Towards Understanding the Predictability of Stock Markets from the Perspective of Computational Complexity*

James Aspnes† David F. Fischer‡ Michael J. Fischer§ Ming-Yang Kao¶
Alok Kumar||

May 31, 2002

Abstract

This paper initiates a study into the century-old issue of market predictability from the perspective of computational complexity. We develop a simple agent-based model for a stock market where the agents are traders equipped with simple trading strategies, and their trades together determine the stock prices. Computer simulations show that a basic case of this model is already capable of generating price graphs which are visually similar to the recent price movements of high tech stocks. In the general model, we prove that if there are a large number of traders but they employ a relatively small number of strategies, then there is a polynomial-time algorithm for predicting future price movements with high accuracy. On the other hand, if the number of strategies is large, market prediction becomes complete for two new computational complexity classes CPP and promise-BCPP, where $\mathrm{P}^{\mathrm{NP}[O(\log n)]} \subseteq \mathrm{BPP}_{\mathrm{path}} \subseteq$ promise-BCPP $\subseteq$ CPP $=$ PP. These computational hardness results open up a novel possibility that the price graph of an actual stock could be sufficiently deterministic for various prediction goals but appear random to all polynomial-time prediction algorithms.

1 Introduction

The issue of market predictability has been debated for more than a century (see [8] for earlier papers and [6, 15, 20, 22] for more recent viewpoints). In 1900, the pioneering work "Theory of Speculation" of Louis Bachelier used Brownian motion to analyze the stochastic properties of security prices [8]. Since then, Brownian motion and its variants have become textbook tools for modeling financial assets. Relatively recently, the radically different methodology of Mandelbrot used fractals to approximate price graphs deterministically [23]. In this paper, we initiate a study into this long-running issue from the perspective of computational complexity.

*A preliminary version will appear in *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2001.

†Department of Computer Science, Yale University, New Haven, CT 06520-8285, USA. Email: aspnes@cs.yale.edu. Supported in part by NSF Grant CCR-9820888.

‡Class of 1999, Yale College, New Haven, CT 06520-8285, USA. Email: fischer@aya.yale.edu.

§Department of Computer Science, Yale University, New Haven, CT 06520-8285, USA. Email: fischer-michael@cs.yale.edu.

¶Department of Computer Science, Yale University, New Haven, CT 06520-8285, USA. Email: kao-ming-yang@cs.yale.edu. Supported in part by NSF Grants CCR-9531028 and CCR-9988376. Current address: Department of Electrical Engineering and Computer Science, Tufts University, Medford, MA 02155, USA. Email: kao@eecs.tufts.edu.

||School of Management, Yale University, New Haven, CT 06520, USA. Email: alok.kumar@yale.edu.

We develop a simple agent-based model for a stock market [9, 21]. The agents are traders equipped with simple trading strategies, and their trades together determine the stock prices. We first consider a basic case of this model where there are only two strategies, namely, momentum and contrarian strategies. The choice of this base model and thus our general model is justified at two levels: (1) Experimental and empirical studies in the finance literature [1, 4, 7, 10–12, 19] show that a large number of traders primarily follow these two strategies. (2) Our own simulation results show that despite its simplicity, the base model is capable of generating price graphs which are visually similar to the recent price movements of high tech stocks (Figures 1 and 2).

With these justifications, we then consider the issue of market predictability in the general model. We prove that if there are a large number of traders but they employ a relatively small number of strategies, then there is a polynomial-time algorithm to predict future price movements with high accuracy (Theorem 5). On the other hand, if there are also a large number of strategies, then the problem of predicting future prices becomes computationally very hard. To describe this hardness, we define two new computational complexity classes called CPP and promise-BCPP (Definitions 8 and 16). We show that some market prediction problems are hard for these two classes (Theorems 17 and 18) and that $\mathrm{P}^{\mathrm{NP}[O(\log n)]} \subseteq \mathrm{BPP}_{\mathrm{path}} \subseteq$ promise-BCPP $\subseteq \mathrm{CPP} = \mathrm{PP}$.

These computational completeness results open up the possibility that the price graph of an actual stock could be sufficiently deterministic for various prediction purposes but appear random to all polynomial-time prediction algorithms. This is in contrast to the most popular academic belief that the future price of a stock cannot be predicted from its historical prices because the latter are statistically random and contain no information. This new possibility also differs from the fractal-based methodology in that the price graph of a stock could be a fractal but the fractal might not be computable in polynomial time. The findings in this paper can by no means settle the debate about market predictability. Our goal is only that our alternative approach could provide new insights to the predictability issue in a systematic manner. In particular, it could provide a general framework to investigate the many documented technical trading rules [25] and to generate novel and significant interdisciplinary research problems for computer science and finance.

The rest of the paper is organized as follows. Section 2 discusses the basic market model. Section 3 formulates the general model. Section 4 proves the complexity results for market prediction in the general model. We conclude the paper with some directions for future research in Section 5.

2 A Basic Market Model

In this section, we present a very simple market model, called the *deterministic-switching MC* (DSMC) model. The letter M stands for a *momentum* strategy, and the letter C for a *contrarian* strategy. These two strategies and the model itself are defined in Section 2.1. Some computer simulations for this model are reported in Section 2.2.

Intuitively, these strategies are heuristics ("rules of thumb") used by traders in the absence of reliable asset valuation models. As discussed in [12], a momentum trader may observe a sequence of "up" trades (price increments) and execute a buy trade in the anticipation that she will not be one of the last buyers, knowing very well that the asset is overpriced. Similarly, she may see some "down" trades (price decrements) and then make a sell trade in the hope that there will be more sellers after her. In contrast, after detecting a number of "up" (respectively, down) trades, a contrarian trader may submit a sell (respectively, buy) trade, anticipating a price reversal.

Both experimental and empirical studies have shown that traders look at past price dynamics to form their expectations of future prices, and a large number of them primarily follow momentum or contrarian strategies [1, 7, 10, 11]. In addition, the traders may switch between these two diametrically opposite strategies. Momentum and contrarian strategies are dominant in the behavior of professional market timers as well [19]. The use of momentum and contrarian strategies sometimes signifies gambling tendencies among traders [7]. In fact, a market model with momentum and contrarian traders can also be interpreted as a market with noise traders and rational traders, where the noise traders essentially follow a momentum strategy while the rational traders attempt to exploit the noise traders by following a contrarian strategy [4, 12].

2.1 Defining the DSMC Model

In the DSMC model, there is only one stock traded in the market. The model is completely specified by three integer parameters $m, L, k > 0$, and a real parameter $\alpha > 0$ as follows.

There are m traders in the market, and each trader's strategy set consists of momentum ($\mathcal{M}$) and contrarian ($\mathcal{C}$) strategies. At the beginning of day 1 of the investment period, each trader randomly chooses her initial strategy from $\{\mathcal{M}, \mathcal{C}\}$ and an integer $\ell_i \in [2, L]$ with equal probability, where L is the *maximum strategy switching period*. This is the only source of randomness in the DSMC model; from this point onwards, there is no random choice.

Rule 1 (Deterministic Strategy Switching Rule) For days $1, \ldots, k+1$, there is no trading. Each trader starts trading from day $k+2$ using her initial strategy. Trader i uses the same strategy for ℓ_i days and switches it at the beginning of every ℓ_i days.

The next rule defines the two strategies with respect to a given memory size k, which is the same for all traders.

