TradeStation における複利型強化学習を用いた Strategy 構築

長瀬舜 [1]* 松井藤五郎 [1] 後藤卓 [2] 和泉潔 [3] 陳ユ [3] 鳥海不二夫 [3]

[1] 中部大学 [2] 株式会社三菱東京 UFJ 銀行 [3] 東京大学

Abstract: 本論文では，株を実際に取引できる TradeStation において，複利型強化学習を用いて取引戦略を獲得する方法について述べる．従来のカブロボでは 1 日に 3 回プログラムが実行されるだけだったが，TradeStation では分足の価格情報が更新されるごとにプログラムが実行されるため，デイトレードを行うことができる．そこで，本論文では，これまで日次取引を対象に開発してきた手法をデイトレードに適用する方法を提案する．また，SPDR S&P 500 ETF Trust を対象とした実験結果を示す．

1 はじめに

我々は，これまで，日本株の仮想取引環境であるカブロボにおいて複利型強化学習 [1, 2, 3] を用いて取引戦略を獲得する手法を開発してきた [4, 5, 6]．カブロボ[*1]は Java 言語によって実装され，パフォーマンス分析やテクニカル指標など株取引に必要な API が提供されていることから，高度な取引戦略を実装することができる [7]．しかしながら，プログラムの実行が市場が閉じた後，前場が開く前，後場が開く前の 1 日 3 回だけに制限されていて，デイトレードを行うことはできない．また，カブロボでは，プログラムを自分で動かして実際に取引を行うことはできない．

これに対し，TradeStation[*2]は，価格情報が更新されるごとにプログラムが実行されるため，更新間隔を短くすることによってデイトレードを行うことができる．また，TradeStation は，開発したプログラムを用いて実際の取引を行うことができる．

そこで, 本論文では, これまで開発してきた複利型強化学習を用いて取引戦略を獲得する手法を TradeStation 上に実装し，デイトレードを行う方法を提案する．ただし，TradeStation は EasyLanguage という独自の言語を用いており，取引に必要な API も十分には提供されていない．そのため，複利型強化学習を用いて取引戦略を獲得するには実装上の工夫が必要となる．従来研究と本研究における環境の違いを表 1 に示す．

表 1 従来研究との比較

	従来研究	本研究
プラットフォーム	カブロボ	TradeStation
使用言語	Java	EasyLanguage
バックテスト	○	○
実取引	×	○
対象商品	日本株	米国株 為替 先物 オプション
時間足	日足以上	日足以上 時間足 分足

2 複利型強化学習を用いた株取引戦略の獲得

複利型強化学習は，割引複利リターン

$$(1+R_{t+1}f)(1+R_{t+2}f)^{\gamma}(1+R_{t+3}f)^{\gamma^2}\ldots = \prod_{k=0}^{\infty}(1+R_{t+k+1}f)^{\gamma^k}$$

の期待値を最大化するような行動規則を学習する．ここで，R_t は時刻 t に観測されたリターン，γ は割引率パラメーター，f は投資比率パラメーターを表す．割引複利リターンは，対数を取ることで，従来の強化学習と同じように再帰的な形で表すことができる．すなわち，行動規則 π の下での状態 s の価値 $V^{\pi}(s)$ と行動規則 π の下での状態 s における行動 a の価値 $Q^{\pi}(s,a)$ は次のよう

* 連絡先: ShunNagase@1056lab.org

*1 http://www.kaburobo.jp

*2 http://www.tradestation.com

Algorithm 1 複利型 OnPS アルゴリズム.

