Learning of the No-Trade Zone

Yifan Sun [1] **Xiaofei Shi** [1] **Yufeng Shen** [2] **Zhipeng Yang** [3]

Abstract

We study the finite-horizon dynamic portfolio management problem with risky assets with mean-reverting (Ornstein- Uhlenbeck) dynamic in the presence of transaction costs. The goal is to maximize the expected constant relative risk aversion (CRRA) utility of terminal wealth. First, we analytically solve the problem without transaction costs for a single risky asset and a risk free asset, and then derive the solution to a portfolio of Ornstein-Uhlenbeck assets via a system of ordinary differential equations (ODEs). The portfolio optimization problem becomes intractable both analytically and computationally when there exist transaction costs and multiple assets. To address transaction costs, we propose a novel numerical approach employing deep neural networks, building on the previous ODE solutions and given the standard No-Trade region policy rules. We establish a model-based reinforcement learning frame work on the dynamic portfolio task, and our method readily extends to high-dimensional portfolio problems wherein traditional methods fail. Experiments with synthetic and market data show the numerical benefits of the developed algorithms.

1. Introduction

Since the 1970s, the dynamic portfolio optimization problem has long been an essential topic in the field of mathematical finance that draws the attention of scholars and partioners alike. Merton (Merton, 1969; 1975) established the framework for dynamic portfolio choice with stochastic variation in investment opportunities, and with the absence of transaction cost, one could explicitly solve the continuous-time portfolio problem where the investor can invest between stocks modeled as geometric Brownian motions and a money market account with a fixed risk-free rate to maximize the expected utility of consumption and terminal wealth.

There has been a vast amount of literature extending Merton's problem from different perspectives. One main stream is to incorporate the stock return predictability into portfolio optimization, for example, modeling the stock returns as Ornstein-Uhlenbeck processes [1], see (Campbell & Viceira, 1999). Another interesting aspect is to take the inflation and risk free rate into account, see (Munk et al., 2004). Moreover, researchers also consider other types of utility function, such as constant relative risk aversion utility (CRRA) (Liu, 2006) or mean-variance utility (Johnstone et al., 2013).

In the studies mentioned above, the optimal trading strategy is solved either analytically or approximately by assuming no transaction cost. However, dynamic portfolio optimization often requires frequent rebalancing, hence transaction costs are usually not negligible. Such trading costs are caused by several factors such as the bid-ask spread, execution commissions, market depth, price impact or tax, etc. Intuitively, with the presence of transaction costs, the optimal trading strategy in the frictionless market needs to be modified since it might not be optimal to adjust the portfolio if the change of the stock is small, which means there might exist the so-called no-trade zone.

As early as in 1976, Magill and Constantinides initiated the research in transaction costs by proposing that the investor only trades in securities when the variation in the underlying security prices forces his portfolio proportions outside this no-trade zone (Magill & Constantinides, 1976). Davis and Norman were the first to provide a detailed formulation and analysis, along with an algorithm and numerical computation of the optimal policy for an infinite-horizon investment and consumption decision problem (Davis & Norman, 1990). Shreve and Soner relaxed several assumptions in Davis and Norman's work and conducted an analysis of the optimal trading and strategies in an infinite horizon. They proved existence, uniqueness and regularity of the value function with respect to the utility (Shreve et al., 1994). Liu and Loewenstein focused on the infinite-horizon optimal trading problem with a single risky asset. The multi-asset

[1]Department of Mathematical Sciences, Carnegie Mellon University [2]Physics Department, Carnegie Mellon University [3]Department of Biological Sciences, Carnegie Mellon University. Correspondence to: Xiaofei Shi <xiaofeis@andrew.cmu.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

[1]https://en.wikipedia.org/wiki/OrnsteinUhlenbeck_process

portfolio optimization problem is more difficult to solve (Liu & Loewenstein, 2002). Liu obtained an almost closed form solution for proportional costs in continuous time for infinite lived constant absolute risk aversion (CARA) investors when asset returns are uncorrelated (Liu, 2004).

On the other hand, during the recent years, deep learning as well as deep reinforcement learning have made breakthroughs in many areas such as image recognition, game playing as well as in the finance industry. Culkin and Das surveyed how and why deep learning can influence the field of finance in a very general way with a specific application to reproducing the Black and Scholes option pricing formula [2] to a high degree of accuracy by training a fully-connected feed-forward deep learning neural network (Culkin & Das, 2017). E et al. proposed a new method for solving high-dimensional fully nonlinear second-order partial differential equations (PDEs) herein (E et al., 2017). The PDEs are reformulated as a control theory problem with the gradient of the unknown solution approximated by neural networks, like deep reinforcement learning with the gradient acting as the policy function. Their technique has inspired us to use deep neural network to solve the free boundary HJB equation.

In this work, we consider a relatively practical setting.

First, for the comparison baseline, we analytically solve the portfolio problem without transaction costs for a risk free asset and a single risky asset. We model the asset prices, rather than asset returns or risk premiums as in many other works in the literature, as Ornstein-Uhlenbeck process, then derive the solution by solving a system of ordinary differential equations (ODEs). Therefore, we would get a comparison baseline as well as a suggestion to propose the hierarchical architecture.

Next, we extend the model to correlated multi-assets portfolio with the presence of transaction cost. Most importantly, when there are multiple assets in the frinctional market, the portfolio optimization problem becomes intractable. Instead, we build a hierarchical architecture and develop a novel numerical method using deep reinforcement learning to parametrize the trading boundaries of the no-trade zone in a dynamic fashion. A very interesting observation, as is pointed out by Matt Emschwiller, Benjamin Petit and Jean-Philippe Bouchaud in their working paper, is that when the number of assets is significantly large, we can use the mean-field approximation to this problem, which we could use as another comparison baseline in our calibration.

By the design of the algorithm, our method is scalable to the high dimensional case. To our best of knowledge, previous work has not been done to study the transaction boundary in the high dimensional case with asset prices modeled as correlated Brownian motions or mean-reverting processes.

[2]https://en.wikipedia.org/wiki/BlackScholes_model

The organization of this work is as follows: in Section 2, we derive the theoretical results of stationary limits of optimal allocation problem under both the frictionless and the frictional assumption. We also show our parametrization and discretization scheme in this section.

Section 3 describes our hierarchical architecture and the associated deep neural network (DNN) approach to parametrize the No-Trade zone and optimal trading strategy based on no-trade zone. It also provides training procedure of the DNN and the simulation results to illustrate how transaction cost affect the No-Trade zone from calibrating the market data, which is the 2013 VIX front month future data, see D.

Section 4 We discuss our results as well as point out possible future directions.

