Expert Systems With Applications 87 (2017) 267–279

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An adaptive portfolio trading system: A risk-return portfolio optimization using recurrent reinforcement learning with expected maximum drawdown

Saud Almahdi, Steve Y. Yang*

Financial Engineering Program, School of Business, Stevens Institute of Technology, 1 Castle Point on Hudson, Hoboken, NJ 07030, USA

ARTICLE INFO

Article history:
Received 24 March 2017
Revised 29 May 2017
Accepted 14 June 2017
Available online 15 June 2017

Keywords:
Recurrent reinforcement learning
Expected maximum drawdown
Optimal portfolio rebalancing
Downside risk

ABSTRACT

Dynamic control theory has long been used in solving optimal asset allocation problems, and a number of trading decision systems based on reinforcement learning methods have been applied in asset allocation and portfolio rebalancing. In this paper, we extend the existing work in recurrent reinforcement learning (RRL) and build an optimal variable weight portfolio allocation under a coherent downside risk measure, the expected maximum drawdown, E(MDD). In particular, we propose a recurrent reinforcement learning method, with a coherent risk adjusted performance objective function, the Calmar ratio, to obtain both buy and sell signals and asset allocation weights. Using a portfolio consisting of the most frequently traded exchange-traded funds, we show that the expected maximum drawdown risk based objective function yields superior return performance compared to previously proposed RRL objective functions (i.e. the Sharpe ratio and the Sterling ratio), and that variable weight RRL long/short portfolios outperform equal weight RRL long/short portfolios under different transaction cost scenarios. We further propose an adaptive E(MDD) risk based RRL portfolio rebalancing decision system with a transaction cost and market condition stop-loss retraining mechanism, and we show that the proposed portfolio trading system responds to transaction cost effects better and outperforms hedge fund benchmarks consistently.

1. Introduction

In financial investing, a general goal is to dynamically allocate a set of assets to maximize the returns over time and minimize risk simultaneously. For investors it is essential to be able to invest in a portfolio that can satisfy their preset goals by building an optimal portfolio initially and subsequently rebalancing it optimally. Portfolio theory began with mean-variance optimization by Markowitz (1952) where he proposed portfolio selection by maximizing the expected return while minimizing risk in the form of covariance matrices. Rebalancing a portfolio re-optimizes the weights of the portfolio over a predefined time horizon. The application of dynamic asset allocation using dynamic programming methods was originally introduced by Bertsekas (1995). Due to the curse of dimensionality in dynamic programming, automated self learning algorithms are normally applied by investors and scholars in designing optimal trading strategies instead. The reinforcement learning method is a type of approximate dynamic programming and a subcategory of machine learning introduced by Sutton and Barto (1998), and has been broadly applied by investors and researchers in building strategic asset allocation decision systems (Dempster & Leemans, 2006; Feuerriegel & Prendinger, 2016; Gold, 2003a; Tan, Quek, & Cheng, 2011).

In this paper, we apply the recurrent reinforcement learning (RRL) method with a statistically coherent downside risk adjusted performance objective function to simultaneously generate both buy/sell signals and optimal asset allocation weights. Moody, Wu, Liao, and Saffell (1998) introduced recurrent reinforcement learning in building a trading system where they examined the performance effect between using the Sharp ratio vs. several economic utility functions. They concluded that the Sharpe ratio behaves like an adaptive utility function, and when maximizing the differential Sharpe ratio as immediate rewards in an online learning mode, the Sharpe ratio significantly outperforms the ones maximizing profits directly. Most of the subsequent work (Gold, 2003b; Maringer & Ramtohul, 2010, 2012; Pu et al., 2016) focused on equally weighted portfolios. Although both Moody et al. (1998) and Bertoluzzo and Corazza (2008) mentioned potential drawdown effects on RRL performance, neither thoroughly examined the actual effects.

* Corresponding author.
E-mail addresses: salmahdi@stevens.edu (S. Almahdi), steve.yang@stevens.edu, steve2yang@yahoo.com (S.Y. Yang).

http://dx.doi.org/10.1016/j.eswa.2017.06.023
0957-4174/

Many practitioners tend to adjust the commonly accepted theoretical models to apply them to their particular situations, or develop measures that focus on their specific interests. They often neglect the theoretical aspects or assumptions of their adjustments, such as in the safety-first risk measures (i.e. the Sharpe ratio, the Sortino ratio, the Sterling ratio, and the Calmar ratio). Bhansali (2007) and Zimmermann, Drobetz, and Oertmann (2003) noted that many risk measures based on estimation of covariance matrices using historical data failed notoriously when they are needed the most. They agreed that the difference in volatility and correlations between up and down market environments implies the risk reduction potential is limited leaving them incapable of foreseeing stress-type events. We argue that large drawdowns usually lead to fund redemption, and hence they should lead to very different optimal decisions. In this paper, we extend the variable weight RRL long only approach by Moody and Saffell (2001) to a long-short approach and examine the expected maximum drawdown E(MDD) (Magdon-Ismail & Atiya, 2004) effect on portfolio performance with joint interaction of transaction costs. Magdon-Ismail, Atiya, Pratap, and Abu-Mostafa (2003) and Magdon-Ismail and Atiya (2004) provided a statistically coherent downside risk measure, the Calmar ratio with the expected maximum drawdown, which provides a theoretical base for us to apply this downside risk measure as a differentiable objective function in RRL. This E(MDD) based Calmar ratio (Magdon-Ismail et al., 2003) is distinctly different from the exponential moving average drawdown approach used by Moody and Saffell (2001).

More specifically, we compare the Calmar ratio[1] with the Sharpe ratio where the risk adjusted measure of performance is calculated by the standard deviation of the returns over a predefined time horizon. Furthermore, we use the recurrent reinforcement learning method with two different objective functions through which we incorporate different risk considerations. We show that the recurrent reinforcement learning with variable weight asset allocation gives a superior performance when applied to a set of highly liquid exchange-traded funds (ETF) with various transaction cost considerations over a 5 year period. We also document that when the expected maximum drawdowns are considered, the RRL can generate a superior portfolio to the ones generated by the average deviation performance measure - the Sharpe ratio. This confirms the intuition that a reasonably low MDD is critical to the success of any fund.

In addition, we propose a portfolio allocation and rebalancing system using RRL with E(MDD) as the performance measure, and this trading system jointly considers transaction costs and market conditions to automatically retrain the system parameters to achieve better performances. We show that a trading system with the stop-loss based on market volatility regime is able to make the portfolio endure higher transaction costs in that the stop-loss strategy will exit the market when the volatility is high and retrain the parameters of the signal generating process and generate new signals to reenter the market. Such a trading decision system is adaptive to the market conditions and is more resilient to transaction cost shocks.

The rest of the paper is organized as follows. In Section 2, we review existing work on dynamic portfolio optimization using reinforcement learning methods. We introduce the expected maximum drawdown and its application to RRL in Section 3. We apply the RRL based portfolio rebalancing approach to a set of ETFs to compare the cost effect of the Sharpe ratio vs. the Calmar ratio using RRL in Section 4. Section 5 conducts a final analysis comparing the performance of the proposed risk-return portfolio optimization with that of two hedge fund indices, and Section 6 concludes the study and identifies some future work.

[1] While there exist multiple definitions of the Sterling ratio, it measures return over maximum drawdown-10%, versus the Calmar ratio, which is similar to the Sterling ratio but normally applied to a 3 year period using a maximum drawdown.

2. Literature review

Machine learning algorithms are widely used for financial market prediction and portfolio constructions, especially for automated trading strategies. Sutton, Barto, and Williams (1992) first introduced the reinforcement learning method (Q-learning) and provided its analytically proven capabilities for one class of adaptive optimal control problems. Recurrent reinforcement learning was introduced by Moody et al. (1998) where it was applied to stock trading as a learning algorithm and they extended a single stock trading into a long only portfolio optimization method using the recurrent reinforcement learning where they used the deferential of the Sharpe ratio as the objective function. In Moody and Saffell (2001) with a direct reinforcement alteration, the authors compared their method with Q-learning and temporal difference algorithms using real data and showed that the deferential Sharpe ratio recurrent reinforcement learning system outperforms Q-learning. The researchers also proposed the deferential Sterling ratio as the performance criterion. However, this version of Sterling ratio neutralizes the downside risk through exponential smoothing.

As a result, a number of trading strategies have been proposed based on recurrent reinforcement learning methods to address issues such as different asset classes, transaction cost and market regime change. Gold (2003b) discusses the application of recurrent reinforcement learning in the foreign exchange market and proposed a two layer network. He compared it with a one layer network and found that the one layer network outperformed the two layer network due to noisy financial data. Others added different algorithms to the recurrent reinforcement learning. Maringer and Ramtohul (2010, 2012) added regime switching to the recurrent reinforcement learning where regime switching captures the different movements of the stock price over time. They added an additional regime switching model to the recurrent reinforcement learning to capture the non-linearity of the financial market and proposed two different methods based on the regime switching: threshold recurrent reinforcement learning (TRRL) in Maringer and Ramtohul (2010) and the smooth transition recurrent reinforcement learning (STRRL) in Maringer and Ramtohul (2012). The authors compared TRRL and STRRL with RRL and used different transaction costs to show the performance of the models. They concluded that the regime switching recurrent reinforcement learning matches the normal recurrent reinforcement learning in a dataset having a single regime but it outperforms the RRL when the dataset has distinctly different regime characteristics.

