Submitted to CIFEr*97.

Optimization of Trading Systems and Portfolios

John Moody and **Lizhong Wu**
Oregon Graduate Institute, CSE Dept.
P.O. Box 91000, Portland, OR 97291–1000
moody@cse.ogi.edu lwu@cse.ogi.edu

Abstract

We propose to train trading systems and portfolios by optimizing objective functions that directly measure trading and investment performance. Rather than basing a trading system on forecasts or training via a supervised learning algorithm using labelled trading data, we train our systems using recurrent reinforcement learning algorithms. The objective functions that we consider as evaluation functions for reinforcement learning are profit or wealth, economic utility, the Sharpe ratio, and our proposed *Differential Sharpe Ratio.* The trading and portfolio management systems require prior decisions as input in order to properly take into account the effects of transactions costs, market impact, and taxes. This temporal dependence on system state requires the use of reinforcement versions of standard recurrent learning algorithms.

We present empirical results in controlled experiments that demonstrate the efficacy of some of our methods. We find that maximizing the differential Sharpe ratio yields more consistent results than maximizing profits, and that both methods outperform a trading system based on forecasts that minimize MSE.

1 Introduction: Objective Functions and Reinforcement Learning for Trading

Many trading systems are optimized using supervised learning, either using labelled trading data or by generating trading signals from price forecasts. Optimizing forecasts typically involves minimizing squared error, while the investor's or trader's ultimate goal is to maximize profits, economic utility or risk-adjusted return. Minimizing forecast error thus corresponds to optimizing an intermediate quantity that is not the ultimate objective of the system and may lead to suboptimal performance. Directly training a system using data labelled with desired trades avoids this intermediate step. The data can be labelled either by an "expert" or by an algorithm designed to constructed an optimal labelling.

Training on labelled data is a two-step procedure. The algorithm for labelling the data attempts to solve the *temporal credit assignment* problem, while subsequently training the system on the labelled data attempts to solve the *structural credit assignment* problem.[1] A system can be optimized to solve both problems simultaneously using *reinforcement learning.* We will adopt this approach here. In reinforcement learning, target outputs are not provided. Rather, the system takes actions (makes trades), receives feedback on its performance (an evaluation signal), and then adjusts its internal parameters to increase its future rewards.

A solution of the structural credit assignment problem (training the system parameters) will generally require using a *recurrent* learning algorithm. Trading system profits depend upon sequences of interdependent decisions, and are

[1] This terminology was proposed in Sutton (1988).

thus path-dependent. Optimal trading decisions when the effects of transactions costs, market impact, and taxes[2] are included require knowledge of the current system state. Including information related to past decisions in the inputs to a trading system results in a *recurrent* decision system.[3] The proper optimization of a recurrent, path-dependent decision system is quite different from the simple supervised optimization techniques used for direct forecasts or for labelled trading data.

The reinforcement learning analogs of recurrent learning algorithms required to train our proposed systems include both off-line (batch) training algorithms like back propagation through time or on-line (adaptive) algorithms like real time recurrent learning or dynamic backpropagation.

2 Structure and Optimization of Traders

2.1 Single Asset with Discrete Position Size

In this section, we consider trading objective functions for trading systems that trade a single security with price series z_t. The trader is assumed to take only long, neutral, or short positions $F_t \in \{-1, 0, 1\}$ of constant magnitude. A conservative strategy for stock or bond investments might be restricted to $F_t \in \{0, 1\}$, while a simplified reversal trading strategy could have no neutral state $F_t \in \{-1, 1\}$. The constant magnitude assumption can be easily relaxed to enable better risk control. (This will be discussed later.) The position F_t is established or maintained at the end of each time interval t, and is reassessed at the end of period $t + 1$. A trade is thus possible at the end of each time period, although nonzero trading costs will discourage excessive trading.