2. Theoretical result

We first derive an explicit solution for the optimal trading strategy when there is no transaction cost. We then introduce proportional transaction cost, and use previously derived strategy as as a warm start for our policy search transaction cost is introduced. When there is transaction cost, the trading strategy depends on a region called no trade zone. We use Deep Neural Network to parametrize this no trade zone hence the optimal trading policy.

2.1. Optimal Strategy With Zero Transaction Cost

Suppose there is one risky asset whose price X_t follows an Ornstein-Uhlenbeck (OU) process and a money market account value by Y_t:

$$dX_t = \alpha(\mu - X_t)dt + \sigma dZ_t \tag{1}$$

$$dY_t = rY_t dt \tag{2}$$

where $\alpha \in \mathbb{R}^+$, $\sigma \in \mathbb{R}^+$ and $\mu \in \mathbb{R}$ are parameters of the OU process and Z_t denotes the standard Brownian motion, $r \in \mathbb{R}$ is the constant risk free rate. Let $\pi(t) \in \mathbb{R}$ be the current proportion of wealth invested in the risky asset at time t. The total wealth W_t follows:

$$\frac{dW_t}{W_t} = \pi_t \frac{dX_t}{X_t} + (1 - \pi_t)\frac{dY_t}{Y_t} \tag{3}$$

Our goal is to maximize the expected utility of the terminal wealth W_T at a given finite horizon T:

$$\max_{\pi} \mathbb{E}[U_\gamma(W_T)] \tag{4}$$

where we choose $U_\gamma(W)$ to be the CRRA utility: $U(W) = \frac{W^{1-\gamma}-1}{1-\gamma}$ for $\gamma > 1$.

The above problem could be solved using Dynamic Programming Principle, in particular we can derive the continuous time version of the Bellman equation called Hamilton-Jacobi-Bellman (HJB) equation. Let $\tau = T - t$ be the horizon of the investment period. Assume $V(W, X, \tau)$ to be the value function which satisfies the boundary condition $V(W, X, 0) = U(W)$. We derive the Hamilton-Jacobi-Bellman equation:

$$\max_{\pi} \Big\{ -V_\tau + \Big[\frac{\pi\alpha}{X}(\mu - X) + (1-\pi)r\Big] W V_W + \frac{1}{2}(\frac{\pi\sigma}{X})^2 W^2 V_{WW} + \alpha(\mu - X)V_X + \frac{1}{2}\sigma^2 V_{XX} + \sigma\frac{\pi\sigma}{X} W V_{WX} \Big\} = 0 \tag{5}$$

where V_τ,V_W, and V_X denote the derivatives of V with respect to t, W, and X respectively. Similarly, V_{WW}, V_{XX} and V_{WX} denote the higher derivatives. We can solve the explicit solution of optimal asset allocation:

$$\pi^*(\tau, X) = \frac{1}{\gamma}\Big[\frac{\alpha(\mu - X)/X - r}{(\sigma/X)^2}\Big] + \frac{1-\gamma}{\gamma}\big[C(\tau)X + B(\tau)\big]X. \tag{6}$$

where formula of $A(\tau)$ and $B(\tau)$ as well as detailed derivation of the solution can be found in Appendix. This $\pi * (\tau, X)$ will serve as our warm start to tackle optimal trading problem with proportional transaction cost.

2.2. Calibration of Parameters

To calibrate the real-world data, we are essentially dealing with the modeling and forecasting of time series, which has fundamental importance to various fields. Therefore, a lot of active research works are going on in this subject during recent years. However, there is a challenge: since for time series, typically only one path, or only one realization to be more precisely, is available, all of the conclusions about the time series must be drawn based on the information extracted from this single path.

In this work, our assumption on the price of the risky assets is given by (1). By fitting the Orstein-Ulenceck process using the traditional MSE loss, we are able to estimate the mean level μ and the return rate α, and we plot the 1-yr stock price against our estimation in order to illustrate the mean-reverting behavior of the stock price.

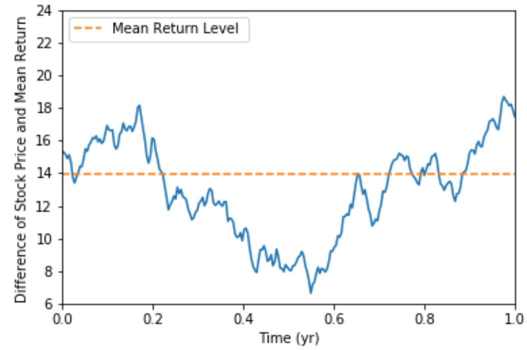


Figure 1. Calibration of the mean level μ and mean-return rate α.

2.3. Trading Strategy With Proportional Cost

2.3.1. Concept of No-Trade Region

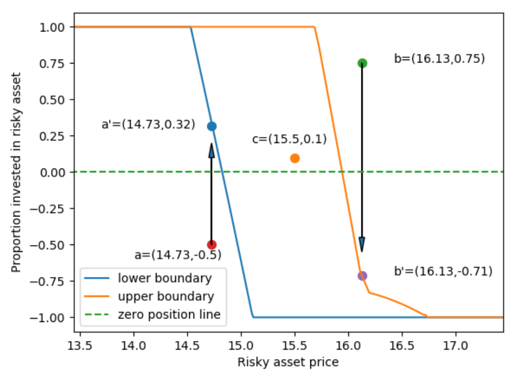


Figure 2. Example of a No-Trade zone and corresponding rebalancing rule.

In practice it is common to to pay proportional transaction cost with a rate λ, meaning that every trade with notional (dollar amount) Z pays λZ transaction fee. It is not optimal to continuously rebalance the portfolio with transaction cost as the profit from rebalancing may not cover the transaction fee. Based on (Shreve et al., 1994), the optimal trading strategy is given by a No-Trade zone. Within the No-Trade zone, no rebalance is needed to avoid the transaction cost. Outside the No-Trade zone, one should rebalance the portfolio to the boundary of the No-Trade zone. However, the boundaries of No-Trade zone is unknown. Traditionally the boundaries are identified by solving Partial Differential Equations (PDE) with free boundary conditions. However this approach is computationally intractable when dimension of asset is greater than 2. In this section we tried to use Deep Neural Network to parametrize the boundary of the No-Trade zone, this method is applicable to high dimensional case when number of assets is greater 2.

