UNIVERSAL SWITCHING PORTFOLIOS UNDER TRANSACTION COSTS

Suleyman S. Kozat

Koc University
Istanbul, Turkey
Email: skozat@ku.edu.tr

Andrew C. Singer

University of Illinois at Urbana Champaign
Urbana, IL 61801, USA
Email: acsinger@uiuc.edu

ABSTRACT

In this paper, we consider online (sequential) portfolio selection in a competitive algorithm framework under transaction costs. We construct a sequential algorithm for portfolio selection that asymptotically achieves the wealth of the best piecewise constant rebalanced portfolio tuned to the underlying individual sequence of price relative vectors where we pay a fixed percent commission for each transaction. Without knowledge of the investment duration, the algorithm can perform as well as the best investment algorithm that can choose both the partitioning of the sequence of the price relative vectors as well as the best constant rebalanced portfolio within each segment based on knowledge of the sequence of price relative vectors in advance. We use a transition diagram similar to that in [1] to compete with an exponential number of switching investment strategies, using only linear complexity in the data length for combination.

***Index Terms*—** Bayesian learning, Adaptive signal processing, Portfolio selection

I. INTRODUCTION

Investment from a competitive approach is an emerging field in signal processing [2], although it has been extensively investigated in information theory [3] and computational learning theory [4]. Here, the objective is to maximize one's invested wealth in a stock market with a finite number of stocks with respect to a class of possible investment strategies, i.e., the competition class. As an example, Cover [3] introduced a game where the objective is to find a sequential investment strategy which achieves the performance of the best constant rebalanced portfolio. A constant rebalanced portfolio (CRP) is an investment strategy that keeps the same proportion of wealth among a set of stocks from one investment period to other. It is shown that under reasonable stochastic assumptions on the sequence of price relatives [3], the portfolio selection model that achieves the maximum wealth is a CRP. Suppose the market is modelled by a sequence of price relative vectors $\boldsymbol{x}^n = \boldsymbol{x}[1], \ldots, \boldsymbol{x}[n]$, $\boldsymbol{x}[t] \in \mathcal{R}_+^m$. The jth entry $x_j[t]$ of a price relative vector $\boldsymbol{x}[t]$ represents the ratio of closing to opening price of the jth stock for the tth trading day. An investment at day t is represented by the portfolio vector $\boldsymbol{b}[t]$, $\boldsymbol{b}[t] \in \mathcal{R}_+^m$ and $\sum_{j=1}^m b_j[t] = 1$ for all t. Each entry $b_j[t]$ corresponds to the portion of the wealth invested in the stock $x_t[j]$ at day t. The achieved wealth after n investment periods is given by $\prod_{t=1}^n \boldsymbol{b}^T[t]\boldsymbol{x}[t]$. For a CRP $\boldsymbol{b}[t]$ is fixed, i.e., $\boldsymbol{b}[t] = \boldsymbol{b}$, for all investment periods, for some $\boldsymbol{b}$. Then, the best CRP is given by $\boldsymbol{b}^* = \arg\max_{\boldsymbol{b}} \prod_{t=1}^n \boldsymbol{b}^T[t]\boldsymbol{x}[t]$. Obviously, this portfolio $\boldsymbol{b}^*$ can only be chosen in hindsight, i.e., one needs to know the future. Cover's sequential algorithm achieves, to first order in the exponent, the wealth of the best constant rebalanced portfolio from the class of all constant rebalanced portfolios for any sequence of price relative vectors. This basic framework is then extended to competition against the best piecewise constant rebalanced portfolios in [2].

