

# 実時間遺伝的アルゴリズムによる 金融時系列データ処理

牟田 英正†

本稿では、為替等の自動取引に用いるパラメータを、遺伝的アルゴリズムを用いて実時間で探索するシステムの実装について述べる。パラメータは 3~5 ビットの期間遺伝子および条件遺伝子にエンコードされ、これらの組み合わせによって生成される戦略染色体によって個体固有のエントリー条件、エグジット条件が決定される。個体の評価では、実際に分足時系列データを用いてバックテストを行い、利益率、勝率、最大ドローダウンなどから適合度を計算する。このような探索を、1) 交叉や突然変異によって生成される染色体から既に評価済のものを排除すること、2) 期間遺伝子によって計算される毎分のテクニカル指標を事前計算すること、3) 1 世代における各個体の適合度評価を並列処理すること、などにより高速化した。これらの改良により、最新の分足時系列データから常にパラメータを探索し続けるシステムを実装した。

## A financial time series data processing by real-time genetic algorithm

Hidemasa MUTA†

This paper describes an implementation of real-time parameter searching system for foreign exchange algorithmic trade by genetic algorithm. The parameter is encoded as 3-5 bit time period gene and condition gene. The trade strategy chromosome which consists of them defines the individual unique entry and exit decisions. To evaluate the individuals, the fitness is calculated from the profit ratio, winning percentage, and maximum drawdown by executing back-test on the actual minutely time series data. I accelerated this search by 1) avoiding duplicated chromosome which is generated again by the recombination and the mutation, 2) pre-calculating the technical indexes which are determined by time period gene for every minutely period, and 3) parallel processing the fitness evaluation for each individual in a single generation. These improvements allow implementation of the real-time parameter searching system from the latest minutely time series data.

### 1. はじめに

#### 1.1 背景

近年、為替取引や株取引を自動化したアルゴリズム・トレーディングが注目されている。FXCM Forex System Selector [1]では、40 種類以上の為替自動取引アルゴリズムが提供されており、顧客は任意のアルゴリズムに任せて自動で為替証拠金取引を行うことができる。また、クリック証券 Web サービス[2]では、顧客からの売買注文の受付や、リアルタイム為替レートの提供を、HTTPS による XML の交換によって行っている。

このように個人投資家であっても手軽に自動取引に参加できるインフラが急速に整いつつある。近い将来、自動取引アルゴリズムなしでは為替取引、株取引等で利益をあげることが難しくなることが予想される。しかしながら、市場は様々な要因により常に変化してお

り、特定のアルゴリズム、特定のパラメータで利益をあげ続けられるほど甘いものではない。本研究では、遺伝的アルゴリズム (以下、GA) を用いて、最新の分足時系列データから常にパラメータを探索し続けることによって、市場の変化に柔軟に適應できるシステムの実装を目指す。

本稿は、このようなシステムの実装法および、その高速化について論じることを目的とする。実際の取引戦略そのものや、その取引戦略にしたがって取引することによって得られる結果について論じるものではない。しかしながら、実際に利益をあげられる現実的なシステムを設計するために、交叉率、突然変異率、個体数などの決定において、実際の分足時系列データを用いた評価を行っている。この評価データは、あくまでも実装のための最適値を求めるためのものであり、この性能の優劣について述べるものではない。

#### 1.2 実時間探索の必要性

取引時間中に分足の粒度で実時間探索を行う必要性について、為替取引における市場環境の変化を例に述べる。為替取引における各通貨ペアの値動きは、その

† 電気通信大学 大学院情報システム学研究所  
Graduate School of Information Systems, The University of  
Electro-Communications

通貨間の政策金利の差に大きく影響される。これは、金利の安い通貨を売って、金利の高い通貨を買うことによって得られるスワップ金利を主な目的としてポジションを立てる投資家が多いためである。

2008年9月のリーマンショックに端を発する金融危機によって、各国通貨の政策金利は大幅に引き下げられた。これにより米ドルは実質ゼロ金利となり、それまでの円と米ドルの金利が逆転した。2008年12月16日、FOMCによるゼロ金利政策発表の瞬間を境に、米ドル円の値動きはそれまでの値動きから一転し、異なるルールに支配されるものとなった。

このような局面で、過去の時系列データのみを対象に探索された条件で自動取引を続けることは致命的な結果をもたらす。常に最新の時系列データを取り入れ、最大ドローダウンを最小化するような探索を行い続けることにより、市場をとりまく様々な外的要因に柔軟に対応し、リスクを最小限に抑えることが可能となる。

