

# Stock Market Trading via Stochastic Network Optimization

Michael J. Neely

University of Southern California

<http://www-rcf.usc.edu/~mjneely>

**Abstract**—We consider the problem of dynamic buying and selling of shares from a collection of  $N$  stocks with random price fluctuations. To limit investment risk, we place an upper bound on the total number of shares kept at any time. Assuming that prices evolve according to an ergodic process with a mild decaying memory property, and assuming constraints on the total number of shares that can be bought and sold at any time, we develop a trading policy that comes arbitrarily close to achieving the profit of an ideal policy that has perfect knowledge of future events. Proximity to the optimal profit comes with a corresponding tradeoff in the maximum required stock level and in the timescales associated with convergence. We then consider arbitrary (possibly non-ergodic) price processes, and show that the same algorithm comes close to the profit of a frame based policy that can look a fixed number of slots into the future. Our approach uses a Lyapunov optimization technique previously developed for optimizing stochastic queueing networks.

**Index Terms**—Queueing analysis, stochastic control, universal algorithms

## I. INTRODUCTION

This paper considers the problem of stock trading in an economic market with  $N$  stocks. We treat the problem in discrete time with normalized time slots  $t \in \{0, 1, 2, \dots\}$ , where buying and selling transactions are conducted on each slot. Let  $\mathbf{Q}(t) = (Q_1(t), \dots, Q_N(t))$  be a vector of the current number of shares owned of each stock, called the *stock queue*. That is, for each  $n \in \{1, \dots, N\}$ , the value of  $Q_n(t)$  is an integer that represents the number of shares of stock  $n$ . Stock prices are given by a vector  $\mathbf{p}(t) = (p_1(t), \dots, p_N(t))$  and are assumed to evolve randomly, with mild assumptions to be made precise in later sections. Each buy and sell transaction incurs trading costs. Stocks can be sold and purchased on every slot. Let  $\phi(t)$  represent the net profit on slot  $t$  (after transaction costs are paid). The goal is to design a trading policy that maximizes the long term time average of  $\phi(t)$ .

For this system model, we enforce the additional constraint that at most  $\mu_n^{max}$  shares of each stock  $n$  can be bought and sold on a given slot. This ensures that our trading decisions only gradually change the portfolio allocation. While this  $\mu_n^{max}$  constraint can significantly limit the ability to take advantage of desirable prices, and hence limits the maximum possible long term profit, we show that it can also reduce investment risk. Specifically, subject to the  $\mu_n^{max}$  constraint, we develop an algorithm that achieves a time average profit

that is arbitrarily close to optimal, with a tradeoff in the maximum number of shares  $Q_n^{max}$  required for stock  $n$ . The  $Q_n^{max}$  values can be chosen as desired to limit the losses from a potential collapse of one or more of the stocks. It also impacts the timescales over which profit is accumulated, where smaller  $Q_n^{max}$  levels lead to faster convergence times.

It is important to note that long term wealth typically grows *exponentially* when the  $Q_n^{max}$  and  $\mu_n^{max}$  constraints are removed. In contrast, it can be shown that these  $Q_n^{max}$  and  $\mu_n^{max}$  constraints restrict wealth to at most a *linear growth*. Therefore, using  $Q_n^{max}$  and  $\mu_n^{max}$  to limit investment risk unfortunately has a dramatic impact on the long term growth curve. However, our ability to bound the timescales over which wealth is earned suggests that our strategy may be useful in cases when, in addition to a good long-term return, we also desire noticeable and consistent short-term gains. At the end of this paper, we briefly describe a modified strategy that increases  $Q_n^{max}$  and  $\mu_n^{max}$  as wealth progresses, with the goal of achieving noticeable short-term gains while enabling exponential wealth increase.

Our approach uses the Lyapunov optimization theory developed for stochastic queueing networks in our previous work [2][3][4]. Specifically, the work [2][3][4] develops resource allocation and scheduling policies for communication and queueing networks with random traffic and channels. The policies can maximize time average throughput-utility and minimize time average power expenditure, as well as optimize more general time average attributes, without a-priori knowledge of the traffic and channel probabilities. The algorithms continuously adapt to emerging conditions, and are robust to non-ergodic changes in the probability distributions [5]. This suggests that similar control techniques can be used successfully for stock trading problems. The difference is that the queues associated with stock shares are controlled to have *positive drift* (pushing them towards the maximum queue size), rather than *negative drift* (which would push them in the direction of the empty state).

The *Dynamic Trading Algorithm* that we develop from these techniques can be intuitively viewed as a variation on a theme of *dollar cost averaging*, where price downturns are exploited by purchasing more stock. However, the actual amount of stock that we buy and sell on each slot is determined by a constrained optimization of a max-weight functional that incorporates transaction costs, current prices, and current stock queue levels.

Much prior work on financial analysis and portfolio opti-

This material is supported in part by one or more of the following: the DARPA IT-MANET program grant W911NF-07-0028, the NSF Career grant CCF-0747525.

mization assumes a known probability model for stock prices. Classical portfolio optimization techniques by Markowitz [6] and Sharpe [7] construct portfolio allocations over  $N$  stocks to maximize profit subject to variance constraints (which model risk) over one investment period (see also [8] and references therein). Solutions to this problem can be calculated if the mean and covariance of stock returns are known. Samuelson considers multi-period problems in [9] using dynamic programming, assuming a known product form distribution for investment returns. Cover in [10] develops an iterative procedure that converges to the constant portfolio allocation that maximizes the expected log investment return, assuming a known probability distribution that is the same on each period. Recent work by Rudoy and Rohrs in [11] [12] considers risk-aware optimization with a more complex *cointegrated vector autoregressive* assumption on stock processes, and uses Monte Carlo simulations over historical stock trajectories to inform stochastic decisions. Stochastic models of stock prices using Lévy processes and multi-fractal processes are considered in [8] [13] [14] and references therein.

A significant departure from this work is the *universal stock trading paradigm*, as exemplified in prior works of Cover and Gluss [15], Larson [16], Cover [17], Merhav and Feder [18], and Cover and Ordentlich [19] [20], where trading algorithms are developed and shown to provide analytical guarantees for *any sample path* of stock prices. Specifically, the work in [15]-[20] seeks to find a non-anticipating trading algorithm that yields the same growth exponent as the best constant portfolio allocation, where the constant can be optimized with full knowledge of the future. The works in [15][16] develop algorithms that come close to the optimal exponent, and the work in [17] *achieves* the optimal exponent under mild assumptions that bound price sample paths away from zero. Similar results are derived in [18] using a general framework of sequential decision theory. Related results are derived in [19] [20], where [19] treats problems with additional “side information,” and [20] treats max-min performance when stock prices are chosen by an adversary.