However, the market considered in [3] or in [2] is an idealized market where there are no transaction costs, short sales or investment on margin. Here, we extend the model considered in [2] to include a fixed percentage commission on each transaction. Clearly, keeping a constant rebalanced portfolio in each segment and adjusting to a new portfolio in every switch requires potentially significant trading. If one starts with a capital of 1 dollars and invests with a constant rebalanced portfolio $\boldsymbol{b} = [b_1, \ldots, b_m]^T$, then at the end of the first period, one has $b_i x_i$ dollars in each stock $i = 1, \ldots, m$, where x_i is the relative price change of the ith stock. Now, the new portfolio vector is given by $[b_1 x_1 / \sum_i (b_i x_i), \ldots, b_m x_m / \sum_i (b_i x_i)]^T$ (which can be significantly different than $\boldsymbol{b}$) and must be adjusted to $\boldsymbol{b}$ before the next trading. An extensive study of how this trading could be optimally done for minimizing the wealth loss due to commission is covered in [5].

We call algorithms (such as Cover's algorithm) that asymptotically achieve the performance of the single-best algorithm (the best CRP) from a given class of algorithms (class of all CRP's) for any sequence of price relatives, "static" universal algorithms, since the competition class contains a fixed set of algorithms, and performance is compared with the best, fixed element of the class. In our previous paper [2], we extended the results for static algorithms to a framework where the underlying competition class includes the ability to switch (in time) among the various static elements. We investigated this problem, when each competing algorithm can divide the sequence of price relatives into, say, k segments and fit each with the best static algorithm for that segment, such as a fixed constant rebalanced portfolio. For k such transitions, there exist $k+1$ segments. The total wealth growth of a class member with such a partition is the product of the wealth growth of all fixed static algorithms associated with each segment. The best partition is the one which gives the maximum total wealth. In this paper, we seek to outperform all such switching algorithms, simultaneously for any number of possible switches, k, and include transactions costs.

For a given sequence of price relative vectors $\boldsymbol{x}^n$, a competing portfolio selection algorithm with a transition path $\mathcal{T}_{k,n}$ with k transitions, represented by $(t_1, \ldots, t_k)$, partitions $\boldsymbol{x}^n$ into $k+1$ segments such that $\boldsymbol{x}^n$ is represented by the concatenation of

$$\{\boldsymbol{x}[1], \ldots, \boldsymbol{x}[t_1-1]\} \ldots \{\boldsymbol{x}[t_k], \ldots, \boldsymbol{x}[n]\}.$$

Given n and k, there exist $\binom{n-1}{k}$ such possible transition paths $\mathcal{T}_{k,n}$. Given the past values of the desired price relatives $\boldsymbol{x}[t]$, $t = 1, \ldots, n-1$, a competing algorithm assigns a portfolio vector $\boldsymbol{b}_i$ in each segment as $\hat{\boldsymbol{b}}[t] = \boldsymbol{b}_i$ where $t_{i-1} \leq t < t_i$, $i = 1, \ldots, k+1$. For simplicity we assume $t_0 = 1$ and $t_{k+1} = n+1$. Here, the competing class contains all constant portfolios in each segment that have the same $\boldsymbol{b}_i$ for each sample of the sequence $\boldsymbol{x}[t]$ for $t = t_{i-1}, \ldots, t_i - 1$, where each $\boldsymbol{b}_i$ can be selected independently. This algorithm will pay a transaction cost in each segment i to rebalance its portfolio to $\boldsymbol{b}_i$ and at each transition to rebalance to the next constant rebalanced portfolio $\boldsymbol{b}_{i+1}$. We define the wealth achieved by this algorithm as $W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$ including commission costs c_{sell} and c_{buy}, where $c = c_{sell} + c_{buy}$ and $\boldsymbol{B}_k = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k+1}]$.The wealth $W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$ can be significantly affected by transaction costs if c is large.

In determining the best algorithm in the competing class, we attempt to outperform all such portfolios, including the one that has been selected by choosing the transition path $\mathcal{T}_{k,n}$ and the constant portfolio vectors $\boldsymbol{b}_i$ in each segment based on observing the entire sequence $\boldsymbol{x}^n$ in advance, simultaneously, for all k. As such we try to minimize the following regret:

$$R_{\boldsymbol{b}}[n] \triangleq \sup_{\boldsymbol{x}^n} \frac{\sup_{\boldsymbol{B}_k, \mathcal{T}_{k,n}} W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})}{W^c(\boldsymbol{x}^n \mid \hat{\boldsymbol{b}})}$$

