Factor Graphs for Universal Portfolios

Andrew J. Bean and Andrew C. Singer
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
Email: ajbean@illinois.edu
Email: acsinger@illinois.edu

***Abstract*—We consider the sequential portfolio investment problem. Building on results in signal processing, machine learning, and other areas, we combine the insights of Cover and Ordentlich's side information portfolio with those of Blum and Kalai's transaction costs algorithm to construct one that performs well under transaction costs while taking advantage of side information. We introduce factor graphs as a computational tool for analysis and design of universal (low regret) algorithms, and develop our algorithm with this insight. Finally, we demonstrate that, in contrast to other algorithms, our portfolio performs well over the full range of costs.**

***Index Terms*—universal, portfolio, investment, transaction costs, piecewise models, factor graph, sum-product**

I. Introduction

Sequential decisions and the sequential investment problem have been extensively studied in signal processing [1]–[5]; computer science [6]–[8]; finance [9]–[11]; information theory [12]–[14]; game theory [15], [16]; and other areas. We consider the problem of sequentially investing in a stock market where we have access to a side information sequence and we also must pay transaction costs in order to adjust the allocation of wealth in our portfolio.

One framework for studying investment strategies under penalty of such costs consists of the following market model and investment. We model the market as a sequence of price relative vectors $\mathbf{x}^n = \mathbf{x}[1], \ldots, \mathbf{x}[n]$, $\mathbf{x}[t] \in \mathbb{R}_+^m$, where $\mathbb{R}_+^m$ is the positive orthant. The j^{th} entry $x_j[t]$ of the price relative vector $\mathbf{x}[t]$ represents the ratio of the opening price of the j^{th} stock on the $(t+1)^{\text{th}}$ trading period to the opening price of the same stock on the t^{th} trading period. The investment at period t is represented by a portfolio vector $\mathbf{b}[t] \in \mathbb{R}_+^m$ with the constraint $\sum_{j=1}^m b_j[t] = 1$ for all t, such that we consider only long positions in each asset. Here, each entry $b_j[t]$ corresponds to the portion of wealth invested in stock j during the t^{th} period. Therefore, under this setup we have that the wealth achieved by the sequence of portfolios $\mathbf{b}^n = \mathbf{b}[1], \ldots, \mathbf{b}[n]$ in the market $\mathbf{x}^n$, without transaction costs, is given by $W(\mathbf{b}^n; \mathbf{x}^n) = \prod_{t=1}^n \mathbf{b}^T[t]\mathbf{x}[t]$. However, if we include transaction costs in our market framework, then the wealth achieved by the sequence of portfolios is given by $W_c(\mathbf{b}^n; \mathbf{x}^n) = \left(\prod_{t=1}^n \mathbf{b}^T[t]\mathbf{x}[t]\right) \times \left(\prod_{t=2}^n C(\mathbf{b}[t]; \mathbf{b}[t-1], \mathbf{x}[t-1])\right)$. Simply put, the wealth achieved by the sequence of portfolios consists of factors $\mathbf{b}^T[t]\mathbf{x}[t]$ for the wealth change for each investment period, and additional penalty factors $C(\mathbf{b}[t]; \mathbf{b}[t-1], \mathbf{x}[t-1]) \leq 1$ incurred for rebalancing between investment periods. In our case, we use the sub-optimal rebalancing procedure refered to in [7] with fixed percentage commission c, which yields

$$C(\mathbf{b}[t]; \mathbf{b}[t-1], \mathbf{x}[t-1]) = 1 - c\sum_{i=1}^m \left| b_i[t] - \frac{b_i[t-1]x_i[t-1]}{\mathbf{b}^T[t-1]\mathbf{x}[t-1]} \right|. \quad (1)$$

In this case, we incur a cost for both buying and selling stock. Finally, each portfolio vector $\mathbf{b}[t]$ must be chosen sequentially, such that the portfolio decision depends only on past information, such as the previous price relatives $\mathbf{x}[1], \ldots, \mathbf{x}[t-1]$, and not on any information from the future. We specifically allow side information modeled as a sequence of values $y^n = y[1], ..., y[n]$ from a finite alphabet.