We use a graph to illustrate the concept of No-Trade zone. Recall that the asset price is mean reverting, hence the optimal strategy is to buy the asset when its price is below the mean level and sell it when its price is above the mean level. With transaction cost, we only buy the asset when its price is below the mean level to some certain extent and sell it conversely. The No-Trade zone in Figure. 2 is simulated with parameters estimated from VIX future in year 2013-2014 and the assumption that there is a 2% transaction cost. The No-Trade zone is for $t_k = 0.4$ when investment horizon is $[0, T] = [0, 1]$. The blue curve denotes the lower boundary of No-Trade zone and orange curve is the upper boundary of the No-Trade zone. The region between two curves is the No-Trade zone. The mean reverting level is 15.4463. X-axis denotes the observed risky asset price, Y-axis denotes the proportion of total wealth invested in risky asset.

- Red point $a = (14.73, -0.5)$. This point means observed asset price is $X_{t_k} = 14.73$, before rebalancing the proportion π_{t_k-} invested in risky asset is -0.5, i.e. we are using 50% of our wealth to short the asset. However, 14.73 is below the mean reverting level 15.4463 hence we should switch our position to long the asset, we rebalance our position to blue point $a' = (14.73, 0.32)$ on the lower boundary of the No-Trade zone which means we put $\pi_{t_k+} = 32\%$ of our wealth to long the asset.

- Orange point $c = (15.5, 0.1)$. This point means $X_{t_k} = 15.5$ and $\pi_{t_k-} = 0.1$. This point lies in the No-Trade zone hence no action is needed. Because 15.5 is close to mean reverting level, the price change in near future will be mainly driven by noise, there is no deterministic trend hence we should not rebalance to avoid transaction cost.

- Green point $b = (16.13, 0.75)$. This point means $X_{t_k} = 16.73$ and before rebalancing we are using using 75% of our wealth to long the asset. Because 16.73 is well above the mean reverting level, we should switch our long position to short position $\pi_{t_k+} = -0.71$ i.e. putting 71% of our wealth to short the asset. We move from the green point b to the purple point b' on the upper boundary of the No-Trade zone.

2.3.2. Theoretical Characterization of No-Trade Region

In this section we use dynamic programming principle to derive the Hamilton-Jacobi-Bellman equation for the value function when proportional transaction cost is present. We give an economic interpretation of the No-Trade region. We will see that the value function is characterized by a variational inequality, whose explicit solution is not available and finding its numerical solution suffers curse of dimensionality, therefore using Neural Network to approximate the No-Trade region becomes desirable.

We keep assumption as we did for frictionless case, i.e. trading a risky asset with price X_t and a risk free asset with price Y_t. We introduce transaction cost λ and two non-decreasing adapted process L_t and M_t that describe the cumulative transfers from the safe asset to the risky asset and vice versa, we always assume the transaction cost is deducted from the safe asset account, then let A_t denote the value of safety asset account, let B_t denote the value of risky asset account, we have:

$$dA_t = rA_t dt - (1+\lambda)dL_t + (1-\lambda)dM_t \qquad (7)$$

$$dB_t = \frac{B_t}{X_t}dX_t + dL_t - dM_t \qquad (8)$$

The value function will take 4 arguments (t, A_t, B_t, X_t) as argument. The agent needs to make decision based on time to maturity $T-t$, its risky / safety asset account value and the current asset price. We can use dynamic programming principle to derive the Hamilton-Jacobi-Bellman equation as we did for frictionless case, we use v_t to denote $\frac{\partial}{\partial t}v(t, a, b, s)$, and similar convention applies to v_a, v_b and v_x:

$$\sup_{dL_t, dM_t}\left[\left[v_a - (1+\lambda)v_b\right]\frac{dL_t}{dt} + \left[(1-\lambda)v_b - v_a\right]\frac{dM_t}{dt}\right] + v_t + rbv_b + \frac{a}{x}\alpha(\mu - x)v_a + \alpha(\mu - x)v_x + \frac{a^2}{x^2}\sigma^2 v_{aa} + \sigma^2\frac{a}{x}v_{ax} = 0$$

We can see that the No-Trade region is characterized by the value function:

$$NT = \{(t, a, b, x) | (1+\lambda)v_b > v_a > (1-\lambda)v_b\} \qquad (9)$$

We consider three different scenarios and explain why NT region is characterized by above formulation, our discussion gives theoretical justification of three bullet points we discussed at the end of last subsection:

- $(1+\lambda)v_b > v_a > (1-\lambda)v_b$: taking sup over dM_t and dL_t yields $dM_t = dL_t = 0$, this means there should be no trade at all, hence the name No-Trade region. When this is the case, the marginal utility gained from increasing one account cannot offset the marginal utility loss of drawing another account upon taking into effect of transaction cost, hence there should be no rebalancing.

- $v_a > (1+\lambda)v_b$: the marginal utility of increasing the risky account is high enough, the optimal strategy is to buy risky asset at infinite rate $dL_t/dt = \infty$ until $v_a = (1+\lambda)v_b$, i.e. we ajust out position back to No-Trade region.

- $v_a < (1-\lambda)v_b$: the mariginal utility of (short) selling the risky asset is high enough, the optimal strategy is to sell riksy aseet at infinite rate $dM_t/d_t = \infty$ until $v_a = (1-\lambda)v_b$, i.e. we are back to the NT region.

In conclusion, the value function solves the following variational inequality:

$$0 = \max\Bigg[v_t + rbv_b + \frac{a}{x}\alpha(\mu - x)v_a + \alpha(\mu - x)v_x + \frac{a^2}{x^2}\sigma^2 v_{aa} + \sigma^2\frac{a}{x}v_{ax},\ (1+\lambda)v_b - v_a,\ v_a - (1-\lambda)v_b\Bigg]$$

The explicit solution for the variational inequality does not exist, and it is computational costly to find its numerical solution. In particular, if we consider two risky asset, then finding numerical solution for variational inequality of above type becomes intractable. Therefore it is reasonable to find approximation method for the No-Trade region.

2.3.3. No-Trade Region Parameterization

We discretize the continuous time interval $[0, T]$ to N subintervals with equal length. At a particular time step t_k, No-Trade region is determined by its lower and upper boundary as we see in Fig.2. In particular, when transaction cost rate $\lambda = 0$, No-Trade zone degenerates to a curve we derived in Eq.28. When λ becomes larger, No-Trade region grows wider. We utilize our prior knowledge of π^* when $\lambda = 0$ as a warm start, and use Deep Neural Network to parameterize lower and upper boundary of the No-Trade Zone:

$$r_u^{t_k}(x) \quad \approx \pi^*(t_k, x) + f_{\theta^u_{t_k}}(x), \tag{10}$$

$$r_d^{t_k}(x) \quad \approx \pi^*(t_k, x) - f_{\theta^d_{t_k}}(x), \tag{11}$$

where $r_u^{t_k}(x)$ is the upper boundary of No-Trade region and $r_d^{t_k}(x)$ is lower boundary of No-Trade region. For implementation, both $r_u^{t_k}(x)$ and $r_d^{t_k}(x)$ are constructed as a 3-layer fully connected Neural Network with ReLU activation and Batch Normalization. The number of hidden units are $[20, 40, 80]$.