where $W^c(\boldsymbol{x}^n \mid \hat{\boldsymbol{b}})$ wealth achieved by a sequential portfolio assignment algorithm $\hat{\boldsymbol{b}}[t]$, i.e., $\hat{\boldsymbol{b}}[t]$ may be a function of $\boldsymbol{x}[1], \ldots, \boldsymbol{x}[t-1]$ but does not depend on the future, $\mathcal{T}_{k,n}$ is any transition path representing $(t_1, \ldots, t_k)$ with an arbitrary number of transitions k. We will show that we can construct a sequential portfolio selection algorithm for which the logarithm of this regret is at most $(k+1)(m-1)\ln(n)/2 + k\ln(n) + k\ln(1/(1-c)) + O(k+1)$ in the exponent for any $\mathcal{T}_{k,n}$, k, c or n and with no prior knowledge of $\mathcal{T}_{k,n}$, k or n. We recognize the term $(k+1)(m-1)\ln(n)/2$ as the parameter regret or additional loss due to the estimation of the best constant portfolio in each of the $k+1$ separate regions and the term $k\ln(n)$ as the transition path regret due to not knowing the best transitions times and $k\ln(1/(1-c))$ as the cost of switching portfolios at each segment.

For this switching framework, investment with transaction costs is investigated in [8], however, when the underlying static algorithms that can be chosen in each segment are restricted to be pure strategies, i.e., strategies just invest in one of the stocks. These results are not valid for CRP's that are optimized for each region. The results we present can also be extended to this special case when we are restricted to choose only from pure strategies. For this special case, our bounds are also superior and we present the actual portfolio selection algorithm instead of only its achieved wealth as in [8].

The organization of the paper is as follows. In Section 2, we provide the main theorem of this paper as an upper bound on the performance of the universal portfolio selection algorithm. The construction of the algorithm and an outline of the proof of the theorem are given in Section 3. Detailed proofs without transactions costs and extensions of this algorithm to switching with an arbitrary side-information sequence are provided in [2].

II. PERFORMANCE RESULTS

The main results of this paper are summarized in the upper bounds of Theorem 1. The corresponding universal and strongly sequential portfolio selection algorithm is constructed at the end of the proof in Section 3.

Theorem 1: *Let $\boldsymbol{x}^n = \boldsymbol{x}[1], \ldots, \boldsymbol{x}[n]$ be an arbitrary sequence of price relative vectors such that $\boldsymbol{x}[t] \in R_+^m$ for all n and some components of $\boldsymbol{x}[t]$ can be zero. Then, for all $\epsilon > 0$, we can construct sequential portfolios $\tilde{\boldsymbol{b}}_u^c[n]$ with complexity linear in n^m such that for any $c = c_{sell} + c_{buy}$, and for all n, k*

$$R^c[n] = \frac{\sup_{\substack{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k+1} \in R_+^m \\ t_1, \ldots, t_k \in \{2, \ldots, n\}}} W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})}{W^c(\boldsymbol{x}^n | \tilde{\boldsymbol{b}}_u)},$$

where $W^c(\boldsymbol{x}^n | \tilde{\boldsymbol{b}}_u)$ is the wealth achieved by the universal algorithm with commission, satisfies

$$\ln \frac{R_[^c n]}{n} \leq (k+1)\frac{\ln((1+c)n)}{n} + k\frac{\ln(1/(1-c))}{n} + 2(k+\epsilon)\frac{\ln(n)}{n} + O(k/n)$$

for any $\mathcal{T}_{k,n}$ representing transition path $(t_1, \ldots, t_k)$ and any k, such that $\tilde{\boldsymbol{b}}_u[t]$ does not depend on $\mathcal{T}_{k,n}$, k or n.