In order to properly incorporate the effects of transactions costs, market impact, and taxes in a trader's decision making, the trader must have internal state information and must therefore be recurrent. An example of a single asset trading system that could take into account transactions costs and market impact would be one with the following decision function:

$$F_t = F(\theta_t; F_{t-1}, I_t) \quad \text{with} \quad I_t = \{z_t, z_{t-1}, z_{t-2}, \ldots; y_t, y_{t-1}, y_{t-2}, \ldots\} \tag{1}$$

where θ_t denotes the (learned) system parameters at time t and I_t denotes the information set at time t, which includes present and past values of the price series z_t and an arbitrary number of other external variables denoted y_t. More general decision functions could include the current trade's profit / loss (for capturing capital gains tax effects), past positions F_{t-s} and past profits and losses (in analogy with the moving average components of $ARIMA$ models), and other factors.

2.2 Optimizing Profit and Wealth

Trading systems can be optimized by maximizing objective functions $U()$ such as profit, wealth, utility functions of wealth, or performance ratios like the Sharpe ratio. Sharpe ratios will be discussed in section 5. The simplest and most natural objective function for a risk-insensitive trader is profit. We consider two cases: additive and multiplicative. The transactions cost rate (per share, per contract, or per dollar amount, depending on context) is denoted δ.

Additive profits are appropriate to consider if each trade is for a fixed number of shares or contracts of security z_t. This is often the case, for example, when trading small futures accounts or when trading standard US$ FX contracts in dollar-denominated foreign currencies. With the definitions $r_t = z_t - z_{t-1}$ and $r_t^f = z_t^f - z_{t-1}^f$ for the price returns of a risky (traded) asset and a risk-free asset (like T-Bills) respectively, the additive profit accumulated over T time periods with trading position size $\mu > 0$ is then defined as:

$$P_T = \sum_{t=1}^{T} R_t = \mu \sum_{t=1}^{T} \left\{ r_t^f + F_{t-1}(r_t - r_t^f) - \delta |F_t - F_{t-1}| \right\} \tag{2}$$

with $P_0 = 0$ and typically $F_T = F_0 = 0$. We have implicitly defined the return for the trade completed at time t as R_t. Note that the transaction costs for switching between short to long positions are twice those for switching between neutral and long/short positions. Equation (2) holds for continuous quantities also. The wealth is defined as $W_T = W_0 + P_T$.

[2] For brevity, we omit further discussion of market impact and tax effects.

[3] Here, recurrence refers to the nature of the algorithms required to optimize the system. For example, optimizing a feed forward NAR(p) model for one-step-ahead prediction does not require a recurrent learning algorithm, while optimizing the same NAR(p) model to perform iterated predictions *does*. The fact that a forecast or decision is made by a feedforward, non-recurrent network does not mean that optimizing it correctly can be done with a standard, non-recurrent training procedure.

Multiplicative profits are appropriate when a fixed fraction of accumulated wealth $\nu > 0$ is invested in each long or short trade. Here, $r_t = (z_t/z_{t-1} - 1)$ and $r_t^f = (z_t^f/z_{t-1}^f - 1)$. If no short sales are allowed and the leverage factor is set fixed at $\nu = 1$, the wealth at time T is:

$$W_T = W_0 \prod_{t=1}^{T} \{1 + R_t\} = W_0 \prod_{t=1}^{T} \left\{1 + (1 - F_{t-1})r_t^f + F_{t-1}r_t\right\} \{1 - \delta|F_t - F_{t-1}|\} \ , \tag{3}$$

where R_t is the return realized for the period ending at time t. In this case, the profit is $P_T = W_T - W_0$. If short sales or leverage $\nu \neq 1$ are allowed, then the correct expression depends in detail on the timing of the sequence of trades. For brevity, we omit the discussion for this case.