### 1.3 関連研究

これまでにも、機械学習などを用いて為替取引、株取引における金融時系列データの方向性を予想する様々な研究が発表されている。Jingtao Yaoらは、[3]で8週間分の時系列データから計算された6種類のテクニカル指標を入力信号としたニューラルネットワークを用いて、為替レートの方角性を予想している。M.A.H.Dempsterらは、[4]で強化学習および遺伝的プログラミングを用いて、複数のテクニカル指標からなる売買ルールを自動生成することを試みている。さらに、彼らは、[5]でこの成果を基にした実時間機械学習システムを提案している。本研究と同様にGAを用いた研究としては、[6][7][8][9]などがあげられる。テクニカル指標の閾値を条件遺伝子とする点などでは、平林らによる[8][9]が本研究の実装形態に近い。

これらの文献においては、学習結果を用いて運用した場合の利益などを評価の対象として、その性能について主に述べている。本稿では、アルゴリズムそのものの性能ではなく、実時間で運用が可能なシステムの実装法について述べることを目的とする。

## 2. 設計・実装

### 2.1 対象

システムを設計するにあたって、はじめに、対象とする金融時系列データとして最もふさわしいものについて論ずる。表1は、個別株式取引、指数先物取引(日経225先物の大阪証券取引所での取引)、為替取引について、その特徴について比較したものである。

一般に取引時間外をまたいでポジションを持ち続け

ることは、予想外のさまざまな外部要因による損失を被るリスクをかかえることになり、自動取引には不向きである。値動きが連続する取引期間内に取引を完結させ、なおかつ利益をあげるためには、十分な取引期間が必要である。その点において、月曜朝から土曜朝までの5日間、値動きの連続性が保障されている為替取引が有利であることがわかる。

また、普遍性のある特徴を発見するためには、十分な流動性・取引規模があることが前提である。個別株式取引の場合、銘柄によっては出来高が少なく、資金力のある特定の投資家の取引によって大きく値動きすることがある。このような値動きは機械学習などの手法では予測不可能である。また、本研究では時系列データを用いたバックテストによって評価を行うため、自分自身の取引によって大きく値が動くことを想定していない。従って、自分自身の取引規模と比較して十分に大きな流動性のある市場を対象とすることが望ましい。

この他に、値動き粒度、スプレッド、値幅制限、注文遅延時間など、さまざまな観点から、為替取引が最も有利であることがわかる。

表1 取引対象の比較

Table 1 Comparison of trading target

	個別株式	指数先物	為替
連続期間	120/180分	120/190分	5日
流動性	△	○	○
値動き粒度	荒い	細かい	細かい
スプレッド	大きい	大きい	小さい
値幅制限	あり	あり	なし
注文遅延	大きい	大きい	小さい

### 2.2 探索手法

探索手法としてエリート選択によるGAを用いる。GAで淘汰の対象とする各個体には、取引においてエントリー条件、エグジット条件を決定するさまざまなパラメータを遺伝子として持たせる。

図1は、値動きの連続性が保障される5日間の分足時系列データを用いて、特定のパラメータに基づくエントリー条件、エグジット条件を持つ戦略A、戦略Bによって、エントリー判定、エグジット判定を行った例である。このように過去の時系列データを用いて、取引戦略のシミュレーションを行うことをバックテストという。本研究では、このような5日単位でのバックテストを一定期間(8週~52週)の時系列データに対して行い、その利益率、勝率、最大ドローダウンなどからスコアを計算する。各個体は淘汰の課程で、バ

ックテストのスコアを適合度として評価され、最終的に最もバックテストの成績の良い個体を持つパラメータが、実時間で取引に用いられる。

このような探索は通貨ペアごと、戦略ごとに足時系列データが更新されるたびに行われることが理想的である。したがって、本研究の目標は1分以内に1通貨ペアあたりの全戦略(3~4種類)のパラメータ探索を完了させることである。実際には、取引の対象となる通貨ペアは複数あるため、これら全てを1分以内に完了するためには现阶段ではクラスターシステムを必要とする。なお、実取引で売買エントリーをしている状態では、エグジットするまでの間、再探索によるパラメータの更新は行われない。これは、エントリー条件とエグジット条件の整合性を保つためである。

