Results of the GECCO 2011 *Industrial Challenge: Optimizing Foreign Exchange Trading Strategies*

Oliver Flasch, Thomas Bartz-Beielstein, Daniel Bicker [1], *Wolfgang Kantschik, and Christian von Strachwitz* [2]

[1] Cologne University of Applied Science, 51643 Gummersbach, Germany (email: {oliver.flasch, thomas.bartz-beielstein, daniel.bicker}@fh-koeln.de)

[2] Quaesta Capital GmbH, 60323 Frankfurt, Germany (email: {wk, cvs}@quaestacapital.de)

November 15, 2011

The GECCO 2011 Industrial Challenge posed a difficult real-world problem in financial time series forecasting, provided by Quaesta Capital GmbH. This report gives a problem overview, outlines the rules of the challenge and provides a result summary of the winning submission.

1 Introduction

The foreign exchange (FX) market is financial market for trading currencies to enable international trade and investment. It is the largest and most liquid financial market in the world. [Weithers, 2006]. Market participants are not only financial institutions, but also large industrial corporations. High frequency intraday FX trading is very common and completely automated through algorithmic trading systems. The GECCO 2011 Industrial Challenge focused on the very challenging problem of profitable profitable intraday FX trading based on hourly data.

Currencies can be traded via a wide variety of different financial instruments, ranging from simple spot trades over to highly complex derivatives. In this competition, a simplified model of the FX market was used to define a simple but still realistic test problem, which will be referred to as the *FX problem* in the remainder of this report. Participants had to solve this problem by generating trading signals for a representative set of currency pairs. These trading signals where assessed by a quality measure that takes profitability (incorporating trading costs) and risk into account.

The FX problem belongs to the class of time series forecasting problems [Brockwell and Davis, 2002]. Although many different methods can be used for time series forecasting, Computational Intelligence (CI) methods, such as Evolutionary Computation, are an attractive option. CI methods are often robust to changes in the underlying system, i.e. changes in the market conditions, which can lead to reduced risk and lower maintenance costs. As an example, Genetic Programming (GP) has been successfully applied to the FX problem, both in research and in industry [Dempster and Romahi, 2002, Wilson and Banzhaf, 2010, 2009, Bhattacharyya et al., 2002, Austin et al., 2004].

These findings let CI-based systems appear as as an interesting alternative to the classical time series analysis methods more widely applied in quantitative finance, and motivated this competition [Hamilton, 1994]. Highlights of the GECCO 2011 Industrial Challenge include:

- *Commercially Relevant Task:* The GECCO Industrial Challenge is based on a commercially relevant real-world problem provided by an industry partner.
- *Complex Problem Domain:* The FX market, with its complex patterns and behavior, offers a fascinating test case for innovative optimization methods.
- *Real-world Data:* Multiple real intraday FX return data sets are provided for training and testing trading strategies.
- *Realistic Quality Measurement:* Trading strategies are scored using a simple but realistic trading simulator that takes trading costs and risk into account.

The remainder of this report is organized in four sections: Section 2 introduces the problem of algorithmic trading in the FX market, its related objective function, and the training and test data sets used in the challenge. Section 3 outlines the rules for ranking submissions. Section 4 gives a summary of the submission results and sets these results in relation to the baseline results provided by Quaesta Capital GmbH. Finally, Section 5 draws conclusions and points to the upcoming GECCO 2012 Industrial Challenge.

2 Problem Description

The objective of the competition was find profitable *trading strategies* for the FX market. A trading strategy f is a function generating a *trading signal* S based on a *FX return time series* R:

$$f(R) \to S \tag{1}$$

A FX return time series R is a time series of absolute returns, i.e. absolute exchange rate changes, for a *currency pair*. The first currency in a currency pair is known as the *base currency*, the second currency is called the *counter currency*. A single data point $R(t)$ in a FX return time series R shows the absolute value change of the counter currency in relation to the base currency, that occurred in the time interval represented by the data point at t. For example, the data point "`2010-03-31 10:00:00 0.0016`" of the hourly FX return time series "EURUSD" encodes that, in the time interval from `10:00:00` to `10:59:59`, the base currency (EUR) has increased in relative value by 0.0016 USD.

A trading signal is a time series of Long (represented as the integer 1), Short (-1), or Flat (0) signals that defines, for each point in time, whether to allocate a fixed size amount of money to the base currency (Long position in the base currency), to allocate that amount to the counter currency (Short position in the base currency), or to stay out of market (Flat signal). Short positions generate profits when the base currency loses in relative value, while

Long positions generate profits when the base currency gains in relative value. The quality of a trading signal is assessed by simulated trading and analysis of the generated returns, as described in the next subsection and shown in Listing 1.