2.3.4. Parameterization Trading Strategy

The rebalancing strategy follows the illustration in Fig.2, when our $(asset\ price, position)$ pair is within the No-Trade region we do not do rebalancing, otherwise we rebalance our position towards the boundary of No-Trade region. More specifically, when we arrive at time t_k, the proportion we invested in risky asset is π_{t_k-}, we rebalance the proportion to π_{t_k+} using the following policy:

$$\pi_{t_k+} = \begin{cases} r_l^{t_k}(X_{t_k}) & \pi_{t_k-} < r_l^{t_k}(X_{t_k}) \\ \pi_{t_k-} & r_l^{t_k}(X_{t_k}) \le \pi_{t_k-} \le r_u^{t_k}(X_{t_k}) \\ r_u^{t_k}(X_{t_k}) & \pi_{t_k-} > r_u^{t_k}(X_{t_k}) \end{cases} \tag{12}$$

2.3.5. Discretization of the Dynamical System

Similar to the theoretical study in the literature (Shreve et al., 1994; Liu, 2004; Muhle-Karbe et al., 2017), we assume the frictional market has proportional transaction cost with rate λ, symmetrically penalizing on both buying and selling activities. Therefore, the transaction cost at each time period with respect to the rebalancing operation is:

$$c_{t_k} = \lambda W_{t_k}|\pi_{t_k+} - \pi_{t_k-}| \tag{13}$$

According to the continuous time dynamics formula, after introducing the transaction costs at each time step, we can write out the system dynamics from t_{k-1} to t_k by discretizing Eq.1-3, detailed formulas can be found in Appendix. Notice that this learning task is essentially a reinforcement learning task, therefore we need to simulate sample paths in the frictional market and use the sample paths as our input in order to learn the optimal trading policy. Moreover, we need to regularize our starting point at the same level. At the beginning of each training epoch, we start at $t_0 = 0$ with n sample paths initialized as:

$$X_0 = [x_0, ..., x_0]^T_{1\times n},$$
$$W_0 = [w_0, ..., w_0]^T_{1\times n},$$
$$\pi_{0-} = [0, ..., 0]^T_{1\times n}.$$

At each time step t_k our position rebalances from the output of the neural network $r_u^{t_k}(x)$ and $r_d^{t_k}(x)$ together with the rebalancing rule (12). We then move to the next time step t_{k+1} using equations (29)-(31). At terminal time T, the $n \times 1$ vector W_T represents the terminal wealth over n sample paths, and the empirical loss over n sample paths is defined as:

$$loss \triangleq \frac{1}{n}\sum_{i=1}^{n} -U(W_T^{(i)}) \tag{14}$$

2.3.6. Design of the Structure of the Deep Neural Network

The computational graph defines the data-flow of the deep neural network (Figure. 5 in Appendix). In the computational graph, 'Rebalance Rule' $(r_l^t(x), r_u^t(x))$ follows equation (12), 'System Dynamics' (X, W, π) follows equations (29)-(31).

Notice that, the 'Rebalance Rule' $(r_l^t(x), r_u^t(x))$ depends not only on the price of the risky asset, but also depends on the time to maturity. In reality this makes sense, because as time getting closer and closer to maturity T, there is less and less time left for the price of the risky asset returns to its mean.

Based on equations (12), the $\pi^*(t_k, x)$ is known. Therefore our learning algorithm is mainly focused on learning a pair of non-negative functions $(f_{\theta^d_{t_k}}(x), f_{\theta^u_{t_k}}(x))$. If we still assume the pair $(f_{\theta^d_{t_k}}(x), f_{\theta^u_{t_k}}(x))$ has time dependence, it will introduce a significantly larger number of tuning parameter, which will make us face the curse of dimentionality as we are using finer discretization scheme. On the other hand, through a rigorous reasoning as well as our simulation results, the time dependence of the 'Rebalance Rule' $(r_l^t(x), r_u^t(x))$ is solely captured by the frictionless optimal strategy π^*. In summary, we can use one single time-independent Deep Neural Network to learn the pair $(f_{\theta^d_{t_k}}(x), f_{\theta^u_{t_k}}(x))$:

$$f_{\theta^u_{t_k}}(x) = f_{\theta^u}(x) \tag{15}$$

$$f_{\theta^d_{t_k}}(x) = f_{\theta^d}(x) \tag{16}$$

We summarize the hyperparameters we use in the following table:

μ	α	σ
15.446	0.113×252	$0.606 \times \sqrt{252}$
T	N	r
1 (year)	252	0.05
X_0	W_0	γ
13.950	100	2

3. Numerical Results

In this section, we provide numerical experiments with real data. We first describe the method of constructing mean-reverting tradable assets. Although the discovery of pairs is not the focus of this paper, the construction process is important for us to estimate the transaction cost. Followed by that, we present the backtest results of a single and multiple risky assets portfolio for year 2014-2017. In particular, we are able to solve the multi-asset portfolio choice problem with 48 Orstein-Ulenbeck assets, 1 risk-free asset and 50 time steps in 3 hours computer time, which is considerable faster than traditional numerical methods that has long been used in this field.

To illustrate our learnt policy, we will present our results with one single risky assets with mean-reverting dynamics.

3.1. Shape of the No-trade Zone

In order to incorporate our model with real-world case scenario, we use parameters calicrated from 2013 VIX front month future data as shown in 2.2, the simulation results illustrate how transaction cost affect the No-Trade zone. The network structure and the optimal hyperparameters we use are shown in the following table:

Structure	# of units	Activation
3 hidden	[20,40,80]	`ReLU`
Batch Normalization	Optimizer	Epochs
`True`	Adam	3000
Learning rate	Learning Rate Decay	
0.01	decay to its 1/2 every 500 epochs	

At each training step, we generate 1000 sample paths. We use Adam optimizer with a starting learning rate $l = 0.1$ and decay to its $1/2$ every 500 training steps. We stop training after 3000 steps. We let $[0, T] = [0, 1]$ to denote a trading year, and discretize it to $N = 252$ intervals, each subinterval represents a single trading day. We run experiments on different transaction cost rate λ and plot the corresponding No-Trade zone at a certain time step:

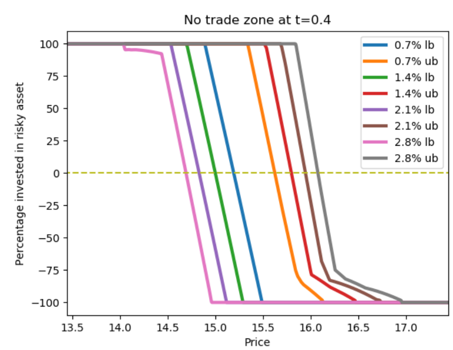
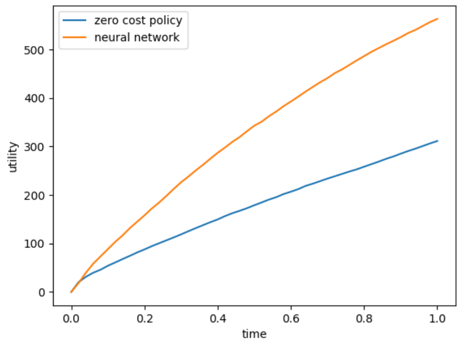


Figure 3. No-Trade Region at $t = 0.4$ for different proportional transaction cost rate.