Theorem 1 states that the logarithm of the regret of the universal sequential portfolios $\tilde{\boldsymbol{b}}_u^c[t]$ is within $O((k+1)\ln(n(1+c)))$ of the best batch piecewise constant rebalanced portfolios with k transitions (tuned to the underlying sequence in hindsight), uniformly, for every sequence of price relatives $\boldsymbol{x}^n$ even under the transaction cost c. The theorem readily generalizes to the case when we compete against a finite set of portfolios in each segment and portfolios with side-information.

III. PROOF AND IMPLEMENTATION

Outline of Proof of Theorem 1: The proof of the theorem closely follows [2], hence, we present only an outline. For each possible transition path $\mathcal{T}_{k,n}$ representing $(t_1, \ldots, t_k)$ with k transitions and data length n, we consider a family of portfolios, each with its own set of constant vectors $\boldsymbol{B}_k = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k+1}]^T$ where each $\boldsymbol{b}_i$ represents a constant portfolio vector for the ith region. For each pairing of $\mathcal{T}_{k,n}$ and $\boldsymbol{B}_k$, we define the wealth of this algorithm as $W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$ with transaction cost $c = c_{sell} + c_{buy}$. Given any $\mathcal{T}_{k,n}$, the competing algorithm with best constant rebalanced portfolios in each region assigns to $\boldsymbol{x}^n$ the largest wealth, i.e., $W^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n}, \boldsymbol{B}_k{}^*)$. Maximizing $W^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n}, \boldsymbol{B}_k{}^*)$ over all $\mathcal{T}_{k,n}$ (with k transitions) yields $W^c(\boldsymbol{x}^n \mid \mathcal{T}^*_{k,n}, \boldsymbol{B}_k{}^*) \triangleq \sup_{\mathcal{T}_{k,n}} W^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n}, \boldsymbol{B}_k{}^*)$. We will demonstrate a sequential algorithm which achieves $W(\boldsymbol{x}^n \mid \boldsymbol{B}_k{}^*, \mathcal{T}_{k,n}{}^*)$ given any k and n, and without a priori knowledge of k or n. We will accomplish this result using a double mixture approach. First we will demonstrate an algorithm achieving the performance of the competing algorithm with the best constant rebalanced portfolios in each region given any $\mathcal{T}_{q,n}$, i.e., $W(\boldsymbol{x}^n \mid \mathcal{T}_{q,n}, \boldsymbol{B}^*_q)$. Then we will show that a proper weighted combination of all such algorithms over all $\mathcal{T}_{q,n}$, $q = 1, \ldots, n$, can be used to find a sequential algorithm that will achieve $W(\boldsymbol{x}^n \mid \mathcal{T}^*_{k,n}, \boldsymbol{B}^*_q)$ for any k.

For any given $\mathcal{T}_{k,n}$, the wealth achieved by the algorithm with the best constant rebalanced portfolios in each region, $W(\boldsymbol{x}^n \mid \mathcal{T}_{k,n}, \boldsymbol{B}_k{}^*)$, can be asymptotically obtained by using the sequential portfolio assignment algorithm [5] (which is universal with respect to the class of all constant rebalanced portfolios under transaction cost), independently for each segment, i.e., apply $\tilde{\boldsymbol{b}}^c_{t_{i-1}}[t]$ between time t_{i-1} up to t_i, where $\tilde{\boldsymbol{b}}^c_{t_{i-1}}[t]$ is the algorithm from [5] with an mth order uniform distribution, that uses the price relative sequence starting from time t_{i-1} up to time t, $t < t_i$. In each segment this universal algorithm achieves the performance of the best constant rebalanced portfolio for that region with transaction cost c. Hence given $\mathcal{T}_{k,n}$, using $\tilde{\boldsymbol{b}}^c_{t_{i-1}}[t]$ in each segment defines a sequential algorithm with achived wealth $\tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n})$ However, we also need to account for the cost of switching the portfolios at each of the transition times. At the end of each segment i, we need to adjust the portfolio vector $\tilde{\boldsymbol{b}}^c_{t_{i-1}}[t_i - 1]$ to the uniform portfolio vector $\frac{\boldsymbol{1}}{m}$ for the start of the next segment. For constant rebalanced portfolios in each segment boundary, we need to adjust $\boldsymbol{b}_i$ to $\boldsymbol{b}_{i+1}$ which is already included in $W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$ by definition. Hence, we need to account for the difference between cost of adjusting $\tilde{\boldsymbol{b}}^c_{t_{i-1}}[t_i - 1]$ to $\frac{\boldsymbol{1}}{m}$ in $\tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n})$ with respect to cost of adjusting $\boldsymbol{b}_i$ to $\boldsymbol{b}_{i+1}$ in $W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$ to compare $\tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n})$ and $W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$. However, at each transition time, even in the worst case to get an upper bound, we can only loose c percent of our wealth while adjusting $\tilde{\boldsymbol{b}}^c_{t_{i-1}}[t_i - 1]$ to $\frac{\boldsymbol{1}}{m}$ in $\tilde{W}^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$ and no loss for the constant rebalanced portfolios while adjusting $\boldsymbol{b}_i$ to $\boldsymbol{b}_{i+1}$, i.e., $\boldsymbol{b}_i = \boldsymbol{b}_{i+1}$, in $W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})$. Hence, applying algorithm from [5] for each segment and further reducing the wealth by $\ln\left((1-c)^k\right)$ (i.e., scaling down the wealth by $(1-c)$ in the worst case per transition) results