Rule 2 (Trading Rule) At the beginning of day t, observe the stock prices P_f of days $f \in [t-(k+1), t-1]$. For $g \in [t-k, t-1]$, count the number k_u of days g when $P_g > P_{g-1}$; and the number k_d of days when $P_g < P_{g-1}$. The *k-day trend* is defied as $\mathrm{Tr}(k, t) = k_u - k_d$. Then, if $\mathrm{Tr}(k, t) \geq 0$ (respectively, < 0), the momentum strategy $\mathcal{M}$ buys (respectively, sells) one share of the stock at the market price determined by Rule 3 below. In contrast, the contrarian strategy $\mathcal{C}$ sells (respectively, buys) one share of the stock.

For instance, suppose that $k = 2$, and investor i picks her initial strategy $\mathcal{M}$ and $\ell_i = 2$ at the beginning of day 1. She then observes the prices of days 1, 2, 3, which are, say, \$80, \$82, \$90. At the beginning of day 4, she issues a market order to buy one share of the stock. The orders issued by the traders on day 4 together determine the price of day 4 as specified by Rule 3. Suppose that the price of day 4 is \$91, then investor i issues another market buy order at the beginning of day 5. Since her ℓ_i is 2, at the beginning of day 6, she switches her strategy from $\mathcal{M}$ to $\mathcal{C}$.

Rule 3 (Price Adjustment Rule) The prices for days $1, \ldots, k+1$ are given. On day $t \geq k+2$, let m_b and m_s be the total numbers of buys and sells, respectively. Then, the price P_t on day t is determined by the following equation:

$$P_t - P_{t-1} = \alpha \cdot (m_b - m_s),$$

where α is the unit of price change.

Figure 1: A one-year price sequence generated using the DSMC model. Parameters: number of traders $m = 20$, memory size $k = 2$, maximum strategy switching period $L = 8$, unit of price change $\alpha = 0.25$, number of trading days = 250. The price graph appears strikingly similar to the recent price movements of high tech stocks.

Figure 2: A one-year price sequence generated using the DSMC model. The parameters are the same as those for Figure 1.

2.2 Computer Simulation on the DSMC Model

We have conducted some computer simulations of the DSMC model to test whether it can generate realistic price graphs. Because we had to examine the graphs visually, our time constraints limited the number of these simulations to only about six hundred. For a large fraction of them, we set $m = 20$, $L = 8$, and the initial k prices in the range of \$70 to \$90. We then focused on testing the effect of memory size k [24]. Two main findings are as follows:

- For $k = 1$, the price graphs were not visually real.
- For $k = 2$, about one out of four graphs were strikingly similar to those of recent high tech stocks, which was a major positive surprise to us. Two representatives of such graphs are shown in Figures 1 and 2.

These two statements are based on our subjective impressions and limited simulations. To further understand the DSMC model, it would be useful to automate statistical analysis on the price graphs generated by this model and compare them with real stock prices.

3 A General Market Model

In this section, we define a market model, called the *AS* model, where the word AS stands for arbitrary strategies. It can be verified in a straightforward manner that the DSMC model is a special case of the AS model.

In the AS model, there is only one stock traded in the market. The model is completely specified as follows with five parameters: (1) the number m of traders, (2) a unit $\alpha > 0$ of price change, (3) a set $\Pi = \{\mathcal{S}^1, \ldots, \mathcal{S}^h\}$ of strategies, (4) a price adjustment rule (Equation 1 or 2 below), and (5) a joint distribution of the population variables $X_1, \ldots, X_h$.

Rule 4 (Market Initialization) There are m traders in the market. At the beginning of day 1 of the investment period, each trader randomly chooses her initial strategy from Π. Let X_i be the number of traders who choose $\mathcal{S}^i$. Then, each X_i is a random variable, which is the only source of randomness in the model. (Unlike the DSMC model, because the allowable generality of Π, the AS model does not need strategy switching.)

Different joint distributions of the variables X_i lead to different specific models and prediction problems. In Section 4.2, we consider joint distributions that tend to Gaussian in the limit as the number m of traders becomes large. In Section 4.3, we consider the case where the variables X_i are independent, and each is 0 or 1 with equal probability.

Rule 5 (Trading Strategies) There is no trading on day 0. At the beginning of day $t \geq 1$, a trader observes the historical prices $P_0, \ldots, P_{t-1}$ and reacts by issuing a market order to buy one share of the stock, hold (i.e., do nothing), or sell one share according her strategy. Formally, a *strategy* is a collection of functions $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_t, \ldots\}$, where each $\mathcal{S}_t$ maps $P_0, \ldots, P_{t-1}$ to $+1$ (buy), 0 (hold), or -1 (sell).

The price P_t of day t is determined at the end of the day by the day's m market orders using Rule 6. Since the traders choose their strategies randomly, the sequence $P_0, P_1, \ldots, P_t, \ldots$ is a stochastic process. We write $\mathcal{F}_t$ for the probability space induced by all possible sequences $\langle P_0, \ldots, P_t \rangle$ [18]. Then, we think of each function $\mathcal{S}_t$ as a random variable on $\mathcal{F}_{t-1}$.

We distinguish between strategies that react to price movements and those that ignore them.

- $\mathcal{S}$ is an *active* strategy if the functions $\mathcal{S}_t$ may or may not be constant functions. An *active* trader is one with an active strategy. Examples of active strategies include many used by day traders, who try to capture extremely short-term price trends.

- $\mathcal{S}$ is a *passive* strategy if the functions $\mathcal{S}_t$ all are constant functions. A *passive* trader is one with a passive strategy. Examples of passive strategies include two very popular ones: (1) dollar averaging, which invests an equal amount every day over a chosen period, and (2) monthly retirement contributions by educational institutions, which are made on the same day every month.

Rule 6 (Price Adjustment) The price P_0 is given. At the end of day $t \geq 1$, the price P_t is determined by the day's market orders to buy or sell from the traders. We consider two simple rules:

With the *proportional increment* (PI) rule,

$$P_t = P_{t-1} + \alpha \cdot \sum_{i=1}^{h} X_i \cdot \mathcal{S}_t^i, \tag{1}$$

where α is the unit of price change. Thus we can observe directly the net difference between the number of buyers and sellers on day t.

With the *fixed increment* (FI) rule,

$$P_t = P_{t-1} + \alpha \cdot \operatorname{sign}\left(\sum_{i=1}^{h} X_i \cdot \mathcal{S}_t^i\right). \tag{2}$$

In this case, the market moves up or down depending on whether the majority of traders are buying or selling, but the amount by which it moves is fixed at α.

For notational brevity, an *AS+FI model* refers to an AS model with the fixed increment rule, and an *AS+PI model* refers to an AS model with the proportional increment rule.

In reality, the price tends to move up if there are more buy orders than sell orders; similarly, the price tends to move down if there are more sell orders than buy orders. The FI rule is meant to model the sign but not the magnitude of the slope of this correlation, while the PI rule attempts to model both. Clearly, there can be many other increment rules, which this paper leaves for future research.

4 Predicting the Market

Informally, the *market prediction problem* at the beginning of day t is defined as follows:

- The data consists of (1) the five parameters of an AS-model, i.e., m, α, Π, X_i, and a price adjustment rule, and (2) a price history $P_0, \ldots, P_{t-1}$.

- The goal is to predict the price P_t by estimating the conditional probabilities $\Pr[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$, $\Pr[P_t < P_{t-1} \mid P_0, \ldots, P_{t-1}]$, and $\Pr[P_t = P_{t-1} \mid P_0, \ldots, P_{t-1}]$.

Note that $\Pr[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$ is symmetric to $\Pr[P_t < P_{t-1} \mid P_0, \ldots, P_{t-1}]$ and $\Pr[P_t = P_{t-1} \mid P_0, \ldots, P_{t-1}] = 1 - \Pr[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}] - \Pr[P_t < P_{t-1} \mid P_0, \ldots, P_{t-1}]$. Thus, from this point onwards, our discussion focuses on estimating $\Pr[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$.