2.1 *Objective Function*

Assessing the quality of a trading signal S for a given training or test FX return time series R is a two-step process:

1. The trading signal return time series $T_{(R,S)}$, i.e. the cost-corrected profits and losses, generated by trading the currency pair of R based on the trading signal S, is calculated.

2. The quality of S is then given by the simplified *Sharpe Ratio*[3] of $T_{(R,S)}$.

[3] Our definition of the Sharpe Ratio is simplified as it does not regard risk-free returns.

Step 1 is implemented by the *trading simulator* shown in Listing 1. The trading simulator calculates the trading signal return time series $T_{(R,S)}$ one step at a time by keeping track of the current position P. At each time step i, the next state, i.e. the next value of P, solely depends on the current state, the fixed trading cost t_c, and on the ith values of the input time series R and S. In Flat position, the trading signal return (output to $T_{(R,S)}$) is always zero. In Long position, the trading signal return equals R. In short Short position, the trading signal return equals the additive inverse of R. When a non-Flat position is exited, i.e. when P changes from Long to Flat, from Long to Short, from Short to Flat, or from Short to Long, the trading cost t_c is subtracted from the current trading signal return value. Trading costs depend on the currency pair traded and are given in Table 2 in Section 2.2. See Listing 1 for a complete definition of the trading simulator outlined above.

BASED on the signal's trading return time series $T_{(R,S)}$ calculated in step 1, in step 2 the signal quality is obtained by calculating the simplified Sharpe Ratio as defined in Equation 2, where *SD* is the standard deviation and *mean* the arithmetic mean:

$$
\begin{aligned}
\text{SignalQuality}(R, S, t_c) &:= \text{SharpeRatio}[\text{SimulateTrading}(R, S, t_c)] \\
&= \frac{\text{mean}(\text{SimulateTrading}(R, S, t_c))}{\text{SD}(\text{SimulateTrading}(R, S, t_c))}
\end{aligned}
\tag{2}
$$

The *score* of a submission is defined as the sum of the signal qualities over the three test data sets given in Section 2.2. A high-level overview of the trading strategy generation and assessment process is given in Figure 2. Implementations of the complete objective function including both steps are available in R and Java.

```
SimulateTrading ← function(R, S, t_c) {
  l ← length(R)
  T_(R,S) ← numeric(l) # initialize T_(R,S) with l zeros
  P ← S[1] # current position

  for (i in 2:l) { # for i from 2 to l...
    # If in FLAT position...
    if (P == 0 && S[i] == 0) {
      # In FLAT position, next signal is FLAT...
      # (no state change)
    } else if (P == 0 && S[i] == 1) {
      # In FLAT position, next signal is LONG...
      P ← 1 # Go LONG.
    } else if (P == 0 && S[i] == −1) {
      # In FLAT position, next signal is SHORT...
      P ← −1 # Go SHORT.
    # If in LONG position...
    } else if (P == 1 && S[i] == 1) {
      # In LONG position, next signal is LONG...
      T_(R,S)[i] ← R[i] # Record return.
    } else if (P == 1 && S[i] == 0) {
      # In LONG position, next signal is FLAT...
      T_(R,S)[i] ← R[i] − t_c # Record return minus costs.
      P ← 0 # Exit position.
    } else if (P == 1 && S[i] == −1) {
      # In LONG position, next signal is SHORT...
      T_(R,S)[i] ← R[i] − t_c # Record return minus costs.
      P ← −1 # Go SHORT.
    # If in SHORT position...
    } else if (P == −1 && S[i] == −1) {
      # In SHORT position, next signal is SHORT...
      T_(R,S)[i] ← −R[i] # Record return.
    } else if (P == −1 && S[i] == 0) {
      # In SHORT position, next signal is FLAT...
      T_(R,S)[i] ← −R[i] − t_c # Record return minus costs.
      P ← 0 # Exit position.
    } else if (P == −1 && S[i] == 1) {
      # In SHORT position, next signal is LONG...
      T_(R,S)[i] ← −R[i] − t_c # Record return minus costs.
      P ← 1 # Go LONG.
    }
  }
  return(T_(R,S))
}
```

Listing 1: The trading simulator used to calculate the trading signal return time series $T_{(R,S)}$ for a trading signal S and a FX return time series R, incorporating trading costs t_c.