$$\ln(\tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n})) \geq \ln\{W^c(\boldsymbol{x}^n \mid \boldsymbol{B}_k, \mathcal{T}_{k,n})\} \tag{1}$$
$$- \sum_{i=1}^{k+1} \ln\left(1 + (1+c)(t_i - t_{i-1})\right) - \ln\left(\frac{1}{(1-c)^k}\right) + O(1).$$

Hence given $\mathcal{T}_{k,n}$, using $\tilde{\boldsymbol{b}}^c_{t_{i-1}}[t]$ in each segment defines a sequential algorithm that asymptotically achieves the performance of the algorithm with the best constant rebalanced portfolio for each segment under transaction cost c. For all $\mathcal{T}_{k,n}$ and k, we construct a similar sequential algorithm yielding a total of 2^{n-1} such sequential portfolio assignment algorithms.

We then define a weighted mixture of the wealth achieved by all such sequential predictors over all possible $\mathcal{T}_{k,n}$ and k

$$\tilde{W}^c_u(\boldsymbol{x}^n) \triangleq \sum_{k=0}^{n-1} \sum_{\mathcal{T}_{k,n}} P(\mathcal{T}_{k,n}) \tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n}), \tag{2}$$

with a suitable prior over the partitions $\mathcal{T}_{k,n}$, $P(\mathcal{T}_{k,n})$. For any transition path $\mathcal{T}_{k,n}$, the weighting (or the assigned probability $P(\mathcal{T}_{k,n})$ to each transition path) would be non-negative and would satisfy $\sum_{k=0}^{n-1} \sum_{\mathcal{T}_{k,n}} P(\mathcal{T}_{k,n}) = 1$. Now we have a total wealth achieved by the class of all possible constant portfolios and for all possible transition paths. By Equation (2), we can conclude that this achieved wealth satisfies $\ln \tilde{W}^c_u(\boldsymbol{x}^n) \geq \ln P(\mathcal{T}_{k,n}) + \ln \tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n})$, for any transition path $\mathcal{T}_{k,n}$, since $\tilde{W}^c_u(\boldsymbol{x}^n) \geq P(\mathcal{T}_{k,n}) \tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}_{k,n})$.

Clearly, the assigned probability or weight for $\mathcal{T}_{k,n}$ directly contributes to the regret as $\ln(P(\mathcal{T}_{k,n}))$ over the best batch algorithm given any path. Hence, it is desirable that the weight of the "best path" be assigned as large as possible. This weight assignment should also be constructed so that the overall weighting and the resulting portfolio assignment algorithm can be sequentially computable. We will use a weighting method based on the Krichevsky-Trofimov (KT) weighting used in [1], [6], [7]. Using this assigned probability, we have, $\ln P(\mathcal{T}_{k,n}) \leq (k + \epsilon)\ln(n) + O(k)$ for all $\epsilon > 0$.

