Analysis of Dynamic Properties of Stock Market Trading Experts Optimized with an Evolutionary Algorithm

Krzysztof Michalak

Department of Information Technologies, Institute of Business Informatics, Wroclaw University of Economics, Wroclaw, Poland

krzysztof.michalak@ue.wroc.pl
Presentation Plan

- Trading rules
- Trading experts
- Dynamic evolutionary optimization
- Analysis of Trading Rule Sets
- Experiments
- Results
- Conclusions & Further Work
Trading Rules (1/2)

- Based on
  - stock price
  - volume
  - technical indicators
    - Chaikin Oscillator
    - Ease of Movement Value (EMV)
    - K-Stochastic
    - MACD
    - Moving Averages (MA)
    - Overbought / Oversold
    - Rate of Change (RoC)
    - Relative Strength Index (RSI)
    - Top / Bottom Reversal (TBR)
    - Williams Oscillator
    - ...

3
Trading Rules (2/2)

- Moving average types
  - SMA
  - EMA

- Parameters
  - period
  - threshold

- Optimization of trading rules parameters
  - separate run of a GA
  - specimen evaluation: average return on 50 stocks
  - intra-day trading
  - commission: 0.4%
Algorithm 1 An example of a trading rule based on two moving averages.

IN:
\[ \tau_{\text{fast}} = 10 \] - the period of the fast moving average
\[ \tau_{\text{slow}} = 80 \] - the period of the slow moving average
\[ t \] - the time instant for which to generate the decision

OUT:
A decision for the time instant \( t \)

\[
\text{if } MA_{\tau_{\text{fast}}}(t) < MA_{\tau_{\text{slow}}}(t) \text{ then}
\]
\[
\quad \text{return } -1 \quad \text{// Sell}
\]
\[
\text{else}
\]
\[
\quad \text{if } MA_{\tau_{\text{fast}}}(t) > MA_{\tau_{\text{slow}}}(t) \text{ then}
\]
\[
\quad \quad \text{return } 1 \quad \text{// Buy}
\]
\[
\quad \text{else}
\]
\[
\quad \quad \text{return } 0 \quad \text{// No suggestion}
\]
\[
\text{end if}
\]
\[
\text{end if}
\]
Stock: Amazon (AMZN)
Date: 2012-06-12

- Black line: stock price
- Blue line: 3-minute EMA
- Pink line: 30-minute EMA
- Green circle: buy signal
- Red circle: sell signal
Trading expert parameters

\[ b_1, \ldots, b_{N_{rules}}, s_1, \ldots, s_{N_{rules}}, \Theta_{buy}, \Theta_{sell} \]

where:

- \( N_{rules} \) - the number of trading rules
- \( b_i \) - a binary variable that determines if the \( i \)-th rule is used for generating „buy” signals
- \( s_i \) - a binary variable that determines if the \( i \)-th rule is used for generating „sell” signals
- \( \Theta_{buy}, \Theta_{sell} \) - decision thresholds for „buy” and „sell” decisions respectively
Trading Experts (2/2)

- Trading expert parameters

\[ b_1, \ldots, b_{N_{rules}}, s_1, \ldots, s_{N_{rules}}, \Theta_{buy}, \Theta_{sell} \]

- Trading decisions

  - applied for each stock separately (no portfolio management)
  - rules generate their „buy” and „sell” signals
  - \( b_i \) and \( s_i \) variables turn the rules „on” and „off”
  - average signal is calculated (for „buy” and „sell” decision separately)
  - decision thresholds are applied
Dynamic optimization
- experts optimized on 8-week periods
- interval $t_k = \{ \text{week}_{k}, \text{week}_{k+1}, \ldots, \text{week}_{k+7} \}$
- goal (at time $k$): find an optimal set of rules for making investments during the $t_k$ interval
- discrete time steps, but...
- the intervals overlap
Dynamic optimization (contd.)
- we allow $N_{gen}$ generations for each time interval $t_k$

Genotype = trading expert parameters

Specimen evaluation
- accumulated return on the interval $t_k$
- intra-day trading simulation
  - sell everything at the end of the day even if no “sell” signal
- commission: 0.4%
Evolutionary Optimization (3/7)

- How to preserve diversity?
  - well-known problem (not only) in dynamic optimization
  - the more the population converges the harder it is to find the new optimum
  - approaches: random immigrants, reinitialization, hypermutation, fitness sharing, subpopulations, ...
How to preserve diversity?

interval $t_i$ \hspace{2cm} interval $t_{i+1}$

$N_{gen}$ generations \hspace{2cm} $N_{gen}$ generations

* = a change in the environment
Evolutionary Optimization (5/7)

- How to preserve diversity?
  - reinitialization

\[ P = \text{InitPopulation}(N_{\text{pop}}) \]

interval \( t_i \) \hspace{1cm} interval \( t_{i+1} \)

\( N_{\text{gen}} \) generations \hspace{1cm} \( N_{\text{gen}} \) generations

\* = a change in the environment
How to preserve diversity?