入力: 割引率 γ, 強化学習率 α, 初期優先度 p, 初期投資比率 f, 投資比率学習率 η
for all s, a **do**
  $P(s,a)$ を p に初期化
  $f(s,a)$ を f に初期化
end for
loop (各エピソードに対して繰り返し)
  $c(s,a) \leftarrow 0$ **for all** s, a
  状態 s を初期化
  repeat (エピソードの各ステップに対して繰り返し)
    P から導かれる行動規則に従って s での行動 a を選択
    $c(s,a) \leftarrow c(s,a) + 1$
    行動 a を実行し，利益率 R と次の状態 s' を観測
    for all s, a **do**
      $P(s,a) \leftarrow P(s,a) + \alpha \log(1 + Rf(s,a))c(s,a)$
      $c(s,a) \leftarrow \gamma c(s,a)$
    end for
    $f(s,a) \leftarrow f(s,a) + \eta \frac{R}{1+Rf(s,a)}$
    $s \leftarrow s'$
  until s が終端状態
end loop

に表される.

$$V^{\pi}(s) = \mathrm{E}_{\pi}\left[\log \prod_{k=0}^{\infty}(1 + R_{t+k+1}f)^{\gamma^k} \middle| s_t = s\right] = \sum_{a\in\mathcal{A}} \pi(s,a) \sum_{s'\in\mathcal{S}} \mathcal{P}^a_{ss'}\left(\mathrm{R}^a_{ss'} + \gamma V^{\pi}(s')\right) \quad (1)$$

$$Q^{\pi}(s,a) = \mathrm{E}_{\pi}\left[\log \prod_{k=0}^{\infty}(1 + R_{t+k+1}f)^{\gamma^k} \middle| s_t = s, a_t = a\right] = \sum_{s'\in\mathcal{S}} \mathcal{P}^a_{ss'}\left(\mathrm{R}^a_{ss'} + \gamma V^{\pi}(s')\right) \quad (2)$$

ここで，$\pi(s,a)$ は行動規則 π の下で状態 s において行動 a が選択される確率（行動選択確率），$\mathcal{P}^a_{ss'}$ は状態 s において行動 a を行ったときに次の状態が s' になる確率（状態遷移確率），$\mathrm{R}^a_{ss'}$ は状態 s において行動 a を行って次の状態が s' になったときに得られるリターンの期待値を表す．複利型強化学習では，すべての s, a に対してこの $Q^{\pi}(s,a)$ を最大化するような行動規則 π を学習する．

本論文では，取引戦略の学習にオンライン勾配法を用いて投資比率最適化をする複利型 OnPS [6, 8] を用いる．複利型 OnPS のアルゴリズムを Algorithm 1 に示す．

複利型強化学習における状態は，終値と移動標準偏差に基づいた二次元空間で表現する．株価は大きく変動するため，直近 60 個のデータと比較した相対的な値として正規化することによって，株価が大きく異なる場合でも学習した行動規則を利用できるようにする．具体的には，移動平均および移動標準偏差の算出期間を k とし，以下のようにして相対化 [9] する．

$$o_t = \frac{v_t - \mu_{t,k}}{4\sigma_{t,k}} \quad (3)$$

ここで，v_t は t における値，$\mu_{t,k}$ は時刻 t の直近 k 個のデータから求めた移動平均，$\sigma_{t,k}$ は同じく移動標準偏差を表す．終値を相対化した値を相対終値（RCP），移動標準偏差を相対化した値を相対移動標準偏差（RMSD）と呼ぶ．RCP が正のときは現在の株価が移動平均株価より大きい，すなわち，株価が上昇していることを表している．RMSD が正のときは現在の標準偏差が移動平均標準偏差より大きい，すなわち，株価の変動が大きくなっていることを表している．これらの値は共に連続値をとるので，15×15 の格子状に配置した動径基底関数を用いて線形関数近似を行う．

エージェントの行動は買いと売りの 2 種類である．株式を購入している状態をロング・ポジション，株式を信用売りしている状態をショート・ポジションという．エージェントは，複利型強化学習によって学習された取引戦略によって行動を選択し，オンライン勾配法によって学習された投資比率 f によってポジションの大きさを調整する．

3 TradeStation における Strategy 構築

3.1 TradeStation

TradeStation 証券は米国の権威ある投資週刊誌「BARRON'S」で毎年高い評価を得ているネット証券で，同社の TradeStation プラットフォームは株式，オプション，先物，FX などの注文執行ならびにシステムトレードをカスタム・デザイン，バックテスティング出来るようになっている．2011 年 4 月にマネックス・グループが同社の株を買収し，日本向けのプラットフォームが公開予定となっている．本研究では，米国でのリリース版 TradeStation9.1 を使用する．