In addition, another strand of literature combines the recurrent reinforcement learning method with other machine learning methods, such as genetic programming and neural networks. Pu et al. (2016) used a genetic algorithm to improve recurrent reinforcement learning for equity trading where the RRL is population-based and the trading system consists of a group of simulation traders. The genetic algorithm (GA) is the selector and the recurrent reinforcement learning is the trading system; the goal is to achieve the optimal combination where the chosen indicators are exported to the recurrent reinforcement learning trading system. The authors find that the GA-RRL system is more stable than the buy and hold strategy but it did not outperform the buy and hold strategy in terms of producing positive Sharpe ratio means. Gorse (2010) worked on transforming the recurrent reinforcement learning into a stochastic learning process by using the stochastic gradient ascent in the optimization and training technique. This improvement helped the process to be less expensive computationally as there is no need to save the previous signals, but only the

$t-1$ signal. Hens and Wöhrmann (2007) applied recurrent reinforcement learning on a long-term equity and bond portfolio, assuming a rational investor with a constant risk aversion and a power utility function. Bertoluzzo and Corazza (2008) developed an artificial neural network based on reinforcement learning algorithm using the reciprocal of the returns weighted direction symmetry index as the measure of profitability. They proposed a procedure for the management of drawdown like phenomena, and concluded that one can take into account a drawdown like phenomenon in the learning process.

In general, most of the current studies on trading decision systems agree that there are a number of critical components that need to be considered in developing such systems in addition to the core algorithms. If not adequately designed, these factors can significantly compromise the advantages of any advanced machine learning or artificial intelligence based trading decision systems. Cavalcante, Brasileiro, Souza, Nobrega, and Oliveira (2016) surveyed the computational intelligence methods, proposed to solve financial market problems, from 2009 to 2015. They specify a framework to be followed by most computational intelligence approaches, consisting of the following major components: a) data preparation (input variables, output variables, acquisition, prepossessing, normalization); b) algorithm definition (choose model, configure architecture); c) training (define algorithm, adjust parameters, perform training); d) model evaluation (define metrics, evaluate accuracy); e) trading strategies; and f) money evaluation. In this survey, the authors note that Chande (2001) identified three characteristics of successful trading strategies: a) a rule set defining entering and exiting trades, b) a risk control method, and c) money management. Other characteristics include taking into account real world constraints such as backtesting with real transaction costs and slippage. Martinez, da Hora, Palotti, Meira, and Pappa (2009) developed a trading system based on forecasting where they used an artificial neural network (ANN) to forecast the asset price, and then designed a trading system specifying the exit and entry rules based on the forecasts. A stop loss strategy based on placing a threshold on negative returns was also incorporated to accommodate market condition changes. In a paper by Beraldi, Violi, and De Simone (2011), the authors proposed a trading support system to help investors solving the strategic asset allocation problem. They focused on the system and its modules and stages rather than the solution method. The modules in their system included data management, statistical analysis, scenario simulation, a model generator, a solution kernel and a solutions analysis module. Their decision system integrated many solution approaches based on statistical and stochastic optimization with a Monte Carlo simulation of the scenarios. Eilers, Dunis, von Mettenheim, and Breitner (2014) developed an automated trading decision system where they combined an artificial neural network with reinforcement learning (RL) and seasonality. The authors trained the ANN using the value iteration method of the RL while only optimizing the immediate reward. This method simplifies the optimal value function to be only the immediate reward. The three layers include input neurons, hidden neurons, and one output neuron; and the feed-forward ANN is trained by minimizing the mean squared error using the back-propagation algorithm. Feuerriegel and Prendinger (2016) proposed a trading system based on news disclosures where the authors design trading strategies that utilize textual news to obtain profits on the basis of novel information entering the market. They developed a system for automated decision making using supervised and reinforcement learning. The system contained two main components: news sentiment extraction and trading strategy execution. They concluded that a trading system can be improved with additional novel market information.

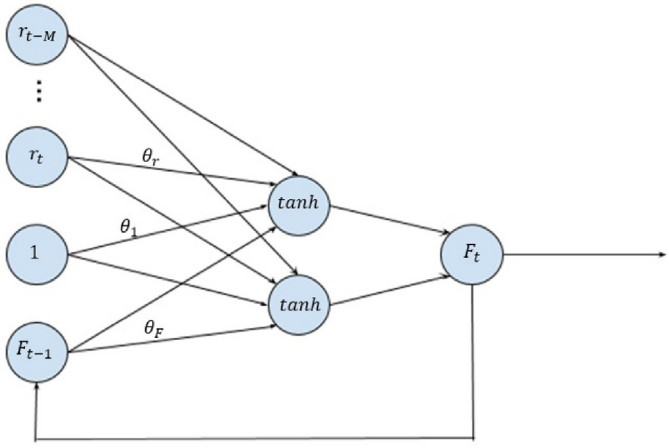

Fig. 1. Recurrent reinforcement learning with portfolio allocation signals.

3. Data and methodology

3.1. Methodology

In this paper, we use the recurrent reinforcement method in portfolio optimization with different risk considerations through two objective functions. Following Moody et al. (1998), we use the differential Sharpe ratio for dynamic optimization of trading system performance. We use performance functions both to increase the convergence of the learning process and to adapt to changing market conditions during live trading (see Fig. 1). During this process, the parameter updates can be done during each forward pass through the training data, and the influence of the performance measure can be computed at any time point. We also assume a small or medium investor who can take fixed or variable sizes of shares of each asset with no price impact in the market but with a fixed transaction cost.

$$F_t = \tanh(\boldsymbol{x}_t'\boldsymbol{\theta}) \tag{1}$$

Before describing the recurrent reinforcement learning approach, we start by comparing the different safety-first risk measures (i.e. the Sharpe ratio, the Sortino ratio, the Calmar ratio, and the Sterling ratio). The Sharpe ratio is widely used and it was developed based on mean-variance optimization; therefore, its risk measure is the standard deviation of returns. The Sortino ratio uses the downside deviation and it includes a target for the investment return. The Sterling ratio is used by Moody and Saffell (2001) where the authors differentiated the Sterling ratio using an exponential moving average. Our definition of the Calmar ratio includes the expected maximum drawdown which is defined by Magdon-Ismail and Atiya (2004), while the basic definition of the Calmar ratio is similar to the Sterling ratio in that both use the maximum drawdown. The issue here is that the Sterling ratio is purely empirical, depending on the dataset it is applied to, and lacking the analytical properties needed in RRL. We choose to use the Calmar ratio defined using the expected maximum drawdown because it is consistent, coherent and differentiable by definition as it can be seen from the definition of the expected maximum drawdown. The relation between the Sharpe and Calmar ratios is shown in Figs. 2 and 3. While Fig. 2 shows the relation between the Sharpe ratio and the expected maximum drawdown per standard deviation unit, Fig. 3 shows the relation between the Calmar ratio with different Sharpe ratio values and a moving time step. From this construction, the Calmar ratio we use here is consistent with Sharpe ratio in a nonlinear way, and it is a statistically coher-

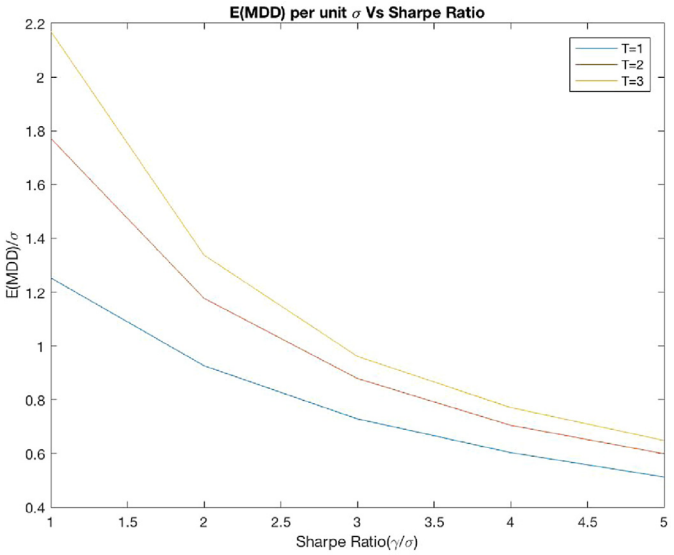

Fig. 2. E(MDD) per unit σ vs. the Sharpe ratio.

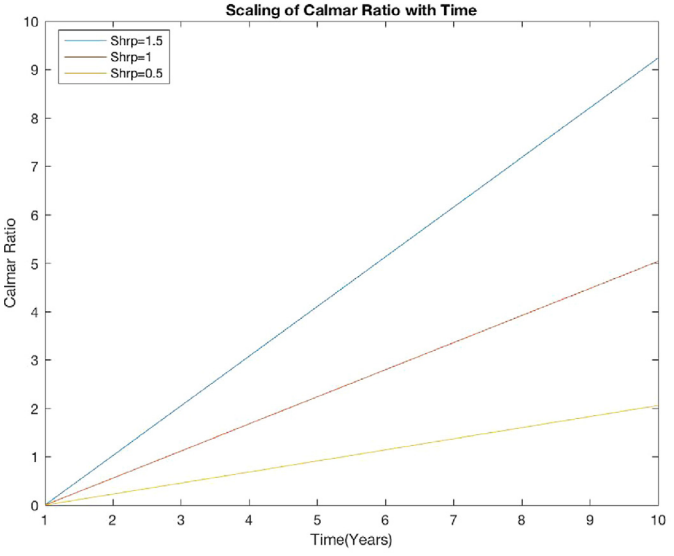

Fig. 3. Scaling the Calmar ratio with time.

ent risk measure as the Sharpe ratio. Its long-term difference from the Sharpe ratio is evident (see Fig. 3).