In Figure 3, 'lb' denotes the lower boundary and 'ub' denotes the upper boundary of the No-Trade zone, percentage number denotes the transaction cost rate λ in (13) and varies from 0.7% to 2.8%. As expected, the No-Trade zone gets wider as the transaction cost increases.

3.2. Comparison Results

We have shown in Sec.3 that DNN learned No-Trade Region meets our intuitions. We have yet to show the optimal trading strategy learned by DNN is better than some baseline strategy.



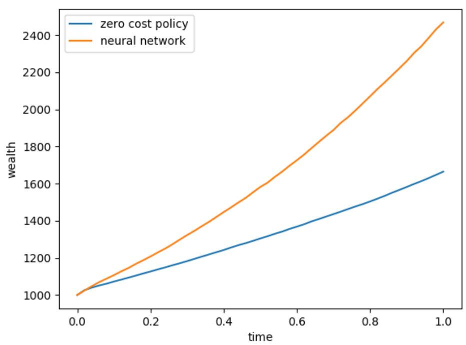


Figure 4. Comparison of the averaged terminal utility and wealth over 10000 sample paths.

4. Discussion and extension

In this work, we present a deep-learning method for solving the portfolio optimization problem where the underlying assets follow Ornstein-Uhlenbeck processes and transaction costs are not negligible. We formulated this as a supervised learning problem using deep neural network to approximate the boundaries of the NT zones and based on the ODE solutions for the no transaction cost case.

Backtest with real data for year 2014-2017 shows the strategy performs well in both one-asset and multi-asset cases, yielding the 35% annual return in the single asset case and doubling the initial wealth each year in the multi-asset case. Importantly, our method based on DNN enjoys superb runtime efficiency. It doesnt suffer from curse of dimensionality as other conventional numerical methods. Hence the trading strategy can be extended to portfolios with a large number of assets This opens a door for future research opportunities as one can further test out other promising neural network architectures or to combine with the reinforcement learning to explore better dynamic trading strategies.

We develop the optimal solution under zero transaction cost assumption, then use the frictionless policy as a warm start for our portfolio optimization problem. We parametrize the No-Trade Zone hence the trading policy using neural network. We repeatedly generate new sample paths according to the system dynamic equations and minimize the loss over all of the sample paths. The experiment on synthetic data shows the learned optimal policy by DNN outperforms the analytic solution we derived under zero transaction cost assumption and the baseline strategy of fully invest in risk-free assets. We pointed out and illustrated that our approach, in the same spirit as the method by (E et al., 2017), could be applied for solving high-dimensional assets allocation problem.

Another interesting possible direction is the small transaction costs scenario. When transaction cost λ is close to 0, there is an asymptotic approximation formula for the No-Trade region given by (Muhle-Karbe et al., 2017). However, since their settings are quite different than our settings, i.e. they are essentially assume a quadratic penalty on the trading rate, whereas in our model the penalty is directly con the total trading volume. In our future work, we would explore the proportional small trading analysis and then show our learned strategy matches the asymptotic strategy derived by (Muhle-Karbe et al., 2017).

Acknowledgements

We thank Yifan for sharing with us his working paper with John M. Mulvey, Mengdi Wang and Jing Ye. We also thank Johannes Muhle-Karbe for introducing the noval comparison baseline as well as the interesting observation by Matt Emschwiller, Benjamin Petit and Jean-Philippe Bouchaud, which is the asymptotic behavior of the optimal strategy when the number of the available assets in the market is going to infinity.

References

Campbell, J. Y. and Viceira, L. M. Consumption and portfolio decisions when expected returns are time varying. *The Quarterly Journal of Economics*, 114(2):433–495, 1999.

Culkin, R. and Das, S. R. Machine learning in finance: The case of deep learning for option pricing. 2017.

Davis, M. H. and Norman, A. R. Portfolio selection with transaction costs. *Mathematics of operations research*, 15(4):676–713, 1990.

E, W., Han, J., and Jentzen, A. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Communications in Mathematics and Statistics*, 5(4):349–380, 2017.

Johnstone, D., Lindley, D., et al. Mean–variance and expected utility: The borch paradox. *Statistical Science*, 28 (2):223–237, 2013.

Liu, H. Optimal consumption and investment with transaction costs and multiple risky assets. *The Journal of Finance*, 59(1):289–338, 2004.

Liu, H. and Loewenstein, M. Optimal portfolio selection with transaction costs and finite horizons. *The Review of Financial Studies*, 15(3):805–835, 2002.

Liu, J. Portfolio selection in stochastic environments. *The Review of Financial Studies*, 20(1):1–39, 2006.

Magill, M. J. and Constantinides, G. M. Portfolio selection with transactions costs. *Journal of economic theory*, 13 (2):245–263, 1976.

Merton, R. C. Lifetime portfolio selection under uncertainty: The continuous-time case. *The review of Economics and Statistics*, pp. 247–257, 1969.

Merton, R. C. Optimum consumption and portfolio rules in a continuous-time model. In *Stochastic Optimization Models in Finance*, pp. 621–661. Elsevier, 1975.

Muhle-Karbe, J., Reppen, M., and Soner, M. A primer on portfolio choice with small transaction costs. *Annual Review of Financial Economics*, 9:301–331, 2017.

Munk, C., Sørensen, C., and Vinther, T. N. Dynamic asset allocation under mean-reverting returns, stochastic interest rates, and inflation uncertainty: are popular recommendations consistent with rational behavior? *International review of economics & finance*, 13(2):141–166, 2004.

Shreve, S. E., Soner, H. M., et al. Optimal investment and consumption with transaction costs. *The Annals of Applied Probability*, 4(3):609–692, 1994.

