are local minima, with the exception that the last transaction must be yen to dollars. We say that a trading strategy, S, is a *money-making* strategy if for any exchange rate sequence E with $R_{OPT}(E) > 1$, then $R_S(E) > 1$, as well.

3 Optimal trading strategies against the statistical adversary

We can specify trading strategies in a "normal form" as follows. Every day, the player might wish to convert some dollars to yen and/or some yen to dollars. It should be clear that only one transaction (i.e. dollars to yen or yen to dollars) is sufficient. It is convenient to conceptualize every such transaction as if the strategy first converted *all* yen back to dollars and then converted a fraction, s, of the total available dollars to yen. Any such transaction can be specified by one number, $s \in [0, 1]$. In this way, we can specify the activity of any conversion strategy by the sequence $s_1, s_2, \ldots, s_n$ where for each day, i, s_i is the fraction of dollars that should be converted to yen, immediately after all available yen are converted to dollars. By the rules of our game, s_n must be zero.

Suppose that the on-line player knows Π, the return of the optimal off-line player for an n-day game. We now derive the optimal on-line strategy for any $n \geq 2$ number of days. To derive the optimal strategy we require the following observation. Consider Figure 1, which illustrates an exchange rate sequence of 10 days (yen per dollar). First, notice that the optimal (off-line) return is

$$\Pi = \left(\frac{e_2}{e_5}\right)\left(\frac{e_6}{e_7}\right)\left(\frac{e_8}{e_9}\right) = \left(\frac{e_2 e_3 e_4}{e_3 e_4 e_5}\right)\left(\frac{e_6}{e_7}\right)\left(\frac{e_8}{e_9}\right)$$

In general, it can be shown that for any n-day sequence,

$$\Pi = \prod_{i=1}^{n-1} \max\{1, e_i/e_{i+1}\}.$$

Any conversion strategy "realizes" its dollar profit only on downward runs of the exchange rate

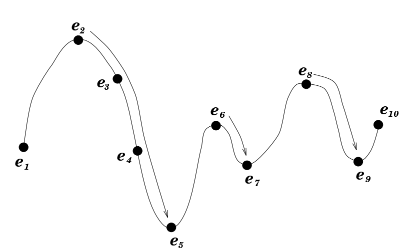

Figure 1: Realization of profit in a 10-day game

sequence. If Π is known at the beginning of the sequence, then on each day, the on-line player can determine exactly what the optimal off-line profit would be for an $(n-1)$-day sequence starting on that day – that is, if the off-line player were to start a "new game" consisting of $n-1$ days. We illustrate this by referring to Figure 1. Since on day 2 we are not in a downward run with respect to the previous day, the optimal off-line return must remain at Π if a new game is to begin at that moment. On the other hand, on day 3 (knowing only the first three rates) we know that the off-line player has just realized a factor e_2/e_3 of his total dollar return, so the optimal off-line return must be $\Pi' = \Pi/\frac{e_2}{e_3}$ if a new game is to be begun at that moment. In this way, the on-line player knows, after each exchange rate is revealed, exactly what the optimal off-line return would be if a new game were to be started that day.

This observation enables a dynamic programming derivation of the optimal on-line strategy using the Principle of Optimality [3]. In our context it can be stated as follows: the optimal on-line strategy has the property that, whatever the initial rate and the initial choice of how many dollars to trade, the remaining trades must constitute an optimal on-line strategy with regard to the state resulting from the first trade. More formally, for $n \geq 2$, let $R_n(\Pi, e_1)$ denote the return of the optimal on-line strategy for an n-day game, given that the first exchange rate is e_1 and that the optimal off-line return for the entire period is Π. Therefore, when exchange rates are chosen by an adversary who tries to minimize the optimal on-line return, we have

$$\begin{aligned} R_n(\Pi, e_1) &= \max_{\substack{\text{fraction of dol-}\\ \text{lars } s_1 \text{ to trade}}} \ \min_{\substack{\text{2nd day}\\ \text{rate } e_2}} \left[\begin{array}{c} \text{total worth in dol-}\\ \text{lars after 2nd day}\\ \text{rate is revealed} \end{array} \right] \cdot R_{n-1}\left(\text{updated}\,\Pi, e_2\right) \\ &= \max_{0 \leq s_1 \leq 1} \ \min_{e_2 \geq \frac{e_1}{\Pi}} \left[\left(\frac{e_1}{e_2} - 1 \right) s_1 + 1 \right] \cdot R_{n-1}\left(\min\left\{ \Pi, \frac{\Pi e_2}{e_1} \right\}, e_2 \right) \qquad (1) \end{aligned}$$

where the lower bound, $\frac{e_1}{\Pi}$, on possible second day rates, e_2, is due to the assumption that the total off-line return is Π. In addition, a boundary condition can easily obtained; namely,

$$R_2(\Pi, \cdot) = \Pi. \qquad (2)$$

Thus, (1) and (2) identify the optimal on-line strategy, which we denote by S^*. The first transaction that S^* performs is the purchase of yen with a fraction s_1^* of its dollars, where s_1^* is the quantity which maximizes the right hand side of (1). The proof of the following lemma appears in the appendix.

Lemma 1 *S^* is a money-making strategy.*

The task of obtaining a closed form expression for (1) seems to be rather hard. The following lemma provides an upper bound on the return of S^* for any Π and n. The proof of the lemma appears in the appendix.

Lemma 2

$$R_n(\Pi, \cdot) \leq \frac{1}{1 - (1 - \frac{1}{\Pi})^{n-1}}$$

The interpretation of Lemma 2 is rather pessimistic. It can be shown that for large Π,

$$\begin{aligned} \left(1 - \frac{1}{\Pi}\right)^{n-1} &= \left(1 - \frac{1}{\Pi}\right)^{\Pi \frac{n-1}{\Pi}} \approx e^{-\frac{n-1}{\Pi}} \\ R_n(\Pi, \cdot) &\approx \frac{1}{1 - e^{-\frac{n-1}{\Pi}}} \end{aligned}$$

Also, if $\Pi = \omega(n)$, then $e^{-\frac{n-1}{\Pi}} \approx 1 - \frac{n-1}{\Pi}$, and $R_n(\Pi, \cdot) \approx \frac{\Pi}{n-1}$. If $\Pi = \Theta(n)$, then $R_n(\Pi, \cdot) = \frac{1}{1-e^c}$. For $\Pi = o(n)$, $R_n(\Pi, \cdot)$ approaches 1.