The Calmar ratio (CR) is similar to the Sharpe ratio (SR) in that it is also a risk adjusted measure of performance. However, it is an MDD risk metric that measures the maximum cumulative loss from a peak to a following bottom. When the downside losses are considered rather than the average deviation from mean return, the trading decisions will certainly be different. Here we derive reward based on this E(MDD) risk based measure, the Calmar ratio. We use the deferential of the Calmar ratio as our objective function:

$$C_T = \frac{\gamma T}{E(MDD)} \tag{2}$$

$$E(MDD) = \begin{cases} \frac{2\sigma^2}{\gamma} Q_p\left(\frac{\gamma^2 T}{2\sigma^2}\right) \overset{T\to\infty}{\to} \frac{\sigma^2}{\gamma}(0.63519 \\ +0.5\log T + \log\frac{\gamma}{\sigma}) & \text{if} \gamma > 0 \\ 1.2533\sigma\sqrt{T} & \text{if} \gamma = 0 \\ \frac{-2\sigma^2}{\gamma} Q_n\left(\frac{\gamma^2 T}{2\sigma^2}\right) \overset{T\to\infty}{\to} -\gamma T - \frac{\sigma^2}{\gamma} & \text{if} \gamma < 0 \end{cases}$$

$$C_T = \frac{\frac{T}{2}Shrp^2}{Q_p(\frac{T}{2}Shrp^2)} \overset{T\to\infty}{\to} \frac{TShrp^2}{0.63519 + 0.5\log T + \log Shrp}$$

where C_T is the Calmar ratio over the time horizon T, $E(MDD)$ is the expected maximum drawdown. The functions $Q_n(x)$ and $Q_p(x)$ are complicated integral expansions that do not have a convenient analytical form and they are independent from γ, σ and T. Their forms can be found in Magdon-Ismail et al. (2003) and in Pratap (2004). γ is the mean of the returns and σ is the standard deviation of returns, $Shrp = \frac{\gamma}{\sigma}$.

Next, we construct a variable weight portfolio with a reinforcement learning signal. Let $F_t \in \{-1, 1\}$ be the trading signal with only two values. For $F_t > 0$, the investor would take a long position, and we set $F_t = 1$. For $F_t < 0$, the investor would then take a short position, and we set $F_t = -1$. θ are the parameters that we want to train $\theta \in \Re^{M+2}$ where M is the time series that we want to trade in. x_t is a vector where $x_t = [1; r_t \dots r_{t-M}; F_{t-1}]$, r_t is the log return $r_t = log(price_t) - log(price_{t-1})$. We calculate the return at time t of our position in Eq. (3):

$$R_t = \mu * [F_{t-1} \cdot r_t - \delta|F_t - F_{t-1}|] \tag{3}$$

μ is the number of shares that is a constant number and it can be the maximum number of shares one can trade. δ is the transaction cost and it is also a constant. Using a risk adjusted measure we will maximize the Sharpe ratio in Eq. (4):

$$S_T = \frac{E[R_t]}{\sigma} \tag{4}$$

where $E[R_t]$ is the mean of the return and σ is the standard deviation of the returns. The objective function is the deferential of the Sharpe ratio and it is calculated by using the average of return and standard deviation of the return. Using the chain rule we get Eq. (5):

$$A = \frac{1}{T}\sum_{t=1}^{T} R_t$$

$$B = \frac{1}{T}\sum_{t=1}^{T} R_t^2$$

$$\frac{dS_T}{d\theta} = \sum_{t=1}^{T}\left\{\frac{dS_T}{dA}\frac{dA}{dR_t} + \frac{dS_T}{dB}\frac{dB}{dR_t}\right\}\cdot\left\{\frac{dR_t}{dF_t}\frac{dF_t}{d\theta} + \frac{dR_t}{dF_{t-1}}\frac{dF_{t-1}}{d\theta}\right\} \tag{5}$$

where we have:

$$\frac{dR_t}{dF_t} = -\mu\delta \cdot sgn(F_t - F_{t-1})$$

$$\frac{dR_t}{dF_{t-1}} = -\mu \cdot r_t + \mu\delta.sgn(F_t - F_{t-1})$$

$$\frac{dF_t}{d\theta} = (1 - tanh(\boldsymbol{x}_t'\boldsymbol{\theta})^2)\cdot\left(\boldsymbol{x}_t + \boldsymbol{\theta}_{M+2}\frac{dF_{t-1}}{d\theta}\right)$$

The above equations conclude that $\frac{dF_t}{d\theta}$ is recurrent, and the weights are updated by the gradient ascent $\theta_{i+1} = \theta_i + \rho.\frac{dS_T}{d\theta}$, and ρ is the learning rate. The recurrent reinforcement learning can be used to optimize a variable weight portfolio. First we change Eq. (1) for each asset to Eq. (6):

$$f_{it} = \text{logsig}(\boldsymbol{x}_{it}'\boldsymbol{\theta}_i) \tag{6}$$

where f_{it} is the action on asset i at time t, and *logsig* is the log-sigmoid transfer function. This then leads to the following equation:

$$\frac{dF_t}{d\theta} = (1 - \text{logsig}(\boldsymbol{x}_t'\boldsymbol{\theta})^2)\cdot\left(\boldsymbol{x}_t + \boldsymbol{\theta}_{M+2}\frac{dF_{t-1}}{d\theta}\right) \tag{7}$$

For a long only portfolio with variable weights, Eq. (8) needs to be applied for each asset:

$$F_{it} = \text{softmax}(f_{it}) \tag{8}$$

where the *softmax* function is applied to the actions on all the assets. It will assign weights to each asset and it already includes the constraint $\sum_i^n F_{it} = 1$, where n is the number of assets in the portfolio.

The decisions from training the parameters θ with a *logsig* activation function will result in the choice of a number in the interval [0, 1] using a $log - sigmoid$ function. The *softmax* function is applied to all the assets decision at time t, and it will distribute asset allocation weights and assign the highest weight to the largest asset decision and redistribute the weights from high to low according to the decision obtained from the *logsig* activation function. In our case, the portfolio will act closely to the equal weight portfolio except if a decision on an asset is close to zero, then the *softmax* will redistribute the weight among the other assets so we are switching the weights between the assets as we move forward with decisions. The *softmax* function is recommended for a portfolio as discussed in Moody et al. (1998). Here we are defining clearly f_{it} and we are using the *softmax* out of the training model. The training and the initial decision are based on the *logsig* function within the model. This model is sensitive to the number of assets as with more assets this model will act as an asset selector by assigning variable weights. Due to the similar statistical properties as the Sharpe ratio, the Calmar ratio based recurrent reinforcement learning as an objective function will be similar but substitute S_T with C_T in Eqs. (4) and (5) accordingly.

3.2. Portfolio constraints

In portfolio optimization, practitioners consider some of the real world constraints in their optimization process such as cardinality constraint, floor and ceiling constraint, round-lot constraint, pre-assignment constraint and class constraint. The cardinality constraint is used to limit the asset selection in the portfolio to a K number of assets. The floor and ceiling constraints limit the weight of each asset allocation to certain boundaries. The pre-assignment constraint allows the investor to pre-select a desired asset in the portfolio. The round-lot constraint restricts the number of any asset to an exact multiple of the normal trading lots. The class constraint limits the proportion invested in assets with common characteristics. In Chang, Meade, Beasley, and Sharaiha (2000), the authors proposed three meta-heuristic algorithms (genetic algorithm, simulated annealing, and particle swarm) to solve the portfolio constraint problems.

Many scholars followed Chang et al. (2000) and developed meta-heuristic methods to solve the constrained portfolio optimization problems. Lwin, Qu, and Kendall (2014) developed a learning guided multi-objective evolutionary algorithm to solve the portfolio optimization problem with the cardinality constraint, floor and ceiling constraint, pre-assignment constraint and round-lot constraint. The authors discussed that these constraints are hard to satisfy at any time as the cardinality constraint by-itself is a mixed quadratic integer NP-hard problem and the portfolio selection with round-lot constraint is an NP-complete problem. The authors compared the performance of their proposed algorithm with four different well-known multi-objective evolutionary algorithms (the Non-dominated Sorting Genetic Algorithm(NSGA-II), the Strength Pareto Evolutionary Algorithm(SPEA-2), Pareto Envelope-based Selection Algorithm(PESA-II), Pareto Archived Evolution Strategy(PAES)). The proposed method is computationally efficient and yields a better result over all the four algorithms. In a paper by Silva, Neves, and Horta (2015), the authors solved the constrained portfolio optimization problem with cardinality constraint, quantity constraint, long only constraint and transaction cost constraint by combining multi-objective evolutionary (MOEA) algorithm with technical indicators, where the indicators are determined by the MOEA algorithm and the selection method is adaptive as the stocks selected are changing with time. The stocks are selected using fundamental indicators, and the trading decisions are based on technical indicators. During the testing phase on the S&P 500 stocks, the authors included a 2% of the stock value as a transaction cost. The proposed method outperforms the index in terms of returns and variance. Liagkouras and Metaxiotis (2016) suggested that due to the intrinsic multi-objective nature of the constrained portfolio optimization problem, the multi-objective evolutionary algorithms proved to be very useful and effective in handling the difficulties imposed by the problem in a reasonable time. Chen, Lin, Zeng, Xu, and Zhang (2017) discussed the exact algorithms for solving the constrained portfolio optimization problem where they mentioned that the disadvantage of the exact algorithm is that it always needs more computational time and can find an optimal solution only in a specified time. The authors then presented a heuristic approach which is an extension to the Non-dominated Sorting and Local Search (NSLS) based multi-objective evolutionary framework and called it (e-NSLS) in order to solve the cardinality constrained portfolio optimization problem. They compared their method with five different algorithms (NSGA-II, SPEA-2, MOEA/D-DE, ABC-FC, GRASP-QUAD) and showed that the proposed method outperforms the other five algorithms in computational results. Moreover, the authors used the Wilcoxon signed ranks test analysis to statistically test the significant performance of e-NSLS with the other algorithms where the results show that the proposed method outperformed the other algorithms.