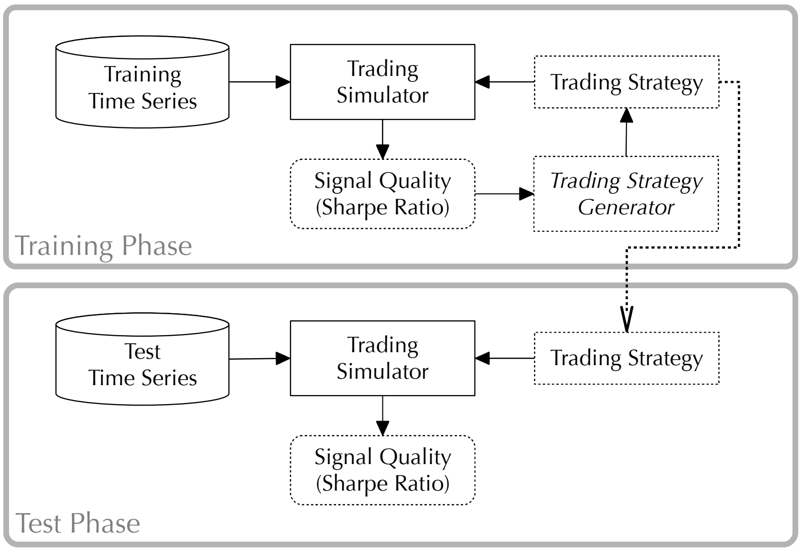

Figure 2: The trading strategy generation and assessment process. In the training phase, a trading strategy for a currency pair is generated by the participant's trading strategy generator. During training, the quality of the trading strategy is assessed by simulated trading on the supplied training data set. In the test phase, the trading strategy is continuously evaluated by simulated trading on the supplied test data set. In this diagram, components that have to be supplied by participants are drawn with a thin dashed outline, while components supplied by the organizers are drawn with a solid outline.

2.2 *Training- and Test-Datasets*

Training- and test time series data for the FX problem consist of 3900 data records from three different currency pairs. For the first currency pair (AUDUSD), data of the year 2005 is provided, while 2010 data is provided for the remaining two currency pairs (EURUSD and GBPUSD).[4] As the score of a trading strategy is obtained by taking the sum of the signal qualities over the three test data sets, signal strategies that are robust to a wide variety of market conditions have an advantage. The data sets with their respective time intervals and sizes are shown in Table 1.

[4] The EURUSD and GBPUSD FX return time series are based on the same time intervals to enable trading strategies that exploit possible lagged correlations of these data sets.

	Start Date	End Date	# Records
AUDUSD Training	2005-04-01 00:00	2005-06-09 23:00	1188
AUDUSD Test	2005-06-10 00:00	2005-06-30 23:00	357
EURUSD Training	2010-04-01 00:00	2010-06-10 23:00	1185
EURUSD Test	2010-06-11 00:00	2010-06-30 23:00	324
GBPUSD Training	2010-04-01 00:00	2010-06-10 23:00	1185
GBPUSD Test	2010-06-11 00:00	2010-06-30 23:00	324

Table 1: The training and test data sets with their respective time intervals and sizes used in the competition. EURUSD and GBPUSD data are given for the same time intervals to enable trading strategies to exploit possible correlations.

Trading costs as defined in Section 2 are given in Table 2. These costs, given in counter currency units, apply each time a trade is exited (see Section 2). Figure 3 shows a plot of the accumulated FX return time series AUDUSD, to give an idea of the data. Note the individual returns have been accumulated in this plot. All data sets are made available in CSV format.

	Trading Cost (USD)
AUDUSD	0.0002
EURUSD	0.00015
GBPUSD	0.0003

Table 2: Trading cost (given in counter currency units) for each currency pair.