- random immigrants every interval

\[
R = \text{InitPopulation}(N_{\text{pop}}) \\
P = P \cup R
\]
How to preserve diversity?

- random immigrants every generation

\[ R = \text{InitPopulation}(N_{\text{pop}}) \]
\[ P = P \cup R \]
Analysis of Trading Rule Sets (1/3)

- Data set for pattern discovery

  - a top 20% of specimens present in the last population in the evolutionary algorithm (after $N_{gen}$ generations)

  - $P_k^{(best)} = 20\%$ of specimens selected separately for each stock and each interval $t_k$, $k = 1, \ldots, N_{time}$

  - a matrix $A_{N_{time} \times N_{rules}}$ is created by averaging the $b_i$ (or $s_i$) parameters of the specimens in $P_k^{(best)}$, $k = 1, \ldots, N_{time}$
Clustering of the $A_{N_{time} \times N_{rules}}$ matrix

- agglomerative hierarchical clustering
- grouped sets of columns stored in a tree
Clustering of the $A_{N_{time} \times N_{rules}}$ matrix

ordering of columns changed recursively

- if $d(C'.first, C''.first)$ is the smallest, $C'$ is reversed,
- if $d(C'.last, C''.last)$ is the smallest, $C''$ is reversed,
- if $d(C'.first, C''.last)$ is the smallest, both $C'$ and $C''$ are reversed,
- if $d(C'.last, C''.first)$ is the smallest, no reordering is performed,

result:

- similar columns merged into clusters
- similar columns from adjacent clusters close to each other
Experiments (1/2)

- High-frequency data from the NYSE
- Date range 2011.10.17 to 2013.05.20
- Minute quotations
- 50 stocks
- 76 weeks
- 82 rules
Experiments (2/2)

- Evolutionary algorithm details
  - single-objective optimization (objective = return)
  - roulette-wheel selection
  - mutation:
    - bit-flip ($P_{mut} = 0.02$) for bi and si parameters
    - polynomial (distribution index $\eta = 20$)
  - crossover: single-point ($P_{cross} = 0.9$)
  - $N_{gen} = 30$, $N_{pop} = 50$
Best method: random immigrants every generation

stock: AAPL
Results after 30 generations
Results after 30 generations
Results (4/5)

- Results of the clustering

- stock: AAPL „buy” signals

Random immigrants every generation

Random immigrants every interval

Population reinitialization
Results (5/5)

- Stability of the obtained return

The dependence between the return obtained in interval $t_k$ (x axis) and interval $t_{k+1}$ (y axis). The specimens that had not changed for at least 3 intervals were used.
Conclusions (1/3)

- Dynamic optimization of trading experts was performed
  - random immigrants every interval
  - random immigrants every generation
  - population reinitialization

- A method was proposed for analyzing the usage of rules in trading experts optimized using a dynamic evolutionary algorithm
  - selection of 20% of the best specimens
  - clustering
  - visualization
Conclusions (2/3)

- Dynamic optimization results
  - adding random immigrants every generation worked best
  - useful information can be extracted from trading experts optimized for past time intervals (reinitialization deteriorates the results)
  - the EA can produce good rule sets that are relatively stable - they produce similar return values for consecutive time intervals
Results of rules usage analysis

- there exist prolonged periods of time when the same set of rules produces the best investment returns.

- among the fittest specimens the fraction of rules used at the same time for generating "buy" and "sell" signals is approximately $\frac{1}{4}$

- in order to build good trading experts one should allow using trading rules separately for generating "buy" and "sell" signals
Further Work

- Detection of time intervals during which the optimal set of rules remains the same or very similar

- Using the analysis of the stability of optimal trading rule sets for deciding if a given trading expert should be used for making decision at a given time or not.