Although these constrained portfolio optimization problems are complex and hard to solve (Chang et al., 2000; Moral-Escudero, Ruiz-Torrubiano, & Suárez, 2006), there exist a number of heuristic search based approaches in the current literature to help practitioners to address their specific needs. In this paper, we primarily focus on developing an effective RRL trading strategy and a trading system using different objective functions in a dynamic portfolio optimization setting. We will direct our attention in an unconstrained problem setting in the present study, and yet we do not foresee major difficulties to combine our proposed approach with the exiting heuristic portfolio constraint methods to address specific practical requirements. In fact, one could replace the gradient ascent search in the current RRL optimization with an evolutionary algorithm using a desirable objective function (e.g. Sharpe ratio or Calmar ratio) as the fitness function to optimize portfolio weights and constraints simultaneously. For future work, we will combine evolutionary algorithms with our proposed Calmar RRL model to investigate the benefit of introducing various portfolio constraints.

3.3. Data collection

In this study, we construct a five asset portfolio using five of the most commonly traded exchange-traded funds from different asset categories. These assets (identified by their ticker symbols and fund names) are as follows:

- IWD: iShares Russell 1000 Value
- IWC: iShares Micro-Cap
- SPY: SPDR S&P 500 ETF
- DEM: WisdomTree Emerging Markets High Dividend
- CLY: iShares 10+ Year Credit Bond

IWD ETF is an equity fund that holds mid and large-cap US stocks. This ETF tracks the performance of the Russell 1000 value index. IWC ETF is a fund that seeks to correspond to the performance of the Russell micro-cap index. It consists of small cap US based companies. SPY ETF tracks the S&P500 index. It represents

Table 1
Statistical features of the ETFs.

Asset name	Mean of returns	Maximum drawdown	Average volume
IWD	0.0015	0.1996	2,355,230
IWC	0.0014	0.2714	55,670
SPY	0.0018	0.1744	78,605,495
DEM	−0.0024	0.5199	315,405
CLY	0.0002	0.1602	185,953

all 500 stocks in the index and pays dividends on a quarterly basis. DEM ETF is a fund that tracks the price and yield of the WisdomTree emerging markets equity income index, and has an international geographical focus. The CLY ETF is a fixed income asset class that follows the investment results of an index consisting of long-term US corporate bonds and dollar dominated bonds with remaining maturities more than ten years. We extract the weekly closing prices for each of five assets from Yahoo Finance using the `fetch` function in MATLAB. The dates are from January 01, 2011 to December 31, 2015. We use three years of weekly returns for training and two years for testing. Table 1 shows some statistical features of the assets selected over the total time horizon of five years.

4. Trading algorithms comparison

In this section, we first compare the performance of three performance ratios as three different objective functions for the model. This will result in three trading algorithms producing different trading decisions for the same set of assets, and then we can readily assess the merits of each performance ratio in generating trading signals. The resulting portfolio rebalancing methods are: the Sharpe ratio RRL (SR-RRL), the Sterling ratio RRL (TR-RRL), and the Calmar ratio RRL (CR-RRL).

We show the comparison between the portfolios formulated using recurrent reinforcement learning with the Sharpe ratio as the objective function vs. the Calmar ratio as the objective function. In addition, we compare the recurrent reinforcement learning based portfolios with the buy-and-hold strategy as a baseline benchmark. We use three years of weekly closing prices (January 01, 2011 - December 31, 2013) to train our θ for each asset, and two years of weekly closing prices (January 01, 2014 - December 31, 2015) for testing. The value of M is set to 104, the number of weeks in the two years of testing data. In order to generate the signals and the weights from the recurrent reinforcement learning model, we set the number of evaluations for the *tanh* model to a maximum of 10,000 and for the *logsig* model to 500. The *logsig* model at this stage is given a small alteration to the equally weighted portfolio weights due to the number of assets.

4.1. Sharpe ratio recurrent reinforcement learning portfolios

In this model, the objective function that we need to optimize is the Sharpe ratio. The Sharpe ratio is a performance measure that can be maximized by maximizing the mean of the return of the portfolio and minimizing the standard deviation of the return. In the SR-RRL we are using the deferential of the Sharpe ratio that is calculated by averaging the Sharpe ratio. The weights in the model will be updated with respect to the gradient of the Sharpe ratio. The standard deviation is used as a measure of volatility. The more the mean returns of the portfolio vary, the higher the volatility. In other words, the volatility will increase if the mean of the returns is varying in a positive or negative direction.

$$\sigma = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(R_t - \gamma)^2} \tag{9}$$

The standard deviation should not be the only measurement of the risk of a given portfolio. For example, a fund accumulating a return between 4% and 6% on average will have a lower standard deviation than a fund accumulating a return between 4% and 14% on average. In a portfolio with different types of assets of different volatilities, the Sharpe ratio will be a setback to the performance of the portfolio and the decision making process.

We use the recurrent reinforcement learning method with the deferential Sharpe ratio as the objective function to obtain two different portfolios: the Sharpe Ratio Equally Weighted (SR-RRL EW) Long/Short (L/S) Portfolio, and the Sharpe Ratio Variable Weights (SR-RRL VW) Long/Short (L/S) Portfolio.

We use Eq. (1) as the activation function to get the signals of each asset over the training period, and we then use equal weights and apply the signals to the equal weights. Let $n =$ the number of assets, and the weight $w = 1/n$ for each asset. This results in:

$$\sum_{i}^{n} |F_{it} w_{it}| = 1 \tag{10}$$

where i is the number of assets at time t.

In the combination of the above two portfolios, F_t in Eq. (10) is the signal from Eq. (1) and w_t is the weight from Eqs. (6) and (8). We select four portfolios from the Markowitz efficient frontier shown in Fig. 4 and the equally weighted buy & hold portfolio to compare them with the different RRL portfolios. Fig. 5 shows the SR-RRL equally weighted and variable weights portfolio compared in terms of cumulative returns with the four portfolios from the efficient frontier using Markowitz mean-variance optimization namely (minimum variance, maximum return, maximum Sharpe ratio and a Pareto optimal) and the buy & hold portfolio. Both of the SR-RRL portfolios are outperformed by the Pareto optimal portfolio by the end of the investment horizon. In this test we choose $\mu = 100$ and $\delta = 0$ bp which is one basis point per stock traded. When examining the return per asset of the SR-equally weighted long/short portfolio, it shows fluctuation of the asset returns. Since it is an equally weighted portfolio, the portfolio is affected by each asset movement equally. The return per asset conclude that the DEM ETF is drawing down our portfolio as it is the only asset that has strong drawdowns within the portfolio. Fig. 6 shows the asset returns in the buy & hold strategy, indicating that the DEM ETF is causing sharp negative returns. By examining the return of each asset in the SR-variable weights long/short portfolio we conclude that the portfolio acts closely to the equally weighted portfolio. Since it is based on the same signals, the minor changes are causing the portfolio to out-perform due to the higher weight of the SPY ETF until week 40 and IWD ETF in weeks 40–100. It is placing less weight on CLY in weeks 60–80, which minimizes the loss caused by the ETF.

4.2. Sterling ratio recurrent reinforcement learning portfolios

TR-RRL is a model with the objective function as the differential of the Sterling ratio introduced in Moody and Saffell (2001). The output of the model produces trading decisions to maximize the Sterling ration in Eq. (11). The change of the weights in the model will be based on the gradient of the Sterling ratio. Let

$$TR = \frac{\gamma}{MDD} \tag{11}$$

where TR is the Sterling ratio, γ is the mean of the return, and MDD is the maximum drawdown. The deferential of the Sterling ratio in Moody and Saffell (2001) is calculated empirically by using the exponential moving average of the ratio. The maximum drawdown is calculated as follows in Eq. (12):

$$MDD_T = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\min[R_t, 0]^2} \tag{12}$$

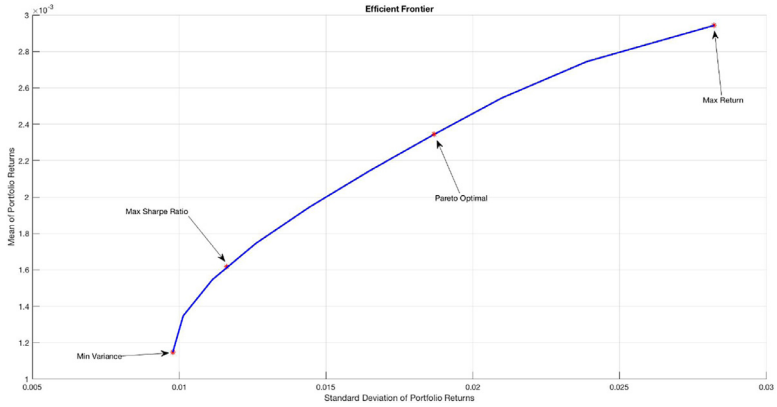

Fig. 4. Efficient frontier portfolios.