A. Solution of HJB equation and Optimal Trading Strategy

When there is no transaction cost, we get following HJB equation of the value function:

$$\max_{\pi} \Big\{ -V_\tau + \Big[\frac{\pi\alpha}{X}(\mu - X) + (1-\pi)r\Big]WV_W + \frac{1}{2}(\frac{\pi\sigma}{X})^2 W^2 V_{WW} + \alpha(\mu - X)V_X + \frac{1}{2}\sigma^2 V_{XX} + \sigma\frac{\pi\sigma}{X}WV_{WX}\Big\} = 0 \tag{17}$$

From HJB equation we can compute the optimal asset allocation:

$$\pi^*(\tau, X, W) = -\frac{V_W}{WV_{WW}}\Big[\frac{\alpha(\mu - X)/X - r}{(\sigma/X)^2}\Big] - \frac{XV_{WX}}{WV_{WW}} \tag{18}$$

The HJB equation is solved by first "guessing" a general form for the solution and then verified later. We assume the value function V takes the form:

$$V(W, X, \tau) = \frac{(W\phi(\tau))^{1-\gamma} - 1}{1-\gamma} \tag{19}$$

$$\phi(\tau) = \exp(A(\tau) + B(\tau)X + C(\tau)X^2/2) \tag{20}$$

$$A(0) = B(0) = C(0) = 0 \tag{21}$$

Substituting π^* in (18) and the guessed value function (19)-(21) into Eq.5 generates a quadratic equation of X_t. Making all the coefficients zeros, we obtain the following ODE system of $A(\tau)$, $B(\tau)$ and $C(\tau)$:

$$C'(\tau) = aC^2(\tau) + bC(\tau) + c \tag{22}$$

$$B'(\tau) = aB(\tau)C(\tau) + \frac{b}{2}B(\tau) + dC(\tau) + g \tag{23}$$

$$A'(\tau) = \frac{a}{2}B(\tau)^2 + dB(\tau) + \frac{\sigma^2}{2}C(\tau) + \frac{(\alpha\mu)^2}{2\gamma\sigma^2} + r \tag{24}$$

with boundary condition $A(0) = B(0) = C(0) = 0$ and parameters:

$$a = \frac{1-\gamma}{\gamma}\sigma^2 \quad b = \frac{2(\gamma r - r - \alpha)}{\gamma}$$

$$c = \frac{(\alpha + r)^2}{\gamma\sigma^2} \quad d = \frac{\alpha\mu}{\gamma} \quad g = -\frac{\alpha\mu(\alpha + r)}{\gamma\sigma^2}$$

The ODE system can be solved sequentially for explicit solution:

$$C(\tau) = \frac{2c(1-e^{-\eta\tau})}{2\eta - (b+\eta)(1-e^{-\eta\tau})} \tag{25}$$

$$B(\tau) = \frac{-4gr(1-e^{-\eta\tau/2})^2 + 2g\eta(1-e^{-\eta\tau})}{\eta[2\eta - (b+\eta)(1-e^{-\eta\tau})]} \tag{26}$$

$$A(\tau) = \int_0^\tau \frac{a}{2}B(t)^2 + dB(t) + \frac{\sigma^2}{2}C(t) + \frac{(\alpha\mu)^2}{2\gamma\sigma^2} + rdt \tag{27}$$

with $\eta = \sqrt{b^2 - 4ac}$, we can then plug in $B(\tau)$ and $C(\tau)$ into EQ.18 to get the optimal trading strategy:

$$\pi^*(\tau, X) = \frac{1}{\gamma}\Big[\frac{\alpha(\mu - X)/X - r}{(\sigma/X)^2}\Big] + \frac{1-\gamma}{\gamma}\big[C(\tau)X + B(\tau)\big]X. \tag{28}$$

B. Discretization Formula for System Dynamics

We use $\Delta_{t_{k-1}}X$ to denote the discretized change in the price of the risky asset and similar convention applies to $\Delta_{t_{k-1}}Y$ and $\Delta_{t_{k-1}}W$,

$$\Delta_{t_{k-1}}X = (e^{-\alpha h} - 1)X_{t_{k-1}} + \mu(1 - e^{-\alpha h}) + N(t,h) \tag{29}$$

$$\Delta_{t_{k-1}}Y = (e^{rh} - 1)Y_{t_{k-1}} \tag{30}$$

$$\Delta_{t_{k-1}}W = \frac{\pi_{t_{k-1}^+}W_{t_{k-1}}}{X_{t_{k-1}}}\Delta_{t_{k-1}}X + \frac{(1-\pi_{t_{k-1}}^+)W_{t_{k-1}}}{Y_{t_{k-1}}}\Delta_{t_{k-1}}Y - c_{t_{k-1}} \tag{31}$$

where

$$N(t,h) := \sigma e^{-\alpha(t+h)}\int_t^{t+h} e^{\alpha u}dZ_u \sim N(0, \frac{\sigma^2(1-e^{-2\alpha h})}{2\alpha}),$$

$$h := t_k - t_{k-1} = \frac{T}{N}.$$

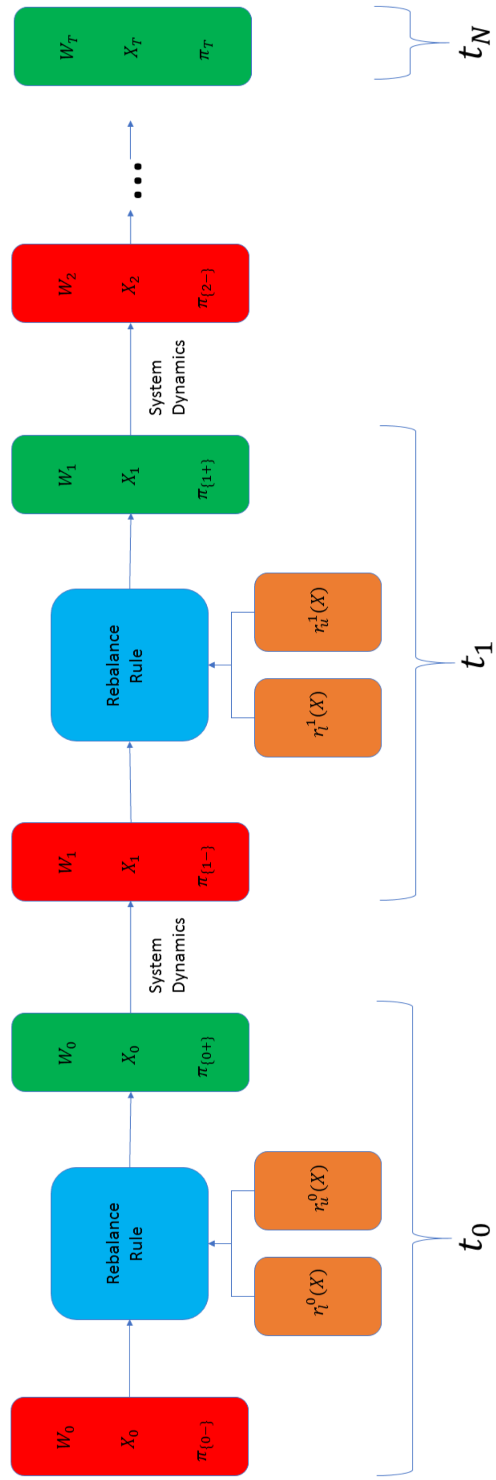


Figure 5. Computational Graph of DNN.