In [10], Cover presents a sequential algorithm that asymptotically achieves the wealth of the best *constant rebalanced portfolio* (CRP), where this best CRP is determined with knowledge of the entire sequence of price relatives. (Of course, the algorithm of [10] does not have access to such information.) Cover and Ordentlich [14] then extend these results to the situation where a side information sequence is available to the user. However, in both [10] and [14], it is assumed that no costs are incurred when reapportioning the wealth of the portfolio. Blum and Kalai [7] address this issue by constructing a sequential portfolio similar to that given by Cover in [10] that asymptotically achieves the wealth of the best CRP in the presence of fixed percentage transaction costs. However, there is no method given for incorporating the use of a side information sequence. The work presented here is an attempt to combine the insights of [14] and [7] in order to construct a portfolio algorithm that, in some sense, performs well under any level of transaction costs and is also able to make use of side information.

We begin the discussion in Section II by presenting the algorithms of [10], [7], and [14]. We offer a new perspective on [14], and how this allows us to incorporate the insight of [7] to construct a new portfolio algorithm that takes into account both side information and transaction costs. In Section III, we briefly discuss the difficulties in directly implementing the proposed algorithm and introduce the use of factor graphs as a tool for deriving computationally efficient implementations of, or approximations to, universal (low regret) algorithms. We proceed by using an appropriate factor graph to develop an approximation to the new transaction costs and side information portfolio. Finally, in Section IV we present simulation results comparing our algorithm to the those of [14] and [7].

II. Universal Portfolios

A. Portfolios without Side Information

Cover's universal portfolio from [10] is the following algorithm:

$$\hat{\mathbf{b}}[t] = \frac{\int_{\mathbf{b}\in\Delta_m} \mathbf{b}W(\mathbf{b};\mathbf{x}^{(t-1)})d\mu(\mathbf{b})}{\int_{\mathbf{b}\in\Delta_m} W(\mathbf{b};\mathbf{x}^{(t-1)})d\mu(\mathbf{b})} \quad (2)$$

where we have that

$$W(\mathbf{b};\mathbf{x}^{(t-1)}) \triangleq \prod_{\tau=1}^{t-1} \mathbf{b}^T\mathbf{x}[\tau], \quad (3)$$

$$\Delta_m \triangleq \left\{\mathbf{b}\in\mathbb{R}_+^m : \sum_{j=1}^m b_j[t] = 1\right\}, \quad (4)$$

and the distribution $\mu(\mathbf{b})$ over Δ_m is either Dirichlet $(1,\ldots,1)$ (the uniform prior) or Dirichlet $\left(\frac{1}{2},\ldots,\frac{1}{2}\right)$. For Dirichlet $(1,\ldots,1)$, Cover and Ordentlich [14] show that the following universal bound is achieved:

$$\frac{1}{n}\ln W(\hat{\mathbf{b}};\mathbf{x}^n) \geq \sup_{\mathbf{b}\in\Delta_m} \frac{1}{n}\ln W(\mathbf{b};\mathbf{x}^n) - \frac{m-1}{n}\ln(n+1). \quad (5)$$

For the Dirichlet $\left(\frac{1}{2},\ldots,\frac{1}{2}\right)$ prior, the following slightly improved bound is achieved:

$$\frac{1}{n}\ln W(\hat{\mathbf{b}};\mathbf{x}^n) \geq \sup_{\mathbf{b}\in\Delta_m} \frac{1}{n}\ln W(\mathbf{b};\mathbf{x}^n) - \frac{m-1}{2n}\ln(n+1) - \frac{\ln(2)}{n}. \quad (6)$$

This portfolio algorithm and the given bounds, however, do not take into account transaction costs.