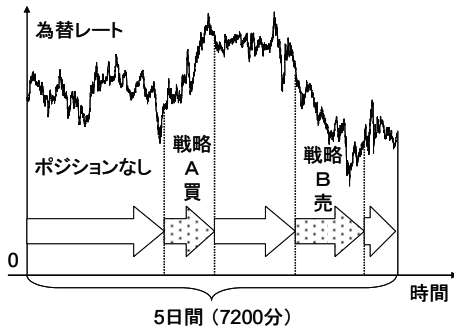


図1 バックテスト例

Figure 1 Back-test example

### 2.3 遺伝子・染色体

図2に各個体を持つ染色体の例を示す。染色体内における遺伝子の種類やそのビット長などは戦略ごとに異なる。ここでは例として、戦略A、戦略B、戦略Cの染色体を挙げる。これらの戦略は、個別に探索プロセスにかけられ、実取引段階で併用される。

染色体全体のビット長は最大30ビットとする。これは、高速化のために32ビット整数型に収めることと、3.1で述べる重複染色体を排除するための評価済フラグテーブルのメモリ制限によるものである。また、限られた30ビットを有効に活用するため、イントロンの挿入は行わない。

各戦略染色体には、期間遺伝子グループと、条件遺伝子グループがある。表2は、期間遺伝子のデコード例である。売買戦略を決定するためには、時系列データから表2のようなテクニカル指標を計算する必要がある。このようなテクニカル指標の計算には期間パラメータが必要であり、期間パラメータの設定次第でテクニカル指標の値も大きく変化する。各テクニカル指

標の意味については、[8][9]に解説があるためここでは省略する。表3は、条件遺伝子のデコード例である。条件遺伝子は、テクニカル指標を用いてエントリー、エグジット判定を行うための閾値として用いられる。

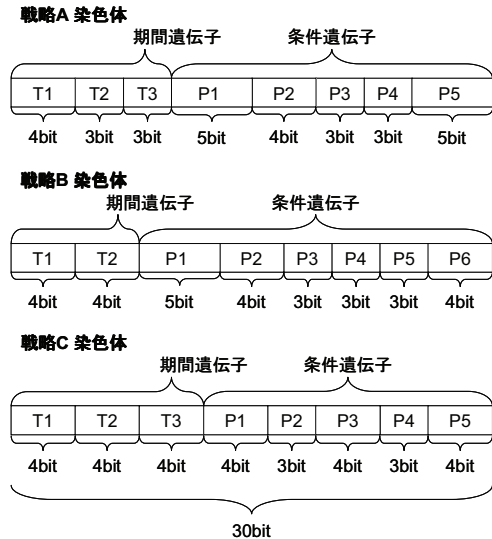


図2 遺伝子・染色体

Figure 2 Genes and chromosomes

表2 期間遺伝子のデコード例

Table 2 Decoding example of time period gene

テクニカル指標	計算期間
移動平均 (MA)	$30 + 10 \cdot T1$ (4bit) 分
標準偏差 ( $\sigma$ )	$30 + 10 \cdot T1$ (4bit) 分
加重移動平均(短期)	$30 + 10 \cdot T1$ (4bit) 分
加重移動平均(長期)	$60 + 20 \cdot T1$ (4bit) 分
MACD	$5 + 5 \cdot T2$ (3bit) 分
RSI	$7 + T3$ (3bit) 分

表3 条件遺伝子のデコード例

Table 3 Decoding example of condition gene

テクニカル指標	条件	アクション
移動平均 標準偏差	現在値 < $MA - 0.1 \sigma \cdot P1$ (5bit)	買エントリー 売エグジット
移動平均 標準偏差	現在値 > $MA + 0.1 \sigma \cdot P1$ (5bit)	売エントリー 買エグジット
加重移動平均	短期 - 長期 > 現在値 $\cdot \alpha P2$ (4bit)	買エントリー 売エグジット
加重移動平均	長期 - 短期 > 現在値 $\cdot \alpha P2$ (4bit)	売エントリー 買エグジット
MACD	$5 + P3$ (3bit) 分以上連続上昇	買エントリー 売エグジット
MACD	$5 + P3$ (3bit) 分以上連続下降	売エントリー 買エグジット
RSI	$20 + 10 \cdot P4$ (3bit) 以上	売エントリー 買エグジット
RSI	$80 - 10 \cdot P4$ (3bit) 以下	買エントリー 売エグジット

なお、表2、表3で挙げられているテクニカル指標、判定条件はほんの一例であり、実際には更に多様なテクニカル指標と、複雑な判定条件が用いられる。

#### 2.4 戦略

具体的な取引戦略を論文等の文献で公にすることは、市場に影響を与える可能性があるため、本稿では詳細な戦略の公開は差し控える。ここでは、本研究におけるすべての戦略に共通する考え方を述べる。