C. Computational Graph of DNN

D. Data

D.1. Λ for Year 2014-2017

Ticker	2014	2015	2016	2017
CMCSA	0.1512044	0.09903023	0.088121957	0.079624755
SBUX	0.119861566	0.106692767	0.098077297	0.089953586
F	0.066364963	0.050982492	0.04489592	0.043963961
MDT	0.107307167	0.102353912	0.086329823	0.08685961
DUK	0.121337593	0.08755698	0.058748914	0.061531854
D	0.089492081	0.098878838	0.067219952	0.066434508
AEP	0.113546658	0.09884012	0.078860184	0.072107818
PG	0.126228638	0.096886326	0.069082293	0.05817882
KO	0.115512178	0.096943248	0.081430882	0.069477828
PEP	0.149839963	0.12689585	0.111955847	0.087106367
CL	0.174192155	0.135077399	0.105763485	0.082967234
MSFT	0.14835767	0.13526843	0.085187717	0.080934768
INTC	0.085382411	0.076182585	0.0530736	0.045079178
CSCO	0.10030578	0.083682861	0.081974905	0.07533517
ORCL	0.078216727	0.072394562	0.062725916	0.058971221
T	0.082766055	0.045339141	0.05609628	0.022236044
VZ	0.096536174	0.090379316	0.078847628	0.073606135
HON	0.194247732	0.128088189	0.12284851	0.08929841
JPM	0.077781933	0.082246427	0.062812423	0.058392144
WFC	0.140411733	0.101826513	0.08560014	0.069309928
IP	0.110442126	0.094170262	0.077487284	0.064448131
WY	0.148304805	0.111528619	0.084138711	0.059737781
PLD	0.127881484	0.124128489	0.10390303	0.083838078
VTR	0.123566047	0.103444615	0.094639029	0.092862104
BK	0.152665573	0.086988846	0.066685143	0.06256798
NTRS	0.085238352	0.09623448	0.080502361	0.078105738
FITB	0.185831425	0.146666291	0.119258936	0.087743548
C	0.121257087	0.097341215	0.085707487	0.080808048
PNC	0.111797979	0.076926986	0.07132905	0.068030345
RF	0.13340155	0.113899191	0.066644951	0.065864507
FII	0.070298788	0.0692512	0.063582922	0.048911198
ETFC	0.106017484	0.076158733	0.08130912	0.07331601
STT	0.069168964	0.067691624	0.076441998	0.068969932
IVZ	0.084301846	0.079766503	0.067302767	0.060173509
LAZ	0.165662314	0.162293273	0.148215258	0.122143918
TOL	0.120344576	0.101937009	0.101144192	0.092829882
JCI	0.134428205	0.12877133	0.121046752	0.077376392
MAS	0.124948355	0.101403999	0.085898027	0.068374814
PHM	0.105976914	0.093875333	0.069488354	0.058985726
LEN	0.102457074	0.06532253	0.05861495	0.054436747
ODP	0.130774618	0.069522574	0.069515129	0.05851729
URBN	0.099900322	0.07510007	0.054682685	0.044866865
CY	0.066689969	0.043098203	0.03907061	0.043387069
ADI	0.085303172	0.08190453	0.089768265	0.077714042
ATI	0.127979297	0.089665411	0.071413592	0.068926531
HAL	0.111331436	0.106928803	0.087892309	0.092583281
PTEN	0.087740809	0.08052275	0.072652146	0.069694441
SPN	0.120296202	0.102141582	0.091339108	0.069543253

Figure 6. Λ for Year 2014-2017.

D.2. Σ_{OU} for Year 2014

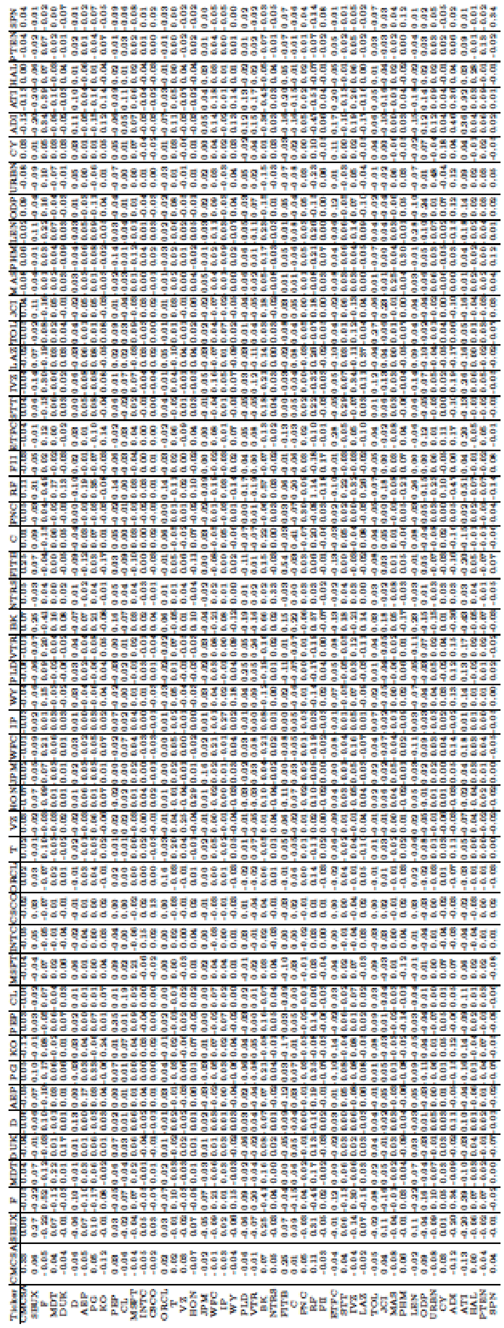


Figure 7. Σ_{OU} for Year 2014.