3 Structure and Optimization of Portfolios

3.1 Portfolios: Continuous Quantities of Multiple Assets

When the risk-free rate of return r_t^f is included in single risky-asset trading models above, one actually has a simple two asset portfolio. For trading multiple assets in general (typically including a risk-free instrument), a multiple output trading system is required. Denoting a set of m markets with price series $\{\{z_t^a\} : a = 1, \ldots, m\}$, the market return r_t^a for price series z_t^a for the period ending at time t is defined as $((z_t^a/z_{t-1}^a) - 1)$. Defining portfolio weights of the a^{th} asset as $F^a()$, a trader that takes only long positions must have portfolio weights that satisfy:

$$F^a \geq 0 \quad \text{and} \quad \sum_{a=1}^{m} F^a = 1 \ . \tag{4}$$

With these constraints, standard Markowitz mean-variance portfolio optimization is a quadratic programming problem. However, when optimizing the parameters of a nonlinear trading system, portfolio optimization becomes a nonlinear programming problem.

One approach to imposing the constraints on the portfolio weights (4) without requiring that a constrained optimization be performed is to use a trading system that has softmax outputs:

$$F^a() = \frac{\exp[f^a()]}{\sum_{b=1}^{m} \exp[f^b()]} \quad \text{for} \quad a = 1, \ldots, m \ . \tag{5}$$

Here, the $f^a()$ could be linear or more complex functions of the inputs, such as a two layer network with linear outputs. Such a trading system can be optimized using unconstrained optimization methods. Denoting the set of raw and normalized outputs collectively as $\mathbf{f}()$ and $\mathbf{F}()$ respectively, a recursive trader will have structure $\mathbf{F}_t =$ softmax $\{\mathbf{f}_t(\theta_{t-1}, \mathbf{F}_{t-1}, I_t)\}$.

3.2 Profit and Wealth for Portfolios

When multiple assets are considered, the effective portfolio weightings change with each time step due to price movements. Thus, maintaining constant or desired portfolio weights requires that adjustments in positions be made at each time step. The wealth after T periods for a portfolio trading system is

$$W_T = W_0 \prod_{t=1}^{T} \{1 + R_t\} = W_0 \prod_{t=1}^{T} \left\{ \left(\sum_{a=1}^{m} F_{t-1}^a \frac{z_t^a}{z_{t-1}^a} \right) \left(1 - \delta \sum_{a=1}^{m} |F_t^a - \tilde{F}_t^a| \right) \right\} \ , \tag{6}$$

where $\tilde{F}_t^a$ is the effective portfolio weight of asset a before readjusting, defined as

$$\tilde{F}_t^a = \frac{F_{t-1}^a (z_t^a / z_{t-1}^a)}{\sum_{b=1}^{m} F_{t-1}^b (z_t^b / z_{t-1}^b)} \ , \tag{7}$$

and we have defined the trading returns R_t implicitly. In (6), the first factor in the curly brackets is the increase in wealth over the time interval t prior to rebalancing to achieve to newly specified weights F_t^a. The second factor is the reduction in wealth due to the rebalancing costs. The profit after T periods is $P_T = W_T - W_0$.

4 Optimizing Economic Utility

The optimization of profit and wealth described above assumes that investors are insensitive to risk. However, most investors are more sensitive to losses than to gains, and are therefore willing to sacrifice some potential gains in order to have less risk of loss. There is also an intrinsic asymmetry between percentage losses and gains: a 25% loss must be followed by 33% gain in order to break even. Utility functions of wealth $U(W)$ can capture various kinds of risk/reward preferences.

In 1738, Daniel Bernoulli proposed the logarithmic utility $U(W_t) = \log W_t$ (see Bernoulli (1954)). A more general class of utility functions that capture varying degrees of risk sensitivity are the gamma utilities:

$$\begin{aligned} U_\gamma(W_t) &= W_t^\gamma/\gamma \quad \text{for } \gamma \neq 0 \\ U_0(W_t) &= \log W_t \quad \text{for } \gamma = 0 \ . \end{aligned} \tag{8}$$

These power law utilities have constant relative risk aversion, defined as

$$\mathcal{R}(W) = -\frac{d \log U'(W)}{d \log W} = 1 - \gamma \ . \tag{9}$$

The case $\gamma = 1$ is risk-neutral, and $U_1(W)$ corresponds to absolute wealth or profit (equation 2), while utilities with $\gamma > 1$ describe risk-seeking or "thrill-seeking" behavior. Most investors have risk-averse utility functions ($\gamma < 1$), with smaller values of γ corresponding to greater sensitivity to loss. The limit $\gamma \mapsto -\infty$ corresponds to absolute risk aversion.