We now have a method of selecting portfolios that achieves, to first order in the exponent, the same wealth as that achieved by the best batch constant portfolios, for any partition $\mathcal{T}_{k,n}$. In this sense, $\tilde{W}^c_u(\boldsymbol{x}^n)$ is a "universal" portfolio selection method. It still remains to find a sequential algorithm whose wealth assignment is as large as $\tilde{W}_u^c(\boldsymbol{x}^n)$, the wealth achieved by all sequential algorithms represented in Equation (1) weighted by the corresponding $P(\mathcal{T}_{k,n})$, $k = 1, \ldots, n$.

We are now ready to find the actual universal portfolio assignment algorithm. By definition $\tilde{W}_u^c(\boldsymbol{x}^n) = \prod_{t=1}^n \frac{\tilde{W}_u^c(\boldsymbol{x}^t)}{\tilde{W}_u^c(\boldsymbol{x}^{t-1})}$. If we look each term in the product closely, we observe that

$$\frac{\tilde{W}_u(\boldsymbol{x}^n)}{\tilde{W}_u(\boldsymbol{x}^{n-1})} = \tilde{\boldsymbol{b}}_u^T[t]\boldsymbol{x}[t].$$

for a strongly sequential portfolio assignment algorithm $\tilde{\boldsymbol{b}}_u[t]$. Hence, $\tilde{\boldsymbol{b}}_u[t]$ is the required portfolio vector at each time t. Nevertheless, in this form the sequential algorithm $\tilde{\boldsymbol{b}}_u[t]$ requires 2^n different sequential algorithms to run in parallel on the sequence of price relatives. We will now demonstrate that this sequential portfolio assignment algorithm can be calculated efficiently by using a linear transition diagram, similar to that used in [1] (after assigning appropriate weights to each branch).

At each time n, we divide the set of all possible paths $\mathcal{T}_{k,n}$, $k = 1, \ldots, n$ into n disjoint sets. We label each set by a state variable s_n representing the most recent transition of a corresponding path within the period $1 \le t \le n$ such that for any $\mathcal{T}_{k,n}$, $s_n = t_k$. Given n, there can be at most n states $s_n = 1, \ldots, n$. At time n, all transition paths with the same last transition instant, $t_k = s$, are represented by the state $s_n = s$. We then define $W_n^c(s_n = s, \boldsymbol{x}^n)$ as the achieved wealth of all sequential algorithms at state s_n at time n. Here, $W_n^c(s_n = s, \boldsymbol{x}^n)$ is the weighted sum of all the wealth achieved by the sequential algorithms as given in [1] whose transition paths ended up at $s_n = s$; i.e., for all paths $\mathcal{T}^{'}$ such that the last transition was at $s_n = s$

$$W_n^c(s_n = s, \boldsymbol{x}^n) \triangleq \sum_{\mathcal{T}^{'}:s_n=s} P(\mathcal{T}^{'})\tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}^{'}).$$

Since the states partition the set of paths $\mathcal{T}_{k,n}$, $\tilde{W}_u^c(\boldsymbol{x}^n) = \sum_{\mathcal{T}} P(\mathcal{T})\tilde{W}^c(\boldsymbol{x}^n \mid \mathcal{T}) = \sum_{s_n=1}^n W_n^c(s_n, \boldsymbol{x}^n)$. From Property (3), any efficient implementation of $\tilde{W}_u^c(\boldsymbol{x}^n)$ by combining the certain paths in states can only improve the performance due to occasional in trading. To obtain a closed form expression for $\frac{\tilde{W}_u^c(\boldsymbol{x}^n)}{\tilde{W}_u^c(\boldsymbol{x}^{n-1})}$, we show that $W_n^c(s_n = s, \boldsymbol{x}^n)$ can be calculated recursively by using the linear transition diagram as in [1]. As such, state s_n represents the most recent transition within the period $1 \le t \le n$. After some algebra, it can be shown that [2], the final universal algorithm is given as

$$\tilde{\boldsymbol{b}}_u[n] = \sum_{s_{n-1}=1}^{n-1} \mu^c(s_{n-1})$$

$$\left\{ P_{tr}(s_n = s_{n-1} \mid s_{n-1})\tilde{\boldsymbol{b}}^c_{s_{n-1}}[n] + P_{tr}(s_n = n \mid s_{n-1})\frac{\mathbf{1}}{m} \right\}.$$