3.2 TradeStation における Strategy 構築

Strategy とは指標に基づいて自動的に売買する取引ルールのことを指す．TradeStation では，チャート，マトリックス，レーダースクリーンなど様々な機能を使用することができ，Strategy はチャートに適用する形で使用する．Strategy は TradeStation に付随する TradeStation Development Environment という EasyLanguage 専用のエディタを使用することで編集・作成が可能となっている．TradeStation では，任意の

銘柄，時間足，期間のチャートを表示することができ，Strategy を適用すると Strategy に記述したルールに従ってバックテストが行われる．

EasyLanguage は，その名の通り簡易的な構文によりプログラムを記述することができる．例えば，

```
Buy 100 shares next bar at market;
```

という一文で，次の足に成行で 100 株買注文を出すことができる．しかし，システムトレード専用で，かつ実取引が前提におかれた言語であるため，本 Strategy を構築する上で必要な API が提供されていない．例えば，自身の資産や現在保有しているポジション情報などをシステムから参照する API は用意されているが，実取引した際に反映される情報であり，バックテスト中の処理には影響されない．そのため，強化学習をする上で必要な情報は自ら導出する必要がある．

本研究における Strategy は，直近 60 足を指標とし，それによって当日の行動 (ロング・ポジション，ショート・ポジション) を決定している．米国証券取引所は日本時間の 23:30-6:00 の間取引が行われるが，前日の終値と当日の始値には関連性がないため，直近 60 足が日をまたいでいると指標を正確に読み取れているとは言えない．そのため，本 Strategy では市場が開いてから指標が正確に読み取れるようになるまでの期間を様子見し，一切の取引を行わない．

実際に Strategy をバックテストした TradeStation の実行画面を図 1 に示す．グラフの横軸が時間，縦軸が株価を示し，ボリンジャーバンドを表示している．足に対して下からの矢印が買注文，上からの矢印が売注文で，数値は取引した株数を示す．グラフ中の縦の破線は営業日の境目を示す．

4 実験

4.1 実験手順

SPDR S&P 500 ETF Trust (SPY) を取引対象とし，時間足を 1 分足，手数料を 0 として実験を行った．学習期間を 1 週間，2 週間，1 か月，3 ヶ月，6 ヶ月，1 年，2 年，運用期間を 1 日とし，それぞれ無作為に 30 回バックテストを行い，利益率，最大ドローダウン，シャープレシオを評価した．参考のため，2012 年から 2013 年にかけての SPY の値動き（日足終値）を図 2 に示す．

図 2　SPDR S&P 500 ETF Trust (SPY) の値動き．

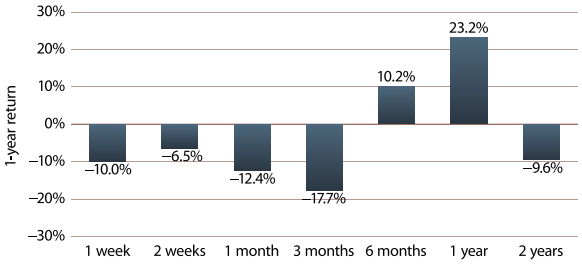

図 3　年換算利益率．

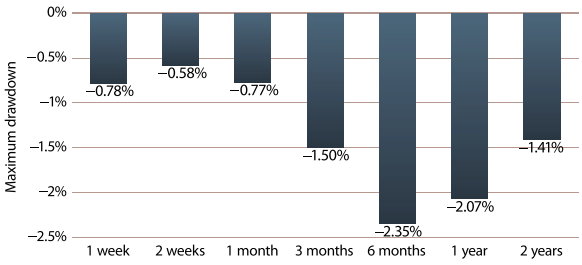

図 4　一日あたりの最大ドローダウン．

4.2 実験結果

30 回のバックテストの結果の幾何平均利益率から 1 年間を 250 営業日として年換算の利益率を求めた結果を図 3 に示す．また，一日あたりの最大ドローダウンを図 4 に示す．最後に，無リスク資産の利益率を 0% として，利益率と同様に年換算のシャープレシオを求めた結果を図 5 に示す．