From an algorithmic perspective, we sometimes assume that the price adjustment rule and the joint distribution of the variables X_i are fixed, and that the input to the algorithm is m, α, a description of Π, and the price history. This allows different algorithms for different model families as well as side-steps the issue of how to represent the possibly very complicated joint distribution of the variables X_i as part of the input. As for the description of Π, we only need $\mathcal{S}_1^i, \ldots, \mathcal{S}_t^i$ for each $\mathcal{S}^i \in \Pi$ instead of the whole Π, and the description of these functions can simplified by restricting their domains to consist of the price sequences consistent with the given price history.

4.1 Markets as Systems of Linear Constraints

In the AS+FI model with parameters m and α, a price sequence $P_0, \ldots, P_t$ and Π can yield a set of linear inequalities in the population variables X_i as follows. If the price changes on day t, we have

$$\operatorname{sign}(P_t - P_{t-1}) \sum_{i=1}^{h} \mathcal{S}_t^i X_i > 0. \tag{3}$$

If the price does not change, we have instead the equation

$$\sum_{i=1}^{h} \mathcal{S}_t^i X_i = 0. \tag{4}$$

Furthermore, any assignment of the variables X_i that satisfies either inequality is feasible with respect to the corresponding price movement on day t. In both cases, $\mathcal{S}_t^i$ is computable from the price sequence $P_0, \ldots, P_{t-1}$. The same statements hold for days $1, \ldots, t-1$. Therefore, given m and α, we can extract from Π and $P_0, \ldots, P_t$ a set of linear constraints on the variables X_i. The converse holds similarly. We formalize these two observations in Lemmas 1 and 2 below.

Lemma 1 *In the AS+FI model with parameters m and α, given Π and a price sequence $P_0, \ldots, P_\beta$, there are matrices A and B with coefficients in $\{-1, 0, +1\}$, h columns each, and β rows in total. The rows of A (respectively, B) correspond to the days when $P_j \neq P_{j-1}$ (respectively, $P_j = P_{j-1}$). Furthermore. the column vectors $x = (X_1, \ldots, X_h)^\top$ consistent with Π and $P_0, \ldots, P_\beta$ are exactly those that satisfy $Ax > 0$ and $Bx = 0$. The matrices A and B can be computed in time $O(h\beta T)$, where T is an upper bound on the time to compute a single $\mathcal{S}_j^i$ from $P_0, \ldots, P_\beta$ over all $j \in [1, \beta]$ and $\mathcal{S}^i$.*

Proof: Follows immediately from Equations 3 and 4. ∎

Lemma 2 *In the AS+FI model with parameters m and α, given a system of linear inequalities $Ax > 0, Bx = 0$, where A and B have coefficients in $\{-1, 0, +1\}$ with h columns each, and β rows in total, there exist (1) a set Π of h strategies corresponding to the h columns of A and B, and (2) a $(\beta + 1)$-day price sequence $P_0, \ldots, P_\beta$ with the latter β days corresponding to the β rows of A and B. Furthermore, the values of the population variables $X_1, \ldots, X_n$ are feasible with respect to the price movement on day j if and only if column vector $x = (X_1, \ldots, X_n)^\top$ satisfies the j-th constraint in A and B. Also, $P_0, \ldots, P_\beta$ and a description of Π can be computed in $O(h\beta)$ time.*

Proof: Follows immediately from Equations 3 and 4. ∎

In the AS+PI model we obtain only equations, of the form:

$$\sum_{i=1}^{h} \mathcal{S}_t^i X_i = \frac{1}{\alpha}(P_t - P_{t-1}). \tag{5}$$

In this case there is a direct correspondence between market data and systems of linear equations. We formalize this correspondence in Lemmas 3 and 4 below.

Lemma 3 *In the AS+PI model with parameters m and α, given Π and a price sequence $P_0, \ldots, P_\beta$, there is a matrix B with coefficients in $\{-1, 0, +1\}$, h columns, and β rows, and a column vector b of length h, such that the column vectors $x = (X_1, \ldots, X_h)^\top$ consistent with Π and $P_0, \ldots, P_\beta$ are exactly those that satisfy $Bx = b$. The coefficients of B and b can be computed in time $O(h\beta T)$, where T is an upper bound on the time to compute a single $\mathcal{S}^i_j$ from $P_0, \ldots, P_\beta$ over all $j \in [1, \beta]$ and $\mathcal{S}^i$.*

Proof: Follows immediately from Equation 5. ∎

Lemma 4 *In the AS+PI model with parameters m and α, given a system of linear equations $Bx = b$, where B is a $\beta \times h$ matrix with coefficients in $\{-1, 0, +1\}$, there exist (1) a set Π of h strategies corresponding to the h columns of B, and (2) a $(\beta + 1)$-day price sequence $P_0, \ldots, P_\beta$ with the last β days corresponding to the β rows of B. Furthermore, the values of the population variables $X_1, \ldots, X_n$ are feasible with respect to the price movement on day j if and only if column vector $x = (X_1, \ldots, X_n)^\top$ satisfies the j-th constraint in B. Also, $P_0, \ldots, P_\beta$ and a description of Π can be computed in $O(h\beta)$ time.*

Proof: Follows immediately from Equation 5. ∎

4.2 An Easy Case for Market Prediction: Many Traders but Few Strategies

In Section 4.2.1, we show that if an AS+FI market has far more traders than strategies, then it takes polynomial time to estimate the probability that the next day's price will rise. In Section 4.2.2, we discuss why the same analysis technique does not work for an AS+PI market.

4.2.1 Predicting an AS+FI Market

For the sake of emphasizing the dependence on m, let $\Pr_m[E]$ be the probability that event E occurs when there are m traders in the market.

This section makes the following assumptions:

E1 The input to the market prediction problem is simply a price history $P_0, \ldots, P_{t-1}$. The output is $\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$.

E2 The market follows the AS+FI model.

E3 Π is fixed. The values $\mathcal{S}^i_j$ over all $i \in [1, h]$ are computable from the input in total time polynomial in j.

E4 Each of the m traders independently chooses a random strategy $\mathcal{S}^i$ from Π with fixed probability $p_i > 0$, where $p_1 + \cdots + p_h = 1$.

The parameter α is irrelevant.

Notice that the column vector $X = (X_1, \ldots, X_h)^\top$ is the sum of m independent identically-distributed vector-valued random variables with a center at $p = m{\cdot}(p_1, \ldots, p_h)^\top$. We recenter and rescale X to $Y = (X - m{\cdot}(p1, \ldots, p_h)^\top)/\sqrt{m}$. Then, by the Central Limit Theorem (see, e.g., [3, Theorem 29.5]), as $m \to +\infty$, Y converges weakly to a normal distribution centered at the h-dimensional vector $(0, \ldots, 0)^\top$. In Theorem 5 below, we rely on this fact to estimate the probability that the market rises for price histories that occur with nonzero probability.

Theorem 5 *Assume that* $\lim_{m\to\infty} \Pr_m[P_0, \ldots, P_{t-1}] > 0$. *Then there is a fully polynomial-time approximation scheme for estimating* $\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$ *from* $P_0, \ldots, P_{t-1}$. *The time complexity of the scheme is polynomial in* (1) *the length* t *of the price history,* (2) *the inverse of the relative error bound* ϵ, *and* (3) *the inverse of the failure probability* η.

Remark. We omit the explicit dependency of the running time in h and $p_1, \ldots, p_h$ in order to concentrate on the main point that market prediction is easy with this section's four assumptions. The parameters h and $p_1, \ldots, p_h$ are constant under the assumptions.