The sensitivity to risk on a per time period or per trade basis can be seen by considering a second order taylor expansion of the contribution of R_t to $U_\gamma(W_T)$:

$$\begin{aligned} U_\gamma(1+R_t) &\approx \frac{1}{\gamma} + R_t + \frac{\gamma - 1}{2} R_t^2 \quad \text{for} \quad \gamma \neq 0 \\ U_\gamma(1+R_t) &\approx 0 + R_t - \frac{1}{2} R_t^2 \quad \text{for} \quad \gamma = 0 \end{aligned} \tag{10}$$

Taking the expectation value of $U_\gamma(1 + R_t)$, and defining risk to be $E(R_t^2)$, we see that for $\gamma > 1$, risk is positively-weighted, while for $\gamma < 1$, risk is negatively-weighted.

Although we have focussed on γ utilities here due to their property of constant relative risk aversion, many other classes of utility functions are worthy of consideration.

5 The Sharpe Ratio and the *Differential* Sharpe Ratio

5.1 Optimizing the Sharpe Ratio

The Sharpe ratio is a measure of risk-adjusted return (Sharpe(1966; 1994)). Denoting as before the trading system returns for period t (including transactions costs) as R_t, the Sharpe ratio is defined to be

$$S_T = \frac{\text{Average}(R_t)}{\text{Standard Deviation}(R_t)} \tag{11}$$

where the average and standard deviation are estimated over returns for periods $t = \{1, \ldots, T\}$. A trading system can be trained to maximize the Sharpe Ratio as follows. First, we define the Sharpe ratio for n returns R_i in terms of estimates of the first and second moments of the returns distributions:

$$S_n = \frac{A_n}{K_n (B_n - A_n^2)^{1/2}} \tag{12}$$

with

$$A_n = \frac{1}{n} \sum_{i=1}^{n} R_i \quad B_n = \frac{1}{n} \sum_{i=1}^{n} R_i^2 \quad K_n = \left(\frac{n}{n-1} \right)^{1/2} \ . \tag{13}$$

The normalizing factor K_n is required for an unbiased estimate of the standard deviation, but is not relevant for optimization. The derivative with respect to the system parameters (using scalar notation for θ, even though it's generally a vector) is:

$$\begin{aligned}\frac{dS_n(\theta)}{d\theta} &= \sum_{i=1}^{n}\left\{\frac{dS}{dA_n}\frac{dA_n}{dR_i}+\frac{dS}{dB_n}\frac{dB_n}{dR_i}\right\}\left\{\frac{dR_i}{dF_i}\frac{dF_i}{d\theta}+\frac{dR_i}{dF_{i-1}}\frac{dF_{i-1}}{d\theta}\right\}\\ &= \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{B_n-A_nR_i}{K_n(B_n-A_n^2)^{3/2}}\right\}\left\{\frac{dR_i}{dF_i}\frac{dF_i}{d\theta}+\frac{dR_i}{dF_{i-1}}\frac{dF_{i-1}}{d\theta}\right\}\ . \end{aligned} \tag{14}$$

The above expression as written with scalar F_i applies to the traders of a single risky asset described in section 2, but can be trivially generalized to the vector case for portfolios, as described in section 3.

The system can be optimized in batch mode by repeatedly computing the value of S_n on forward passes through the data and adjusting the trading system parameters by using gradient descent (with learning rate ρ)

$$\Delta\theta = \rho\frac{dS_n(\theta)}{d\theta} \tag{15}$$

or some other optimization method. A simple incremental optimization might consider only the term in (14) dependent on the last realized return R_n. Note that the quantities $dF_i/d\theta$ are *total derivatives* that depend upon the entire sequence of previous trades. To correctly compute and optimize these total derivatives requires that a recurrent algorithm like BPTT, RTRL, or dynamic backpropagation be used.