Blum and Kalai [7] offer a solution to the problem of transaction costs in the following portfolio algorithm:

$$\hat{\mathbf{b}}[t] = \frac{\int_{\mathbf{b}\in\Delta_m} \mathbf{b}W_c(\mathbf{b};\mathbf{x}^{(t-1)})d\mu(\mathbf{b})}{\int_{\mathbf{b}\in\Delta_m} W_c(\mathbf{b};\mathbf{x}^{(t-1)})d\mu(\mathbf{b})} \quad (7)$$

where we have that

$$W_c(\mathbf{b};\mathbf{x}^{(t-1)}) \triangleq \prod_{\tau=1}^{t-1}\left(\mathbf{b}^T\mathbf{x}[\tau]\times C(\mathbf{b};\mathbf{b},\mathbf{x}[\tau])\right) \quad (8)$$

and the distribution $\mu(\mathbf{b})$ over Δ_m is Dirichlet $(1,\ldots,1)$. It is shown [7] that the following universal bound is achieved:

$$\frac{1}{n}\ln W_c(\hat{\mathbf{b}};\mathbf{x}^n) \geq \sup_{\mathbf{b}\in\Delta_m} \frac{1}{n}\ln W_c(\mathbf{b};\mathbf{x}^n) - \frac{m-1}{n}\ln((1+2c)n+1). \quad (9)$$

B. Portfolios with Side Information

Typically, an investor is able to base investment decisions on some *side information*. For our purposes, this side information is modeled as a sequence of values $y^n = y[1],...,y[n]$ from a finite alphabet, which we can take to be $\mathcal{Y} = \{1,...,K\}$ without loss of generality. We assume that the investor is able to base the decision for investment period t on the value of $y[t]$. In [14], the authors present an adaptation of the algorithm in (2) resulting in a universal portfolio algorithm that is able to make use of the side information sequence y^n. The resulting side information portfolio is the following:

$$\hat{\mathbf{b}}[t] = \frac{\int_{\mathbf{b}\in\Delta_m} \mathbf{b}W^{y[t]}(\mathbf{b};\mathbf{x}^{(t-1)})d\mu(\mathbf{b})}{\int_{\mathbf{b}\in\Delta_m} W^{y[t]}(\mathbf{b};\mathbf{x}^{(t-1)})d\mu(\mathbf{b})} \quad (10)$$

where we have that

$$W^{y[t]}(\mathbf{b};\mathbf{x}^{(t-1)}) \triangleq \prod_{\tau\in\mathcal{T}_{y[t]}^t} \mathbf{b}^T\mathbf{x}[\tau] \quad (11)$$

and

$$\mathcal{T}_y^t \triangleq \{\tau\in\mathbb{N} : \tau < t\ ,\ y[\tau] = y\}. \quad (12)$$

The distribution $\mu(\mathbf{b})$ over Δ_m can be either Dirichlet $(1,\ldots,1)$ or Dirichlet $\left(\frac{1}{2},\ldots,\frac{1}{2}\right)$. It should be noted that the algorithm amounts to running independent universal portfolios from (2) on each of the K subsequences of $\mathbf{x}^n$, $\mathbf{x}_y^n$, consisting of the price relatives $\mathbf{x}[t]$ when $y[t] = y$. For both of the distributions $\mu(\mathbf{b})$, there is a corresponding universal bound indicating that the algorithm asymptotically performs as well as the best state-constant rebalanced portfolio $\{\mathbf{b}_1,...,\mathbf{b}_K\}$, meaning the same portfolio $\mathbf{b}_y$ is used whenever $y[t] = y$. The natural extension is to develop a portfolio algorithm that both uses a given side information sequence and takes into account transaction costs. This is the contribution of our work.