Proof: We use Lemma 1 to convert the price history $P_0, \ldots, P_{t-1}$ and the strategy set Π into a system of linear constraints $AX > 0$ and $BX = 0$, with the next day's price change $P_t - P_{t-1}$ determined by $\mathrm{sign}(c{\cdot}X')$ for some c. Since the values $\mathcal{S}^i_j$ are computable in time polynomial in j, this conversion takes time polynomial in t.

Then, $\Pr_m[P_0, \ldots, P_{t-1}] = \Pr_m[AX > 0 \wedge BX = 0]$. Since $\lim_{m\to\infty} \Pr_m[AX > 0 \wedge BX = 0] > 0$, the constraints in B must be vacuous; in other words, for each $P_i = 0$ with $i \in [0, t-1]$, the corresponding constraint in B is $0{\cdot}X_1 + \cdots + 0{\cdot}X_h = 0$. Therefore, $\Pr_m[P_0, \ldots, P_{t-1}] = \Pr_m[AX > 0]$. Furthermore, since both A and c are constant with respect to m,

$$\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}] = \frac{\lim_{m\to\infty} \Pr_m[AX > 0 \wedge c{\cdot}X > 0]}{\lim_{m\to\infty} \Pr_m[AX > 0]}. \tag{6}$$

So to compute the desired $\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$, we compute $\lim_{m\to\infty} \Pr_m[AX > 0 \wedge c{\cdot}X > 0]$ and $\lim_{m\to\infty} \Pr_m[AX > 0]$ as follows.

To avoid the degeneracy caused by $\sum_{i=1}^h X_i = m$, we work with $X' = (X_1, \ldots, X_{h-1})^\top$ instead of X by replacing X_h with $m - \sum_{i=1}^{h-1} X_i$ and making related changes. Let $p' = (p_1, \ldots, p_{h-1})^\top$, which is the center of X'. As is true for Y, as $m \to +\infty$, the vector $Y' = (X' - m{\cdot}p')/\sqrt{m}$ converges weakly to a normal distribution centered at the $(h-1)$-dimensional point $(0, \ldots, 0)^\top$. Under the assumption that each p_i is nonzero, the distribution of Y' is full-dimensional (within its restricted $(h-1)$-dimensional space), as in the limit the variance of each coordinate Y'_i is nonzero conditioned on the values of the other coordinates, which implies that the smallest subspace containing the distribution must contain all $h-1$ axes. We can calculate the covariance matrix of Y' directly from the p_i, as it is equal to the covariance matrix for a single trader: on the diagonal, $C_{ii} = p_i - p_i^2$; and for off-diagonal elements, $C_{ij} = -p_i p_j$. Given C, Y' has density $\rho(x) = ae^{x^\top Cx}$ for some constant a, and we can evaluate this density in $O(h^2)$ time given x, which is $O(1)$ time under our assumption that Π is fixed.

Let A_i be the i-th constraint of A, i.e., $A_{i,1}X_1 + \cdots + A_{i,h}X_h > 0$. Let A'_i denote the constraint $(A_{i,1} - A_{i,h}, \ldots, A_{i,h-1} - A_{i,h})$. Let $c' = (c_1 - c_h, \ldots, c_{h-1} - c_h)$.

We next convert the constraints of A on X into constraints on Y'. First of all, notice that $A_iX = \sqrt{m}{\cdot}(A'_iY') + m{\cdot}A_ip$. So $A_iX > 0$ if and only if $A'_iY' > -\sqrt{m}{\cdot}A_ip$. The term $-\sqrt{m}{\cdot}A_ip$ may not be constant. In such a case, as $m \to \infty$, the hyper plane bounding the half space $A'_iY' > -\sqrt{m}{\cdot}A_ip$ keeps moving away from the origin, which presents some technical complication. To remove this problem, we analyze the term in three cases. If $A_ip < 0$, then since $m{\cdot}p$ is the center of X, as $m \to \infty$, $\Pr_m[A_iX < 0]$ converges to 1. In other words, A_i is infeasible with probability 1 in the limit. Then, since $\lim_{m\to\infty} \Pr_m[P_0, \ldots, P_{t-1}] > 0$, such A_i cannot exist in A. Similarly, if $A_ip > 0$, then $\lim_{m\to\infty} \Pr_m[A_iX > 0] = 1$ and A_i is vacuous. The interesting constraints are those for which $A_ip = 0$; in this case, by algebra, $A_iX > 0$ if and only if $A'_iY' > 0$. Thus, let D be the matrix formed by these constraints; D can be computed in $O(ht)$ time. Then, since D is constant with respect to m, $\lim_{m\to\infty} \Pr_m[AX > 0] = \lim_{m\to\infty} \Pr_m[DY' > 0]$. Similarly, $\Pr_m[AX > 0 \wedge c{\cdot}X > 0]$ converges to (1) 0, (2) $\Pr_m[DY' > 0]$, or (3) $\Pr_m[DY' > 0 \wedge c'{\cdot}Y' > 0]$ for case (1) $c{\cdot}p < 0$, case (2) $c{\cdot}p > 0$, or case (3) $c{\cdot}p = 0$, respectively.

Therefore, by Equation 6, $\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$ equals 0 for case (1) and equals 1 for case (2). Case (3) requires further computation.

$$\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}] = \frac{\lim_{m\to\infty} \Pr_m[DY' > 0 \wedge c' \cdot Y' > 0]}{\lim_{m\to\infty} \Pr_m[DY' > 0]}. \tag{7}$$

The numerator and denominator of the ratio in Equation 7 are both integrals of the distribution of Y' in the limit over the bodies of possibly infinite convex polytopes. To deal with the possible infiniteness of the convex bodies $DY' > 0 \wedge c' \cdot Y' > 0$ and $DY' > 0$, notice that the density drops exponentially. So we can truncate the regions of integration to some finite radius around the $(h-1)$-dimensional origin $(0, \ldots, 0)^\top$ with only exponentially small loss of precision. Finally, since the distribution of Y' in the limit is normal, by applying the Applegate-Kannan integration algorithm for log-concave distributions [2] to the numerator and denominator separately, we can approximate $\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$ within the desired time complexity. ∎

4.2.2 Remarks on Predicting an AS+PI Market

The probability estimation technique based on taking m to ∞ does not appear to be applicable to the AS+PI model for the following reasons.

First of all, by Lemma 3, the input price history induces a system of linear equations $BX = b$. If any equation in $BX = b$ is not equivalent to $X_1 + \cdots + X_h = m$ or $0 \cdot X_1 + \cdots + 0 \cdot X_h = 0$, then $\lim_{m\to\infty} \Pr_m[P_0, \ldots, P_{t-1}] = 0$.

A natural attempt to overcome this seemingly technical difficulty would be to (1) solve $BX = b$ to choose a maximal set U of independent variables X_i and (2) evaluate $\Pr_m[P_0, \ldots, P_{t-1}]$ in the probability space induced by this set. Still, a single constraint such as $B_{i,1} \cdot X_1 + \cdots + B_{i,h} \cdot X_h = \alpha \cdot m_0$ with $B_{i,j} \geq 0$ for all $j \in [1, h]$ and $B_{i,j'} > 0$ for some $X_{j'} \in U$ forces $\lim_{m\to\infty} \Pr_m[P_0, \ldots, P_{t-1}] = 0$ in the new probability space. This is due to the fact that m_0 is constant with respect to m.