5.2 Running and Moving Sharpe Ratios

In order to facilitate on-line learning, an incremental Sharpe ratio is required. First, we define a *running Sharpe ratio* by making use of recursive estimates of the first and second moments of the returns distributions:

$$A_n = \frac{1}{n}R_n + \frac{n-1}{n}A_{n-1} \quad \text{and} \quad B_n = \frac{1}{n}R_n^2 + \frac{n-1}{n}B_{n-1} \tag{16}$$

with $A_0 = B_0 = 0$. Next, we extend this definition to an *exponential moving average Sharpe ratio* on time scale η^{-1} by making use of moving average estimates of the first and second moments of the returns distributions:

$$S_\eta(t) = \frac{A_\eta(t)}{K_\eta(B_\eta(t) - A_\eta^2(t))^{1/2}} \tag{17}$$

with

$$A_\eta(t) = \eta R_t + (1-\eta)A_\eta(t-1) \quad B_\eta(t) = \eta R_t^2 + (1-\eta)B_\eta(t-1) \quad K_\eta = \left(\frac{1-\eta/2}{1-\eta}\right)^{1/2} \tag{18}$$

initialized with $A_\eta(0) = B_\eta(0) = 0$. In (17), the normalization factor K_η is required for an unbiased estimate of the moving standard deviation. For purposes of trading system optimization, however, this constant factor can be ignored.

5.3 *Differential* Sharpe Ratios for On-Line Optimization

While both the running and moving Sharpe ratios can be used to optimize trading systems in batch or off-line mode, proper on-line learning requires that we compute the influence on the Sharpe ratio of the return at time t. With the running or moving Sharpe ratios defined above, we can derive *Differential Sharpe Ratios* for on-line optimization of trading system performance. It is advantageous to use on-line performance measures to (1) speed the convergence of the learning process (since parameter updates can be done *during* each forward pass through the training data), and (2) to adapt to changing market conditions during live trading.

The update equations for the exponential moving estimates can be rewritten

$$\begin{aligned} A_\eta(t) &= A_\eta(t-1) + \eta\Delta A_\eta(t) = A_\eta(t-1) + \eta(R_t - A_\eta(t-1))\\ B_\eta(t) &= B_\eta(t-1) + \eta\Delta B_\eta(t) = B_\eta(t-1) + \eta(R_t^2 - B_\eta(t-1))\ , \end{aligned} \tag{19}$$

where we have implicitly defined the update quantities $\Delta A_\eta(t)$ and $\Delta B_\eta(t)$. Treating $A_\eta(t-1)$, $B_\eta(t-1)$, and K_η as numerical constants, note that η in the update equations (19) controls the magnitude of the influence of the return R_t on the Sharpe ratio $S_\eta(t)$. Taking $\eta \mapsto 0$ turns off the updating, and making η positive turns it on.

With this in mind, we can obtain a *differential* Sharpe ratio by expanding (17) to first order in η:[4]

$$S_\eta(t) \approx S_\eta(t-1) + \eta \frac{dS_\eta(t)}{d\eta}|_{\eta=0} + O(\eta^2) \ . \tag{20}$$

Noting that only the first order term in this expansion depends upon the return R_t at time t (through $\Delta A_\eta(t)$ and $\Delta B_\eta(t)$), we define the *Differential Sharpe Ratio* as:

$$D_\eta(t) \equiv \frac{dS_\eta(t)}{d\eta} = \frac{B_\eta(t-1)\Delta A_\eta(t) - \frac{1}{2}A_\eta(t-1)\Delta B_\eta(t)}{(B_\eta(t-1) - A_\eta(t-1)^2)^{3/2}} \ . \tag{21}$$

The influences of risk and return on the differential Sharpe ratio are readily apparent. The first term in the numerator is positive if R_t exceeds the moving average $A_\eta(t-1)$, while the second term is negative if R_t^2 exceeds the moving average $B_\eta(t-1)$. Assuming that $A_\eta(t-1) > 0$, the largest possible improvement in $D_\eta(t)$ occurs when

$$R_t^* = B_\eta(t-1)/A_\eta(t-1) \ . \tag{22}$$

Thus, the Sharpe ratio actually penalizes returns larger than R_t^*.