In order to derive our algorithm, we begin by recalling that both algorithms of (2) and (7) may be written in the following way:

$$\begin{aligned} P_t(\mathbf{d}) &= \frac{P_{t-1}(\mathbf{d})f_t(\mathbf{d})}{Z} \\ \hat{\mathbf{d}}[t] &= \mathrm{E}_{P_t(\mathbf{d})}[\mathbf{d}] \\ \hat{\mathbf{b}}[t] &= g_t(\hat{\mathbf{d}}[t]) \end{aligned} \quad (13)$$

for some non-negative factor $f_t(\mathbf{d})$ and normalizing constant Z. (We use the symbol Z whenever it is necessary to normalize to a proper distribution. It's value will be obvious based on the context.) The variable $\mathbf{d}$ represents one from a set $\mathcal{D}$ of constant strategies, such as constant rebalanced portfolios, and the function $g_t(\cdot)$ maps the decision $\hat{\mathbf{d}}$ to a portfolio vector $\hat{\mathbf{b}} \in \Delta_m$. In (2) and (7), we have that $\mathcal{D} = \Delta_m$ (i.e., $\mathbf{d} = \mathbf{b}$) and $g_t(\cdot)$ is the identity function (i.e., $\hat{\mathbf{d}} = \hat{\mathbf{b}}$). The distribution $P_t(\mathbf{b})$ is simply the wealth function $W(\mathbf{b};\mathbf{x}^{(t-1)})$ or $W_c(\mathbf{b};\mathbf{x}^{(t-1)})$, multiplied by the distribution $\mu(\mathbf{b})$, and then normalized so that we have a proper probability distribution. Hence, we can take $P_0(\mathbf{b})$ as the uniform distribution, and $f_1(\mathbf{b}) = \mu(\mathbf{b})$.

Now, as presented in [14], the algorithm in (10) is not of the form in (13), as it involves jumping between the various independently running algorithms for each side information value. However, we can consider the following algorithm, using side information y^n, which is in the form of (13) and is defined as follows:

The constant strategy space is

$$\mathcal{D} = \Delta_m^K = \{\mathbf{d} = (\mathbf{b}_1,...,\mathbf{b}_K) : \mathbf{b}_i \in \Delta_m \text{ for } i = 1,...,K\}. \quad (14)$$

We now define

$$f_1(\mathbf{d}) = \mu(\mathbf{b}_1) \times \ldots \times \mu(\mathbf{b}_K) \quad (15)$$

where μ is either the Dirichlet$(1, \ldots, 1)$ prior or the Dirichlet$\left(\frac{1}{2}, \ldots, \frac{1}{2}\right)$ prior, and $P_0(\mathbf{d})$ is the uniform distribution over Δ_m^K. Furthermore, for $t > 1$, define

$$f_t(\mathbf{d}) = \mathbf{b}_{y[t-1]}^T \mathbf{x}[t-1]. \quad (16)$$

Finally, with $\hat{\mathbf{d}} = (\hat{\mathbf{b}}_1, ..., \hat{\mathbf{b}}_K)$, we define

$$g_t(\hat{\mathbf{d}}) = \hat{\mathbf{b}}_{y[t]}. \quad (17)$$

Now, computing this algorithm directly can be very expensive, as it involves evaluating an integral over the potentially high dimensional space Δ_m^K. However, it can be shown that this algorithm reduces to the side information portfolio of (10), which involves only integrals over the space Δ_m.

The purpose of this exercise is to allow us to extend the side information portfolio of [14] to the situation with transaction cost by drawing direct analogy to the insight of [7], as follows: The universal portfolio of [10] is constructed using

$$f_t(\mathbf{b}) = \mathbf{b}^T \mathbf{x}[t-1]. \quad (18)$$

This algorithm is generalized to incorporate transaction costs by modifying (18) to be

$$f_t(\mathbf{b}) = \mathbf{b}^T \mathbf{x}[t-1] \times C(\mathbf{b}; \mathbf{b}, \mathbf{x}[t-1]). \quad (19)$$

Similarly, we can modify (16) to be

$$f_t(\mathbf{d}) = \mathbf{b}_{y[t-1]}^T \mathbf{x}[t-1] \times C(\mathbf{b}_{y[t]}; \mathbf{b}_{y[t-1]}, \mathbf{x}[t-1]). \quad (20)$$

Hence, when we combine the definitions in (13), (14), (15), (17), and (20), we arrive at the algorithm we propose, which takes into account both side information and transaction costs. The result is a performance-weighted convex combination of all the constant strategies in the set $\mathcal{D} = \Delta_m^K$. It should be noted that our algorithm reduces to (10) when there are no transaction costs, to (7) when $K = 1$, and to (2) when $K = 1$ and there are no transaction costs.