Our work is similar in spirit to this universal trading paradigm, in that we do not base decisions on a known (or estimated) probability distribution. However, our context, solution methodology, and solution structure is very different. Indeed, the works in [15]-[20] assume that the entire stock portfolio can be sold and reallocated on every time period, and allow stock holdings to grow arbitrarily large. This means that the accumulated profit is always at risk of one or more stock failures. In our work, we take a more conservative approach that restricts reallocation to gradual changes, and that pockets profits while holding no more than  $Q_n^{max}$  shares of each stock  $n$ . We also explicitly account for trading costs and integer constraints on stock shares, which is not considered in the works [15]-[20]. In this context, we first design an algorithm under the assumption that prices are ergodic with an unknown distribution. In this case, we develop a simple non-anticipating algorithm that comes arbitrarily close to the optimal time average profit that could be earned by an ideal policy with complete knowledge of the future. The ideal policy used for comparison can make different allocations at different times,

and is not restricted to constant allocations.

We then show that the *same* algorithm can be used for general price sample paths, even non-ergodic sample paths without well defined time averages. A more conservative guarantee is shown in this case: The algorithm yields profit that is arbitrarily close to that of a frame based policy with “ $T$ -slot lookahead,” where the future is known up to  $T$  slots. Our approach is different from prior work on universal algorithms and is inspired by Lyapunov optimization and decision theory for stochastic queueing networks [2]. However, the Lyapunov theory we use here involves sample path techniques that are different from those in [2]. These techniques might have broader impacts on queueing problems in other areas.

In the next section we present the system model. In Section III we develop the Dynamic Trading Algorithm and analyze performance for the simple (and possibly unrealistic) case when price vectors  $\mathbf{p}(t)$  are ergodic and i.i.d. over slots. While this i.i.d. case does *not* accurately model actual stock prices, its analysis provides valuable insight. Section IV shows the algorithm also provides performance guarantees for completely arbitrary price processes (possibly non-ergodic). Simple extensions are described in Section V.

## II. SYSTEM MODEL

Let  $\mathbf{A}(t) = (A_1(t), \dots, A_N(t))$  be a vector of decision variables representing the number of new shares purchased for each stock on slot  $t$ , and let  $\boldsymbol{\mu}(t) = (\mu_1(t), \dots, \mu_N(t))$  be a vector representing the number of shares sold on slot  $t$ . The values  $A_n(t)$  and  $\mu_n(t)$  are non-negative integers for each  $n \in \{1, \dots, N\}$ . Each purchase of  $A$  new shares of stock  $n$  incurs a transaction cost  $b_n(A)$  (called the *buying cost function*). Likewise, each sale of  $\mu$  shares of stock  $n$  incurs a transaction cost  $s_n(\mu)$  (called the *selling cost function*). The functions  $b_n(A)$  and  $s_n(\mu)$  are arbitrary, and are assumed only to satisfy  $b_n(0) = s_n(0) = 0$ , and to be non-negative, non-decreasing, and bounded by finite constants  $b_n^{max}$  and  $s_n^{max}$ , so that:

$$\begin{aligned} 0 \leq b_n(A) &\leq b_n^{max} && \text{for } 0 \leq A \leq \mu_n^{max} \\ 0 \leq s_n(\mu) &\leq s_n^{max} && \text{for } 0 \leq \mu \leq \mu_n^{max} \end{aligned}$$

where for each  $n \in \{1, \dots, N\}$ ,  $\mu_n^{max}$  is a positive integer that limits the amount of shares of stock  $n$  that can be bought and sold on slot  $t$ .

### A. Example Transaction Cost Functions

The functions  $b_n(A)$  might be *linear*, representing a transaction fee that charges per share purchased. Another example is a *fixed cost model* with some fixed positive fee  $b_n$ , so that:

$$b_n(A) = \begin{cases} b_n & \text{if } A > 0 \\ 0 & \text{if } A = 0 \end{cases}$$

Similar models can be used for the  $s_n(\mu)$  function. The simplest model of all is the *zero transaction cost model* where the functions  $b_n(A)$  and  $s_n(\mu)$  are identically zero.

## B. System Dynamics

The stock price vector  $\mathbf{p}(t)$  is assumed to be a random vector process that takes values in some finite set  $\mathcal{P} \subset \mathbb{R}^N$ , where  $\mathcal{P}$  can have an arbitrarily large number of elements.<sup>1</sup> For each  $n$ , let  $p_n^{max}$  represent a bound on  $p_n(t)$ , so that:

$$0 \leq p_n(t) \leq p_n^{max} \quad \text{for all } t \text{ and all } n \in \{1, \dots, N\} \quad (1)$$

We assume that buying and selling decisions can be made on each slot  $t$  based on knowledge of  $\mathbf{p}(t)$ . The selling decision variables  $\boldsymbol{\mu}(t)$  are made every slot  $t$  subject to the following constraints:

$$\mu_n(t) \in \{0, 1, \dots, \mu_n^{max}\} \quad \text{for all } n \in \{1, \dots, N\} \quad (2)$$

$$\mu_n(t)p_n(t) \geq s_n(\mu_n(t)) \quad \text{for all } n \in \{1, \dots, N\} \quad (3)$$

$$\mu_n(t) \leq Q_n(t) \quad \text{for all } n \in \{1, \dots, N\} \quad (4)$$

Constraint (2) ensures that no more than  $\mu_n^{max}$  shares can be sold of stock  $n$  on a single slot. Constraint (3) restricts to the reasonable case when the money earned from the sale of a stock must be larger than the transaction fee associated with the sale (violating this constraint would clearly be sub-optimal).<sup>2</sup> Constraint (4) requires the number of shares sold to be less than or equal to the current number owned.

The buying decision variables  $\mathbf{A}(t)$  are constrained as follows:

$$A_n(t) \in \{0, 1, 2, \dots, \mu_n^{max}\} \quad \text{for all } n \in \{1, \dots, N\} \quad (5)$$

$$\sum_{n=1}^N A_n(t)p_n(t) \leq x \quad (6)$$

where  $x$  is a positive value that bounds the total amount of money used for purchases on slot  $t$ . For simplicity, we assume there is always at least a minimum of  $x$  and  $\sum_{n=1}^N [\mu_n^{max} p_n^{max} + b_n(\mu_n^{max})]$  dollars available for making purchasing decisions. This model can be augmented by adding a *checking account queue*  $Q_0(t)$  from which we must draw money to make purchases, although we omit this aspect for brevity.

The resulting queueing dynamics for the stock queues  $Q_n(t)$  for  $n \in \{1, \dots, N\}$  are thus:

$$Q_n(t+1) = \max[Q_n(t) - \mu_n(t) + A_n(t), 0] \quad (7)$$

Strictly speaking, the  $\max[\cdot, 0]$  operator in the above dynamic equation is redundant, because the constraint (4) ensures that the argument inside the  $\max[\cdot, 0]$  operator is non-negative. However, the  $\max[\cdot, 0]$  shall be useful for mathematical analysis when we compare our strategy to that of a queue-independent strategy that neglects constraint (4).

<sup>1</sup>The cardinality of the set  $\mathcal{P}$  does not enter into our analysis. We assume it is finite only for the convenience of claiming that the supremum time average profit  $\phi^{opt}$  is achievable by a single “ $p$ -only” policy, as described in Section II-D. Theorems 1, 2 are unchanged if the set  $\mathcal{P}$  is infinite, although the proofs of Theorem 1 would require an additional limiting argument over  $p$ -only policies that approach  $\phi^{opt}$ .