The Sharpe ratio is not a standard economic utility function, since it's value depends not just on current wealth (or current changes in wealth), but also on past performance. This can be seen by comparing the numerator of the differential Sharpe ratio (21)

$$-\frac{1}{2}A_\eta(t-1)B_\eta(t-1) + B_\eta(t-1)R_t - \frac{1}{2}A_\eta(t-1)R_t^2 \tag{23}$$

with (10). In a sense, the Sharpe ratio is an adaptive utility function, since the relative weightings of risk and return depend on past performance as measured by A_η and B_η.

Note that a second expression for a differential Sharpe ratio can be obtained by expanding $S_\eta(t)$ of equation (17) in a taylor series about $R_t = A_\eta(t-1)$ to second order in $\Delta A_\eta(t)$, and expressing the coefficients in terms of information available at time $t-1$. Using simplified notation, the result is:

$$\begin{aligned} S_t &\approx \frac{1}{(1-\eta)^{1/2}}\left\{S_{t-1} + \frac{\eta}{K_\eta}D_\eta(t) + O((\Delta A_t)^3)\right\} \\ D_\eta(t) &\equiv \frac{(B_{t-1} - A_{t-1}^2)\Delta A_t - \frac{1}{2}A_{t-1}(\Delta A_t)^2}{(B_{t-1} - A_{t-1}^2)^{3/2}} \ . \end{aligned} \tag{24}$$

The numerators of (21) and (24) differ only by a constant. As for (21), this expression too is maximized when $R_t^* = B_{t-1}/A_{t-1}$. Whether optimizing a trading system with (21) or (24), the relevant derivatives have the same simple form:

$$\frac{dD_\eta(t)}{dR_t} = \frac{B_{t-1} - A_{t-1}R_t}{(B_{t-1} - A_{t-1}^2)^{3/2}} \ . \tag{25}$$

6 Empirical Results

We have tested techniques for optimizing both profit and the Sharpe ratio in a variety of settings. In this section, we present results for optimizing three simple trading systems using data generated for an artificial market. The systems are short/long trading systems with recurrent state similar to that described in section 2. Two of the systems are trained via reinforcement learning and RTRL to maximize the differential Sharpe ratio (21) or profit. These two systems are compared to a third trading system built on a forecasting system that minimizes forecast MSE.

In our simulations, we find that maximizing the differential Sharpe ratio yields more consistent results than maximizing profits, and that both methods outperform the trading system based on forecasts.

6.1 Data

We generated a log price series as a random walk with autoregressive trend process. The two parameter model is thus:

$$\begin{aligned} p(t) &= p(t-1) + \beta(t-1) + k * \delta(t) & (26) \\ \beta(t) &= \alpha * \beta(t-1) + \gamma(t) \ , & (27) \end{aligned}$$

[4] Again, we treat $A_\eta(t-1)$, $B_\eta(t-1)$, and K_η as constants. As a technical point, differentiation of K_η with respect to η can be avoided by considering an expansion with terms $K_\eta^{-1}d^m(K_\eta S_\eta(t))/d\eta^m$.

where α and k are constants, and $\delta(t)$ and $\gamma(t)$ are normal random deviates with zero mean and unit variance. We defined the artificial price series as

$$z(t) = \exp\left(\frac{p(t)}{R}\right) \tag{28}$$

where R is a scale defined as the range of $p(t)$: $\max(p(t)) - \min(p(t))$ over a simulation with 10,000 samples.[5] In our current simulation, we set $\alpha = 0.9$ and $k = 3$. The artificial price series are trending on short time scales and have a high level of noise. An example of the artificial price series is shown in the top panel of Figure 1.