4 Games against stronger adversaries

One can think of several meaningful ways to strengthen the original adversary. Here we consider two stronger adversaries which correspond to the cases where the on-line player does not know Π or does not know n *a priori*. In either case we prove the nonexistence of a money-making strategy for non-degenerate strategies. A *non-degenerate strategy* is one that makes at least one non-zero transaction. The proof of the following lemmas appear in the appendix.

Lemma 3 *For any $n > 2$, and any non-degenerate deterministic on-line strategy S that only knows n in advance there is an exchange rate sequence, $E = e_1, e_2, \ldots, e_n$ for which $R_S(E) < 1$ and $R_{OPT}(E) > 1$, even if S also knows in advance that there is a positive off-line profit.*

Lemma 4 *For any $\Pi > 1$, and any non-degenerate deterministic on-line strategy S that only knows Π in advance, there exists an exchange rate sequence, $E = e_1, e_2, \ldots$ for which $R_S(E) < 1$ and $R_{OPT}(E) > 1$.*

5 Games against weaker adversaries

In this section we impose more constraints on the adversary. In each of the following examples, the constraints may be estimated from relevant past sequences using simple statistical analysis.

- *maximum daily fluctuation ratio:* a number $\alpha > 1$ such that for every day i, the next day's rate, e_{i+1}, is in $[e_i\alpha, e_i\alpha]$. Although we measure the time difference between two successive exchange rates by "days", these time differences may be of any size (seconds, minutes, etc.), and, in fact, they need not be of a fixed size.

- *minimum and maximum bounds on exchange rates:* numbers, m and M, such that all exchange rates are within the interval $[m, M]$.

- *maximum run length:* a number ρ such that there is no monotone increasing (decreasing) subsequence of consecutive exchange rates of length longer than ρ.

- *number of extrema points:* a number k such that the number of minima and maxima in the exchange rate sequence is k.

- *statistical functions of exchange rate sequences:* "standard" statistical functions like mean and standard deviation may be considered.

It is possible to incorporate any of the above constraints in (1) to yield an optimal on-line strategy against the corresponding, more constrained adversary. In each case, we have to replace the bounding interval for possible choices of e_2 which was originally $[\frac{e_1}{\Pi}, \infty)$. Intuitively, by including more constraints, we should obtain better performance. The appeal of this scheme is that the users of our strategies may choose their own set of statistical features and obtain optimal on-line performance against an adversary that reflects "financial nature" according to their own beliefs.

Using our scheme, we now derive and analyze the optimal strategy against a weak adversary that is restricted by a statistical feature which is, in a sense, a hybrid of the (n, Π) feature and the *maximum daily fluctuation ratio* as discussed above. To motivate the use of this new feature, let us first discuss the limitations of the previous model (in which the feature (n, Π) yielded the strategy S^*). In this model, the on-line player is forced to invest very little on most days, since the adversary can eliminate any day's investment by raising the rates arbitrarily high. By imposing additional features such as the maximum fluctuation ratio or the number of extrema points, we can reduce these kinds of unrealistic threats. It is possible to incorporate the (n, Π) feature and the maximum fluctuation ratio feature into a single feature as follows.

The parameters of this new statistical feature are (α, m, n) where α represents a fixed ratio between any two successive exchange rates, m denotes the number of downward changes, and n is the total number of changes. Since each downward change in the exchange rate corresponds to a realization of dollar profit, we know that for each exchange rate sequence conforming to (α, m, n), the optimal off-line profit is α^m. Notice that n in this statistical feature measures the total number of the α-changes whereas in the the (n, Π) feature, n is the length of the exchange rate sequence.

Of course, in real exchange rate sequences, successive exchange rates do not maintain fixed ratios. However, sometime after the exchange rate becomes some value v, the exchange rate will eventually change to a value greater than or equal to αv or a value less than or equal to $\frac{v}{\alpha}$. In either case, when this occurs, S^{**} moves into a new "day". Note that the days are no longer fixed measures of time, but instead change when the exchange rate changes by a desired amount.

The advantage of considering such a *fixed* change is twofold: first, it simplifies the analysis, and second, it is sometimes very useful from a practical point of view to filter out negligible transactions that correspond to miniscule changes. For instance, by choosing a sufficiently large α, one may "filter out" some of the effect of *spreads*[1]. Note that choosing α too large can decrease returns as well; S^{**} may ignore profitable fluctuations of size less than α.

We assume that the on-line player knows (α, m, n). The knowledge of this triplet is of significant value. In fact, for sequences conforming to (α, m, n), we expect the exchange rate to change at a rate of $\alpha^{2\frac{m}{n}-1}$. Hence, even knowledge of the ratio $\frac{m}{n}$ may be extremely valuable as it represents the trend during the period in question. Given the knowledge of a particular trend (either downward or upward) one can use standard techniques (via the use of *future contracts*) to guarantee the profit of the buy-and-hold strategy. Moreover, using standard hedging techniques (via *put* and *call* options) one does not need to know the direction of the trend and can guarantee the by-and-hold profit corresponding to one direction and hedge against any risk corresponding to the other direction. Hence, of particular interest is the case $m = \frac{1}{2}n$ in which exactly half of the changes are upward and half the changes are downward. If this is the case, we say that the exchange rate is *stable* and *active*.

Let $R_\alpha(m, n)$ be the optimal on-line return with parameters α, m, and n. When the on-line player invests s, his return is either $(\alpha s + 1 - s)R_\alpha(m-1, n-1)$ or $(\frac{s}{\alpha} + 1 - s)R_\alpha(m, n-1)$, which, respectively, correspond to a downward change and an upward change. The adversary will choose the minimum of these two values. Hence, the following recurrence identifies the optimal

[1] A *spread* is the difference between the *bid* and *ask* prices of a certain commodity and reflects "transaction fees".

on-line strategy which we call S^{**}.

$$\begin{aligned} R_\alpha(m,n) &= \max_{0\le s\le 1} \min \left\{ (\alpha s + 1 - s)\cdot R_\alpha(m-1,n-1),\ \ (\frac{s}{\alpha}+1-s)\cdot R_\alpha(m,n-1) \right\}, \quad (3) \\ R_\alpha(n,n) &= \alpha^n, \\ R_\alpha(0,n) &= 1 \end{aligned}$$

5.1 Some properties of S^{**}

We now derive some interesting properties of the strategy S^{**}. First, notice that the left operand of the "min" in (3) is increasing with s while the right operand is decreasing with s. Hence, the optimal strategy sets s so that $(\alpha s+1-s)\cdot R_\alpha(m-1,n-1) = (\frac{s}{\alpha}+1-s)\cdot R_\alpha(m,n-1)$. Solving for s,

$$s = \frac{R_\alpha(m,n-1) - R_\alpha(m-1,n-1)}{(\alpha-1)\cdot R_\alpha(m-1,n-1) - (\frac{1}{\alpha}-1)\cdot R_\alpha(m-1,n-1)}$$