A further attempt would be to evaluate $\lim_{m\to\infty} \Pr_m[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$ by directly working with the probability space induced by $P_0, \ldots, P_{t-1}$. This also does not work because we show below that the market prediction problem can be reduced to the case where taking a limit in m has no effect on the distribution of the strategy counts. Suppose that we are given a market which follows the assumptions E1, E3, and E4 of Section 4.2.1 except that this market uses the PI rule and has m_0 traders. We construct a new market with any $m \geq m_0$ traders with the following modifications:

1. The price history $P_0, \ldots, P_{t-1}$ is extended with an extra day into $P'_0, \ldots, P'_{t-1}, P'_t$, where $P'_j = P_j$ for $0 \leq j \leq t-1$. Each strategy $\mathcal{S}_i$ is extended into a new strategy $\mathcal{S}'_i$ where (1) on day $j \in [1, t-1]$, $\mathcal{S}'_i(P_0, \ldots, P_{j-1}) = \mathcal{S}_i(P_0, \ldots, P_{j-1})$, (2) on day t, $\mathcal{S}'_i$ always buys, and (3) on day $t+1$, $\mathcal{S}'_i(P'_0, \ldots, P'_t) = \mathcal{S}_i(P_0, \ldots, P_{t-1})$. Thus, $P'_t = P'_{t-1} + \alpha \cdot m_0$.

2. Add a passive strategy $\mathcal{S}'_{h+1}$ that always holds.

3. Let $p'_i = \frac{1}{2} p_i$ for $1 \leq i \leq h$ and $p'_{h+1} = \frac{1}{2}$.

Note that since $P'_t - P'_{t-1} = \alpha \cdot m_0$, $m - m_0$ traders choose the passive strategy $\mathcal{S}_{h+1}$. Also, the new market and the new price history can accommodate any $m \geq m_0$ traders. Note that because of the constraint $P'_t - P'_{t-1} = \alpha \cdot m_0$, the probability distribution of $(X_1, \cdots, X_h)^\top$ conditioned on $P'_0, \ldots, P'_t$ in the new market for each $m \geq m_0$ is identical to the probability distribution of $(X_1, \cdots, X_h)^\top$ conditioned on $P_0, \ldots, P_{t-1}$ in the original market with $m = m_0$. Furthermore,

$\Pr_m[P'_{t+1} > P'_t \mid P'_0, \ldots, P'_t] = \Pr_{m_0}[P_t > P_{t-1} \mid P_0, \ldots, P_{t-1}]$. So we have obtained the desired reduction.

Consequently, we are left with a situation where the number of active strategies may be comparable to the number of traders. Such a market turns out to be very hard to predict, as shown next in Section 4.3.

4.3 A Hard Case for Market Prediction: Many Strategies

Section 4.2 shows that predicting an AS+FI market is easy (i.e., takes polynomial time) when the number m of traders vastly exceeds the number h of strategies. In this section, we consider the case where every trader may have a distinct strategy, and show that predicting an AS+FI or AS+PI market becomes very hard indeed.

We now define two decision-problem versions of market prediction. Both versions make the following assumption:

- Each X_i is independently either 0 or 1 with equal probability.

The *bounded* market prediction problem is:

- Input: a set of n passive strategies and a price history spanning n days such that the probability that the market rises on day $n+1$ conditioned on the price history is either (1) greater than 2/3 or (2) less than 1/3.
- Question: Which case is it, case (1) or case (2)?

The output of bounded market prediction is not defined when the input does not yield a bounded probability of a rise or fall on the next day. Bounded market prediction is thus an example of a *promise problem* [13, 14], defined as a pair of predicates (Q, R) where Q, the *promise*, specifies which inputs are permitted, and R specifies which inputs in Q are contained in the language.

The *unbounded* market prediction problem is:

- Input: a set of n passive strategies and a price history spanning n days.
- Question: Is the probability that the market rises on day $n+1$ conditioned on the price history greater than 1/2 (without the usual ϵ term)?

The unbounded market prediction problem has less financial payoff than the bounded one due to different probability thresholds. For each of these two problems, there are in effect two versions, depending on which price increment rule is used; however, both versions turn out to be equally hard. These two problems can be analyzed by similar techniques, and our discussion below focuses on the bounded market prediction problem with a hardness theorem for the unbounded market prediction problem in Section 4.3.5.

We show in Section 4.3.1 how to construct passive strategies and price histories such that solving bounded market prediction is equivalent to estimating the probability that a Boolean circuit outputs 1 on a random input conditioned on a second circuit outputting 1. In Section 4.3.2, we show that this problem is hard for $\mathrm{P}^{\mathrm{NP}[O(\log n)]}$ and complete for a class that lies between $\mathrm{P}^{\mathrm{NP}[O(\log n)]}$ and PP. Thus bounded market prediction is not merely NP-hard, but cannot be solved in the polynomial-time hierarchy at all unless the hierarchy collapses to a finite level.

4.3.1 Reductions from Circuits to Markets

Lemma 6 converts a circuit into a system of linear inequalities, while Lemma 7 converts a system of linear inequalities into a system of linear equations. These systems can then be converted into AS+FI and AS+PI market models using Lemmas 2 and 4, respectively.

Note that the restriction in Lemma 6 to circuits consisting of 2-input NOR gates is not an obstacle to representing arbitrary combinatorial circuits (with constant blow-up), as 2-input NOR gates are universal.

Lemma 6 *For any n-input Boolean circuit C consisting of m 2-input NOR gates, there exists a system $Ax > 0$ of $3m+2$ linear constraints in $n+m+2$ unknowns and a length $n+m+2$ column vector c with the following properties:*

1. *Both A and c have coefficients in $\{-1, 0, +1\}$ that can be computed in time $O((n+m)^2)$.*
2. *Any 0-1 vector $(x_1, \ldots, x_n)$ has a unique 0-1 extension $x = (x_1, \ldots, x_n, x_{n+1}, \ldots x_{n+m+2})$ satisfying $Ax > 0$.*
3. *If $Ax > 0$, then $cx > 0$ if and only if $C(x_1, x_2, \ldots, x_n) = 1$.*

Proof: Let x_{n+k} represent the output of the k-th NOR gate, where $1 \leq k \leq m$. Without loss of generality we assume that gate m is the output gate.

The variables x_{n+m+1} and x_{n+m+2} are dummies to allow for a zero right-hand-side in $Ax > 0$; our first two constraints are $x_{n+m+1} > 0$ and $x_{n+m+2} > 0$.

Suppose gate k has inputs x_i and x_j. The NOR operation is implemented by the following three linear inequalities:

$$\begin{array}{ccccccc} x_i & & & + & x_{n+k} & < & 2; \\ & & x_j & + & x_{n+k} & < & 2; \\ x_i & + & x_j & + & x_{n+k} & > & 0. \end{array}$$

The first two constraints ensure that the output is never 1 if an input is 1, while the last requires that the output is 1 if both inputs are 0; the constraints are thus satisfied if and only if $x_{n+k} = \neg(x_i \vee x_j)$. Using the dummy variables, the first two constraints are written as

$$\begin{array}{ccccccccc} -x_i & & - & x_{n+k} & + & x_{n+m+1} & + & x_{n+m+2} & > & 0; \\ & -x_j & - & x_{n+k} & + & x_{n+m+1} & + & x_{n+m+2} & > & 0. \end{array}$$

Let $Ax > 0$ be the system obtained by combining all of these inequalities. Then for each $(x_1, \ldots, x_n)$, $Ax > 0$ determines x_{n+k} for all $k \geq 1$. The vector c is chosen so that $cx = x_{n+m}$. ∎

One might suspect that the fixed increment rule's ability to hide the exact values of the left-hand side of each constraint is critical to disguise the inner workings of the circuit. However, by adding slack variables we can translate the inequalities into equations, allowing the use of a proportional increment rule without revealing extra information.