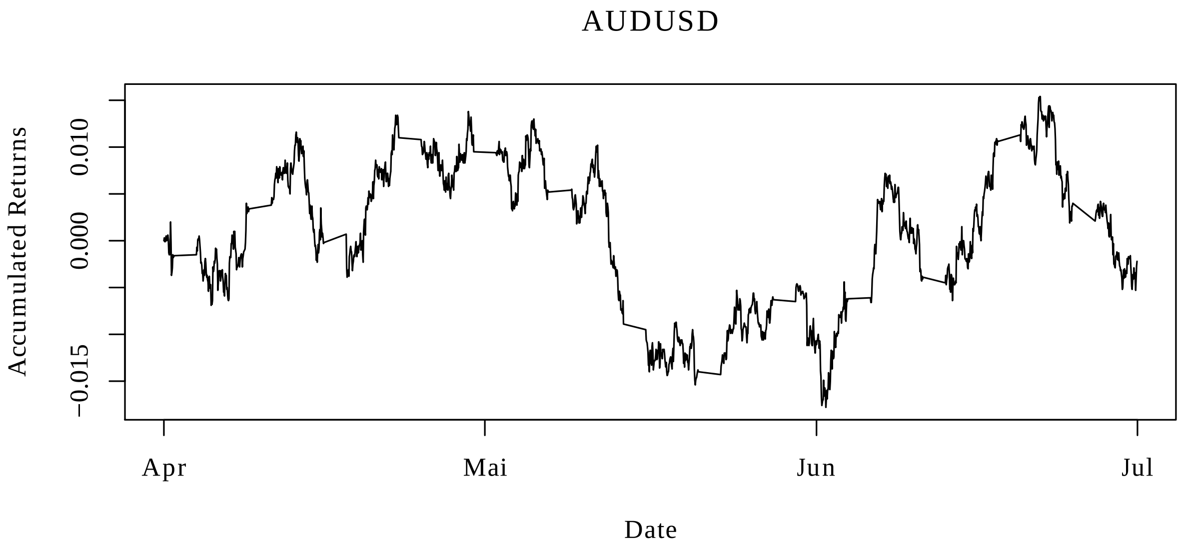

Figure 3: The accumulated FX return data set AUDUSD. Shown are both training and test data ranges.

3 *Rules and Ranking*

The challenge was organized in two rounds: In the first round, participants submitted their score as defined in Section 2.1. These scores are based on the three test data sets given in Section 2.2. They also submitted their executable trading strategy generator. Submissions where ranked by score. In the second and final round, the ten best submissions where be ranked by their scores computed on training and test data sets collected after the submission date, as shown in Table 3.

	Start Date	End Date	# Records
Training	2011-04-01 00:00	2011-06-09 23:00	1155
Test	2011-06-10 00:00	2011-06-29 23:00	322

Table 3: The training and test data set time intervals and sizes used in the second and final round of the competition. All currency pairs (AUDUSD, EURUSD, and GBPUSD where sampled in the same time interval and had the same number of records.

A submission's overall runtime for training per currency pair must not exceed 4 hours on a modern compute server. The compute servers used in the final round where equipped with Intel Xeon E5540 CPUs (2.53 GHz) and 4 GB RAM.

Implementations of the objective function in both Java and R, accompanying tools for result analysis, example trading signals, and the training and test data sets will be kept available for download at `http://gociop.de/gecco-2011-industrial-challenge/`.

THE winner of the GECCO 2011 Industrial Challenge is the participant with the highest score in the second round.

4 Results

The organizers of the GECCO 2011 Industrial Challenge received multiple submissions in the first round. Two submissions generated reproducible results of high enough quality to qualify for the second round. Table 4 shows the final scores and ranking of the submissions. These results are also summarized graphically, for both competition rounds,in Figure 4. Table 5 shows the profit and loss generated by the submissions. To put these results into perspective, a baseline buy and hold and a simple proprietary strategy provided by Quaesta Capital GmbH (Simple Proprietary) are shown as baselines.

Rank	Submission	Author	Score	
			Round 1	Round 2
1	Dynamic Linear Trading	Qinyuan Hong	0.2243	-0.0080
2	CMA-ES Tuned Automatic FX Trading	Zhi Yuan	0.0876	-0.0293
3	Buy and Hold	(baseline)	0.0050	-0.0387
4	Simple Proprietary	(baseline)		-0.0690

Table 4: Sharpe ratios (scores) of GECCO 2011 Industrial Challenge submissions compared with baseline strategies.

THE buy and hold baseline strategy is realized by a trading strategy that generates a constant Long signal. In this strategy, the portfolio returns directly follow the base currency returns. Trading costs are minimized, as no signal changes occur. Simple Proprietary is a typical trend following strategy used as a building block of the much larger strategy portfolio of Quaesta Capital GmbH. The behavior of this strategy should be similar to other simple strategies available in commercial trading systems.

Rank	Submission	Author	PnL	
			Round 1	Round 2
1	Dynamic Linear Trading	Qinyuan Hong	0.1029	-0.0033
2	CMA-ES Tuned Automatic FX Trading	Zhi Yuan	0.0455	-0.0157
3	Buy and Hold	(baseline)	0.0056	-0.0210
4	Simple Proprietary	(baseline)		-0.0411

Table 5: Profit and loss (PnL) of GECCO 2011 Industrial Challenge submissions compared with baseline strategies.