III. Computation by Factor Graphs

Unfortunately, the evaluation of the expectation in (13), which again involves an integral over Δ_m^K, does not simplify as was the case for (10), and the direct evaluation of the integral is exponential in K for both storage and computation. For this reason, we propose using factor graphs and a sum-product algorithm [17] in order to approximate our algorithm.

Graphical representations are not new in universal portfolios and universal prediction. Examples include the transition diagrams used in [2], [5], [6] and the tree representations used in [3], [4]. However, the authors are not aware of any previous use of factor graphs and sum-product algorithms in this area of research.

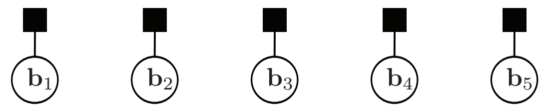

Fig. 1. Factor graph for portfolio with side information of $K = 5$.

A. Factor Graph for Universal Portfolio with Side Information

We begin this development by demonstrating the factor graph concept on the instance of the side information portfolio from [14] defined by (13)-(17). First, we note that

$$\begin{aligned} P_t(\mathbf{d}) &= \frac{1}{Z} \prod_{\tau=1}^{t} f_t(\mathbf{d}) \\ &= \frac{1}{Z} \left(\prod_{i=1}^{K} \mu(\mathbf{b}_i) \right) \left(\prod_{\tau=1}^{t-1} \mathbf{b}_{y[\tau]}^T \mathbf{x}[\tau] \right) \end{aligned} \quad (21)$$

Upon regrouping the factors, we see that

$$\begin{aligned} P_t(\mathbf{d}) &= \frac{1}{Z} \prod_{i=1}^{K} \left(\mu(\mathbf{b}_i) \prod_{\tau \in \mathcal{T}_i^t} \mathbf{b}_i^T \mathbf{x}[\tau] \right) \\ &= \frac{1}{Z} \prod_{i=1}^{K} \left(W^i(\mathbf{b}_i; \mathbf{x}^{(t-1)}) \mu(\mathbf{b}_i) \right) \end{aligned} \quad (22)$$

It is now obvious that $P_t(\mathbf{d})$ may be represented by a factor graph (forest) such as we show in Fig. 1, where in the depicted case we have that $K = 5$.

Furthermore, note that due to the form of $g_t(\hat{\mathbf{d}})$, we have that $\hat{\mathbf{b}}[t] = \mathrm{E}_{P_t(\mathbf{d})}[\mathbf{b}_{y[t]}]$. It is then clear by inspection of the factor graph that the algorithm reduces to that of (10), where this expectation only needs to be taken with respect to the following distribution:

$$\frac{W^{y[t]}(\mathbf{b}_{y[t]}; \mathbf{x}^{(t-1)}) \mu(\mathbf{b}_{y[t]})}{\int_{\mathbf{b}_{y[t]} \in \Delta_m} W^{y[t]}(\mathbf{b}_{y[t]}; \mathbf{x}^{(t-1)}) d\mu(\mathbf{b}_{y[t]})} \quad (23)$$

B. Factor Graph for Universal Portfolio with Side Information and Transaction Costs

Similar derivations can be used to show that the algorithm we propose here results in distributions $P_t(\mathbf{d})$ of the following form:

$$P_t(\mathbf{d}) = \frac{1}{Z} \prod_{i=1}^{K} \left(h_i(\mathbf{b}_i) \prod_{j=i+1}^{K} h_{(i,j)}(\mathbf{b}_i, \mathbf{b}_j) \right) \quad (24)$$

where the factors $h_i(\mathbf{b}_i)$ are further composed of factors either of the form $\mu(\mathbf{b}_i)$, $\mathbf{b}_i^T \mathbf{x}$, or $C(\mathbf{b}_i; \mathbf{b}_i, \mathbf{x})$, and the factors $h_{(i,j)}(\mathbf{b}_i, \mathbf{b}_j)$ are further composed of factors either of the form $C(\mathbf{b}_i; \mathbf{b}_j, \mathbf{x})$ or $C(\mathbf{b}_j; \mathbf{b}_i, \mathbf{x})$. Hence, we have that $P_t(\mathbf{d})$ may be represented by such a factor graph as we show in Fig. 2, where in the depicted case we have that $K = 5$.