<sup>2</sup>Constraint (3) can be augmented by allowing equality only if  $\mu_n(t) = 0$ .

## C. The Maximum Profit Objective

Define  $\phi(t)$  as the net profit on slot  $t$ :

$$\begin{aligned} \phi(t) \triangleq & \sum_{n=1}^N [\mu_n(t)p_n(t) - s_n(\mu_n(t))] \\ & - \sum_{n=1}^N [A_n(t)p_n(t) + b_n(A_n(t))] \end{aligned} \quad (8)$$

Define  $\bar{\phi}$  as the time average expected value of  $\phi(t)$  under a given trading algorithm (temporarily assumed to have a well defined limit):

$$\bar{\phi} \triangleq \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E} \{ \phi(\tau) \}$$

The goal is to design a trading policy that maximizes  $\bar{\phi}$ . It is clear that the trivial strategy that chooses  $\boldsymbol{\mu}(t) = \mathbf{A}(t) = \mathbf{0}$  for all  $t$  yields  $\phi(t) = 0$  for all  $t$ , and results in  $\bar{\phi} = 0$ . Therefore, we desire our algorithm to produce a long term profit that satisfies  $\bar{\phi} > 0$ .

## D. The Stochastic Price Vector and $p$ -only Policies

We first assume the stochastic process  $\mathbf{p}(t)$  has well defined time averages (this is generalized to non-ergodic models in Section IV). Specifically, for each price vector  $\mathbf{p}$  in the finite set  $\mathcal{P}$ , we define  $\pi(\mathbf{p})$  as the time average fraction of time that  $\mathbf{p}(t) = \mathbf{p}$ , so that:

$$\lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} 1\{\mathbf{p}(\tau) = \mathbf{p}\} = \pi(\mathbf{p}) \quad \text{with probability 1} \quad (9)$$

where  $1\{\mathbf{p}(\tau) = \mathbf{p}\}$  is an indicator function that is 1 if  $\mathbf{p}(\tau) = \mathbf{p}$ , and zero otherwise.

Define a  $p$ -only policy as a buying and selling strategy that chooses *virtual* decision vectors  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$  as a stationary and possibly randomized function of  $\mathbf{p}(t)$ , constrained only by (2)-(3) and (5)-(6). That is, the virtual decision vectors  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$  associated with a  $p$ -only policy do not necessarily satisfy the constraint (4) that is required of the *actual* decision vectors, and hence these decisions can be made independently of the current stock queue levels.

Under a given  $p$ -only policy, define the following time average expectations  $d_n^*$  and  $\phi^*$ :

$$d_n^* \triangleq \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E} \{ A_n^*(\tau) - \mu_n^*(\tau) \} \quad (10)$$

$$\begin{aligned} \phi^* \triangleq & \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E} \left\{ \sum_{n=1}^N [\mu_n^*(\tau)p_n(\tau) - s_n(\mu_n^*(\tau))] \right. \\ & \left. - \sum_{n=1}^N [A_n^*(\tau)p_n(\tau) + b_n(A_n^*(\tau))] \right\} \end{aligned} \quad (11)$$

It is easy to see by (9) that these time averages are well defined for any  $p$ -only policy. For each  $n$ , the value  $d_n^*$  represents the *virtual drift* of stock queue  $Q_n(t)$  associated with the virtual decisions  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$ . The value  $\phi^*$  represents the *virtual profit* under virtual decisions  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$ . Note that the

trivial  $p$ -only policy  $\mathbf{A}^*(t) = \boldsymbol{\mu}^*(t) = \mathbf{0}$  yields  $d_n^* = 0$  for all  $n$ , and  $\phi^* = 0$ . Thus, we can define  $\phi^{opt}$  as the supremum value of  $\phi^*$  over all  $p$ -only policies that yield  $d_n^* \geq 0$  for all  $n$ , and we note that  $\phi^{opt} \geq 0$ . Using an argument similar to that given in [3], it can be shown that:

- 1)  $\phi^{opt}$  is *achievable* by a single  $p$ -only policy that satisfies  $d_n^* = 0$  for all  $n \in \{1, \dots, N\}$ .
- 2)  $\phi^{opt}$  is greater than or equal to the supremum of the limsup time average expectation of  $\phi(t)$  that can be achieved over the class of all *actual* policies that satisfy the constraints (2)-(6), including ideal policies that use perfect information about the future. Thus, no policy can do better than  $\phi^{opt}$ .

That  $\phi^{opt}$  is achievable by a single  $p$ -only policy (rather than by a limit of an infinite sequence of policies) can be shown using the assumption that the set  $\mathcal{P}$  of all price vectors is finite. That  $\phi^{opt}$  bounds the time average profit of *all* policies, including those that have perfect knowledge of the future, can be intuitively understood by noting that the optimal profit is determined only by the time averages  $\pi(\mathbf{p})$ . These time averages are the same (with probability 1) regardless of whether or not we know the future. The detailed proofs of these results are similar to those in [3] and are provided in [1]. In the next section we develop a *Dynamic Trading Algorithm* that satisfies the constraints (2)-(6) and that does not know the future or the distribution  $\pi(\mathbf{p})$ , yet yields time average profit that is arbitrarily close to  $\phi^{opt}$ .

To develop our Dynamic Trading Algorithm, we first focus on the simple case when the vector  $\mathbf{p}(t)$  is independent and identically distributed (i.i.d.) over slots, with a general probability distribution  $\pi(\mathbf{p})$ . This is an overly simplified model and does *not* reflect actual stock time series data. Indeed, a more accurate model would be to assume the differences in the logarithm of prices are i.i.d. (see [8] and references therein). However, we show in Section IV that the *same* algorithm is also efficient for arbitrary (possibly non-ergodic) price models.

### E. The i.i.d. Model

Suppose  $\mathbf{p}(t)$  is i.i.d. over slots with  $Pr[\mathbf{p}(t) = \mathbf{p}] = \pi(\mathbf{p})$  for all  $\mathbf{p} \in \mathcal{P}$ . Because the value  $\phi^{opt}$  is achievable by a single  $p$ -only policy, and because the expected values of any  $p$ -only policy are the same every slot under the i.i.d. model, we have the following: There is a  $p$ -only policy  $\mathbf{A}^*(t)$ ,  $\boldsymbol{\mu}^*(t)$  that yields for all  $t$  and all  $\mathbf{Q}(t)$ :

$$\mathbb{E}\{A_n^*(t) - \mu_n^*(t) \mid \mathbf{Q}(t)\} = 0 \quad (12)$$

and

$$\mathbb{E}\left\{\sum_{n=1}^N [\mu_n^*(t)p_n(t) - s_n(\mu_n^*(t))] - \sum_{n=1}^N [A_n^*(t)p_n(t) + b_n(A_n^*(t))] \mid \mathbf{Q}(t)\right\} = \phi^{opt} \quad (13)$$