D.3. Σ_{OU} for Year 2015

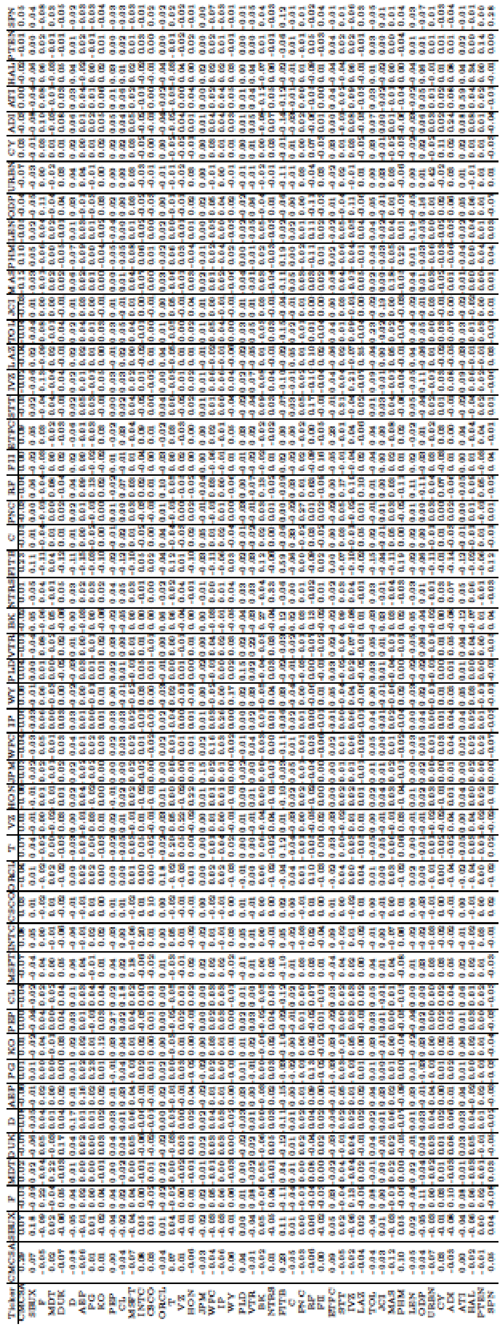


Figure 8. Σ_{OU} for Year 2015.

D.4. Σ_{OU} for Year 2016

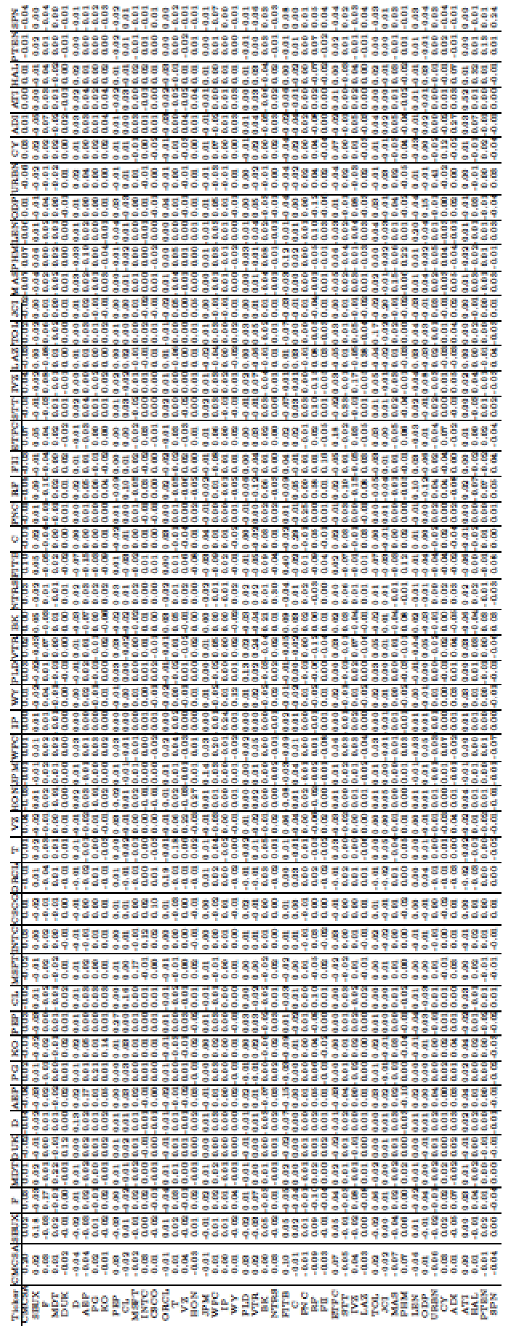


Figure 9. Σ_{OU} for Year 2016.

D.5. Σ_{OU} for Year 2017

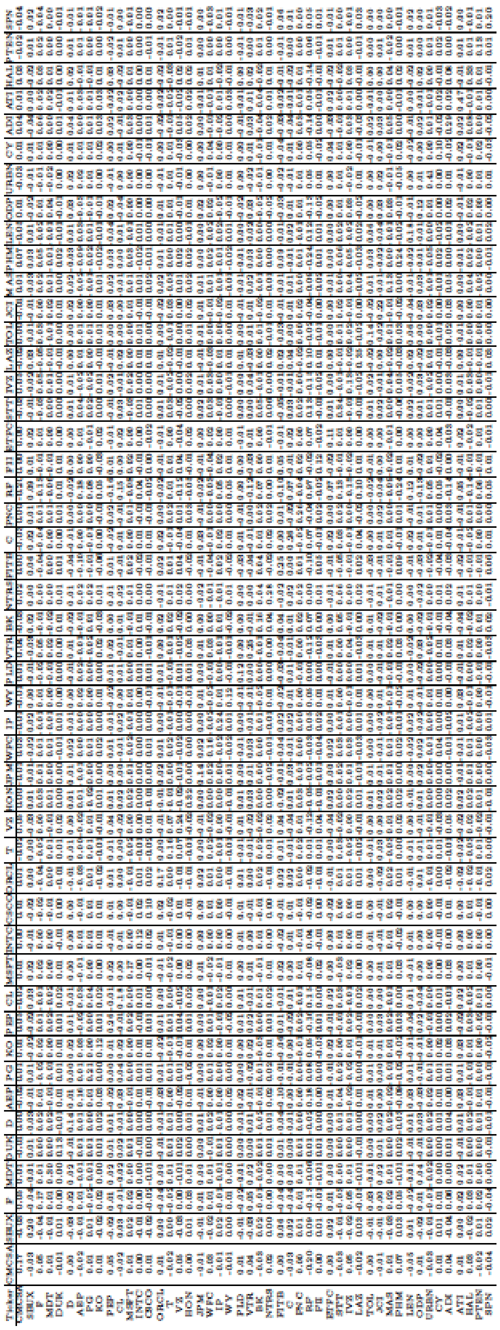


Figure 10. Σ_{OU} for Year 2017.