In order to make use of the graph, we note that, as in the previous subsection, our portfolio for an investment period is $\hat{\mathbf{b}}[t] = \mathrm{E}_{P_t(\mathbf{d})}[\mathbf{b}_{y[t]}]$. Hence, we only need to compute the marginal distribution over $\mathbf{b}_{y[t]}$ and then compute the expectation. Since the factor graph we have constructed has

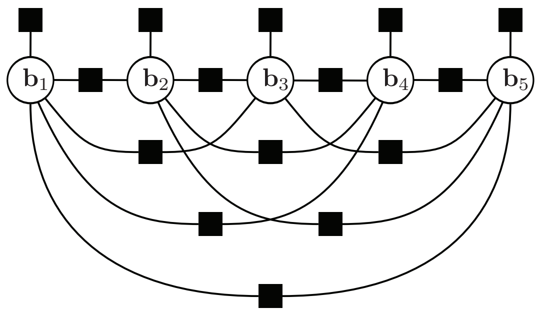

Fig. 2. Factor graph for portfolio with side information of $K = 5$ and transaction costs.

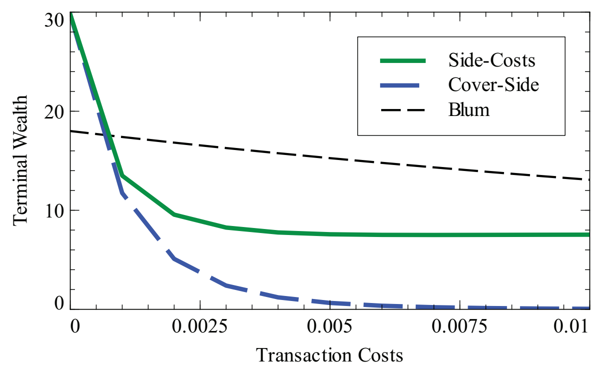

Fig. 3. Terminal achieved wealth (averaged over all stock pairs) versus transaction costs c.

many cycles (for $K > 2$), it is not possible to compute the marginal distribution exactly. However, we have found through simulations that the following message passing (sum-product) algorithm works well on our data sets:

First of all, we choose the distribution $\mu(\cdot)$ to be uniform. We then begin with the first investment period by initializing all of the messages over the edges of the graph to be the uniform distribution, i.e., constant functions over the simplex Δ_m. Note that with this initialization the messages correspond with exact function summaries for $P_1(\mathbf{d})$ up to a constant scale factor.

Now, let us suppose that we have the factor graph and messages from the previous distribution $P_{t-1}(\mathbf{d})$, and that these messages are close to exact function summaries on $P_{t-1}(\mathbf{d})$. For $P_t(\mathbf{d})$, we first initialize the messages in the new factor graph with the messages from $P_{t-1}(\mathbf{d})$. This is a good initialization because the two factor graphs differ only in 1 or 2 out of the $\frac{1}{2}K(K+1)$ function nodes, depending on whether $y[t-1] = y[t]$.

In order to carry out message passing, we begin by queueing the messages going outward from the 1 or 2 function nodes that differ between $P_{t-1}(\mathbf{d})$ and $P_t(\mathbf{d})$. This is a queue of pending message updates. Now that there is at least one message on the queue, we proceed with a serial message passing schedule:

1) Begin at the front of the queue.

2) Update the pending message.

3) Suppose the message just updated goes along the edge from node A to node B. Queue up all of the messages going out of node B, except the one going back to node A. Limit the total queue size to P message passes.

4) If there are more messages on the queue, move on to the next one and go back to step 2.