Substituting for s, we obtain

$$\begin{aligned} R_\alpha(m,n) &= \left(\frac{(\alpha-1)\cdot(R_\alpha(m,n-1) - R_\alpha(m-1,n-1))}{(\alpha-1)\cdot R_\alpha(m-1,n-1) + \frac{\alpha-1}{\alpha}\cdot R_\alpha(m,n-1)} + 1 \right) \cdot R_\alpha(m-1,n-1) \\ &= \frac{\frac{\alpha+1}{\alpha} R_\alpha(m,n-1)\cdot R_\alpha(m-1,n-1)}{R_\alpha(m-1,n-1) + \frac{1}{\alpha} R_\alpha(m-1,n-1)} \end{aligned}$$

Setting $R_\alpha^{-1}(m,n) \stackrel{\text{def}}{=} \frac{1}{R_\alpha(m,n)}$ and inverting both sides,

$$R_\alpha^{-1}(m,n) = \frac{\alpha}{\alpha+1} R_\alpha^{-1}(m,n-1) + \frac{1}{\alpha+1} R_\alpha^{-1}(m-1,n-1) \quad (4)$$

Set $\beta \stackrel{\text{def}}{=} \frac{1}{\alpha+1}$, and let $B(k;n,p) \stackrel{\text{def}}{=} \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}$, the partial binomial sum. The following lemma (whose proof appears in the appendix), provides a solution to (4).

Lemma 5 $R_\alpha^{-1}(cn,n) = B(n(1-c)-1; n-1, 1-\beta) + \alpha^{n(1-2c)} B(cn-1; n-1, 1-\beta)$

Using the result of Lemma 5, the next lemma characterizes the performance of S^{**}. The proof of the following lemma appears in the appendix.

Lemma 6 *For $m = cn$ with $c \in (0,1)$, the following asymptotic relations hold.*

- *If $0 \le c \le \beta$, then $R_\alpha(cn,n) \to 1$.*
- *If $\beta < c \le \frac{1}{2}$, then $R_\alpha(cn,n) \to e^{\Omega(n)}$.*
- *If $\frac{1}{2} < c < 1-\beta$, then $R_\alpha(cn,n) \to \alpha^{n(2c-1)} e^{\Omega(n)}$*
- *If $1-\beta \le c \le 1$, then $R_\alpha(cn,n) \to \alpha^{n(2c-1)}$.*

The interpretation of Lemma 6 is quite surprising. Consider the behavior of the optimal buy-and-hold strategy. Buy-and-hold will invest all its capital when $c > \frac{1}{2}$. On the other hand, when $c \le \frac{1}{2}$ it will avoid any transaction. Hence, the return is 1 for $c \le \frac{1}{2}$, and $\alpha^{n(2c-1)}$ for $c > \frac{1}{2}$. In the case where $0 \le c \le \beta$ or $1-\beta \le c \le 1$, S^{**} asymptotically performs the same as buy-and-hold. However, for $\beta < c < 1-\beta$, S^{**} performs exponentially better. In particular, for $c = \frac{1}{2}$,

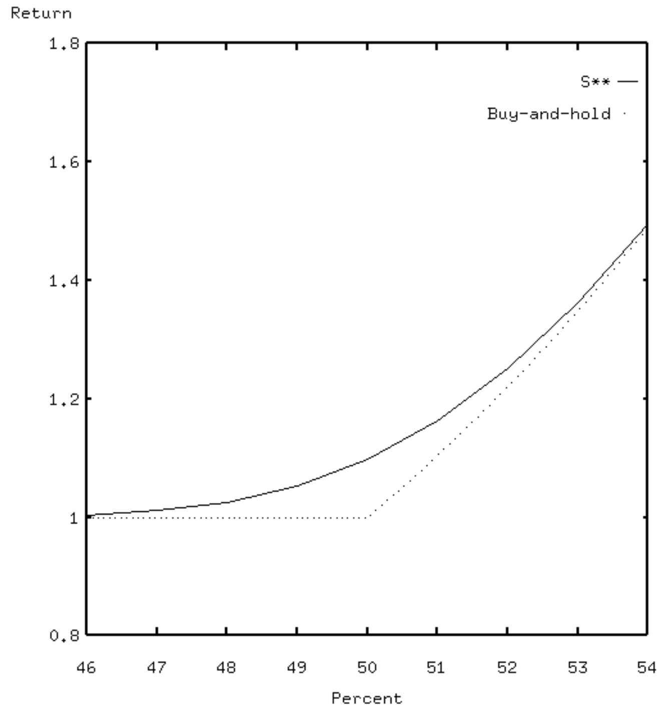

Figure 2: S^{**} vs. buy-and-hold as c varies ($\alpha = 1.01$ and $n = 500$)

buy-and-hold will return 1, while S^{**} yields exponential return. Note that in the case $\beta < c < \frac{1}{2}$, the market is moving unfavorably yet the return is exponential in n. The relative advantage of S^{**} over buy-and-hold is the greatest when the market is perfectly stable. This fact is illustrated in the graph of Figure 2.

5.2 Implementation and experimental notes

Before running S^{**} on real data, one must set the parameters n, m, and α. α may be chosen to capture "significant" changes in the exchange rate sequence. (e.g. one may choose sufficiently large α to filter out "insignificant" fluctuations). For a particular choice of n and α, the on-line player can chooses a value for m according to the player's beliefs, analysis (and risk aversion). In any case, it would be unrealistic to assume that one knows the exact value of m. Let m^* be the actual number of profitable changes among the n changes.

In Figure 3 and Figure 4 , we plot the return of S^{**} as a function of m. At the point where $m = m^*$, S^{**} obtains a maximum. On one hand, if S^{**} underestimates m^*, then S^{**} invests conservatively, since it "believes" that the number of remaining positive changes will be small. As m approaches zero, the return approaches 1, which is analogous to not trading at all. On the other hand, if S^{**} overestimates m^*, S^{**} invests more "aggressively" as it expects the exchange rate to be favorable. As m approaches n, the return approaches α^{2m^*-n}. This case is analogous to investing all the money on the first trading day and converting back on the last trading day (buy-and-hold). In both cases (overestimating and underestimating), we see exponential convergence to the limit cases.