Lemma 7 *Let $Ax > 0$ be a system of m linear inequalities in n variables where A has coefficients in $\{-1, 0, +1\}$. Then there is a system $By = 1$ of $mn - m + 1$ linear equations in $2mn - 3m + n + 1$ variables with the following properties:*

1. *B has coefficients in $\{-1, 0, +1\}$ that can be computed in time $O((mn)^2)$.*
2. *There is a bijection $f : x \mapsto y$ between the 0-1 solutions x to $Ax > 0$ and the 0-1 solutions y to $By = 1$, such that $x_j = y_j$ for $1 \leq j \leq n$ whenever $y = f(x)$.*

Proof: For each $1 \leq i \leq m$, let A_i be the constraint $\sum_j A_{ij} x_j > 0$. To turn these inequalities into equations, we add slack variables to soak up any excess over 1, with some additional care taken to ensure that there is a unique assignment to the slack variables for each setting of the variables x_j.

We will use the following 0-1 variables, which we think of as alternate names for y_1 through $y_{2mn-3m+n+1}$:

Variables	Purpose	Indices	Count
x_j	original variables	$1 \leq j \leq n$	n
u	constant 1	none	1
s_{ij}	slack variables for A_i	$1 \leq i \leq m, 1 \leq j \leq n-1$	$m(n-1)$
t_{ij}	slack variables for $s_{ij} \geq s_{i,j+1}$	$1 \leq i \leq m, 1 \leq j \leq n-2$	$m(n-2)$

Name	Equation	Purpose	Indices	Count
U	$u = 1$	set u	none	1
B_i	$\sum_j A_{ij} x_j - \sum_j s_{ij} = 1$	represent A_i	$1 \leq i \leq m$	m
S_{ij}	$s_{ij} - s_{i,j+1} - t_{ij} + u = 1$	require $s_{ij} \geq s_{i,j+1}$	$1 \leq i \leq m$, $1 \leq j \leq n-2$	$m(n-2)$

Observe that for each i, $\sum_j s_{ij}$ can take on any integer value σ_i between 0 and $n-1$, and that for any fixed value of σ_i, the S_{ij} constraints uniquely determine the values of s_{ij} and t_{ij} for all j. So each constraint B_i permits $\chi_i = \sum_j A_{ij} x_j$ to take on precisely the same values 1 to n that A_i does, and each χ_i uniquely determines σ_i and thus the assignment of all s_{ij} and t_{ij}. ∎

4.3.2 Conditional Probability Complexity Classes

Suppose that we take a polynomial-time probabilistic Turing machine, fix its inputs, and use the usual Cook's Theorem construction to turn it into a circuit whose inputs are the random bits used during its computation. Then, we can feed the resulting circuit to Lemmas 6 and 2 to obtain an AS+FI market model in which there is exactly one assignment of population variables for each set of random bits, and the price rises on the last day if and only if the output of the Turing machine is 1. By applying Lemma 7 to the intermediate system of linear inequalities, we can similarly convert a circuit to an AS+PI model. It follows that bounded market prediction is BPP-hard for either model. But with some cleverness, we can exploit the conditioning on past history to show that bounded market prediction is in fact much harder than this. We do so in Section 4.3.4, after a brief detour through computational complexity in this section.

We proceed to define some new counting classes based on conditional probabilities. One of these, BCPP, has the useful feature that bounded market prediction solves all problems in BCPP, and is complete for the "promise problem" version of BCPP, which we will write as promise-BCPP and which we define in Section 4.3.3. We will use this fact to relate the complexity of bounded market prediction to more traditional complexity classes.

The usual counting classes of complexity theory (PP, BPP, R, ZPP, $C_=$, etc.) are defined in terms of counting the relative numbers of accepting and rejecting states of a nondeterministic Turing machine. We will define a new family of counting classes by adding a third decision state that does not count for the purposes of determining acceptance or rejection.

A *noncommittal* Turing machine is a nondeterministic Turing machine with three decision states: *accept*, *reject*, and *abstain*. We represent a noncommittal Turing machine as a deterministic Turing machine which takes a polynomial number of random bits in addition to its input; each assignment of the random bits gives a distinct computation path. A computation path is accepting/rejecting/abstaining if it ends in an accept/reject/abstain state, respectively. We often write 1, 0, or $\perp$ as shorthand for the output of an accepting, rejecting, or abstaining path.

Conditional versions of the usual counting classes are obtained by carrying over their definitions from standard nondeterministic Turing machines to noncommittal Turing machines, with some care in handling the case of no accepting or rejecting paths. We can still think of these modified classes as corresponding to probabilistic machines, but now the probabilities we are interested in are conditioned on not abstaining.

Definition 8 The *conditional probabilistic polynomial-time* class (CPP) consists of those languages L for which there exists a polynomial-time noncommittal Turing machine M such that $x \in L$ if and only if the number of accepting paths when M is run with input x exceeds the number of rejecting paths.

Definition 9 The *bounded conditional probabilistic polynomial-time* class (BCPP) consists of those languages L for which there exists a constant $\epsilon > 0$ and a polynomial-time noncommittal Turing machine M such that (1) $x \in L$ implies that a fraction of at least $\frac{1}{2} + \epsilon$ of the total number of accepting and rejecting paths are accepting and (2) $x \notin L$ implies that a fraction of at least $\frac{1}{2} + \epsilon$ of the total number of accepting and rejecting paths are rejecting.

Definition 10 The *conditional randomized polynomial-time* class (CR) consists of those languages L for which there exists a constant $\epsilon > 0$ and a polynomial-time noncommittal Turing machine M such that (1) $x \in L$ implies that a fraction of at least ϵ of the total number of accepting and rejecting paths are accepting, and (2) $x \notin L$ implies that there are no accepting paths.

As we show in Theorems 11 and 12, CPP and CR turn out to be the same as the unconditional classes PP and NP, respectively.

Theorem 11 CPP = PP.

Proof: First of all, PP $\subseteq$ CPP because a PP machine is a CPP machine that happens not to have any abstaining paths. For the inverse direction, represent each abstaining path of a CPP machine by a pair consisting of one accepting and one rejecting path, and each accepting or rejecting path by two accepting or rejecting paths. Then the resulting PP machine accepts if and only if the CPP machine does. ∎

Theorem 12 CR = NP.

Proof: To show NP $\subseteq$ CR, replace each rejecting path of an NP machine with an abstaining path in a CR machine. For the inverse direction, replace each abstaining path of the CR machine with a rejecting path in the NP machine. ∎

The class BCPP is more obscure; it is equivalent to the threshold version of BPP, $\mathrm{BPP}_{\mathrm{path}}$ [17].[1]. The class $\mathrm{BPP}_{\mathrm{path}}$ is defined as the class of all languages accepted by a *threshold machine* with threshold $\frac{1}{2} + \epsilon$ for some $\epsilon > 0$, where a threshold machine accepts or rejects if at least a fixed proportion of its computation paths accept or reject, with each computation path counted as one without regard to its probability.

[1]We are grateful to Lance Fortnow[16] for pointing out this equivalence

Theorem 13 $\mathrm{BCPP} = \mathrm{BPP}_{\mathrm{path}}$.