6.2 Simulated Trading Results

Figure 1 shows results for a single simulation of an artificial market as described above. The trading system is initialized randomly at the beginning, and adapted using real-time recurrent learning to optimize the differential Sharpe ratio (21). The transactions costs are fixed at a half percent during the whole real-time learning and trading process. Transient effects of the initial learning while trading process can be seen in the first 2000 time steps of figure 1.

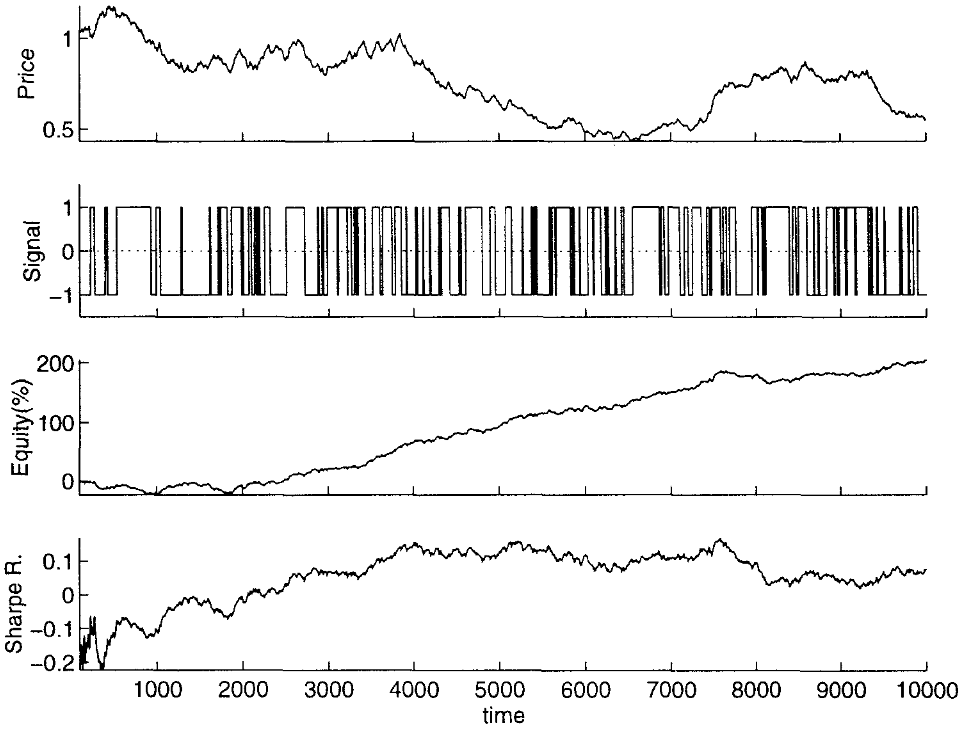

Figure 1: Artificial prices (top panel), trading signals (second panel), cumulative sums of profits (third panel) and the differential Sharpe ratio with $\eta = 0.01$ (bottom panel). (Note that the values of the Sharpe ratio are not annualized.) The system performs poorly while learning from scratch during the first 2000 time periods, but its performance remains good thereafter.

Figure 2 compares three kinds of trading systems. "Max.SR" maximizes the differential Sharpe ratio, "Max.Profit" maximizes the cumulative profit, and "Min.MSE" minimizes the mean-squared forecast error. In this comparison, the transaction costs are set to 0.5%. As shown, the "Max.SR" and "Max.Profit" trading systems significantly out-perform the "Min.MSE" systems. "Max.SR" achieves slightly better mean return than "Max.Profit" and is substantially more consistent in its performance over 100 such trials.

[5] This is slightly more than the number of hours in a year (8760), so the series could be thought of as representing hourly prices in a 24 hour artificial market. Alternatively, a series of this length could represent slightly less than five years of hourly data in a market that trades about 40 hours per week.