The graph in Figure 3 illustrates the behavior when $m^* < \frac{1}{2}n$. In this case, the off-line buy-and-hold strategy does not invest and receives a return of 1. S^{**} always exceeds the buy-and-hold return when it underestimates the value of m^*. However, if S^{**} overestimates by too much, S^{**}

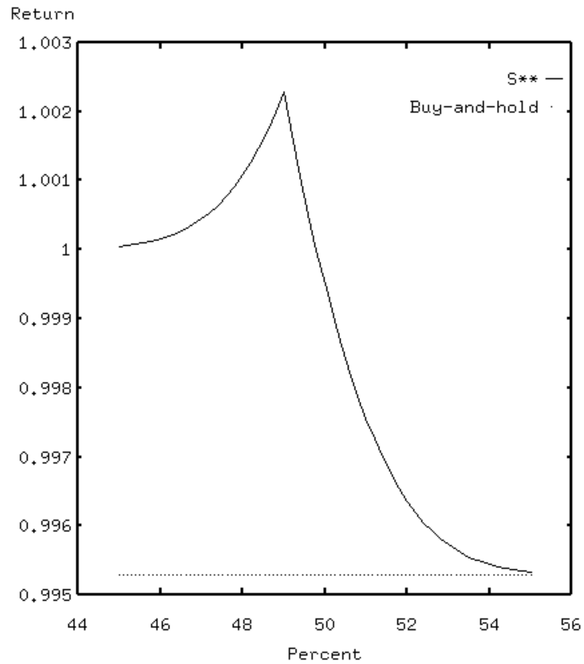

Figure 3: Returns of S^{**} as a function of m/n, $m^* < \frac{1}{2}$

may yield a return less than 1. Therefore, if S^{**} expects $m^* < \frac{1}{2}n$, then it is safe to underestimate.

A similar phenomenon is shown in Figure 4 ($m^* > \frac{1}{2}n$). The off-line buy-and-hold strategy will buy in the initial period and sell in the final period. Its return will be α^{2m^*-n}. If we incorrectly overestimate m^*, we will always exceed the buy-and-hold return. However, if we underestimate, then our return may be less than the off-line buy-and-hold return.

Based upon these graphs, it would appear that we need very accurate predictions to be successful. If we incorrectly estimate m^*, we can get returns that are worse than the off-line buy-and-hold. However, consider what it means for m^* to be different than m, where m is our estimate. Then, after n days, the exchange rate will differ from our expectation by a factor of $\alpha^{(m-m^*)}$. It is no surprise that if we experience an unanticipated expontential change in the exchange rate, then the algorithm (or any other algorithm) will perform poorly. Fortunately, actual exchange rate sequences rarely exhibit this behavior. In fact, the simple strategy where we assume $m = \frac{1}{2}n$ performs fairly well on small samples of real data.

S^{**} was tested on historical intra-day data for both US dollars vs. Japanese Yen and US dollars vs. German Marks. The intra-day data consisted of the reported exchange rate every 10-120 seconds. Decision points were then inserted every time the rate changed by α (We used values of α ranging from the minimum change to five times the minimum change). We then ran S^{**} on these exchange rate sequences and received very promising results. So far, the data appears to conform to our model. Although the exchange rate changes frequently, the overall trends have been quite stable. In a sample exchange rate sequence, the exchange rate changed by 5-10 points [2] a minute. But, the total daily change was usually less than 50 points. At this time, our experimental results are too preliminary to be statistically sound.

[2] A point is the smallest unit used to measure exchange rates.

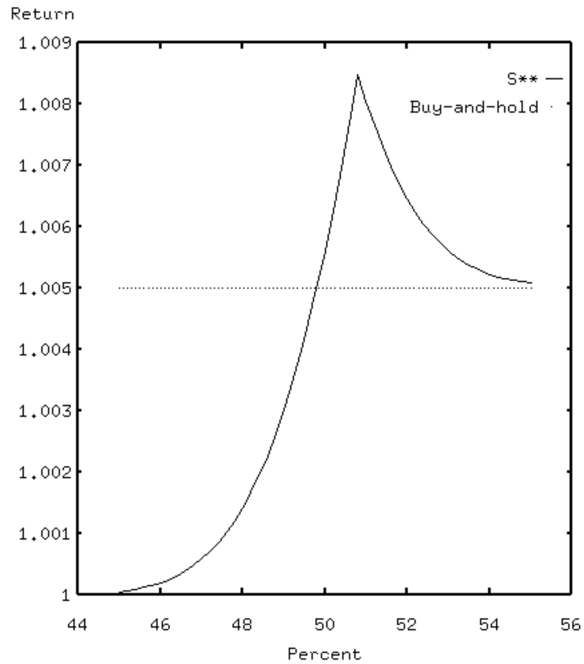

Figure 4: Returns of S^{**} as a function of m/n, $m^* > \frac{1}{2}$

References

[1] N. Alon and J. Spencer. *The Probabilistic Method.* John Wiley & Sons, Inc., 1991.

[2] D.S. Johnson amd A. Demers, J.D. Ullman, M.R. Garey, and R.L. Graham. Worst-case performance bounds for simple one-dimensional packing algorithms. *SIAM Journal on Computing*, 3:299–325, 1974.

[3] R. Bellman. *Dynamic Programming.* Princeton University Press, 1957.

[4] S. Ben-David and A. Borodin. A new measure for the study of online algorithms. *Algoritmica*, 11, January 1994.

[5] J.L. Bentley and C.C. Mcgeoch. Amortized analyses of self-organizing sequential search heuristics. *Communication of the ACM*, 28(4):404–411, 1985.

[6] A. Borodin, S. Irani, P. Raghavan, and B. Schieber. Competitive paging with locality of reference. In *23rd Annual ACM Symposium on Theory of Computing*, pages 249–259, 1991.

[7] A. Borodin, N. Linial, and M. Saks. An Optimal On-line Algorithm for Metrical Task System. *Journal of the ACM*, 39(4):745–763, 1992 1987.

[8] J.R. Calderon-Rossel and M. Ben-Horim. The behavior of the foreign exchange rates. *Journal of International Business Studies*, 13:99–111, 1982.

[9] F.X. Diebold. *Empirical modeling of exchange rate dynamics. Lecture notes in economic and mathematical systems*, volume 303. Springer, Berlin, 1988.

[10] R. El-Yaniv, A. Fiat, R.M. Karp, and G. Turpin. Competitive analysis of financial games. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, pages 327–333, 1992.

[11] R.L. Graham. Bounds for certain multiprocessing anomalies. *SIAM Journal on Computing*, 17:263–269, 1969.