一般に取引戦略には、順張りや逆張りがある。順張りは値動きのトレンドと同じ方向に売買する戦略で、逆張りはトレンドの反転を期待して行き過ぎた値動きに対して反対の方向に売買する戦略である。自動売買で用いる戦略として考えた場合、逆張りは短期間で大きな利益をあげられる可能性がある反面、リーマンショックのような想定外の値動きで大きな損失を被るリスクがある。そのため、本研究のシステムでは主に順張り戦略で最大ドローダウンを最小限に抑えることを重視している。

このような複数の戦略について個別にパラメータの探索を行い、実取引では、市場のトレンドに応じてエントリー条件を満たした戦略を取引に用いる。取引中はエントリー条件で用いた戦略のエグジット条件のみをエグジット判定に用いる。

このようなエグジット条件に加えて、図3のようなデフォルト条件も用いてリスクを限定する。ここで、 $Wmax$ は利益確定値、 $Lmax$ は損切り値、 $Tmax$ はタイムアウト時間である。また、値動きが連続しない取引時間外でのポジション保持を避けるため、土曜日早朝などのあらかじめ決められた時間以降ではエントリーは行わず、また土曜日朝7時の取引終了時間直前にはエグジット条件を満たしていなくても、エグジット処理を行う。

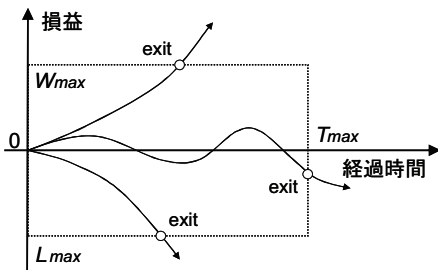


図3 デフォルト・エグジット条件  
Figure 3 Default exit conditions

#### 2.5 適合度評価関数

GAで望ましい収束解を得るためには、淘汰の過程で個体の適合度評価にどのような評価関数を用いるか

が極めて重要である。評価の対象が利益率のみであれば、評価関数は単純である。しかし、為替取引では任意のレバレッジを設定できるため、たとえ1%の負けであっても、レバレッジ次第で10%~100%の損失を被ることになる。このような場合、2~3回連続して負けるだけで資産が半減することもあり得る。そのため、利益率だけでなく、勝率や最大ドローダウンも考慮する必要がある。さらに、少ない取引回数、少ない手数料で利益をあげることを目指す場合には、1取引あたりの利益も考慮に入れる。

利益率と勝率を考慮に入れた最も簡単な評価関数は、利益率  $p$ 、勝数  $w$ 、負数  $l$  とすると、式1で表される。

$$f(p, w, l) = p \frac{w}{w+l} \quad (1)$$

式1では利益率と勝率を線形的に評価しているが、為替取引においては、たとえ利益率が高くても勝率が低い戦略は好ましくない場合がある。このような場合、式2のようなシグモイド関数を用いて、ある値の勝率を境に大きく評価が上がるように誘導する。

$$\zeta_a(x) = \frac{1}{1+e^{-ax}} \quad (2)$$

式3は、シグモイド関数(ゲイン  $a=10$ )を用いて勝率を  $r_{ex}$  以上に誘導したものである。

$$f(p, w, l) = p \zeta_{10} \left( \frac{w}{w+l} - r_{ex} \right) \quad (3)$$

戦略によっては、利益率を式4のように3乗することで、より高い利益率と  $r_{ex}$  以上の勝率を同時に得られる場合もある。

$$f(p, w, l) = p^3 \zeta_{10} \left( \frac{w}{w+l} - r_{ex} \right) \quad (4)$$

1取引において  $p_{ex}$  以上の利益を期待する場合、式5のようなシグモイド関数を乗じることが考えられる。

$$f(p, w, l) = p \zeta_{10} \left( \frac{w}{w+l} - r_{ex} \right) \zeta_{10} \left( \frac{p}{w+l} - p_{ex} \right) \quad (5)$$

さらに、最大ドローダウン  $d_{max}$  が、 $d_{acc}$  以下となるように誘導する場合は、式6のようになる。

$$f(p, w, l, d_{max}) = p \zeta_{10} \left( \frac{w}{w+l} - r_{ex} \right) (1 - \zeta_{10}(d_{max} - d_{acc})) \quad (6)$$

以上のような評価関数は、戦略によって相性の良いもの悪いものがあり、一概にどれが優れているとは言えない。本研究のシステムでは、これらの評価関数を戦略ごとに使い分けている。

## 2.6 交叉率・突然変異率・個体数

GAにおける収束までの探索時間は、交叉率、突然変異率、個体数の設定に大きく左右される。しかしながら、これらの最適値は扱う問題によって異なり、普遍的な最適値は存在しない。