## III. CONSTRUCTING A DYNAMIC TRADING ALGORITHM

The goal is to ensure that all stock queues  $Q_n(t)$  are maintained at reasonably high levels so that there are typically enough shares available to sell if an opportune price should arise. To this end, define  $\theta_1, \dots, \theta_n$  as positive real numbers

that represent target queue sizes for the stock queues (soon to be related to the maximum queue size). The particular values  $\theta_1, \dots, \theta_n$  shall be chosen later. As a scalar measure of the distance each queue is away from its target value, we define the following *Lyapunov function*  $L(\mathbf{Q}(t))$ :

$$L(\mathbf{Q}(t)) \triangleq \frac{1}{2} \sum_{n=1}^N (Q_n(t) - \theta_n)^2 \quad (14)$$

Suppose that  $\mathbf{Q}(t)$  evolves according to some probability law, and define  $\Delta(\mathbf{Q}(t))$  as the *one-slot conditional Lyapunov drift*:<sup>3</sup>

$$\Delta(\mathbf{Q}(t)) \triangleq \mathbb{E}\{L(\mathbf{Q}(t+1)) - L(\mathbf{Q}(t)) \mid \mathbf{Q}(t)\} \quad (15)$$

As in the stochastic network optimization problems of [2][3][4], our approach is to take control actions on each slot  $t$  to minimize a bound on the “drift-plus-penalty” expression:

$$\Delta(\mathbf{Q}(t)) - V\mathbb{E}\{\phi(t) \mid \mathbf{Q}(t)\}$$

where  $V$  is a positive parameter to be chosen as desired to affect the proximity to the optimal time average profit  $\phi^{opt}$ . To this end, we first compute a bound on the Lyapunov drift.

*Lemma 1:* (Lyapunov drift bound) For all  $t$  and all possible values of  $\mathbf{Q}(t)$ , we have:

$$\Delta(\mathbf{Q}(t)) \leq B - \sum_{n=1}^N (Q_n(t) - \theta_n)\mathbb{E}\{\mu_n(t) - A_n(t) \mid \mathbf{Q}(t)\}$$

where  $B$  is a finite constant that satisfies:

$$B \geq \frac{1}{2} \sum_{n=1}^N \mathbb{E}\{(\mu_n(t) - A_n(t))^2 \mid \mathbf{Q}(t)\} \quad (16)$$

Such a finite constant  $B$  exists because of the boundedness assumptions on buy and sell variables  $\mu_n(t)$  and  $A_n(t)$ . In particular, we have:

$$B \leq \frac{1}{2} \sum_{n=1}^N (\mu_n^{max})^2 \triangleq D \quad (17)$$

*Proof:* The proof follows by squaring the queue dynamics (7) (see [1]).  $\square$

Using Lemma 1 with the definition of  $\phi(t)$  in (8), a bound on the drift-minus-reward expression is given as follows:

$$\begin{aligned} \Delta(\mathbf{Q}(t)) - V\mathbb{E}\{\phi(t) \mid \mathbf{Q}(t)\} &\leq B \\ &- \sum_{n=1}^N (Q_n(t) - \theta_n)\mathbb{E}\{\mu_n(t) - A_n(t) \mid \mathbf{Q}(t)\} \\ &- V \sum_{n=1}^N \mathbb{E}\{\mu_n(t)p_n(t) - s_n(\mu_n(t)) \mid \mathbf{Q}(t)\} \\ &+ V \sum_{n=1}^N \mathbb{E}\{A_n(t)p_n(t) + b_n(A_n(t)) \mid \mathbf{Q}(t)\} \end{aligned} \quad (18)$$

We desire an algorithm that, every slot, observes the  $\mathbf{Q}(t)$  values and the current prices, and makes a greedy trading action subject to the constraints (2)-(6) that minimizes the right-hand-side of (18).

<sup>3</sup>Strictly speaking, proper notation is  $\Delta(\mathbf{Q}(t), t)$ , as the drift may arise from a non-stationary algorithm. However, we use the simpler notation  $\Delta(\mathbf{Q}(t))$  as a formal representation of the right-hand-side of (15).

### A. The Dynamic Trading Algorithm

Every slot  $t$ , observe  $\mathbf{Q}(t)$  and  $\mathbf{p}(t)$  and perform the following actions.

1) *Selling*: For each  $n \in \{1, \dots, N\}$ , choose  $\mu_n(t)$  to solve:

$$\begin{aligned} \text{Minimize: } & [\theta_n - Q_n(t) - Vp_n(t)]\mu_n(t) + Vs_n(\mu_n(t)) \\ \text{Subject to: } & \text{Constraints (2)-(4)} \end{aligned}$$

2) *Buying*: Choose  $\mathbf{A}(t) = (A_1(t), \dots, A_N(t))$  to solve:

$$\begin{aligned} \text{Minimize: } & \sum_{n=1}^N [Q_n(t) - \theta_n + Vp_n(t)]A_n(t) \\ & + \sum_{n=1}^N Vb_n(A_n(t)) \\ \text{Subject to: } & \text{Constraints (5)-(6)} \end{aligned}$$

The buying algorithm uses the integer constraints (5)-(6), and is related to the well known *bounded knapsack problem* (it is exactly the bounded knapsack problem if the  $b_n(\cdot)$  functions are linear). Implementation of this integer constrained problem can be complex when the number of stocks  $N$  is large. However, if we use  $x \triangleq \sum_{n=1}^N [\mu_n^{max} p_n^{max} + b_n(\mu_n^{max})]$ , then constraint (6) is effectively removed. In this case, the stocks are decoupled and the buying algorithm reduces to making separate decisions for each stock  $n$ . Alternatively, the constraint (6) can be replaced by the constraint  $\sum_{n=1}^N A_n(t) \leq A_{tot}$ , where  $A_{tot}$  is an integer that bounds the total number of stocks that can be bought on a single slot. In this case, it is easy to see that if buying costs are linear, so that  $b_n(A) = b_n A$  for all  $n$  (for some positive constants  $b_n$ ), then the buying algorithm reduces to successively buying as much stock as possible from the queues with the smallest (and negative)  $[Q_n(t) - \theta_n + V(p_n(t) + b_n)]$  values. An alternative relaxation of the constraint (6) is discussed in [1].