[12] D.S. Johnson. *Near-Optimal Bin Packing Algorithms*. PhD thesis, Massachusetts Institute of Technology, 1973.

[13] A.R. Karlin, M.S. Manasse, L. Rudolph, and D.D. Sleator. Competitive snoopy caching. *Algorithmica*, 3(1):70–119, 1988.

[14] A.R. Karlin, S.J. Phillips, and P. Raghavan. Markov paging. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, pages 208–217, 1992.

[15] B.B. Mandelbrot and H.M. Taylor. On the distribution of stock price differences. *Operations Research*, 15:1057–1062, 1967.

[16] J.W. McFarland, R.R. Pettit, and S.K. Sung. The distribution of foreign exchange price changes: Trading day effects and risk measurement. *Journal of Finance*, 37:693–715, 1982.

[17] P. Raghavan. A statistical adversary for on-line algorithms. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 7:79–83, 1991.

[18] R.J. Rogalski and J.D. Vinso. Empirical properties of foreign exchange rates. *Journal of International Business Studies*, 9:69–79, 1978.

[19] D.D. Sleator and R.E. Tarjan. Amortized efficiency of list update and paging rules. *Communication of the ACM*, 28:202–208, 1985.

[20] A.L. Tucker and E. Scott. A study of diffusion processes for foreign exchange rates. *Journal of International Money and Finance*, 6:465–478, 1987.

[21] D.J Wilde and C.S. Beighther. *Foundations of Optimization*. Prentice–Hall, Inc., Englewood Cliffs, N.J., 1967.

[22] A.C. Yao. New algorithms for bin packing. *Journal of the ACM*, 27:207–227, 1980.

[23] N. Young. competitive paging as cache-size varies. In *Proceedings of the 2nd ACM-SIAM Symposium on Discrete Algorithms*, 1991.

Appendix

Proof of Lemma 1

We prove by induction on $n \geq 2$ that if $\Pi > 1$ then $R_n(\Pi, \cdot) > 1$. The base case, $n = 2$, clearly holds.

Induction step: Assume that $\Pi > 1$. We divide the proof into three cases:

- case (i) $e_1 = e_2$: then $\min\{\Pi, \Pi e_2/e_1\} = \Pi$, and $R_n(\Pi, e_1) = R_{n-1}(\Pi, e_2)$. By the induction hypothesis on $n-1$ we have that $R_{n-1}(\Pi, e_2) > 1$.

- case (ii) $e_1 < e_2$: here again, since $e_2/e_1 > 1$, $\min\{\Pi, \Pi e_2/e_1\} = \Pi$ and by the induction hypothesis, $R_{n-1}(\Pi, e_2) > 1$. Let us write $R_{n-1}(\Pi, e_2) = 1 + \varepsilon$ where $\varepsilon > 0$. Even though $e_1/e_2 - 1 < 0$ it is clear that by choosing a sufficiently small s_1, the on-line player can guarantee that $R_n > 1$. More specifically, we shall show that for any choice of $e_2 > e_1$ and any choice of $s_1 < \frac{e_2\varepsilon}{(e_2-e_1)(1+\varepsilon)}$, $R_n > 1$. Let $s_1 = \frac{e_2\varepsilon}{(e_2-e_1)(1+\varepsilon)} - \delta$ for some small positive δ. Then,

$$\begin{aligned} R_n(\Pi, e_1) &\geq \min_{e_2 > e_1} \left[\left(\frac{e_1}{e_2} - 1 \right) \frac{e_2\varepsilon}{(e_2 - e_1)(1+\varepsilon)} - \delta + 1 \right] \cdot (1+\varepsilon) \\ &= \min_{e_2 > e_1} \left(\varepsilon - \delta \cdot \frac{e_1 - e_2}{e_2} + 1 \right) \cdot (1+\varepsilon) \\ &= \min_{e_2 > e_1} (1+\varepsilon)^2 + \delta \cdot \frac{e_2 - e_1}{e_2} \cdot (1+\varepsilon) \\ &> (1+\varepsilon)^2. \end{aligned}$$

- case (iii) $e_1 > e_2$: now for every positive fraction s_1, $(e_1/e_2 - 1)s_1 + 1 > 1$. If $\Pi e_2/e_1 > 1$ then by the induction hypothesis, $R_{n-1}(\Pi e_2/e_1, e_2) > 1$ as required. Otherwise, S^* ceases all activity from the second day onward, since it is known that no profit potential exists in the remainder of the sequence. In any case, $R_{n-1} \geq 1$, so $R_n > 1$.

Thus, we have completed the proof.

Proof of Lemma 2

We begin by restricting the adversary to two possible moves. The adversary can either realize a profit Π (the entire off-line profit) or cause the investment to be a total loss (set e_2 so high that $\frac{e_1}{e_2}s$ is negligible). Let $R_n(\Pi)$ be the optimal on-line return against this restricted adversary. When $n = 2$, $R_2(\Pi) = \Pi$. In the case where the adversary realizes a profit Π, the on-line player receives a return of $s\Pi + 1 - s$. Since the entire off-line profit is realized, the trading will stop. In the case where the adversary causes the on-line player to completely lose his previous investment, the on-line player will receive a return of $(1-s)R_{n-1}(\Pi)$. The amount he invested, s, is lost, but he can still receive the optimal return on the remaining $1 - s$.

The on-line player will set s to maximize his return, while the adversary will choose between his two options to minimize the return. Therefore, the optimal return satisfies the following

$$R_n(\Pi, e_1) = \max_{0 \leq s_1 \leq 1} \min\{s\Pi + 1 - s, (1-s)R_{n-1}(\Pi)\}$$

Note that $s\Pi + 1 - s$ is increasing in s while $(1-s)R_{n-1}(\Pi)$ is decreasing in s, so the on-line player will choose s such that $s\Pi + 1 - s = (1-s)R_{n-1}(\Pi)$.

Solving for s,

$$s = \frac{R_{n-1}(\Pi) - R_n(\Pi)}{R_{n-1}(\Pi)}$$

It is not hard to see that $0 \le R_n(\Pi) \le R_{n-1}(\Pi)$, so $0 \le s \le 1$.