ここでは実際に豪ドル/米ドル、NZドル/円、米ドル/円の分足時系列データ（2008年第1週～第52週）を用いたバックテストで適合度評価を行い、本研究で扱う問題に有効な交叉率、突然変異率、個体数を求めた。

表4は突然変異率を3%に固定して、交叉率を10%から90%まで変化させた場合の収束解および探索時間である。個体数は十分に余裕を持って500とし、適合度評価関数には式3を採用した。交叉方法は1点交叉法、突然変異方法はランダムビットの反転である。

表4 交叉率を変えた場合の探索結果

Table 4 Search results with variety of recombine ratios

交叉率	AUD/USD '08		NZD/JPY '08		USD/JPY '08	
	勝率 利益率	世代 時間	勝率 利益率	世代 時間	勝率 利益率	世代 時間
10 %	90.9% 15.8%	3 47s	77.8% 11.5%	2 43s	75.0% 10.1%	13 60s
20 %	68.9% 29.2%	19 110s	90.0% 17.4%	33 113s	75.9% 11.2%	2 50s
30 %	84.6% 27.6%	35 179s	90.0% 17.4%	19 103s	67.2% 15.5%	45 227s
40 %	84.6% 27.6%	23 130s	90.0% 17.4%	19 187s	67.8% 15.3%	26 199s
50 %	84.6% 27.6%	28 233s	90.0% 17.4%	14 171s	67.2% 14.9%	36 312s
60 %	84.6% 27.6%	23 241s	90.0% 17.4%	17 244s	67.2% 15.5%	28 252s
70 %	84.6% 27.6%	30 385s	90.0% 17.4%	15 208s	66.2% 14.1%	19 206s
80 %	84.6% 27.6%	23 227s	90.0% 17.4%	12 216s	67.2% 15.5%	55 634s
90 %	74.5% 28.9%	25 333s	90.0% 17.4%	15 267s	76.5% 12.1%	16 278s

豪ドル/米ドルでは40%、NZドル/円では50%、米ドル/円では40%の交叉率の時に最も少ない世代で収束した。これらの結果から、40%の交叉率がこの問題に最も適していることが分かる。

次に、交叉率を40%に固定して、突然変異率を0%～5%に変化させた結果が表5である。得られた収束解はほぼ同じものであったが、収束するまでの世代、探索時間は突然変異率によって異なった。ここでは突然変異率が2～3%のときに最も短時間で収束が見られ

たため、突然変異率は3%とした。

最後に、交叉率を40%、突然変異率を3%として、個体数を20～1000まで変化させた結果が表6である。ここでは、個体数が100以上のときにほぼ同一の収束解が得られている。しかしながら、GAの収束までの探索過程は乱数によって左右されるため、個体数を100に設定すると、乱数の発生させ方によっては局所解で終わる可能性がある。そのため、個体数は余裕を持って200とし、探索精度を高めることを優先した。

表5 突然変異率を変えた場合の探索結果

Table 5 Search results with variety of mutation ratios

突然変異率	AUD/USD '08		NZD/JPY '08		USD/JPY '08	
	勝率 利益率	世代 時間	勝率 利益率	世代 時間	勝率 利益率	世代 時間
0 %	84.6% 27.6%	31 206s	90.0% 17.4%	22 163s	67.2% 14.6%	28 195s
1 %	84.6% 27.6%	37 240s	90.0% 17.4%	23 175s	68.0% 13.2%	20 156s
2 %	84.6% 27.6%	26 194s	90.0% 17.4%	18 157s	67.3% 14.5%	41 268s
3 %	84.6% 27.6%	23 130s	90.0% 17.4%	19 187s	67.8% 15.3%	26 199s
4 %	84.6% 27.6%	25 205s	90.0% 17.4%	21 163s	67.3% 13.9%	25 202s
5 %	84.6% 27.6%	31 179s	90.0% 17.4%	20 190s	64.2% 14.2%	30 222s

表6 個体数を変えた場合の探索結果

Table 6 Search results with variety of individual sizes

個体数	AUD/USD '08		NZD/JPY '08		USD/JPY '08	
	勝率 利益率	世代 時間	勝率 利益率	世代 時間	勝率 利益率	世代 時間
20	82.4% 20.6%	17 22s	81.8% 13.9%	30 25s	69.7% 10.6%	15 22s
50	75.0% 25.9%	42 38s	73.3% 17.9%	9 29s	66.7% 12.7%	14 28s
100	84.6% 27.6%	30 52s	90.0% 17.4%	18 44s	76.5% 11.3%	19 49s
200	84.6% 27.6%	37 90s	90.0% 17.4%	35 106s	76.5% 12.1%	24 69s
500	84.6% 27.6%	23 130s	90.0% 17.4%	19 187s	67.8% 15.3%	26 199s
1000	84.6% 27.6%	25 349s	90.0% 17.4%	14 223s	67.2% 15.5%	32 470s