Stocks	c=0.005	c=0.005	c=0.01	c=0.01
	CRP	Wtr	CRP	Wtr
Ir&Kn	33.9	59.0	28.6	54.4
Co&Me	65.9	91.7	58.2	87.0
Fi&Me	25.2	33.1	22.5	31.8

Table I. Performance on historical stock pairs.

where $\mathbf{1}$ is a vector of size $(m \times 1)$ of all ones, $\tilde{\boldsymbol{b}}^c_{s_{n-1}}[n]$ is Cover's algorithm that started at time s_{n-1} and the weights $\mu^c(s_{n-1})$ are defined as

$$\mu^c(s_{n-1} = j) \triangleq \frac{W_{n-1}^c(s_{n-1} = j, \boldsymbol{x}^{n-1})}{\sum_{s_{n-1}=1}^{n-1} W_{n-1}^c(s_{n-1}, \boldsymbol{x}^{n-1})}$$

and are a form of performance-weighting for the states s_{n-1}. $P_{tr}(s_n \mid s_{n-1})$ are the transition probabilities from state s_{n-1} to state s_n that are needed to sequentially calculate the path weights where $P_{tr}(s_n \mid s_{n-1})$ will be different depending on the weighting used for $P(\mathcal{T}_{k,n})$. ■.

IV. SIMULATIONS AND CONCLUSIONS

We compare the performance of our algorithm on benchmark historical data. These stocks are collected over 25 years ending in 1985. The data set includes stocks for Iroquois, Kinark, Commercial Metals, Fisch and Meico. We present the results in Table I, for $c = 0.005$ and $c = 0.01$. In the table, wealth achieved by algorithm from [5] is displayed under CRP and by our algorithm under Wtr, resulting after 1 dollar of initial investment. We have presented a strongly sequential algorithm that can achieve the wealth of the best piece-wise constant rebalanced portfolio chosen in hindsight under transactions costs. This algorithm neither requires knowledge of the number of piecewise constant regions, nor the data length, yet asymptotically outperforms the best such portfolios chosen with complete knowledge of the data.

V. REFERENCES

[1] F. M. J. Willems, "Coding for a binary independent piecewise-identically-distributed source," *IEEE Trans. on Info. Theory*, vol. 42, pp. 2210-2217, 1996.

[2] S. S. Kozat and A. C. Singer, "Universal constant rebalanced portfolios with switching," in *ICASSP*, 2007, pp. 1129-1132.

[3] T. Cover and E. Ordentlich, "Universal portfolios with side-information," *IEEE Trans. on Info. Theory*, vol. 42, no. 2, pp. 348-363, 1996.

[4] D. P. Helmbold, R. E. Schapire, Y. Singer, andM. K.Warmuth, "Online portfolio selection usingmultiplicative updates," *Mathematical Finance*, vol. 8, no. 4, pp. 325-347, 1998.

[5] A. Blum and A. Kalai, "Universal portfolios with and without transaction costs," in *COLT*, 1997, pp. 1-13.

[6] G. I. Shamir and N. Merhav, "Low-complexity sequential lossless coding for piecewise-stationary memoryless sources," *IEEE Trans. on Info. Theory*, vol. 45, no. 5, pp. 1498-1519, 1999.

[7] R. E. Krichevsky and V. K. Trofimov, "The performance of universal encoding," *IEEE Trans. on Info. Theory*, vol. 27, pp. 190-207, 1981.

[8] Y. Singer, "Swithcing portfolios," in *Proc. of Conf. on Uncertainty in AI*, 1998, pp. 1498=1519.