Substituting for s, we obtain

$$R_n(\Pi, \cdot) = \frac{(\Pi - 1)\,(R_{n-1}(\Pi) - R_n(\Pi))}{R_{n-1}(\Pi)} + 1$$

Rearranging terms,

$$\frac{1}{R_n(\Pi)} = \frac{1}{\Pi} + \frac{\Pi - 1}{\Pi R_{n-1}(\Pi)}$$

Let $R_n^{-1}(\Pi) \stackrel{\text{def}}{=} \frac{1}{R_n(\Pi)}$. Thus,

$$R_n^{-1}(\Pi) = \frac{1}{\Pi} + \frac{\Pi - 1}{\Pi} R_{n-1}^{-1}(\Pi).$$

Solving this linear recurrence using the base case $R_2^{-1}(\Pi) = \frac{1}{\Pi}$, we obtain

$$R_n^{-1}(\Pi) = 1 - \left(1 - \frac{1}{\Pi}\right)^{n-1}$$

and thus,

$$R_n(\Pi) = \frac{1}{1 - (1 - \frac{1}{\Pi})^{n-1}}. \tag{5}$$

Since (5) is the optimal on-line return against this restricted adversary, it must *upper* bound the return against the unrestricted adversary.

Proof of Lemma 3

Let $n = 3$. We show how the adversary can construct a sequence, $E = e_1, e_2, e_3$ for which $R_S(E) < 1$ and $R_{OPT}(E) > 1$. Let e_1 be any positive real. If S does not purchase any yen on the first day, then, since S is non-degenerate, it must buy some yen on the second day and the adversary can take $e_1 \gg e_2 < e_3$. Clearly, $R_{OPT}(E)$ can be made arbitrarily large and $R_S(E) < 1$. Therefore, assume that S trades $s_1 > 0$ dollars on the first day (with rate e_1). If $s_1 = 1$, the adversary can take any $e_1 < e_3 < e_2$ with a clear loss to S and a return of e_2/e_3 to OPT. Thus, assume that $s_1 < 1$. Let δ be any positive real such that $\delta < s_1$. For any $0 < \varepsilon < \frac{s_1 - \delta}{1 - s_1}$ let

$$e_2 = \frac{s_1 e_1 (1+\varepsilon)}{s_1(1+\varepsilon) - \varepsilon - \delta}; \tag{6}$$

$$e_3 = e_2/(1+\varepsilon) \tag{7}$$

First, notice that since $\delta < s_1$ and $\varepsilon < \frac{s_1-\delta}{1-s_1}$, e_2 and e_3 are positive and hence, well-defined exchange rates. Also, it is easy to see that $e_2 > e_1$. Therefore, to perform optimally from this stage onward, S must convert the remaining dollars to yen on the second day and all yen back to dollars on the last day. Thus,

$$R_S(E) \leq \frac{s_1 e_1 + (1-s_1)e_2}{e_3}. \tag{8}$$

Substituting (6) and (7) for e_2 and e_3 in (8) respectively, it is not hard to verify that $R_S(E) \leq 1-\delta$. Clearly, $R_{OPT}(E) = e_2/e_3 = 1+\varepsilon$.

It is possible to extend this exchange rate sequence to any length $n > 3$. Moreover, one can show that for larger n, the adversary can construct (more complicated) sequences for which the guaranteed off-line profit is larger, while the on-line profit remains negative.

Proof of Lemma 4

We present a sketch. Fix any $\Pi > 1$. The adversary presents to the on-line player a strictly monotone increasing exchange rate sequence $e_1, e_1\Pi, e_1\Pi^2, \ldots$ where e_1 is an arbitrary real. Let i be the first day for which $s_i\Pi < 1$. There must be such a day (after at most $\lceil \Pi \rceil$ days) since otherwise it means that the on-line player spends an amount greater than $1/\Pi$ every day for an infinite number of days. Then, on the $(i+1)$st day the adversary drops the rates by a factor of Π. Thus, the off-line return is Π and the on-line return is $s_i < 1$.

Proof of Lemma 5

Recall the initial conditions of S^{**} (3). For all n,

$$\begin{aligned} R_\alpha^{-1}(0,n) &= 1, \\ R_\alpha^{-1}(n,n) &= \alpha^{-n}. \end{aligned}$$

Intuitively, $R_\alpha^{-1}(m,n)$ has no meaning for $m > n$ or $m < 0$. We now extend $R_\alpha^{-1}(m,n)$ to these cases, while still satisfying both the recurrence and initial conditions.

Let $R_\alpha^{-1}(m,n) = 1, m < 0$ and $R_\alpha^{-1}(m,n) = \alpha^{(n-2m)}, m > n$. Note that for $n = m$, $\alpha^{(n-2m)} = \alpha^{-n}$, so the two conditions combine to $R_\alpha^{-1}(m,n) = \alpha^{(n-2m)}, m \geq n$

Claim: The extended $R_\alpha^{-1}(m,n)$ satisfies the recurrence and the initial conditions.

Proof: By induction on n. For the base case, $n = 1$, we have $R_\alpha^{-1}(m,1) = 1$ for $m \leq 0$, and $R_\alpha^{-1}(m,1) = \alpha^{(1-2m)}$ for $m > 0$. The initial conditions $R_\alpha^{-1}(0,1) = 1$ and $R_\alpha^{-1}(1,1) = \alpha^{-1}$ are satisfied. We assume the induction hypothesis for $n-1$, and prove it for n.

1. For $m \leq 0$,

$$\begin{aligned} R_\alpha^{-1}(m,n) &= \frac{\alpha}{\alpha+1} R_\alpha^{-1}(m,n-1) + \frac{1}{\alpha+1} R_\alpha^{-1}(m-1,n-1) \\ &= \frac{\alpha}{\alpha+1} \cdot 1 + \frac{1}{\alpha+1} \cdot 1 \\ &= 1 \end{aligned}$$

2. For $m \geq n$,

$$R_\alpha^{-1}(m,n) = \frac{\alpha}{\alpha+1} R_\alpha^{-1}(m,n-1) + \frac{1}{\alpha+1} R_\alpha^{-1}(m-1,n-1)$$

$$
\begin{aligned}
&= \frac{\alpha}{\alpha+1}\alpha^{n-1-2m} + \frac{1}{\alpha+1}\cdot\alpha^{n-1-2m+2} \\
&= \frac{1}{\alpha+1}(\alpha^{n-2m} + \alpha^{n-2m+1}) \\
&= \alpha^{n-2m}
\end{aligned}
$$