Proof: To show $\mathrm{BCPP} \subseteq \mathrm{BPP}_{\mathrm{path}}$, replace each abstaining path with one accepting and one rejecting path. To show $\mathrm{BPP}_{\mathrm{path}} \subseteq \mathrm{BCPP}$, we must normalize the $\mathrm{BPP}_{\mathrm{path}}$ computation so that all paths include the same number of branches. Suppose that in some $\mathrm{BPP}_{\mathrm{path}}$ computation, the number of branches on any path is bounded by some polynomial $T(n)$. Extend each path in the $\mathrm{BPP}_{\mathrm{path}}$ machine with $k < T(n)$ branches into $2^{T(n)-k}$ paths in the BCPP machine, of which all but one are abstaining and the remaining path accepts or rejects depending on the output of the corresponding $\mathrm{BPP}_{\mathrm{path}}$ path. ∎

$\mathrm{BCPP} = \mathrm{BPP}_{\mathrm{path}}$ is a much stronger class than the analogous non-conditional class BPP. For example, if one takes a NP machine and replaces each accepting path with exponentially many accepting paths and each rejecting path with an equally large family of abstaining paths sprinkled with a single rejecting path, the result is a BCPP machine that accepts the same language as the NP machine. By repeating this sort of amplification of "good" paths, BCPP can in fact simulate $O(\log n)$ queries of an NP-oracle. Because of the equivalence of BCPP and $\mathrm{BPP}_{\mathrm{path}}$, we can show this formally by using similar results for $\mathrm{BPP}_{\mathrm{path}}$ from [17].

Corollary 14 $\mathrm{P}^{NP[O(\log n)]} \subseteq \mathrm{BCPP} \subseteq \mathrm{PP}$.

Proof: The first inclusion is immediate from Theorem 13 and the fact that $\mathrm{P}^{\mathrm{NP}[O(\log n)]} \subseteq \mathrm{BPP}_{\mathrm{path}}$, shown in Corollary 3.4 in [17]. The second inclusion follows from Theorem 13 and the observation that $\mathrm{BPP}_{\mathrm{path}} \subseteq \mathrm{PP}_{\mathrm{path}} = \mathrm{PP}$, also from [17]. ∎

An interesting open question is where exactly $\mathrm{BCPP} = \mathrm{BPP}_{\mathrm{path}}$ lies between $\mathrm{P}^{\mathrm{NP}[O(\log n)]}$ and PP. It is conceivable that by cleverly exploiting the power of conditioning to amplify low-probability events one could show $\mathrm{BCPP} = \mathrm{PP}$. However, we will content ourselves with the much easier observation that the usual amplification technique for BPP also applies to BCPP; as with other results in this section, this observation follows from the equivalence of BCPP and $\mathrm{BPP}_{\mathrm{path}}$.

Corollary 15 *If $L \in \mathrm{BCPP}$, then there exists a noncommittal Turing machine M such that the probability that M accepts conditioned on not abstaining is at least $1 - f(n)$ if $x \in L$ and at most $f(n)$ if $x \notin L$, where $n = |x|$ and $f(n)$ is any function of the form $2^{-O(n^c)}$ for some constant $c > 0$.*

Proof: Immediate from Theorem 13 and Theorem 3.1 of [17]. ∎

4.3.3 Promise Problems and Promise-BCPP

Part of the motivation for defining BCPP and CPP was to identify exactly the complexity of solving bounded and unbounded market prediction. Unfortunately, while we can show that bounded market prediction is hard for BCPP, in the sense that any problem in BCPP reduces to bounded market prediction, it is not clear that bounded market prediction is actually *contained* in BCPP.

The reason is that the definition of BCPP does not allow excluding bad inputs. Though we don't care what our BCPP machine does when given an instance of market prediction in which the next day's price movement is not predictable, the definition of the class still requires that the machine produce more than $\frac{1}{2} + \epsilon$ accepting or rejecting paths. The natural solution to bounded market prediction using a noncommittal machine does not have this property, and it is not clear that we can guarantee it in general. Instead, we define a promise-problem version of BCPP, and show (in Section 4.3.4) that bounded market prediction is complete for this class.

Definition 16 The class promise-BCPP consists of all pairs of predicates (Q, R) for which there exists a constant $\epsilon > 0$ and a polynomial-time noncommittal Turing machine M such that for all $x \in Q$, (1) $x \in R$ implies that a fraction of at least $\frac{1}{2} + \epsilon$ of the total number of accepting and rejecting paths are accepting and (2) $x \notin R$ implies that a fraction of at least $\frac{1}{2} + \epsilon$ of the total number of accepting and rejecting paths are rejecting.

A pair of predicates (Q, R), in which Q specifies which inputs are valid and R specifies which valid inputs should be accepted, is called a *promise problem* [13, 14]. Polynomial-time reductions, as defined for languages, have a natural analog for promise problems: (Q, R) is polynomial-time reducible to (Q', R') if and only if there is a polynomial-time function f such that (a) $f(Q) \subseteq f(Q')$, and (b) for all $x \in Q$, $f(x) \in R'$ if and only if $x \in R$.[2] Similarly, a particular promise problem is *hard* for a class of such problems if every problem in the class reduces to it in polynomial-time, and that it is *complete* for a class if it is both hard for the class and contained in the class.

There is also a natural correspondence between promise problems and standard languages. A *solution* to a promise problem (Q, R) is a language L for which L and R agree on inputs in Q; in this way promise problems can be turned into languages. In the other direction, any standard language L can be through of as a promise problem (true, L).

With this correspondence, we can easily see that BCPP $=$ BPP_{path} is contained in promise-BCPP, in the sense that for any L in BCPP, (true, L) is in promise-BCPP; and that promise-BCPP is in turn contained in CPP, in the sense that any problem (Q, R) in promise-BCPP has a solution in CPP(we can just run the noncommittal machine that accepts (Q, R)). We will abuse notation slightly by writing $\text{BPP}_{\text{path}} \subseteq$ promise-BCPP $\subseteq$ CPP, eliding the implicit conversions between languages and promise problems.

4.3.4 Bounded Market Prediction is Promise-BCPP-Complete

In Section 4.3.2, we have defined the complexity class BCPP and observed that it is equal to BPP_{path}, which implies that it contains the powerful class $\text{P}^{\text{NP}[O(\log n)]}$. In this section, we show that solving bounded market prediction solves all problems in BCPP.

In a sense, this result says that market prediction is a universal prediction problem: if we can predict a market, we can predict any event conditioned on past history as long as we can sample from an underlying discrete probability space whose size is at most exponential.

It also says that bounded market prediction is very hard. That is, using Corollaries 15 and 14, even if the next day's price is determined with all but an exponentially small probability, it cannot be solved in the polynomial-time hierarchy unless the hierarchy collapses to a finite level.

Theorem 17 *The bounded market prediction problem is complete for* promise-BCPP, *in either the AS+FI or the AS+PI model.*

Proof: First we show that bounded market prediction is a member of promise-BCPP. Given a market, construct a noncommittal Turing machine M whose input is the price history and strategies, and whose random inputs supply the settings for the population variables X_i. Let M abstain if the price history is inconsistent with the input and population variables; depending on the model, this is either a matter of checking the linear inequalities produced by Lemma 1 or the equations produced by Lemma 3. Otherwise, M accepts if the market rises and rejects if the market falls on the next day. The probability that M accepts thus equals the probability that the

[2]There are many ways to define more complicated reductions involving promise problems; a detailed discussion of this issue can be found in [5].

market rises: either more than 2/3 or less than 1/3. Since the problem is to distinguish between these two cases, M solves the problem within the definition of promise-BCPP.

In the other direction, we will show how to reduce from any promise-BCPP-language L to bounded market prediction. Suppose (Q, R) is accepted by some BCPP-machine M for all $x \in Q$. We will translate M and its input x into a bounded market prediction problem. First use Corollary 15 to amplify the conditional probability that M accepts to either more than 2/3 or less than 1/3 as bounded market prediction demands. Then convert M into two polynomial-size circuits, one computing

$$C_{\not\perp}(r) = \begin{cases} 0 & \text{if } M(x,r) = \perp; \\ 1 & \text{if } M(x,r) \neq \perp, \end{cases}$$

and the other computing

$$C_1(r) = \begin{cases} 0 & \text{if } M(x,r) \neq 1; \\ 1 & \text{if } M(x,r) = 1. \end{cases}$$

Without loss of generality we may assume that $C_{\not\perp}$ and C_1 are built from NOR gates. Applying Lemma 6 to each yields two sets of constraints $A_{\not\perp}y > 0$ and $A_1 y > 0$ and column vectors $c_{\not\perp}$ and c_1 such that $c_{\not\perp}y > 0$ if and only if $C_{\not\perp}y = 1$ and $c_1 x > 0$ if and only if $C_1(x) = 1$, where y satisfies the previous linear constraints and x is the initial prefix of y consisting of variables not introduced by the construction of Lemma 6. We also have from Lemma 6 that there is a one-to-one correspondence between assignments of x and assignments of y satisfying the A constraints, so probabilities are not affected by this transformation.