In our simulations, we allow a queue size on the order of K^2 message passes. This is natural since there are K^2 edges in the graph. In particular, we have experimentally determined that $P = 3K^2$ gives good results. The final algorithm has run time and storage complexity polynomial in K per investment period.

IV. Simulation Results

In this section, we present results comparing our algorithm with the factor graph implementation to the side information portfolio of [14] and the transaction costs portfolio of [7]. The data sets we use consist of historical stock prices collected from 34 stocks[1] in the New York Stock Exchange over a 22 year period until 1985.[2] The simulations consist of portfolios containing two stocks and $K = 4$. The side information is formed by quantizing the price relative space $\mathbf{x}[t-1]$, i.e., $y[t] = q(\mathbf{x}[t-1])$ where we have that

$$q(\mathbf{x}) = \begin{cases} 1 & & x_1 \geq 1 \text{ and } x_2 \geq 1 \\ 2 & \text{if} & x_1 \geq 1 \text{ and } x_2 < 1 \\ 3 & & x_1 < 1 \text{ and } x_2 \geq 1 \\ 4 & & x_1 < 1 \text{ and } x_2 < 1 \end{cases} \quad (25)$$

and we arbitrarily choose $y[1] = 1$. Each price relative represents one day.

In Fig. 3, we show results comparing the terminal wealths after the full 22 years, with 1 unit initially invested, of the three algorithms as a function of the parameter c of transaction costs. In particular, these terminal wealths are the averages over all $34 \times 33/2 = 561$ pairs of stocks. The graph shows characteristics common to most of the simulation runs. For example, our algorithm ("Side-Costs" in the plot) consistently achieves equal or greater wealth than the side information algorithm of [14] ("Cover-Side") under all simulated values of the transaction costs parameter c. More specifically, Cover-Side achieves a terminal wealth that falls off exponentially as c increases, whereas our portfolio's wealth decreases to a certain point, and then after that point it essentially achieves the same terminal wealth. This is because the algorithm "learns" that rebalancing the portfolio, even based on the side information, is too costly, and the algorithm reverts to a buy-and-hold portfolio.

Fig. 3 also shows a comparison between the transaction costs portfolio of [7] ("Blum" in the plot) and our algorithm. We point out that, even though Blum performs consistently better than our algorithm for transaction costs above approximately $c = 0.001$, it should be noted that it cannot be know a-priori whether the particular setup of stocks, side

[1]We have excluded the anomalous Kin-Ark data set from our simulations.

[2]We thank Dr. Erik Ordentlich for providing us with the historical data.

information, and transaction costs will place us above or below the critical threshold of c that is observed between our algorithm or Blum having better performance. In particular, the performance of our algorithm is generally no worse than a factor of approximately 3 below that of Blum in the cases where Blum has the better performance. However, for a well chosen side information sequence, it is possible for our algorithm to achieve wealth orders of magnitude above the algorithm of [7].

V. Conclusion

In this paper, we consider the problem of sequentially investing in a stock market where we have access to a side information sequence and we must pay a fixed percentage commission on every transaction. By generalizing the formulations of the algorithms of Cover [10], Cover and Ordentlich [14], and Blum and Kalai [7], we have shown that the key insights of these algorithms can be combined to construct a portfolio algorithm that is shown to perform consistently well in simulations under various setups of side information and transaction costs. Furthermore, though we have not attempted to do so in this paper, it may be possible to perform a more complete analysis of our algorithm by applying the results of [7] in the hopes of deriving a universal performance bound.

We have also introduced the use of factor graphs for the design of portfolio algorithms. In our case, this has allowed us to generalize from the portfolio of [14], where the strategy for each of the side information states is independent of the others, to a new portfolio where all of the side information states are pairwise connected due to the $C(\mathbf{b}_i; \mathbf{b}_j, \mathbf{x})$ terms.

However, this is not the only situation where we can conceive of the strategies for different side information states being dependent. For example, the fundamental nature of the side information sequence may not be finite valued. The current setup forces us to take continuous valued side information and quantize it. However, there is the obvious tradeoff between coarse and fine quantization of this side information space. In particular, the coarse quantization allows for faster learning of the strategy for a particular value of the quantized side information, but it does not allow for finding the small detail dependencies between the continuous side information and a good portfolio strategy.