## 3. 高速化

### 3.1 重複染色体の排除

GAによる探索の過程では、交叉や突然変異により新たな染色体が生成される。通常、現代に同一の染色体パターンを持つ個体がある場合は、生成された染色体パターンを破棄し、再度、交叉や突然変異により染色体パターンを生成し直すことが一般的である。

しかし、前世代までに淘汰された染色体パターンが



再出現した場合、その染色体は適合度が低く評価に値しないにもかかわらず、再度バックテストにかけられ、高い計算コストを費やして評価されることになる。このような重複染色体の無駄な評価を排除し高速化を図るため、図4のような評価済フラグテーブルを設けた。

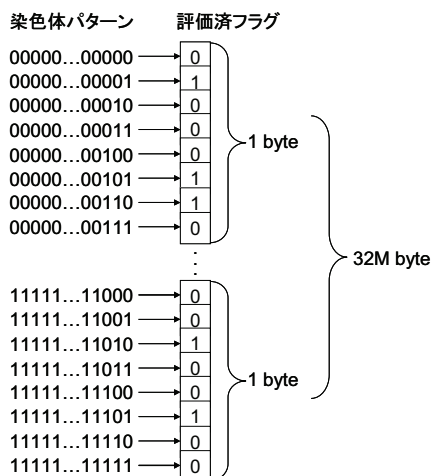


図4 評価済フラグテーブル  
Figure 4 Evaluated flag table

評価済フラグテーブルの各ビットはそれぞれ1種類の染色体パターンに対応する。30ビット分の全てのパターンに対応するためには、32Mバイトの容量の評価済フラグを必要とする。染色体パターンの適合度評価をするたびに、対応するフラグを立てていくことにより、無駄な重複評価を排除し探索効率を高めた。表7は、このような評価済フラグテーブルを用いた場合と、用いなかった場合で、同一の収束解に至るまでの世代数を比較したものである。評価済フラグテーブルを用いることで、1~5世代早く収束している。

表7 評価済フラグテーブルの効果

Table 7 Improvement by evaluated flag table

時系列データ	テーブルなし 収束世代	テーブルあり 収束世代
GBP/CHF '06	34	29
AUD/USD '07	24	21
AUD/USD '08	38	37

### 3.2 事前計算

GAでは、ビット列で表された遺伝子からデコードされた有限の離散的な値を選択肢として探索を行うため、他の探索アルゴリズムと比べて探索精度は悪い。しかし、一方で有限の離散的な値のみを選択肢とする

ことがあらかじめ分かっているため、様々な事前計算による高速化が可能である。

本研究では、個体の適合度評価に時系列データを用いたバックテストを行っている。このバックテストのある時刻  $t$  において、それ以前の時系列データから様々なテクニカル指標が計算され、エントリー、エグジットの判断材料として用いられる。

これらのテクニカル指標の計算は、2.3 で述べた期間遺伝子による期間パラメータに依存する。しかし、この期間パラメータは有限の離散的な値であるため、メモリに余裕があれば、あらかじめ全ての期間パラメータに対応するテクニカル指標を事前計算しておき、図5のように時系列データのメタデータとして登録しておくことが可能である。ちなみに、16通りの期間パラメータに対応したテクニカル指標を、短精度浮動小数点値として52週間分計算するためには、23Mバイトのメモリを必要とする。表2の戦略の場合、テクニカル指標は6種類あるため、全て事前計算するためには138Mバイトのメモリが必要である。

このようにして、バックテストによる個体の適合度評価から大部分のテクニカル指標の計算を排除することができた。表8は、このような事前計算を行った場合と、行わなかった場合の、26世代にわたる探索プロセスにおけるプロファイリング結果である。対象とする足時系列データには豪ドル/米ドル(2009年第1週~第11週)を用いた。コードは64ビット版 Fedora 8 Linux 上で、gcc 4.1.2 によってコンパイルした。コンパイルオプションは `-O3 -pg` である。プロファイラには `gprof 2.17.50` を用いた。