学習期間が 3 ヶ月以下のときと 2 年のときは利益をあげることができなかったが，学習期間が 6 ヶ月のときは年換算で 10.2%，学習期間が 1 年のときは年換算で 23.2% の利益をあげることができた．また，最も利益率が高かった学習期間が 1 年のときの 1 日あたりの最大ドローダウンは −2.07%，シャープレシオは 1.96 だった．

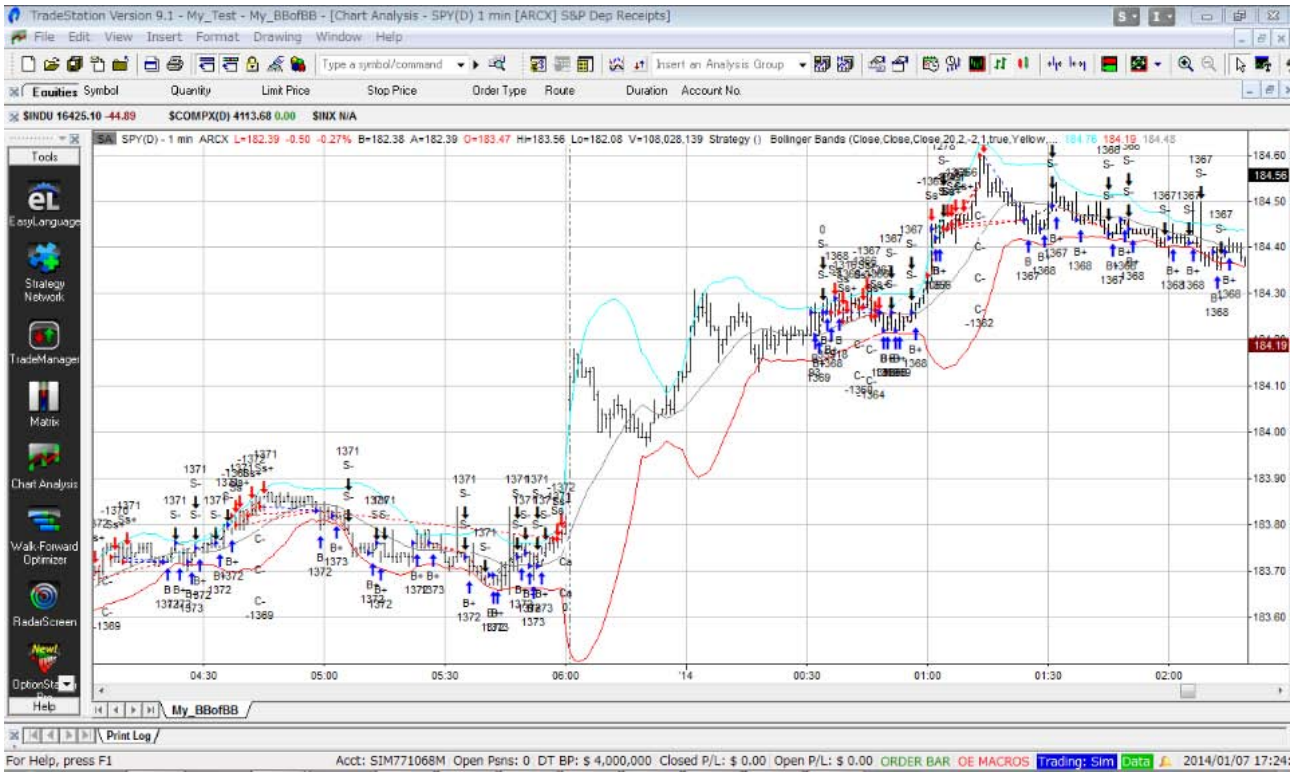

図 1　TradeStation の実行画面.

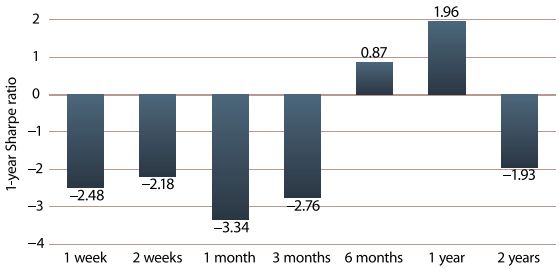

図 5　シャープレシオ.