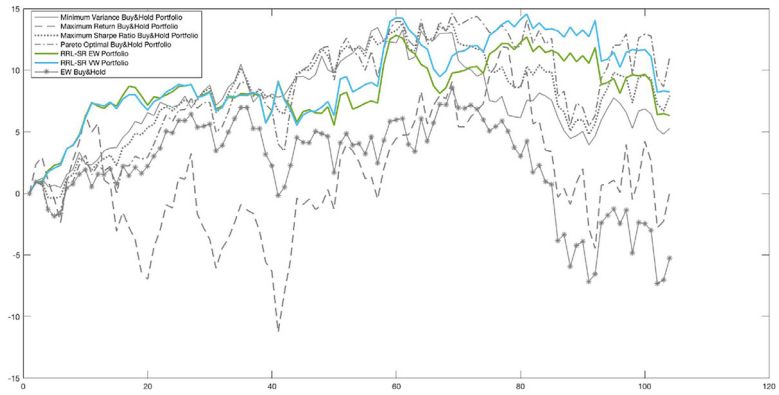

Fig. 5. Portfolio performance RRL with Sharpe ratio (104 weeks).

Fig. 6. Asset returns buy & hold strategy (104 weeks).

In order to train this model to have the same number of cycles as the others, we need to choose a number close to zero (e.g. 0.0001) to avoid division by zero during training when evaluating the minimum of the return (R_t) and zero. We developed two portfolios: the Sterling Ratio Equally Weighted (TR-RRL EW) Long/Short (L/S) Portfolio, and the Sterling Ratio Variable Weight (TR-RRL VW) Long/Short (L/S) Portfolio. In Fig. 7, the TR-RRL equally weighted and variable weights portfolios are compared with the four portfolios from the efficient frontier and the buy & hold portfolio. The TR-RRL portfolios outperform all the portfolios by the end of the investment horizon, where $\mu = 100$ and $\delta = 0bp$.

4.3. Calmar ratio recurrent reinforcement learning portfolios

As in the previous experiment, the training set is three years of weekly closing prices and the testing set is two years of weekly closing prices. We use the Calmar ratio (defined in Eq. (2)) instead of the Sharpe ratio to obtain the signals and weights of our portfolios, and the objective function is the derivative of the Calmar ratio. The difference between the two objective functions is that the Calmar ratio is more sensitive to extreme losses while the Sharpe ratio considers average deviations. Our goal is to identify whether the large losses would make differences in the dynamic optimization process. Using the expected maximum drawdown in the Calmar ratio allows us to increase the number of function evaluations to 10,000 because the expected maximum drawdown is based on the mean and standard deviation of returns where by definition the expected maximum drawdown will not cause a division by zero error. On the other hand, the basic Sterling ratio used by Moody will stop at some point due to a division by zero error. We need to use a number that is close to zero in the maximum drawdown evaluation when computing the minimum of the return (R_t) and zero. In this experiment, we test the following two portfolios: the Calmar Ratio Equally Weighted (CR-RRL EW) Long/Short (L/S) Portfolio, and the Calmar Ratio Variable Weights (CR-RRL VW) Long/Short (L/S) Portfolio.

In Fig. 8, we show the performance of the portfolios developed using the recurrent reinforcement learning and the differential Calmar ratio as the objective function where $\mu = 100$ and $\delta = 0$ bp. Where the CR-RRL portfolio is compared with the four portfolios

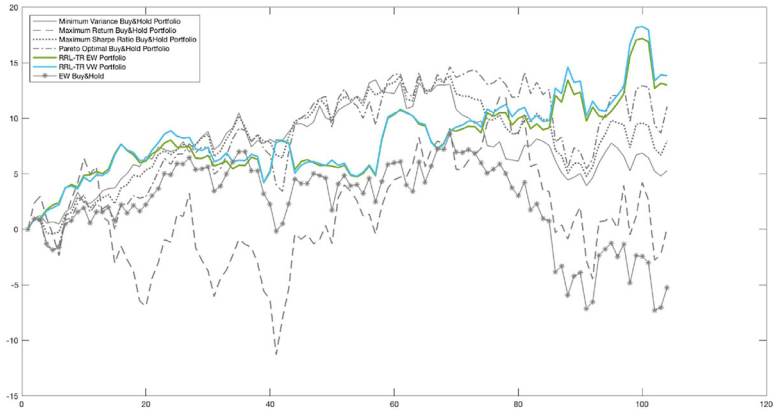

Fig. 7. Portfolio performance RRL with Sterling ratio (104 weeks).

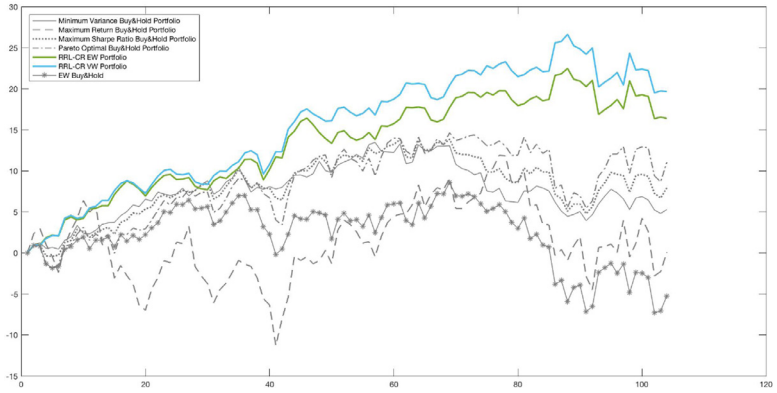

Fig. 8. Portfolio performance Calmar ratio RRL (104 weeks).

Table 2
Model training computation time using eight core processors, 8GB RAM, Windows-64bit, MATLAB.

Trading strategy	Computational time (HH:MM:SS)
SR-RRL	02:39:39
TR-RRL	02:37:50
CR-RRL	02:43:19

from the efficient frontier and the buy & hold portfolio. The CR-RRL portfolios are superior to the other portfolios over most of the investment horizon till the end of the horizon.

In Table 2, we show the training computational time for each model where we used a computer with multiple cores. The computational times are presented in the hour, minute, and second format (HH:MM:SS). We observe that the differences between the methods in terms of computational time is minimal where they differ only in a few minutes. Nevertheless, the TR-RRL method has the least computational training time and the CR-RRL method has the longest training time, and the SR-RRL method sits in between the other two. Overall, the computational efficiency of all the strategies are not too far apart from each other since they are all based on the RRL method with the exception of the objective function. The choosing of the objective function would affect the calculation of the gradient and that would then affect the training of the parameters. In the case that the objective function cannot be increased in the direction of it's gradient, the algorithm may stop at a local maximum with no efficient training of the parameters θ.

4.4. Transaction cost sensitivity analysis

It is well-established that when designing a realistic trading system, one has to account for all transaction costs (Madhavan, 2002; Tetlock, Saar-Tsechansky, & Macskassy, 2008). Although prior studies have conducted trading simulations, many neglect the influence of transaction costs. The primary reason for such omission is due to the difficulty in estimating realistically different types of transaction costs involved, and these costs most likely differ for different asset classes and depend on many other market characteristics. In this section, we examine the impact of trading costs on the profitability of different portfolio strategies. Empirical evidence shows that the average round-trip trading cost of large-cap stocks on NYSE is at least 20 bps (Chan & Lakonishok, 1997; Keim & Madhavan, 1998; Mittermayer, 2004). In the cost sensitivity analysis, we applied one-way trading costs of 10, 15, 20, and 25 bps. Even with realistic transaction costs of 10 bps per round-trip, the portfolio strategies are superior to the hedge fund industry index performance. For transaction costs of 15 bps, the Calmar ratio RRL strategy is on average still profitable, but sustains a substantial loss potential. In general, these strategies cannot compensate transaction costs of more than 20 bps. It requires a system level design to accommodate high transaction costs and further improve portfolio performances, which we will discuss in the next section.

In Fig. 9, we compare the Calmar ratio RRL portfolios performance with the Sharpe ratio RRL portfolios. Due to transaction costs, the Calmar ratio RRL outperforms the Sharpe ratio RRL in terms of cumulative returns. We can conclude from the signals generated that the Calmar ratio RRL portfolios are changing positions in some assets less frequently than the Sharpe ratio RRL portfolios. This is clearer with the CLY ETF where the portfolio suf-

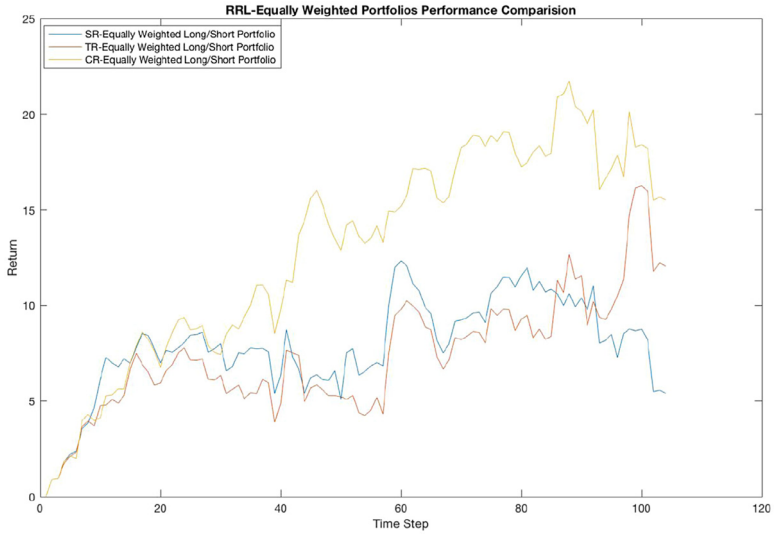

Fig. 9. Comparison in performance $\mu = 100, \delta = 0$ bp (104 weeeks).