Now use Lemma 2 to construct a market model in which $A_{\not\perp}y > 0$, $A_1 y > 0$, and $c_{\not\perp}y > 0$ are enforced by the strategies and price history, and $\text{sign}(c_1 y)$ determines the price change on the next day of trading. Thus the consistent settings of the variables X_i are precisely those corresponding to settings of r for which $C_{\not\perp}(r) = 1$, or, in other words, those yielding computation paths that do not abstain. The market rises when $C_1(r) = 1$, or when M accepts. So if we can predict whether the market rises or falls with conditional probability at least 2/3, we can predict the likely output of M. It follows that bounded market prediction for the AS+FI model is promise-BCPP-hard.

To show the similar result for the AS+PI model, use Lemma 7 to convert the constraints $A_{\not\perp}y > 0$, $A_1 y > 0$ into a system of linear equations $Bz = 1$, and then proceed as before, using Lemma 4 to convert this system to a price history and letting $c_1 z$ determine the price change (and thus the sign of the price change) on the next day of trading. ∎

4.3.5 Unbounded Market Prediction is CPP-Complete

The unbounded market prediction problem seems harder because the probability threshold in question is $\frac{1}{2}$ with no ϵ bound in contrast to the thresholds $\frac{2}{3}$ and $\frac{1}{3}$ for the bounded market prediction problem. The following theorem reflects this intuition. However, since we do not know whether BCPP is distinct from PP, we do not know whether unbounded prediction is strictly harder.

Theorem 18 *The unbounded market prediction problem is complete for* CPP = PP*, in either the AS+FI or the AS+PI model.*

Proof: Similar to the proof of Theorem 17. ∎

5 Future Research Directions

There are many problems left open in this paper. Below we briefly discuss some general directions for further research.

We have reported a number of simulation and theoretical results for the AS model. As for empirical analysis, it would be of interest to fit actual market data to the model. We can then use the estimated parameters to (1) test whether the model has any predicative power and (2) test the effectiveness of new or known trading algorithms. This direction may require carefully choosing "realistic" strategies for Π. Besides the momentum and contrarian strategies, there are some popular ones which are worth considering, such as those based on support levels. Investment newsletters could be a useful source of such strategies.

The AS model is an idealized one. We have chosen such simplicity as a matter of research methodology. It is relatively easy to design highly complicated models which can generate very complex market behavior. A more challenging and interesting task is to design the simplest possible model which can generate the desired market characteristics. For instance, a significant research direction would be to find the simplest model in which market prediction is computationally hard. On the other hand, it would be of great interest to find the most general models in which market prediction takes only polynomial time. For this goal, we can consider injecting more realism into the model by introducing resource-bounded learning (the generality of Π is equivalent to unbounded learning), variable memory size, transaction costs, buying power, limit orders, short sell, options, etc.

Acknowledgments

We would like to thank Lance Fortnow for pointing out the equivalence of BCPP and BPP_{path}, and Lane Hemaspaandra for explaining the mechanics of promise problems.

This work originated with David Fischer's senior project in 1999, advised by Ming-Yang Kao. David would like to thank his father and role model, Professor Michael Fischer, for teaching, mentoring, and inspiring him throughout college.

References

[1] P. B. Andreassen and S. Krause. Judgemental extrapolation and the salience of change. *Journal of Forecasting*, 9(4):347–372, 1990.

[2] D. Applegate and R. Kannan. Sampling and integration of near log-concave functions. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 156–163, 1991.

[3] P. Billingsley. *Probability and Measure.* John Wiley and Sons, second edition, 1986.

[4] F. Black. Noise. *Journal of Finance*, 41(3):529–543, 1986.

[5] J.-Y. Cai, L. A. Hemachandra, and J. Vyskoč. Promises and fault-tolerant database access. In K. Ambos-Spies, S. Homer, and U. Schöning, editors, *Complexity Theory: Current Research*, pages 101–146. Cambridge University Press, 1993.

[6] J. Y. Campbell, A. W. Lo, and A. C. MacKinlay. *The Econometrics of Financial Markets.* Princeton University Press, Princeton, NJ, 1997.

[7] R. G. Clarke and M. Statman. Bullish or bearish? *Financial Analysts Journal*, May/June:63–72, 1998.

[8] P. A. Cootner. *The Random Character of Stock Market Prices.* MIT Press, Cambridge, MA, 1964.

[9] R. H. Day and W. H. Huang. Bulls, bears, and market sheep. *Journal of Economic Behavior and Organization*, 14:299–330, 1990.

[10] W. DeBondt. What do economists know about the stock market? *Journal of Portfolio Management*, Winter:84–91, 1991.

[11] W. DeBondt. Betting on trends: Intuitive forecasts of financial risk and return. *International Journal of Forecasting*, 9:355–371, 1993.

[12] J. B. DeLong, A. Shleifer, L. H. Summers, and R. J. Waldmann. Noise trader risk in financial markets. *Journal of Political Economy*, 98(4):703–738, 1990.

[13] S. Even, A. Selman, and Y. Yacobi. The complexity of promise problems with applications to public-key cryptography. *Information and Control*, 61(2):159–173, 1984.

[14] S. Even and Y. Yacobi. Cryptocomplexity and NP-completeness. In *Proceedings of the 7th International Colloquium on Automata, Languages, and Programming*, pages 195–207. Springer-Verlag, 1980.

[15] E. Fama. Effieicnt capital markets: II. *Journal of Finance*, 46:1575–1618, 1991.

[16] L. Fortnow. Private communication, Mar. 2001.

[17] Y. Han, L. A. Hemaspaandra, and T. Thierauf. Threshold computation and cryptographic security. *SIAM J. Comput.*, 26(1):59–78, 1997.

[18] I. Karatzas and S. E. Shreve. *Methods of Mathematical Finance*, volume 39 of *Applications of Mathematics.* Springer Verlag, New York, NY, 1998.

[19] A. Kumar. Behavior of momentum following and contrarian market timers. Working Paper 99-01, International Center for Finance, School of Management, Yale University, Jan. 1999.

[20] A. W. Lo and A. C. MacKinlay. *A Non-Random Walk Down the Wall Street.* Princeton University Press, Princeton, NJ, 1999.

[21] T. Lux. The socio-economic dynamics of speculative markets: Interacting agents, chaos, and the fat tails of return distributions. *Journal of Economic Behavior and Organization*, 33:143–165, 1998.

[22] B. G. Malkiel. *A Random Walk down Wall Street.* Norton, New York, NY, 1990.

[23] B. B. Mandelbrot. *Fractals and scaling in finance.* Springer-Verlag, New York, 1997. Discontinuity, concentration, risk, Selecta Volume E, with a foreword by R. E. Gomory.

[24] S. Mullainathan. A memory based model of bounded rationality. Working paper, Department of Economics, MIT, Apr. 1998.

[25] R. Sullivan, A. Timmermann, and H. White. Data-snooping, technical trading rules and the bootstrap. *Journal of Finance*, 54:1647–1692, 1999.