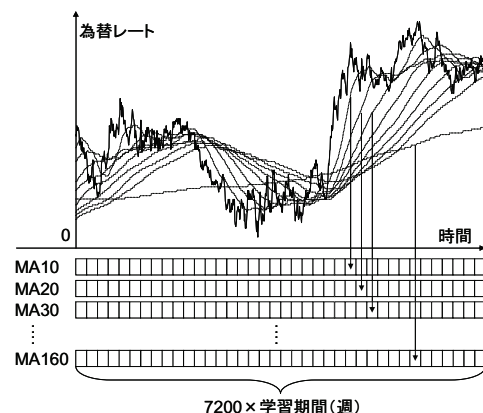


図5 事前計算例

Figure 5 An example of pre-calculation

表 8 プロファイリング結果

Table 8 Profiling result		
ジョブ	事前計算なし	事前計算あり
事前計算	-	3.28s
更新処理	10.90s	10.69s
指標計算	454.47s	4.36s
エントリー判定	1.68s	2.12s
エグジット判定	0.21s	0.19s
その他	52.20s	6.85s
合計	519.46s	27.52s

この結果から、3 秒程度の事前計算を行うことで、それまで 454 秒かかっていたテクニカル指標の計算が 100 分の 1 以下に高速化されていることが分かる。全所要時間でみても 19 倍弱の高速化が図られた。

### 3.3 並列化

GA の探索において、同一世代の各個体の適合度評価は並列に行うことができる。本研究の実装では時系列データを用いたバックテストによって適合度評価を行うため、探索プロセスのワークロード中に占める適合度評価のワークロードの割合が最も高い。

本研究では、個体の適合度評価計算を、POSIX スレッドによるマスターワーカー型並列処理で実装し、1 世代の各個体の適合度評価を任意のワーカーズレッド数で並列に計算できるようにした。図 6 は前節で述べた事前計算を行わない実装を複数のワーカーズレッドで並列処理した場合の実行時間である。実行環境には、3.0GHz の Intel Xeon 5160 (デュアルコア CPU) を搭載した 2-Way の SMP 環境を使用した。Hyper-Threading はサポートされていないため、ハードウェア的に並列実行できるスレッドは 4 つである。

図 6 では、4 ワーカーズレッドまでほぼ 2 倍、3 倍、4 倍とリニアに性能が向上していることが分かる。通常、並列処理では入出力データの読み書きなどのオーバーヘッドにより、並列度を上げててもリニアに性能が上がらない場合が多い。しかし、この実装では入力データはスレッド起動時に読み込まれスレッド終了まで保持される時系列データ (465K バイト)、出力データは適合度評価値 (4 バイト) のみである。並列処理に向けた典型的な密問題であり、分散メモリ型のクラスター・システムを用いても同様のリニアな性能向上が期待できる。

次に、前節の事前計算をマスタースレッドで行った後で、バックテストによる適合度評価を複数のワーカーズレッドで並列処理した場合の実行時間を図 7 に示す。この実装では、テクニカル指標の計算などのワークロードと比べて、毎分の更新処理などメモリ操作の

ワークロードが大きく、並列度が上がっても前実装のような 2 倍、3 倍といった性能向上は見られない。それでも 4 スレッドまでは性能向上が見られ、4 スレッド時にこれまでで最短の 13 秒弱で探索を終えることができた。この実装では、大容量の事前計算済メタデータへアクセスする必要があるため、分散メモリ環境には不向きであり、SMP 向けの実装であると言える。

以上のような並列化により、1 通貨ペアあたりの全ての戦略 (3~4 種類) を対象とする探索を、分足時系列データの更新時に 1 分以内に終わることを実現した。

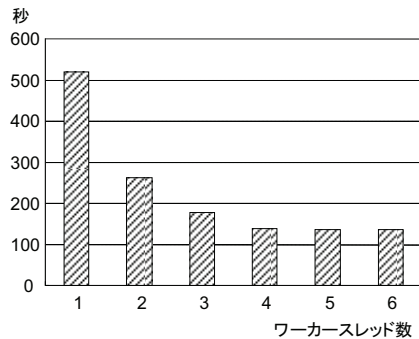


図 6 実行時間 (事前計算なし)

Figure 6 Execution time (with no pre-calculation)

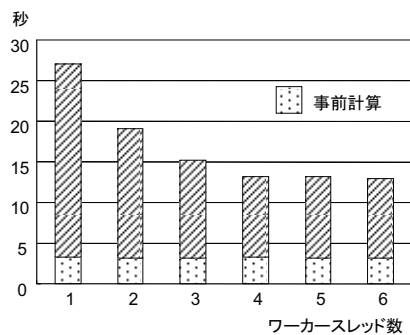


図 7 実行時間 (事前計算あり)