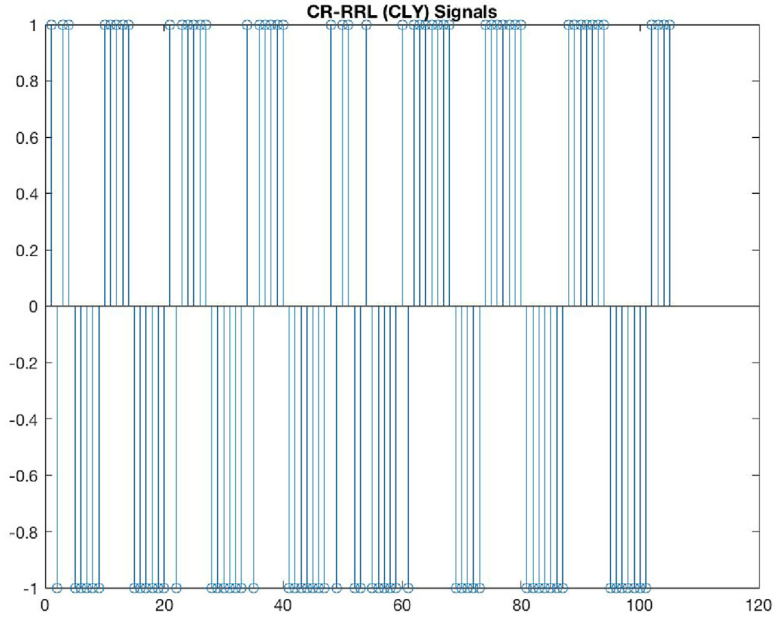

Fig. 10. CLY ETF Signals generated from RRL-Calmar.

fers from losses due to transaction costs. The consistency in signals means that we are holding the asset at the same position for a longer period of time which reduces the transaction cost. Figs. 10 and 11 show clearly the differences of signals generated for the CLY ETF using the Calmar ratio and the Sharpe ratio, respectively. This is reasonable because the Calmar ratio based objective function is sensitive to large losses which occur less frequently than in the Sharpe ratio. As a result, due to the frequent rebalancing signals generated from Sharpe ratio objective function, transaction costs are relatively higher compared with the Calmar ratio portfolios.

Table 3 shows the Sharpe ratio of each portfolio through backtesting with both the Sharpe ratio, Sterling ratio and the Calmar ratio as objective functions. Overall, the Calmar ratio portfolios outperform the Sharpe ratio and Sterling ratio portfolios consistently. When the transaction cost increases, the performance of the Sharpe ratio based portfolios decreases, while the Calmar ratio based portfolios maintain almost the same performance (see Tables 4 and 5). The Calmar ratio portfolios are actually increasing in Sharpe ratio measure due to the fact that the transaction cost is affecting the standard deviation of the returns more than the mean of the returns due to the high returns generated by the portfolio with a low standard deviation. The Sterling ratio based portfolios perform between the Sharpe ratio and the Calmar ratio portfolios both with and without transaction costs. When the transaction cost increases to 15 bps, both the Sharpe ratio and Sterling ratio portfolios start to generate negative annualized returns. Under no transaction cost, the Calmar ratio based portfolio retains high performance. Under a high transaction cost ($\delta = 20$ bps), the Sharpe ratio based portfolio suffers a large loss, while the Calmar ratio based portfolios performance is impacted only by a slight decrease in returns. In Section 5, we propose a stop-loss control to the system to limit the transaction cost effect (see Table 6).

5. Trading system and discussion

We develop an adaptive trading system based on the recurrent reinforcement learning using three different objective functions.

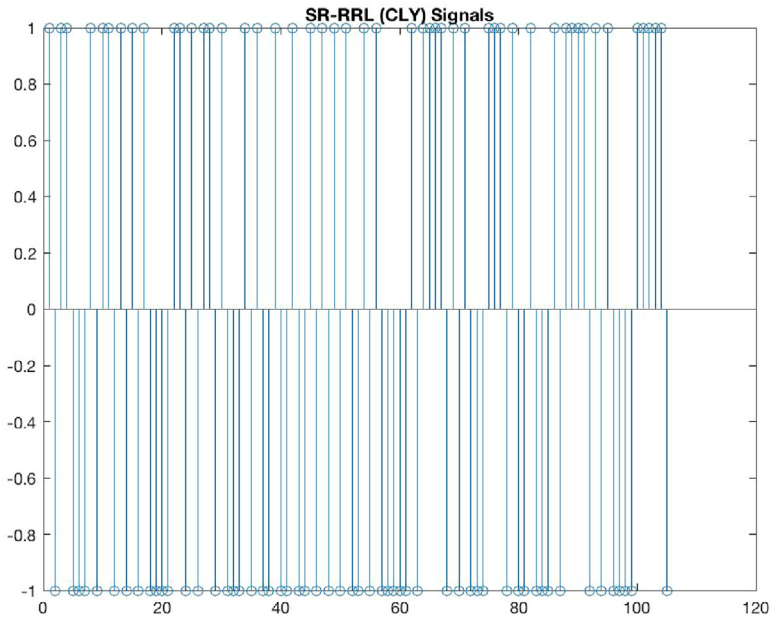

Fig. 11. CLY ETF Signals generated from RRL-Sharpe.

Table 3
Long-short portfolio comparison with different learning objective functions. Note: EW and VW represent Equally Weighted and Variable Weight strategies, respectively.

L/S portfolio ($\delta = 0$ bps)	Sharpe ratio	Return (%) accumulative (annualized)	Maximum drawdown	Num. of trades
Sharpe ratio RRL EW	3.1229	6.31 (3.11)	0.5080	230
Sharpe ratio RRL VW	2.7630	8.24 (4.04)	0.4363	230
Sterling ratio RRL EW	2.3780	12.99 (6.29)	0.4699	235
Sterling ratio RRL VW	2.2816	13.83 (6.69)	0.5262	235
Calmar ratio RRL EW	2.3245	16.39 (7.88)	0.2721	221
Calmar ratio RRL VW	2.1781	19.65 (9.39)	0.2675	221

Table 4
Long-short portfolio comparison with different learning objective functions and transaction costs ($\delta = 10$ bps). Note: EW and VW represent Equally Weighted and Variable Weight strategies, respectively.

L/S portfolio ($\delta = 10$ bps)	Sharpe ratio	Return (%) accumulative (annualized)	Maximum drawdown	Num. of trades
Sharpe ratio RRL EW	1.6926	−2.77 (−1.39)	0.9771	230
Sharpe ratio RRL VW	2.0105	−1.55 (−0.78)	0.8575	230
Sterling ratio RRL EW	1.8071	3.67 (1.82)	0.9968	235
Sterling ratio RRL VW	1.5061	3.46 (1.71)	1.0000	235
Calmar ratio RRL EW	2.5760	7.63 (3.74)	0.4914	221
Calmar ratio RRL VW	2.3367	9.75 (4.76)	0.4692	221

Table 5
Long-short portfolio comparison with different learning objective functions and transaction costs ($\delta = 15$ bps). Note: EW and VW represent Equally Weighted and Variable Weight strategies, respectively.

L/S portfolio ($\delta = 15$ bps)	Sharpe ratio	Return (%) accumulative (annualized)	Maximum drawdown	Num. of trades
Sharpe ratio RRL EW	0.4669	−7.31 (−3.72)	0.9960	230
Sharpe ratio RRL VW	0.6165	−6.45 (−3.28)	0.9878	230
Sterling ratio RRL EW	0.3943	−0.99 (−0.50)	0.9931	235
Sterling ratio RRL VW	0.1044	−1.73 (−0.87)	0.9769	235
Calmar ratio RRL EW	2.6433	3.25 (1.61)	0.7116	221
Calmar ratio RRL VW	2.3925	4.79 (2.37)	0.6619	221

Table 6
Long-short portfolio comparison with different learning objective functions and transaction costs ($\delta = 20$ bps). Note: EW and VW represent Equally Weighted and Variable Weight strategies, respectively.

L/S portfolio ($\delta = 20$ bp)	Sharpe ratio	Return (%) accumulative (annualized)	Maximum drawdown	Num. of trades
Sharpe ratio RRL EW	−0.2281	−11.85 (−6.11)	0.9661	230
Sharpe ratio RRL VW	−0.2551	−11.35 (−5.84)	0.9630	230
Sterling ratio RRL EW	−0.5334	−5.65 (−2.87)	0.9973	235
Sterling ratio RRL VW	−0.7198	−6.91 (−3.52)	0.9869	235
Calmar ratio RRL EW	2.1424	−1.13 (−0.57)	0.8966	221
Calmar ratio RRL VW	2.1320	−0.16 (−0.08)	0.9990	221