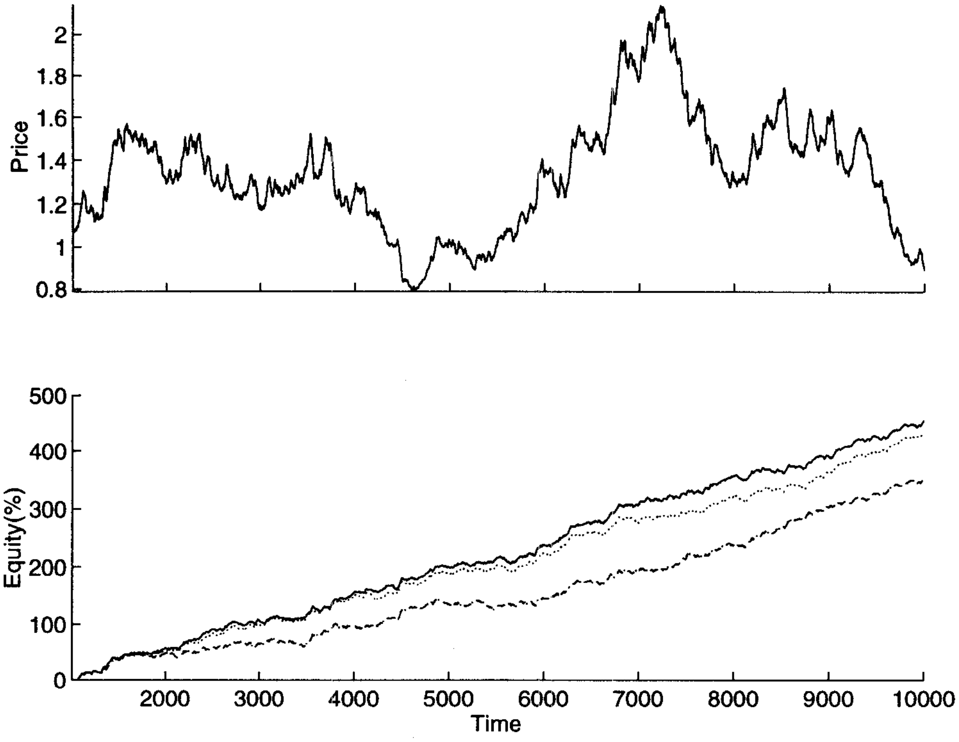

Figure 2: Comparison of the cumulative profits of three trading systems for 9,000 time steps out of sample. The transaction cost is 0.5%. The price series is plotted in the upper panel. The lower panel shows the cumulative sum of median profits over 10 different trials. The solid curve is for the "Max.SR" system, the dotted curve is for the "Max.Profit" system, and the dashed curve is for the "Min.MSE" system.

7 Conclusions and Extensions

Our empirical results to date verify the efficacy of training trading systems via reinforcement learning to maximize profits and the Sharpe Ratio. The results also demonstrate the effectiveness of on-line optimization using our proposed *Differential Sharpe Ratio.* In addition, we have proposed to maximize economic utility functions, such as the Bernoulli and γ utilities. Extensions to the methods presented here include maximizing other economic utility functions and maximizing other performance ratios that use different definitions of risk. These include the Sterling ratio (that uses maximum drawdown) and measures based on the *semi-variance* (Markowitz, 1959) and the *second lower partial moment* (SLPM) (White, 1996). Our ongoing empirical work includes refining, testing, and comparing such methods for optimizing both trading systems and portfolios on both simulated and real data series. Using simulated data for which we have a good understanding enables extensive controlled experiments.

References

Bernoulli, D. (1954), 'Exposition of a new theory on the measurement of risk', *Econometrica* **32**, 23–26.

Markowitz, H. (1959), *Portfolio Selection: Efficient Diversification of Investments*, New York: Wiley.

Sharpe, W. F. (1966), 'Mutual fund performance', *Journal of Business* pp. 119–138.

Sharpe, W. F. (1994), 'The sharpe ratio - properly used, it can improve investment management', *The Journal of Portfolio Management* pp. 49–58.

Sutton, R. S. (1988), 'Learning to predict by the methods of temporal differences', *Machine Learning* **3**, 9–44.

White, H. (1996), Personal communication.