Figure 7 Execution time (with pre-calculation)

## 4. おわりに

本稿では、実時間での GA 探索により、自動取引に用いるパラメータを更新するシステムの実装について述べた。金融時系列データとして、値動きが連続する期間が長く、かつ流動性が高い為替取引の分足時系列データを対象とした。

各個体の染色体には期間遺伝子と条件遺伝子を持たせ、それぞれテクニカル指標の計算、エントリー判定・

エグジット判定の閾値に用いた。このような情報を持つ個体によって実際に分足時系列データ上でバックテストを行い、利益率、勝率、最大ドローダウンなどから式1~式6により適合度を求めた。取引戦略には主に順張り戦略を用いて、条件遺伝子によるエントリー判定、エグジット判定に加えて、デフォルト・エグジット条件を設定してリスクを限定した。交叉率、突然変異率、個体数の決定にあたっては、実データを用いたバックテストによる適合度評価を行い、最短時間で収束できる最適値を求めた。

このようにして実装したGAによる探索を、分足時系列データの更新時に1分以内に終わるために、様々な高速化を試みた。まず、交叉や突然変異によって生成される染色体から既に評価済みの染色体を排除するために評価済フラグテーブルを実装した。次に、GAで選択肢として用いる有限の離散的な値それぞれに対応するテクニカル指標をあらかじめ事前計算し、これを時系列データのメタデータとすることで、個体の適合度評価を高速化した。最後にPOSIXスレッドを用いてマスターワーカー型並列処理を実装し、1世代の各個体の評価を複数のワーカースレッドで並列に計算できるようにした。以上のような高速化により、1戦略あたりの探索を13秒弱で終え、1分以内に3~4種類の戦略のパラメータを更新することに成功した。

今後、さらに細かい粒度の金融時系列データを対象とした探索を実時間で行うために、現在のシステムの並列度をさらに高める必要がある。SMPで実装する場合、最新のNehalem-EP Xeonを用いた2-Way SMPシステムでは、ハードウェア的に16スレッドを並列実行でき、現在のシステムの2~3倍の処理能力が期待できる。しかし、10秒ごとに更新されるティックデータに対応するためには、単純計算で現在の36倍以上の処理能力が必要となる。そのためには、SMPの並列度を上げるだけでは限界があり、適合度評価対象の個体数以上のノードを持った分散メモリ型クラスターシステムや、GPGPUなどにより、1世代分の全ての個体の適合度評価を同時に計算する方法が考えられる。

最後に、本稿で提案した3つのGA高速化手法は、金融時系列データ処理に限らず、個体の適合度評価の負荷が高い探索問題に対して、一般的に有効な手法である。このようなGAを高速化する手法の研究が、様々な分野での実時間探索に応用されることを期待したい。

## 参考文献

- 1) FXCM Forex System Selector:  
<http://www.fxcm.com/forex-system-selector.jsp>
- 2) クリック証券 Web service:  
<https://www.click-sec.com/corp/guide/kabu/web-service/>
- 3) Jingtao Yao, Chew Lim Tan: "A case study on using neural networks to perform technical forecasting of forex," *Neurocomputing*, Vol. 34, pp. 79-98, 2000
- 4) M.A.H. Dempster et al: "Computational learning techniques for intraday FX trading using popular technical indicators," *IEEE Transactions on neural networks*, Vol. 12, Issue 4, pp. 744-754, 2001
- 5) M.A.H. Dempster and C.M. Jones: "A real-time adaptive trading system using genetic programming," *Quantitative Finance*, Vol. 1, pp. 397-413, 2001
- 6) T.Lux and S.Schornstein: "Genetic learning as an explanation of stylized facts of foreign exchange markets," *Journal of Mathematical Economics*, Vol. 41, Issues 1-2, pp. 169-196, 2005
- 7) Pablo Fernández-Blanco et al: "Technical market indicators optimization using evolutionary algorithms," *Proceedings of the 2008 GECCO conference companion on Genetic and evolutionary computation*, pp. 1851-1858, 2008
- 8) 平林明憲, 伊庭斉志: "遺伝的アルゴリズムによる外国為替取引手法の最適化," 人工知能学会, 第1回ファイナンスにおける人工知能応用研究会, 2008
- 9) 平林明憲, 伊庭斉志: "遺伝的アルゴリズムによる外国為替取引手法の最適化(続報)," 人工知能学会, 第2回ファイナンスにおける人工知能応用研究会, 2009