*Lemma 2*: For a given  $\mathbf{Q}(t)$  on slot  $t$ , the above dynamic trading algorithm satisfies:

$$\begin{aligned} B - V\phi(t) - \sum_{n=1}^N (Q_n(t) - \theta_n)(\mu_n(t) - A_n(t)) &\leq \\ B - V\phi^*(t) - \sum_{n=1}^N (Q_n(t) - \theta_n)(\mu_n^*(t) - A_n^*(t)) &\quad (19) \end{aligned}$$

where  $\mathbf{A}(t)$ ,  $\boldsymbol{\mu}(t)$  are the actual decisions made by the algorithm, which define  $\phi(t)$  by (8), and  $\mathbf{A}^*(t)$ ,  $\boldsymbol{\mu}^*(t)$  are any alternative (possibly randomized) decisions that can be made on slot  $t$  that satisfy (2)-(6), which define  $\phi^*(t)$  by (8). Furthermore, we have:

$$\begin{aligned} \Delta(\mathbf{Q}(t)) - V\mathbb{E}\{\phi(t) | \mathbf{Q}(t)\} &\leq B \\ - \sum_{n=1}^N (Q_n(t) - \theta_n)\mathbb{E}\{\mu_n^*(t) - A_n^*(t) | \mathbf{Q}(t)\} & \\ - V \sum_{n=1}^N \mathbb{E}\{\mu_n^*(t)p_n(t) - s_n(\mu_n^*(t)) | \mathbf{Q}(t)\} & \\ + V \sum_{n=1}^N \mathbb{E}\{A_n^*(t)p_n(t) + b_n(A_n^*(t)) | \mathbf{Q}(t)\} &\quad (20) \end{aligned}$$

where the expectation on the right-hand-side of (20) is with respect to the random price vector  $\mathbf{p}(t)$  and the possibly random actions  $\mathbf{A}^*(t)$ ,  $\boldsymbol{\mu}^*(t)$  in response to this price vector.

*Proof*: Given  $\mathbf{Q}(t)$  on slot  $t$ , the dynamic algorithm makes buying and selling decisions to minimize the left-hand-side of (19) over all alternative decisions that satisfy (2)-(6). Therefore, the inequality (19) holds for all realizations of the random

quantities, and hence also holds when taking conditional expectations of both sides. The conditional expectation of the left-hand-side of (19) is equivalent to the right-hand-side of the drift-minus-reward expression (18), which proves (20).  $\square$

The main idea behind our analysis is that the Dynamic Trading Algorithm is simple to implement and does not require knowledge of the future or of the statistics of the price process  $\mathbf{p}(t)$ . However, it can be *compared* to alternative policies  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$  (such as in Lemma 2), and these policies possibly have knowledge both of the price statistics and of the future.

### B. Bounding the Stock Queues

The next lemma shows that the above algorithm does not sell any shares of stock  $n$  if  $Q_n(t)$  is sufficiently small.

*Lemma 3*: Under the above Dynamic Trading Algorithm and for arbitrary price processes  $\mathbf{p}(t)$  that satisfy (1), if  $Q_n(t) < \theta_n - Vp_n^{max}$  for some particular queue  $n$  and slot  $t$ , then  $\mu_n(t) = 0$ . Therefore, if  $Q_n(0) \geq \theta_n - Vp_n^{max} - \mu_n^{max}$ , then:

$$Q_n(t) \geq \theta_n - Vp_n^{max} - \mu_n^{max} \quad \text{for all } t$$

*Proof*: Suppose that  $Q_n(t) < \theta_n - Vp_n^{max}$  for some particular queue  $n$  and slot  $t$ . Then for any  $\mu \geq 0$  we have:

$$\begin{aligned} [\theta_n - Q_n(t) - Vp_n(t)]\mu + Vs_n(\mu) & \\ \geq [\theta_n - Q_n(t) - Vp_n^{max}]\mu + Vs_n(\mu) & \\ \geq [\theta_n - Q_n(t) - Vp_n^{max}]\mu & \\ \geq 0 & \end{aligned}$$

where the final inequality holds with equality if and only if  $\mu = 0$ . Therefore, the Dynamic Trading Algorithm must choose  $\mu_n(t) = 0$ .

Now suppose that  $Q_n(t) \geq \theta_n - Vp_n^{max} - \mu_n^{max}$  for some time  $t$ . We show it also holds for  $t+1$ . If  $Q_n(t) \geq \theta_n - Vp_n^{max}$ , then it can decrease by at most  $\mu_n^{max}$  on a single slot, so that  $Q_n(t+1) \geq \theta_n - Vp_n^{max} - \mu_n^{max}$ . Conversely, if  $\theta_n - Vp_n^{max} > Q_n(t) \geq \theta_n - Vp_n^{max} - \mu_n^{max}$ , then we know  $\mu_n(t) = 0$  and so the queue cannot decrease on the next slot and we again have  $Q_n(t+1) \geq \theta_n - Vp_n^{max} - \mu_n^{max}$ . It follows that this inequality is always upheld if it is satisfied at  $t = 0$ .  $\square$

We note that the above lemma is a *sample path* statement that holds for arbitrary (possibly non-ergodic) price processes. The next lemma also deals with sample paths, and shows that all queues have a finite maximum size  $Q_n^{max}$ .

*Lemma 4*: Under the above Dynamic Trading Algorithm and for arbitrary price processes  $\mathbf{p}(t)$  that satisfy (1), if  $Q_n(t) > \theta_n$  for some particular queue  $n$  and slot  $t$ , then  $A_n(t) = 0$  and so the queue cannot increase on the next slot. It follows that if  $Q_n(0) \leq \theta_n + \mu_n^{max}$ , then:

$$Q_n(t) \leq \theta_n + \mu_n^{max} \quad \text{for all } t$$

*Proof*: Suppose that  $Q_n(t) > \theta_n$  for a particular queue  $n$  and slot  $t$ . Let  $\mathbf{A}(t) = (A_1(t), \dots, A_N(t))$  be a vector of buying decisions that solve the optimization associated with the Buying algorithm on slot  $t$ , so that they minimize the expression:

$$\sum_{m=1}^N [Q_m(t) - \theta_m + Vp_m(t)]A_m(t) + \sum_{m=1}^N Vb_m(A_m(t)) \quad (21)$$

subject to (5)-(6). Suppose that  $A_n(t) > 0$  (we shall reach a contradiction). Because the term  $[Q_n(t) - \theta_n + Vp_n(t)]$  is *strictly positive*, and because the  $b_n(A)$  function is non-decreasing, we can strictly reduce the value of the expression (21) by changing  $A_n(t)$  to 0. This change still satisfies the constraints (5)-(6) and produces a strictly smaller sum in (21), contradicting the assumption that  $\mathbf{A}(t)$  is a minimizer. Thus, if  $Q_n(t) > \theta_n$ , then  $A_n(t) = 0$ .

Because the queue value can increase by at most  $\mu_n^{max}$  on any slot, and cannot increase if it already exceeds  $\theta_n$ , it follows that  $Q_n(t) \leq \theta_n + \mu_n^{max}$  for all  $t$ , provided that this inequality holds at  $t = 0$ .  $\square$

### C. Analyzing Time Average Profit

*Theorem 1:* Fix any value  $V > 0$ , and define  $\theta_n$  as follows:

$$\theta_n \triangleq Vp_n^{max} + 2\mu_n^{max} \quad (22)$$

Suppose that initial stock queues satisfy:

$$\mu_n^{max} \leq Q_n(0) \leq Vp_n^{max} + 3\mu_n^{max} \quad (23)$$

If the Dynamic Trading Algorithm is implemented over  $t \in \{0, 1, 2, \dots\}$ , then:

(a) Stock queues  $Q_n(t)$  (for  $n \in \{1, \dots, N\}$ ) are deterministically bounded for all slots  $t$  as follows:

$$\mu_n^{max} \leq Q_n(t) \leq Vp_n^{max} + 3\mu_n^{max} \quad \text{for all } n \text{ and all } t \quad (24)$$