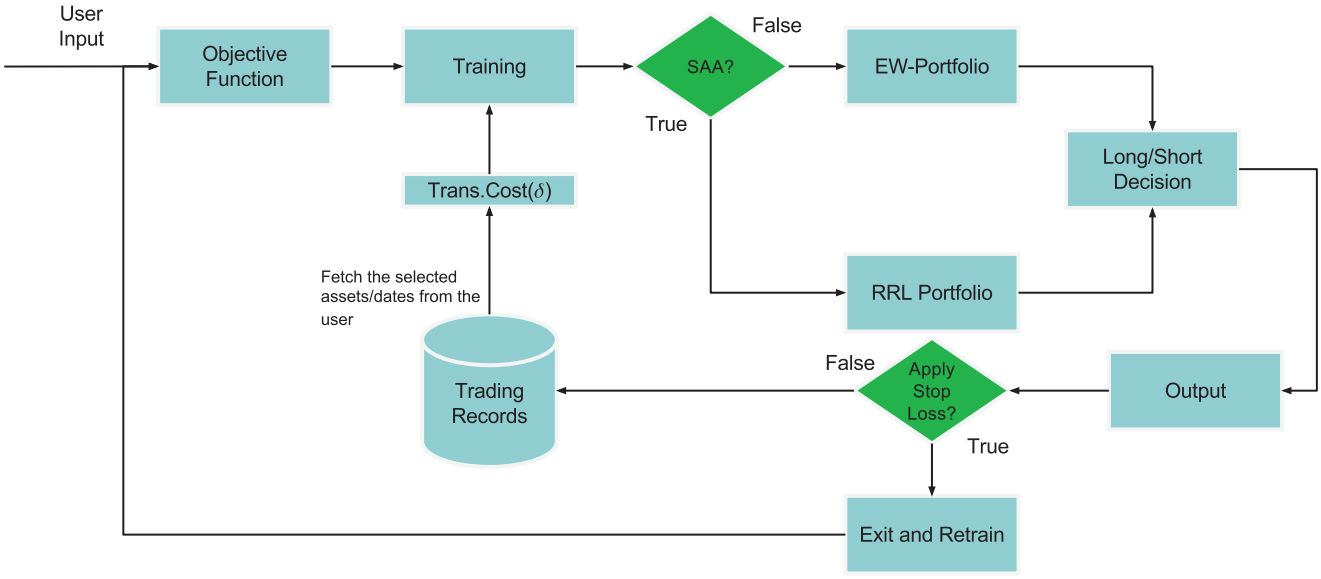

Fig. 12. RRL based trading decision system.

The recurrent reinforcement learning system is a recursive learning system, where the system learns from every output every time step. In this system, the trader can select an objective function that would be the best for the assets of his portfolio. The system parameters are trained based on the objective function desired. We have introduced three objective functions and showed the difference based on a portfolio of five commonly traded ETFs. In a paper by DeMiguel, Garlappi, and Uppal (2009), the authors showed that an equally weighted portfolio can be an efficient portfolio and they compared it to other strategies. In our trading system the default choice of the portfolio weights is the equally weighted portfolio. In Fig. 12, we show the design of the trading system where the user will select the objective function (the Sharpe Ratio, the Calmar Ratio, and the Sterling Ratio) that the RRL system will maximize, and the assets along with the time frame T of the prices. The user will also select the number of decision steps M where $M < T$. The data of the assets will be gathered from Yahoo Finance. The RRL system will learn and train the parameters using the historical returns of time T. After training, the system will allow the user to define the asset allocation from two types of strategic asset allocation (Equally Weighted Portfolio (default), RRL Defined Portfolio). The RRL system will output the long and short decisions of each asset along with the strategic allocation. The system will ask the investor if he would like to use the dynamic stop-loss exit strategy which will stop the trading and go to retraining the system again. If the investor does not want to use the stop-loss then the output will be stored for the next use of the system where it will continue to learn from the given outputs. The system is trained with a predefined transaction cost of $\delta = 10$ bps per share and $\mu = 100$ with no stop-loss during the training phase. In a real trading system, the investor would be able to estimate their transaction costs based on their past trading records, and these costs can change from period to period on the same set of assets. The proposed system will then be able to adapt to these changes through retraining the system with a new cost estimation. The system recommends that the user utilize the Calmar ratio as the objective function when $\delta \geq 15bps$ per share, where using this objective function will help the system endure the transaction cost effect. Also, if the investor is concerned about the drawdown of the portfolio, the Calmar ratio is perfect due to the fact that the system will be trained to minimize the expected maximum drawdown.

Table 7
Portfolio comparison with hedge funds and buy & hold strategy.

Portfolio	Sharpe ratio	Return (%) Accu. (Ann.)	Maximum drawdown
Calmar ratio RRL VW	1.93	28.6 (13.4)	0.1474
SRUSOGP	1.71	9.29 (4.54)	0.7384
HFRIEHI	1.17	0.82 (0.41)	0.8745
Buy & hold	0.62	−5.25 (−2.66)	0.9728

In Fig. 13, we compare the performance of the CR-RRL variable weights (L/S) portfolio with Hedge Fund Research's HFRI Equity Hedge Index (HFRIEHI) and Sunrise's U.S. Equity Optimized Growth Program (SGUSOGP) hedge fund, on a monthly basis over two years (2014–2015) with $\delta = 1$ bp and $\mu = 100$. Table 7 shows the comparison in performance between the CR-RRL variable weights long/short portfolio, the HFRI Equity Hedge Index, and the Sunrise

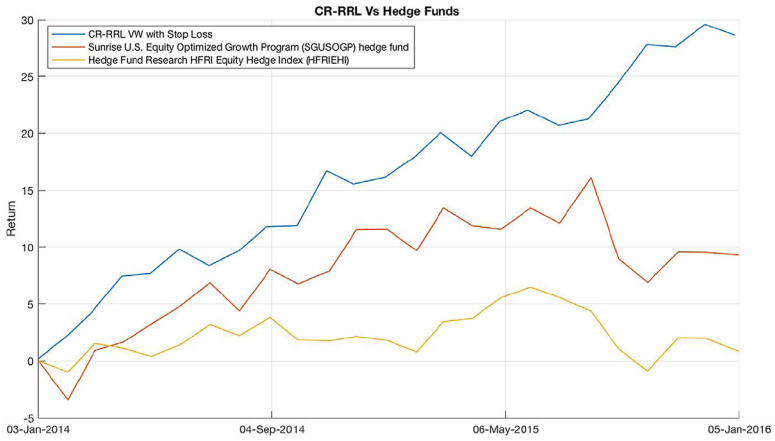

Fig. 13. CR-RRL variable weights portfolio vs. hedge funds (24 months).

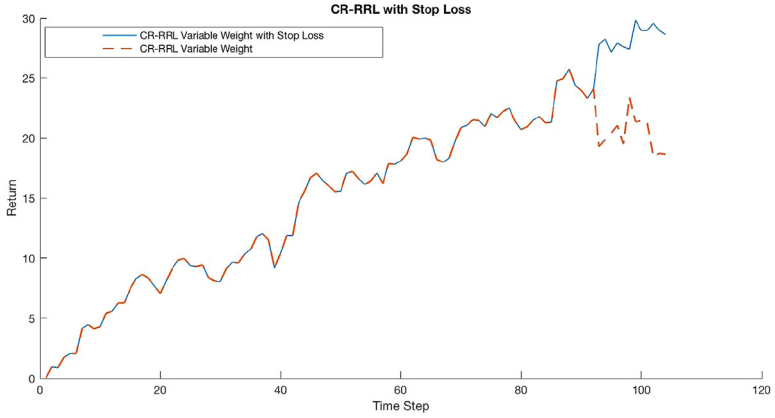

Fig. 14. CR-RRL variable weightes with stop-loss (104 weeks).

U.S. equity hedge fund. This table shows that the CR-RRL portfolio is outperforming in terms of Sharpe ratio, annualized return and maximum drawdown. In this comparison, we highlight the performance of the Calmar ratio RRL system for investors' portfolio designs.

5.1. Dynamic stop−loss strategy

The stop-loss strategy used in the RRL trading decision system is a simple dynamic stop-loss strategy. The notion of the simple dynamic stop-loss is introduced by Chevallier, Ding, and Ielpo (2012) where they applied it to a long-only portfolio. Here we apply the concept in our trading system using the cumulative return in Eq. (13):

$$\frac{r_{t-1}}{\sigma_{t-1}} \leq -n \tag{13}$$

where r_{t-1} is the cumulative return up to time $t-1$, σ_{t-1} is the moving volatility up to time $t-1$, and n is the number of volatility days prompting stop-loss. The stop-loss is applied only during the testing phase at the decision making process; it is not used for training the parameters of the recurrent reinforcement learning model. In Fig. 14, we show the stop-loss strategy effect on returns of the CR-RRL portfolio where the CR-RRL portfolio with stop-loss exits the market in week 91 and stops trading. The system will retrain the parameters based on the latest market movements. In Table 8, we show the Sharpe ratio with a stop-loss for CR-RRL equally weighted portfolio with different transactions costs compared.

Table 8
Calmar ratio recurrent reinforcement learning (CR-RRL) portfolio comparison with/without stop-loss and different transaction costs (δ).

Portfolio ($\delta = 0$ bps)	Sharpe ratio	Return (%) accu. (ann.)	Maximum drawdown	Num. of trades
Stop-Loss	1.9915	29.65 (13.87)	0.2279	230
No Stop-Loss	2.1781	19.65 (9.39)	0.2675	221
Portfolio ($\delta = 10$ bps)	Sharpe ratio	Return (%) accu. (ann.)	Maximum drawdown	Num. of trades
Stop-Loss	2.0852	19.27 (9.21)	0.3628	230
No Stop-Loss	2.3367	9.75 (4.76)	0.4692	221
Portfolio ($\delta = 15$ bps)	Sharpe ratio	Return (%) accu. (ann.)	Maximum drawdown	Num. of trades
Stop-Loss	2.1752	14.08 (6.81)	0.5083	230
No Stop-Loss	2.3925	4.79 (2.37)	0.6619	221
Portfolio ($\delta = 20$ bps)	Sharpe ratio	Return (%) accu. (ann.)	Maximum drawdown	Num. of trades
Stop-Loss	2.2958	8.90 (4.35)	0.7844	230
No Stop-Loss	2.1320	-0.16 (-0.08)	0.9990	221
Portfolio ($\delta = 25$ bps)	Sharpe ratio	Return (%) accu. (ann.)	Maximum drawdown	Num. of trades
Stop-Loss	2.1382	3.71 (1.84)	0.9550	230
No Stop-Loss	1.0273	−5.11 (−2.59)	0.9550	221

From Table 8, we see that the stop-loss will be able to make the portfolio endure higher transaction costs in the case that $\delta \geq 25$ bps as the stop-loss strategy will exit the market when the volatility is high, and retrain the parameters of the model, and then generate new signals to reenter the market. The training of

the new parameters will be done using the latest returns until the exit point to handle changing market conditions.