Methods have been proposed to deal with this problem, which is essentially a problem of model order selection. A notable example is the context tree portfolio of [4]. However, the factor graph formulation may present a new way of approaching this problem. In particular, our portfolio, as presented, uses the following a-priori distribution over the product space $\mathcal{D} = \Delta_m^K$ of strategies assigned to each side information state:

$$P_1(\mathbf{d}) = \prod_{i=1}^{K} \mu(\mathbf{b}_i). \tag{26}$$

This can be interpreted as assuming the strategies in each state should be unrelated. However, it would be possible to use a fine-scale partitioning of the continuous side information space as the quantization into the finite valued side information space, but then offset the high model order by assuming that the strategies of neighboring partitions should be close to each other. Such local correlations can be readily incorporated into the factor graph framework by modifying the a-priori distribution $P_1(\mathbf{d})$ to include correlation factors between these neighboring states.

Acknowledgment

This research was performed under an appointment to the U.S. Department of Homeland Security (DHS) Scholarship and Fellowship Program, administered by the Oak Ridge Institute for Science and Education (ORISE) through an interagency agreement between the U.S. Department of Energy (DOE) and DHS. ORISE is managed by Oak Ridge Associated Universities (ORAU) under DOE contract number DE-AC05-06OR23100. All opinions expressed in this paper are the author's and do not necessarily reflect the policies and views of DHS, DOE, or ORAU/ORISE.

References

[1] A. C. Singer and M. Feder, "Universal linear prediction by model order weighting," *IEEE Transactions on Signal Processing*, vol. 47, no. 10, pp. 2685–2699, October 1999.

[2] S. S. Kozat and A. C. Singer, "Universal constant rebalanced portfolios with switching," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2007, pp. 1129 – 1132.

[3] S. S. Kozat, A. C. Singer, and G. C. Zeitler, "Universal piecewise linear prediction via context trees," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3730–3745, July 2007.

[4] S. S. Kozat, A. C. Singer, and A. J. Bean, "Universal portfolios via context trees," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 2093 – 2096.

[5] S. S. Kozat and A. C. Singer, "Universal switching portfolios under transaction costs," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 5404 – 5407.

[6] Y. Singer, "Switching portfolios," in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, 1998, pp. 1498–1519.

[7] A. Blum and A. Kalai, "Universal portfolios with and without transaction costs," *Machine Learning*, vol. 35, no. 3, pp. 193–205, June 1999.

[8] A. Agarwal, E. Hazan, S. Kale, and R. E. Schapire, "Algorithms for portfolio management based on the newton method," in *Proceedings of the 23rd International Conference on Machine Learning*, 2006, pp. 9–16.

[9] H. Markowitz, "Portfolio selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, March 1952.

[10] T. M. Cover, "Universal portfolios," *Mathematical Finance*, vol. 1, no. 1, pp. 1–29, January 1991.

[11] D. P. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth, "Online portfolio selection using multiplicative updates," *Mathematical Finance*, vol. 8, no. 4, pp. 325–347, October 1998.

[12] M. Feder, N. Merhav, and M. Gutman, "Universal prediction of individual sequences," *IEEE Transactions on Information Theory*, vol. 38, no. 4, pp. 1258–1270, July 1992.

[13] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," *Information and Computation*, vol. 108, no. 2, pp. 212–261, February 1994.

[14] T. M. Cover and E. Ordentlich, "Universal portfolios with side information," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 348–363, March 1996.

[15] R. Bell and T. M. Cover, "Game theoretic optimal portfolios," *Management Science*, vol. 34, no. 6, pp. 724–733, June 1988.

[16] D. P. Foster and R. Vohra, "Regret in the on-line decision problem," *Games and Economic Behavior*, vol. 29, no. 1-2, pp. 7–35, October 1999.

[17] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, February 2001.