(b) If  $\mathbf{p}(t)$  is i.i.d. over slots with general distribution  $Pr[\mathbf{p}(t) = \mathbf{p}] = \pi(\mathbf{p})$  for all  $\mathbf{p} \in \mathcal{P}$ , then for all  $t \in \{1, 2, \dots\}$  we have:

$$\bar{\phi}(t) \geq \phi^{opt} - \frac{B}{V} - \frac{\mathbb{E}\{L(\mathbf{Q}(0))\}}{Vt} \quad (25)$$

where the constant  $B$  is defined by (16) (and satisfies the inequality (17)),  $\phi^{opt}$  is the optimal time average profit, and  $\bar{\phi}(t)$  is the time average expected profit over  $t$  slots:

$$\bar{\phi}(t) \triangleq \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{\phi(\tau)\} \quad (26)$$

Therefore:

$$\liminf_{t \rightarrow \infty} \bar{\phi}(t) \geq \phi^{opt} - B/V \quad (27)$$

Theorem 1 shows that the time average expected profit is within  $B/V$  of the optimal value  $\phi^{opt}$ . Because the  $B$  constant is independent of  $V$ , we can choose  $V$  to make  $B/V$  arbitrarily small. This comes with a tradeoff in the maximum size required for each stock queue that is linear in  $V$ . Specifically, the maximum stock level  $Q_n^{max}$  required for stock  $n$  is given as follows:

$$Q_n^{max} \triangleq Vp_n^{max} + 3\mu_n^{max}$$

Now suppose that we start with initial condition  $Q_n(0) = \mu_n^{max}$  for all  $n$  and all  $t$ . Then for  $t \in \{1, 2, \dots\}$  the error term  $L(\mathbf{Q}(0))/(Vt)$  is given by:

$$\frac{L(\mathbf{Q}(0))}{Vt} = \frac{\sum_{n=1}^N (Vp_n^{max} + \mu_n^{max})^2}{2Vt} = O(V)/t \quad (28)$$

This shows that if  $V$  is chosen to be large, then the amount of time  $t$  required to make this error term negligible must also be large. One can minimize this error term with an initial

condition  $Q_n(0)$  that is close to  $\theta_n$  for all  $n$ . However, this is an artificial savings, as it does not include the *startup cost* associated with purchasing that many initial units of stock. Therefore, the timescales are more accurately described by the transient given in (28).

One may wonder how the Dynamic Trading Algorithm is achieving near optimal profit without knowing the distribution of the price vector  $\mathbf{p}(t)$ , and without estimating this distribution. The answer is that it uses the queue values themselves to guide decisions. These queue values  $Q_n(t)$  only deviate significantly from the target  $\theta_n$  when inefficient decisions are made. The values then act as a ‘‘sufficient statistic’’ on which to base future decisions. The same sufficient statistic holds for the non-i.i.d. case, as shown in Section IV, so that we do not need to estimate price patterns or time-correlations, provided that we allow for a sufficiently large control parameter  $V$  and corresponding large timescales for convergence.

Finally, one may also wonder if the limiting time average *expected profit* given in (27) also holds (with probability 1) for the limiting time average profit (without the expectation). When  $\mathbf{p}(t)$  evolves according to a finite state irreducible Markov chain (as is the case in this i.i.d. scenario), then the Dynamic Trading Algorithm in turn makes  $\mathbf{Q}(t)$  evolve according to a finite state Markov chain, and it can be shown that the limiting time average expected profit is the same (with probability 1) as the limiting time average profit.

### D. Proof of Theorem 1

*Proof:* (Theorem 1 part (a)) By Lemma 3 we know that  $Q_n(t) \geq \theta_n - Vp_n^{max} - \mu_n^{max}$  for all  $t$  (provided that this holds at  $t = 0$ ). However,  $\theta_n - Vp_n^{max} - \mu_n^{max} = \mu_n^{max}$ . Thus,  $Q_n(t) \geq \mu_n^{max}$  for all  $t$ , provided that this holds for  $t = 0$ . Similarly, by Lemma 4 we know that  $Q_n(t) \leq \theta_n + \mu_n^{max}$  for all  $t$  (provided that this holds for  $t = 0$ ), and  $\theta_n + \mu_n^{max} = Q_n^{max}$ .  $\square$

*Proof:* (Theorem 1 part (b)) Fix a slot  $t \in \{0, 1, 2, \dots\}$ . To prove part (b), we plug an alternative set of control choices  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$  into the drift-minus-reward bound (20) of Lemma 2. Because  $\mathbf{p}(t)$  is i.i.d., we can choose  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$  as the  $p$ -only policy that satisfies (12), (13). Note that we must first ensure this  $p$ -only policy satisfies the constraint (4) needed to apply the bound (20). However, we know from part (a) of this theorem that  $Q_n(t) \geq \mu_n^{max}$  for all  $n$ , and so the constraint (4) is *trivially* satisfied. Therefore, we can plug this policy  $\mathbf{A}^*(t)$  and  $\boldsymbol{\mu}^*(t)$  into (20) and use equalities (12) and (13) to yield:

$$\Delta(\mathbf{Q}(t)) - V\mathbb{E}\{\phi(t) | \mathbf{Q}(t)\} \leq B - V\phi^{opt}$$

Taking expectations of the above inequality over the distribution of  $\mathbf{Q}(t)$  and using the law of iterated expectations yields:

$$\mathbb{E}\{L(\mathbf{Q}(t+1)) - L(\mathbf{Q}(t))\} - V\mathbb{E}\{\phi(t)\} \leq B - V\phi^{opt}$$

The above holds for all  $t \in \{0, 1, 2, \dots\}$ . Summing the above over  $\tau \in \{0, \dots, t-1\}$  (for some positive integer  $t$ ) yields:

$$\mathbb{E}\{L(\mathbf{Q}(t)) - L(\mathbf{Q}(0))\} - V \sum_{\tau=0}^{t-1} \mathbb{E}\{\phi(\tau)\} \leq tB - tV\phi^{opt}$$

Dividing by  $tV$ , rearranging terms, and using non-negativity of  $L(\cdot)$  yields:

$$\bar{\phi}(t) \geq \phi^{opt} - B/V - \mathbb{E}\{L(\mathbf{Q}(0))\}/Vt$$

where  $\bar{\phi}(t)$  is defined in (26). This proves the result.  $\square$

#### IV. ARBITRARY PRICE PROCESSES

The Dynamic Trading Algorithm can be shown to provide similar performance for more general non-i.i.d. processes that are ergodic with a mild *decaying memory property* [1]. Here we consider the performance of the Dynamic Trading Algorithm for an *arbitrary* price vector process  $\mathbf{p}(t)$ , possibly a non-ergodic process without a well defined time average such as that given in (9). In this case, there may not be a well defined ‘‘optimal’’ time average profit  $\phi^{opt}$ . However, one can define  $\phi^{opt}(t)$  as the maximum possible time average profit achievable over the interval  $\{0, \dots, t-1\}$  by an algorithm with perfect knowledge of the future and that conforms to the constraints (2)-(6). For the ergodic settings described in the previous sections,  $\phi^{opt}(t)$  has a well defined limiting value, and our algorithm comes close to its limiting value. In this (possibly non-ergodic) setting, we do not claim that our algorithm comes close to  $\phi^{opt}(t)$ . Rather, we make a less ambitious claim that our policy yields a profit that is close to (or greater than) the profit achievable by a frame-based policy that can look only  $T$  slots into the future.