6. Conclusion

In this paper, we use the recurrent reinforcement learning method to solve a dynamic portfolio optimization problem where we develop four portfolios using the RRL and compare them with each other and the buy & hold portfolio. We use RRL methods to optimize the portfolio weights and rebalance the portfolio over a predefined time horizon. We compare the deferential of the Sharpe ratio and the Calmar ratio as the objective functions in the recurrent reinforcement learning process and examine the performance effect by the transaction costs. We compare the performance differences between the Sterling ratio proposed by Moody and Saffell (2001) where they defined the downside risk as an exponential moving average of drawdown. Due to its lack of necessary statistical properties, the Sterling ratio based RRL suffers computational breakdowns during the optimization process. More importantly, it neutralizes the downside risks and therefore it is limited in reaching an optimal trading strategy. Through backtesting of the constructed portfolio using ETFs, we conclude that: a) variable weight long/short portfolios outperform the equally weighted long/short portfolios; b) the RRL Calmar ratio based portfolios outperform the RRL Sharpe ratio based portfolios consistently; c) the E(MDD) RRL based trading system with market condition stop-loss retraining responds to transaction cost effects better and outperforms hedge fund benchmarks consistently. Overall, we show that the portfolios constructed using RRL with the expected maximum drawdown based Calmar ratio result in a significantly superior performance and are more transaction cost resilient than the portfolios constructed with the Sharpe ratio.

In addition, we propose an adaptive trading decision system based on the proposed RRL portfolio rebalance strategies with both transaction costs and market condition changes, and we show that the system consistently outperforms the benchmark and hedge fund industry average index. We specifically demonstrate how this expected maximum drawdown based reinforcement learning approach can filter market noise and identify the significant trading signals, and how the trading decision system with transaction cost and stop-loss retraining can adapt to different market conditions.

For future studies, we plan to define a relative strength performance measure using the expected maximum drawdown to minimize tracking errors with respect to certain benchmarks. We also believe that *logsig, softmax* RRL model can be extended by adding layers to the model in order to help make a good variable weight decision. The Calmar ratio using the expected maximum drawdown can be applied in other reinforcement learning models and on a large set of asset classes.

References

Beraldi, P., Violi, A., & De Simone, F. (2011). A decision support system for strategic asset allocation. *Decision Support Systems, 51*(3), 549–561.

Bertoluzzo, F., & Corazza, M. (2008). Financial trading systems: Is recurrent reinforcement learning the way? In *Reflexing interfaces: The complex coevolution of information technology ecosystems* (pp. 246–256). IGI Global.

Bertsekas, D. P. (1995). *Dynamic programming and optimal control*: 1. Athena Scientific Belmont, MA.

Bhansali, V. (2007). Putting economics (back) into quantitative models. *The Journal of Portfolio Management, 33*(3), 63–76.

Cavalcante, R. C., Brasileiro, R. C., Souza, V. L., Nobrega, J. P., & Oliveira, A. L. (2016). Computational intelligence and financial markets: A survey and future directions. *Expert Systems with Applications, 55*, 194–211.

Chan, L. K. C., & Lakonishok, J. (1997). Institutional equity trading costs: NYSE versus nasdaq. *The Journal of Finance, 52*(2), 713–735.

Chande, T. S. (2001). *Beyond technical analysis: How to develop and implement a winning trading system*: 101. John Wiley & Sons.

Chang, T.-J., Meade, N., Beasley, J. E., & Sharaiha, Y. M. (2000). Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research, 27*(13), 1271–1302.

Chen, B., Lin, Y., Zeng, W., Xu, H., & Zhang, D. (2017). The mean-variance cardinality constrained portfolio optimization problem using a local search-based multi-objective evolutionary algorithm. *Applied Intelligence*, 1–21.

Chevallier, J., Ding, W., & Ielpo, F. (2012). Implementing a simple rule for dynamic stop-loss strategies. *The Journal of Investing, 21*(4), 111–114.

DeMiguel, V., Garlappi, L., & Uppal, R. (2009). Optimal versus naive diversification: How inefficient is the 1/n portfolio strategy? *Review of Financial Studies, 22*(5), 1915–1953.

Dempster, M. A., & Leemans, V. (2006). An automated fx trading system using adaptive reinforcement learning. *Expert Systems with Applications, 30*(3), 543–552.

Eilers, D., Dunis, C. L., von Mettenheim, H.-J., & Breitner, M. H. (2014). Intelligent trading of seasonal effects: A decision support algorithm based on reinforcement learning. *Decision Support Systems, 64*, 100–108.

Feuerriegel, S., & Prendinger, H. (2016). News-based trading strategies. *Decision Support Systems, 90*, 65–74.

Gold, C. (2003a). Fx trading via recurrent reinforcement learning. In *Computational intelligence for financial engineering, 2003. proceedings. 2003 IEEE international conference on* (pp. 363–370). IEEE.

Gold, C. (2003b). FX trading via recurrent reinforcement learning. In *IEEE/IAFE conference on computational intelligence for financial engineering, proceedings (cifer): 2003* (pp. 363–370).

Gorse, D. (2010). Application of stochastic recurrent reinforcement learning to index trading. In *ESANN 2011 proceedings, 19th European symposium on artificial neural networks, computational intelligence and machine learning* (pp. 123–128).

Hens, T., & Wöhrmann, P. (2007). Strategic asset allocation and market timing: A reinforcement learning approach. *Computational Economics, 29*(3-4), 369–381.

Keim, D. B., & Madhavan, A. (1998). The cost of institutional equity trades. *Financial Analysts Journal, 54*(4), 50–69.

Liagkouras, K., & Metaxiotis, K. (2016). A new efficiently encoded multiobjective algorithm for the solution of the cardinality constrained portfolio optimization problem. *Annals of Operations Research*, 1–39.

Lwin, K., Qu, R., & Kendall, G. (2014). A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization. *Applied Soft Computing, 24*, 757–772.

Madhavan, A. N. (2002). Implementation of hedge fund strategies. *Special Issues (Hedge Fund Strategies), 2002*(1), 74–80.

Magdon-Ismail, M., Atiya, A., Pratap, A., & Abu-Mostafa, Y. (2003). The maximum drawdown of the brownian motion. In *Computational intelligence for financial engineering, 2003. proceedings. 2003 IEEE international conference on* (pp. 243–247). IEEE.

Magdon-Ismail, M., & Atiya, A. F. (2004). Maximum drawdown. *Risk Magazine, 17*(10), 99–102.

Maringer, D., & Ramtohul, T. (2010). Threshold recurrent reinforcement learning model for automated trading. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6025 LNCS*(PART 2), 212–221.

Maringer, D., & Ramtohul, T. (2012). Regime-switching recurrent reinforcement learning for investment decision making. *Computational Management Science, 9*(1), 89–107.

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance, 7*(1), 77–91.

Martinez, L. C., da Hora, D. N., Palotti, J. R. d. M., Meira, W., & Pappa, G. L. (2009). From an artificial neural network to a stock market day-trading system: A case study on the bm&f bovespa. In *Neural networks, 2009. IJCNN 2009. international joint conference on* (pp. 2006–2013). IEEE.

Mittermayer, M.-A. (2004). Forecasting intraday stock price trends with text mining techniques. In *system sciences, 2004. proceedings of the 37th annual hawaii international conference on* (pp. 10–pp). IEEE.

Moody, J., & Saffell, M. (2001). Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks, 12*(4), 875–889.

Moody, J., Wu, L., Liao, Y., & Saffell, M. (1998). Performance functions and reinforcement learning for trading systems and portfolios. *Science, 17*(February 1997), 441–470.

Moral-Escudero, R., Ruiz-Torrubiano, R., & Suárez, A. (2006). Selection of optimal investment portfolios with cardinality constraints. In *Evolutionary computation, 2006. CEC 2006. IEEE congress on* (pp. 2382–2388). IEEE.

Pratap, A. (2004). *Maximum drawdown of a Brownian motion and AlphaBoost: a boosting algorithm*. California Institute of Technology Ph.D. thesis..

Pu, Q., Ananthanarayanan, G., Bodik, P., Kandula, S., Akella, A., Bahl, P., … Schmidhuber, J. (2016). Using a genetic algorithm to improve recurrent reinforcement learning for equity trading. *Computational Economics, 47*(4), 421–434.

Silva, A., Neves, R., & Horta, N. (2015). A hybrid approach to portfolio composition based on fundamental and technical indicators. *Expert Systems with Applications, 42*(4), 2036–2048.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*: 1. MIT press Cambridge.

Sutton, R. S., Barto, A. G., & Williams, R. J. (1992). Reinforcement learning is direct adaptive optimal control. *IEEE Control Systems, 12*(2), 19–22.

Tan, Z., Quek, C., & Cheng, P. Y. (2011). Stock trading with cycles: A financial application of anfis and reinforcement learning. *Expert Systems with Applications, 38*(5), 4741–4755.

Tetlock, P. C., Saar-Tsechansky, M., & Macskassy, S. (2008). More than words: Quantifying language to measure firms' fundamentals. *The Journal of Finance, 63*(3), 1437–1467.

Zimmermann, H., Drobetz, W., & Oertmann, P. (2003). *Global asset allocation: New methods and applications*: 197. John Wiley & Sons.