∎

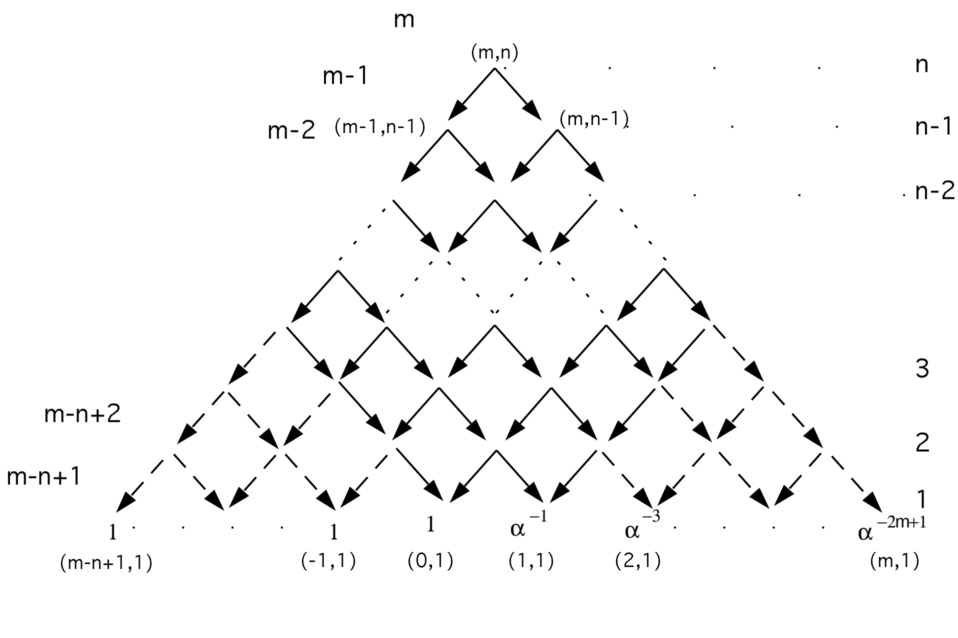

Figure 5: Directed graph showing the expansion of the recurrence

Consider the directed graph in Figure 5 . Each node is labeled (x, y) with the "root" labeled (m, n). The value stored at each node is $R_\alpha^{-1}(x, y)$.

For node (x, y), x corresponds to the vertical height in the grid. "Leaf" nodes have height 1, and the "root" has height n. y labels the left-to-right diagonals. The rightmost diagonal is m, the diagonal immediately below is $m-1$, etc. The left most diagonal (a single node) is labeled $m-n+1$.

For a node (x, y), its left child is $(x-1, y-1)$ and its right child is $(x, y-1)$. To compute $R_\alpha^{-1}(x, y)$ from its children, we add $\frac{1}{\alpha+1}R_\alpha^{-1}(x-1, y-1)$ (the contribution of the left child) to $\frac{\alpha}{\alpha+1}R_\alpha^{-1}(x, y-1)$ (the contribution of the right child) (i.e. $R_\alpha^{-1}(x, y) = \frac{1}{\alpha+1}R_\alpha^{-1}(x-1, y-1) + \frac{\alpha}{\alpha+1}R_\alpha^{-1}(x, y-1)$). Thus, we can consider each left branch to be weighed by $\frac{1}{\alpha+1}$ and each right branch by $\frac{\alpha}{\alpha+1}$.

If we expand the recurrence $n-1$ times, we obtain an expression in $R_\alpha^{-1}(m, 1), R_\alpha^{-1}(m-1, 1), R_\alpha^{-1}(m-2, 1), \ldots R_\alpha^{-1}(m-n+1, 1)$. The number of times $R_\alpha^{-1}(m-k, 1)$ occurs is exactly the number of paths from (m, n) to $(m-k, 1)$, which is $\binom{n-1}{k}$. In addition, each term is weighed by $\frac{1}{\alpha+1}$ for each left branch and $\frac{\alpha}{\alpha+1}$ for each right branch. Each path to $(m-k, 1)$ has the same number of left and right moves, so the weight of each path is identical. Therefore,

$$
\begin{aligned}
R_\alpha^{-1}(m,n) &= \sum_{\text{leaf nodes}} [R_\alpha^{-1}(x,y)] \cdot [\text{Number of paths}] \cdot [\text{Weight of path}] \\
&= \sum_{i=0}^{n-1} R_\alpha^{-1}(m-n+1+i,1) \binom{n-1}{i} \left(\frac{\alpha}{\alpha+1}\right)^i \left(\frac{1}{\alpha+1}\right)^{(n-1-i)} \\
&= \sum_{i=0}^{n-m-1} 1 \cdot \binom{n-1}{i} (\frac{\alpha}{\alpha+1})^i (\frac{1}{\alpha+1})^{(n-1-i)} \\
&\quad + \sum_{i=n-m}^{n-1} \alpha^{(2n-2m-2i-1)} \binom{n-1}{i} (\frac{\alpha}{\alpha+1})^i (\frac{1}{\alpha+1})^{(n-1-i)}
\end{aligned}
$$

In the second sum, we substitute j for $n-1-i$,

$$
\begin{aligned}
R_\alpha^{-1}(m,n) &= \sum_{i=0}^{n-m-1} \binom{n-1}{i} (\frac{\alpha}{\alpha+1})^i (\frac{1}{\alpha+1})^{(n-1-i)} \\
&\quad + \sum_{j=0}^{m-1} \alpha^{(2j-2m+1)} \binom{n-1}{j} (\frac{\alpha}{\alpha+1})^{(n-1-j)} (\frac{1}{\alpha+1})^j \\
&= \sum_{i=0}^{n-m-1} \binom{n-1}{i} (\frac{\alpha}{\alpha+1})^i (\frac{1}{\alpha+1})^{(n-1-i)} \\
&\quad + \alpha^{(n-2m)} \sum_{j=0}^{m-1} \binom{n-1}{j} (\frac{\alpha}{\alpha+1})^j (\frac{1}{\alpha+1})^{(n-1-j)}
\end{aligned}
$$

Proof of Lemma 6

Recall that

$$
\begin{aligned}
B(k;n,p) &= \sum_{i=0}^{k} \binom{n}{k} p^i (1-p)^{n-i} \\
R_\alpha^{-1}(m,n) &= B((1-c)n-1; n-1, 1-\beta) + \alpha^{(1-2c)n} B(cn-1; n-1, 1-\beta)
\end{aligned}
$$

For the sake of brevity, define

$$
\begin{aligned}
B_1 &= B((1-c)n-1; n-1, 1-\beta) \\
B_2 &= B(cn-1; n-1, 1-\beta)
\end{aligned}
$$

To compute the upperbounds on B_1 and B_2, we make use of the following Chernoff bound [1]:

Theorem 1 *Let $X_1, \cdots, X_n$ be n mutually independent random variables with*

$$
\begin{aligned}
Pr\,[X_i = 1] &= p \\
Pr\,[X_i = 0] &= 1-p
\end{aligned}
$$

Let $X = X_1 + \cdots + X_n$. Then for $a > 0$,

$$Pr\,[X < pn - a] < e^{-a^2/2pn}$$

$B(k; n, p)$ is simply the probability that at most k successes occur in a series of n Bernoulli trial and success probability p. We can use this Chernoff bound to bound $B_1 + \alpha^{n(1-2c)}B_2$. We provide bounds based on the value of c: By the theorem above, when $c > \beta$,

$$\begin{aligned} B_1 &= B(n(1-c)-1; n-1, 1-\beta) \\ &< e^{-(n-1)^2(c-\beta)^2/2(1-\beta)(n-1)} \\ &= e^{-(n-1)(c-\beta)^2/2(1-\beta)} \\ &= e^{-\Omega(n)} \end{aligned}$$

When $c < \beta$

$$\begin{aligned} B_1 &= B(nc-1; n-1, \beta) \\ &> 1 - e^{-(n-1)^2(\beta-c)^2/2\beta(n-1)} \\ &= 1 - e^{-(n-1)(\beta-c)^2/2\beta} \\ &= 1 - e^{-\Omega(n)} \end{aligned}$$

When $c < 1-\beta$

$$\begin{aligned} B_2 &= B(nc-1; n-1, 1-\beta) \\ &< e^{-(n-1)^2(1-\beta-c)^2/2(1-\beta)(n-1)} \\ &= e^{-(n-1)(1-\beta-c)^2/2(1-\beta)} \\ &= e^{-\Omega(n)} \end{aligned}$$

When $c > 1-\beta$

$$\begin{aligned} B_2 &= 1 - B((1-c)n-1; n-1, \beta) \\ &> 1 - e^{-(n-1)^2(c-(1-\beta))^2/2\beta(n-1)} \\ &= 1 - e^{-(n-1)(c-(1-\beta))^2/2\beta} \\ &= 1 - e^{-\Omega(n)} \end{aligned}$$

We will need tighter bounds than the Chernoff bounds can provide in some of the case analysis below. The following theorem provides the necessary bounds. The following theorem can be found in [1]:

Theorem 2 *For any constants $1 \geq p > c \geq 0$,*

$$\begin{aligned} B(cn; n, p) &= \sum_{i=0}^{cn} \binom{n}{i} p^i (1-p)^{n-i} \\ &= 2^{n(H(c)+o(1))} p^{cn} (1-p)^{(1-c)n} \\ &= 2^{o(n)} \left(\frac{p}{c}\right)^{cn} \left(\frac{1-p}{1-c}\right)^{(1-c)n} \end{aligned}$$

where $H(c) = -c\log c - (1-c)\log(1-c)$ is the entropy function.

Using the above bounds we can now derive bounds on $B_1 + \alpha^{n(1-2c)}B_2$ for all values of $0 \leq c \leq 1$.

$0 \leq c \leq \beta$: In this case, $B_1 \to 1$ as n becomes large. Because the entire sum is at most 1, and $\alpha^{n(1-2c)}B_2$ is positive, $B_1 + \alpha^{n(1-2c)}B_2 \to 1$. Therefore, $R_\alpha(m,n) \to 1$.

$\beta < c \leq 1/2$: Here B_1 is exponentially small. We wish to show the same for $\alpha^{n(1-2c)}B_2$. Here we will need the tight bound from Theorem 2.

$$\begin{aligned}\alpha^{n(1-2c)}B_2 &= O(\alpha^{n(1-2c)}B(cn;n,1-\beta)) \\ &= 2^{o(n)}\alpha^{(1-2c)n}\left(\frac{1-\beta}{c}\right)^{cn}\left(\frac{\beta}{1-c}\right)^{(1-c)n} \\ &= 2^{o(n)}\left(\frac{1-\beta}{\alpha c}\right)^{cn}\left(\frac{\alpha\beta}{1-c}\right)^{(1-c)n} \\ &= 2^{o(n)}\left(\frac{\beta}{c}\right)^{cn}\left(\frac{1-\beta}{1-c}\right)^{(1-c)n}\end{aligned}$$

To determine that this function is exponentially small, we need only show that $V(c,\beta) = \left(\frac{\beta}{c}\right)^{c}\left(\frac{1-\beta}{1-c}\right)^{1-c} < 1$. First note that $V(\beta,\beta) = 1$. To complete the proof, we show that $V(c,\beta)$ is strictly decreasing as c increases beyond β. To do this we show that the derivative with respect to c of $\ln(V(c,\beta))$ is negative for these values of c.

$$\begin{aligned}\ln'(V(c,\beta)) &= \ln\beta - 1 - \ln c - \ln(1-\beta) + \ln(1-c) + 1 \\ &= \ln\beta - \ln c - \ln(1-\beta) + \ln(1-c)\end{aligned}$$

Now, $\ln\beta - \ln c - \ln(1-\beta) + \ln(1-c) < 0 \iff \frac{\beta}{c}\frac{1-c}{1-\beta} < 1$. But $\frac{\beta}{c} < 1$ and $\frac{1-c}{1-\beta} < 1$ for $\beta < c \leq 1/2$. So $B_1 + \alpha^{n(1-2c)}B_2 \to e^{-\Omega(n)}$.

Therefore, $R_\alpha(m,n) \to e^{\Omega(n)}$

$1/2 < c < 1-\beta$: In this region, B_2 is still exponentially small, so $\alpha^{n(1-2c)}B_2$ is $\alpha^{n(1-2c)}e^{-\Omega(n)}$. We need only show that B_1 takes the same form. Consider $\alpha^{n(2c-1)}B_1$. We make a substitution of variables to show that this is exponentially small. Consider $d = 1-c$. Then $\alpha^{n(2c-1)}B_1 = \alpha^{n(1-2d)}B((1-d)-1, n-1, 1-\beta)$ for $\beta < d < 1/2$. This is precisely the function analyzed in the previous case, which we showed to be exponentially small. Thus, $B_1 + \alpha^{n(1-2c)}B_2 \to \alpha^{n(1-2c)}e^{-\Omega(n)}$. Therefore, $R_\alpha(m,n) \to \alpha^{n(2c-1)}e^{\Omega(n)}$.

$c \geq 1-\beta$: B_1 is exponentially small. B_2 is moving exponentially close to 1, so $B_1 + \alpha^{n(1-2c)}B_2 \to \alpha^{n(1-2c)}$. Therefore, $R_\alpha(m,n) \to \alpha^{n(2c-1)}$.