4.3　考察

学習期間が短いときは利益をあげることができなかったことから，強化学習によって Strategy を構築するにはある程度の学習期間が必要であると考えられる．また，学習期間が長いときも利益をあげることができなかった．これは，強化学習においては初期の学習が大切であるが，学習期間が長いときはより遠い過去の時点から学習を始めることになるため，評価日と市場環境がかなり異なる期間に初期の学習が行われることに原因があると考えられる．

今回の場合，次の 2 点から学習期間を 1 年とするのが最も良いだろう．

1. 学習期間が 1 年のときに複利利益率とシャープレシオが最大となった．
2. 複利利益率がプラスとなったうち，最大ドローダウンが最も小さい[*3]のが学習期間を 1 年としたときだった．

5　まとめ

本論文では，複利型強化学習を用いて取引戦略を獲得する手法を TradeStation 上に実装し，デイトレードを行う方法を提案した．

はじめに，複利型強化学習を株取引に応用する手法について述べた．

次に，TradeStation での Strategy 構築手法を述べた．デイトレードで日をまたいで指標を作ることを避けるため，市場が開いてから指標を作り出す直近 60 足の間取引をしないことを提案した．

また，強化学習の株取引での評価として，学習期間の長さを変え検証を行った．実験の結果から，複利利益率

[*3] 値の上では最も大きい

とシャープレシオが最大となり，かつ複利利益率がプラスとなったうち，最大ドローダウンが最も小さいことから，学習期間を 1 年とするのが最も良いといえる．

現在は手数料を考慮していないため，今後は手数料を考慮した取引戦略を獲得する方法を検討したい．

最後に，TradeStation はカブロボに比べて API が非常に使いにくい．また，TradeStation においてプログラミング言語が EasyLanguage である必要はないように思われる．Java あるいは Python のような広く使われている言語で Strategy をプログラミングできるようになれば，利用者が増えるだろう．

留意事項

本論文は三菱東京 UFJ 銀行の公式見解を表すものではありません．

謝辞

本研究で使用している TradeStation のアカウントはマネックス証券株式会社より提供していただいています．ここに感謝の意を表します．

参考文献

[1] 松井藤五郎. 複利型強化学習. 人工知能学会論文誌, Vol. 26, No. 2, pp. 330–334, 2011.

[2] 松井藤五郎, 後藤卓, 和泉潔, 陳ユ. 複利型強化学習の枠組みと応用. 情報処理学会論文誌, Vol. 52, No. 12, pp. 3300–3308, 2011.

[3] T. Matsui, T. Goto, K. Izumi, and Y. Chen. Compound reinforcement learning: Theory and an application to finance. *EWRL 2011*, LNCS 7188, pp. 321–332, Springer, 2012.

[4] 松井藤五郎. カブロボへの招待–人工知能を用いた株式取引–. 人工知能学会誌, Vol. 22, No. 4, pp. 540–547, 2007.

[5] 松井藤五郎, 後藤卓. 強化学習を用いた金融市場取引戦略の獲得と分析. 人工知能学会誌, Vol. 24, No. 3, pp. 400–407, 2009.

[6] 後藤卓, 松井藤五郎, 大澄祥広. 複利型強化学習の株式取引への応用, 第 27 回人工知能学会全国大会 (JSAI 2013), 4I1-OS-16-4 (2013)

[7] 鳥海不二夫. 株式自動売買ソフトウェア スーパー・株ロボを作ろう, 秀和システム (2006)

[8] 松井藤五郎, 後藤卓, 和泉潔, 陳ユ. 複利型強化学習における投資比率の最適化. 人工知能学会論文誌, Vol. 28, No. 3, pp. 267–272, 2013.

[9] T. Matsui, T. Goto, K. Izumi. Acquiring a government bond trading strategy using reinforcement learning. *Journal of Advanced Computational Intelligence and Intelligent Informatics* (JACIII), Vol. 13, No. 6, pp. 691-696 (2009)