##### A. The $T$ -Slot Lookahead Performance

Let  $T$  be a positive integer, and fix any slot  $t_0 \in \{0, 1, 2, \dots\}$ . Define  $\psi_T(t_0)$  as the optimal profit achievable over the interval  $\{t_0, \dots, t_0 + T - 1\}$  by a policy that has perfect a-priori knowledge of the prices  $\mathbf{p}(\tau)$  over this interval, and that ensures for each  $n \in \{1, \dots, N\}$  that the total amount of stock  $n$  purchased over this interval is greater than or equal to the total amount sold. Specifically,  $\psi_T(t_0)$  is mathematically defined according to the following optimization problem that has decision variables  $\mathbf{A}(\tau)$ ,  $\boldsymbol{\mu}(\tau)$ , and that treats the stock prices  $\mathbf{p}(\tau)$  as deterministically known quantities:

$$\text{Max: } \psi \triangleq \sum_{\tau=t_0}^{t_0+T-1} \sum_{n=1}^N [\mu_n(\tau)p_n(\tau) - s_n(\mu_n(\tau))] - \sum_{\tau=t_0}^{t_0+T-1} \sum_{n=1}^N [A_n(\tau)p_n(\tau) + b_n(A_n(\tau))] \quad (29)$$

$$\text{Subj. to: } \sum_{\tau=t_0}^{t_0+T-1} A_n(\tau) = \sum_{\tau=t_0}^{t_0+T-1} \mu_n(\tau) \quad \forall n \quad (30)$$

$$\text{Constraints (2), (3), (5), (6)} \quad (31)$$

The value  $\psi_T(t_0)$  is equal to the maximizing value  $\psi$  in the above problem (29)-(31). Note that the constraint (30) only requires the amount of type- $n$  stock purchased to be equal to the amount sold by the end of the  $T$ -slot interval, and does not require this at intermediate steps of the interval. This allows the  $T$ -slot lookahead policy to *sell short* stock that is not yet owned, provided that the requisite amount is purchased by the end of the interval.

Note that the trivial decisions  $\mathbf{A}(\tau) = \boldsymbol{\mu}(\tau) = \mathbf{0}$  for  $\tau \in \{t_0, \dots, t_0 + T - 1\}$  lead to 0 profit over the interval, and hence  $\Psi_T(t_0) \geq 0$  for all  $T$  and all  $t_0$ . Consider now the interval  $\{0, 1, \dots, MT - 1\}$  that is divided into a total of  $M$  frames of  $T$ -slots. We show that for any positive integer  $M$ , our Dynamic

Trading Algorithm yields an average profit over this interval that is close to the average profit of a  $T$ -slot lookahead policy that is implemented on each  $T$ -slot frame of this interval.

##### B. The $T$ -Slot Sample Path Drift

Let  $L(\mathbf{Q}(t))$  be the Lyapunov function of (14). For a given slot  $t$  and a given positive integer  $T$ , define the  $T$ -slot *sample path drift*  $\hat{\Delta}_T(t)$  as follows:

$$\hat{\Delta}_T(t) \triangleq L(\mathbf{Q}(t+T)) - L(\mathbf{Q}(t)) \quad (32)$$

This differs from the one-slot conditional Lyapunov drift in (15) in two respects:

- It considers the difference in the Lyapunov function over  $T$  slots, rather than a single slot.
- It is a random variable equal to the difference between the Lyapunov function on slots  $t$  and  $t+T$ , rather than a conditional expectation of this difference.

*Lemma 5:* Suppose the Dynamic Trading Algorithm is implemented, with  $\theta_n$  values satisfying (22), and initial condition that satisfies (23). Then for any given slot  $t_0$  and all integers  $T > 0$ , we have:

$$\hat{\Delta}_T(t_0) - V \sum_{\tau=t_0}^{t_0+T-1} \phi(\tau) \leq DT^2 - V \sum_{\tau=t_0}^{t_0+T-1} \phi^*(\tau) - \sum_{n=1}^N [Q_n(t_0) - \theta_n] \sum_{\tau=t_0}^{t_0+T-1} [\mu_n^*(\tau) - A_n^*(\tau)]$$

where the constant  $D$  is defined in (17),  $\phi(\tau)$  is defined in (8), and  $\phi^*(\tau)$ ,  $\boldsymbol{\mu}^*(\tau)$ ,  $\mathbf{A}^*(\tau)$  represent any alternative control actions for slot  $\tau$  that satisfy the constraints (2), (3), (5), (6).

*Proof:* See [1][21] (the  $D$  constant stated here uses a technique in [21] to improve the bound given in [1]).  $\square$

*Theorem 2:* Suppose the Dynamic Trading Algorithm is implemented, with  $\theta_n$  values satisfying (22), and initial condition that satisfies (23). Then for any arbitrary price process  $\mathbf{p}(t)$  that satisfies (1), we have:

(a) All queues  $Q_n(t)$  are bounded according to (24).

(b) For any positive integers  $M$  and  $T$ , the time average profit over the interval  $\{0, \dots, MT - 1\}$  satisfies the deterministic bound:

$$\frac{1}{MT} \sum_{\tau=0}^{MT-1} \phi(\tau) \geq \frac{1}{MT} \sum_{m=0}^{M-1} \psi_T(mT) - \frac{DT}{V} - \frac{L(\mathbf{Q}(0))}{MTV} \quad (33)$$

where the  $\psi_T(mT)$  values are defined according to the  $T$ -slot lookahead policy that uses knowledge of the future to solve (29)-(31) for each  $T$ -slot frame. The constant  $D$  is defined in (17), and if  $\mathbf{Q}(0) = (\mu_1^{max}, \dots, \mu_N^{max})$  then  $L(\mathbf{Q}(0))/(MTV)$  has the form (28) with  $t = MT$ .

*Proof:* Part (a) has already been proven in Theorem 1. To prove part (b), fix any slot  $t_0$  and any positive integer  $T$ . Define  $\mathbf{A}^*(\tau)$  and  $\boldsymbol{\mu}^*(\tau)$  as the solution of (29)-(31) over the interval  $\tau \in \{t_0, \dots, t_0 + T - 1\}$ . By (31), these decision variables satisfy constraints (2), (3), (5), (6), and hence can be plugged in to the bound in Lemma 5. Because (29), (30) hold for these variables, by Lemma 5 we have:

$$\hat{\Delta}_T(t_0) - V \sum_{\tau=t_0}^{t_0+T-1} \phi(\tau) \leq DT^2 - V\psi_T(t_0)$$

Using the definition of  $\hat{\Delta}_T(t_0)$  given in (32) yields:

$$L(Q(t_0+T)) - L(Q(t_0)) - V \sum_{\tau=t_0}^{t_0+T-1} \phi(\tau) \leq DT^2 - V\psi_T(t_0)$$

The above inequality holds for all slots  $t_0 \in \{0, 1, 2, \dots\}$ . Letting  $t_0 = mT$  and summing over  $m \in \{0, 1, \dots, M-1\}$  (for some positive integer  $M$ ) yields:

$$L(Q(MT)) - L(Q(0)) - V \sum_{\tau=0}^{MT-1} \phi(\tau) \leq MDT^2 - V \sum_{m=0}^{M-1} \psi_T(mT)$$

Rearranging terms and using non-negativity of  $L(\cdot)$  proves the theorem.  $\square$

Theorem 2 is stated for general price processes, but has explicit performance bounds for queue size in terms of the chosen  $V$  parameter, and for profit in terms of  $V$  and of the profit  $\psi_T(mT)$  of  $T$ -slot lookahead policies. Plugging a large value of  $T$  into the bound (33) increases the first term on the right-hand-side because it allows for a larger amount of lookahead. However, this comes with the cost of increasing the term  $DT/V$  that is required to be small to ensure close proximity to the desired profit. One can use this theorem with any desired model of stock prices to compute statistics associated with  $\psi_T(mT)$  and hence understand more precisely the timescales over which near-optimal profit is achieved.

## V. EXTENSIONS

Theorems 1 and 2 require an initial stock level of at least  $\mu_n^{max}$  in all of the  $N$  stocks. This can be achieved by initially purchasing these shares (say, at time  $t = -1$ ). It turns out that we can achieve the *same* performance as specified in Theorems 1 and 2 without paying this startup cost. This can be done using the concept of *place-holder backlog* from [22], which becomes *place-holder stock* in our context. This is detailed in [1]. Our technical report [1] also considers extensions to cases when: (i) the maximum stock price observed jumps above our  $p_n^{max}$  value, (ii) stock prices can split, (iii) the  $\theta_n$  and  $Q_n^{max}$  parameters are scaled as time progresses, so that exponential wealth growth can be achieved.

## VI. CONCLUSION

This work uses Lyapunov optimization theory, developed for stochastic optimization of queueing networks, to construct a dynamic policy for buying and selling stock. When prices are ergodic, a single non-anticipating policy was constructed and shown to perform close to an ideal policy with perfect knowledge of the future, with a tradeoff in the required amount of stock kept in each queue and in the timescales associated with convergence. For arbitrary price sample paths, the same algorithm was shown to achieve a time average profit close to that of a frame based  $T$ -slot lookahead policy that can look  $T$  slots into the future. Our framework constrains the maximum number of stock shares that can be bought and sold at any time. While this restricts the long term growth

curve to a linear growth, it also limits risk by ensuring no more than a constant value  $Q_n^{max}$  shares of each stock  $n$  are kept at any time. A modified policy was briefly discussed that achieves exponential growth by scaling  $Q_n^{max}$  in proportion to increased risk tolerance as wealth increases. These results add to the theory of universal stock trading, and are important for understanding optimal decision making in the presence of a complex and possibly unknown price process.

## REFERENCES

- [1] M. J. Neely. Stock market trading via stochastic network optimization. *ArXiv Technical Report*, arXiv:0909.3891v1, Sept. 2009.
- [2] L. Georgiadis, M. J. Neely, and L. Tassiulas. Resource allocation and cross-layer control in wireless networks. *Foundations and Trends in Networking*, vol. 1, no. 1, pp. 1-149, 2006.
- [3] M. J. Neely. Energy optimal control for time varying wireless networks. *IEEE Transactions on Information Theory*, vol. 52, no. 7, pp. 2915-2934, July 2006.
- [4] M. J. Neely. *Dynamic Power Allocation and Routing for Satellite and Wireless Networks with Time Varying Channels*. PhD thesis, Massachusetts Institute of Technology, LIDS, 2003.
- [5] M. J. Neely and R. Urgaonkar. Cross layer adaptive control for wireless mesh networks. *Ad Hoc Networks (Elsevier)*, vol. 5, no. 6, pp. 719-743, August 2007.
- [6] H. Markowitz. Portfolio selection. *Journal of Finance*, vol. 7, no. 1, pp. 77-91, March 1952.
- [7] W. F. Sharpe. A simplified model for portfolio analysis. *Management Science*, vol. 9, no. 2, pp. 277-293, Jan. 1963.
- [8] J-P. Bouchaud and M. Potters. *Theory of Financial Risk and Derivative Pricing: From Statistical Physics to Risk Management, 2nd ed.* Cambridge University Press, 2003.
- [9] P. A. Samuelson. Lifetime portfolio selection by dynamic stochastic programming. *The Review of Economics and Statistics*, vol. 51, no. 3, pp. 239-246, Aug. 1969.
- [10] T. M. Cover. An algorithm for maximizing expected log investment return. *IEEE Transactions on Information Theory*, IT-30, pp. 369-373, 1984.
- [11] M. B. Rudoy and C. E. Rohrs. A dynamic programming approach to two-stage mean-variance portfolio selection in cointegrated vector autoregressive systems. *IEEE Conf. on Decision and Control*, 2008.
- [12] M. B. Rudoy. *Multistage Mean-Variance Portfolio Selection in Cointegrated Vector Autoregressive Systems*. PhD thesis, Massachusetts Institute of Technology, Feb. 2009.
- [13] A. Turiel and C. J. Pérez-Vicente. Multifractal geometry in stock market time series. *Physica A: Statistical Mechanics and its Applications*, vol. 322, pp. 629-649, May 2003.
- [14] B. Mandelbrot and H. M. Taylor. On the distribution of stock price differences. *Operations Research*, vol. 15, no. 6, pp. 1057-1062, 1967.
- [15] T. M. Cover and D. Gluss. Empirical bayes stock market portfolios. *Adv. Appl. Math.*, vol. 7, pp. 170-181, 1986.
- [16] D. C. Larson. *Growth Optimal Trading Strategies*. PhD thesis, Stanford University, 1986.
- [17] T. M. Cover. Universal portfolios. *Mathematical Finance*, vol. 1, no. 1, pp. 1-29, Jan. 1991.
- [18] N. Merhav and M. Feder. Universal schemes for sequential decision from individual data sequences. *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1280-1292, July 1993.
- [19] T. M. Cover and E. Ordentlich. Universal portfolios with side information. *IEEE Transactions on Information Theory*, vol. 42, no. 2, 1996.
- [20] E. Ordentlich and T. M. Cover. The cost of achieving the best portfolio in hindsight. *Mathematics of Operations Research*, vol. 23, no. 4, Nov. 1998.
- [21] M. J. Neely and L. Huang. Dynamic product assembly and inventory control for maximum profit. *ArXiv Technical Report*, arXiv:1004.0479v1, April 2010.
- [22] M. J. Neely and R. Urgaonkar. Opportunism, backpressure, and stochastic optimization with the wireless broadcast advantage. *Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA*, Oct. 2008.