THE winning submission, "Dynamic Linear Trading" by Qinyuan Hong of TU Dortmund University, used an advanced trend following strategy incorporating time series embedding preprocessing, prediction based on generalized linear models, and algorithm parameter tuning based on grid search. The submission ranked second, "CMA-ES Tuned Automatic FX Trading" by Zhi Yuan of Université Libre de Bruxelles, evolved trading strategies based on three standard technical indicators [5] via the well-known Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES).

[5] Moving Average (MA), Relative Strength Index (RSI), and Bollinger Bands (BB)

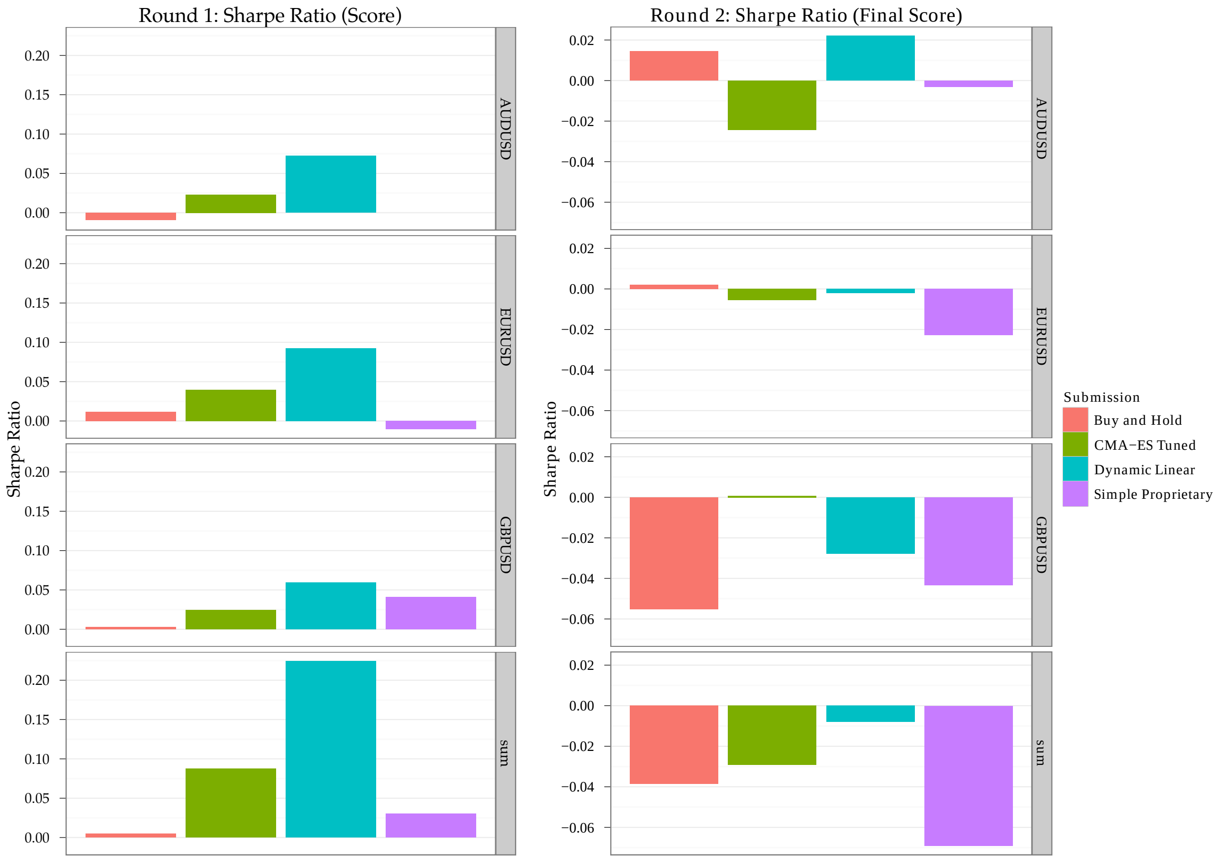

Figure 4: Summary of the Sharpe ratios (scores) of the submitted strategies in relation with the Buy and Hold and Simple Proprietary baselines. Results of the first round are shown on the left, results of the second round on the right. The ranking is based on the sum of the second round scores.

Figure 5 shows the trading behavior of the submitted trading strategies, as well as the behavior of the baselines, on the AUDUSD test data set of the second round. Both submissions outperformed the baseline strategies by a significant margin on the test data sets of both competition rounds.

5 *Conclusions and Outlook*

An important goal of the GECCO 2011 Industrial Challenge was to shed light on the question whether it is possible to generate profitable algorithmic trading strategies based on FX return data alone. While both submissions that qualified for the second round significantly outperformed the baseline strategies in both rounds, the results also indicate that the profitability of trading strategies

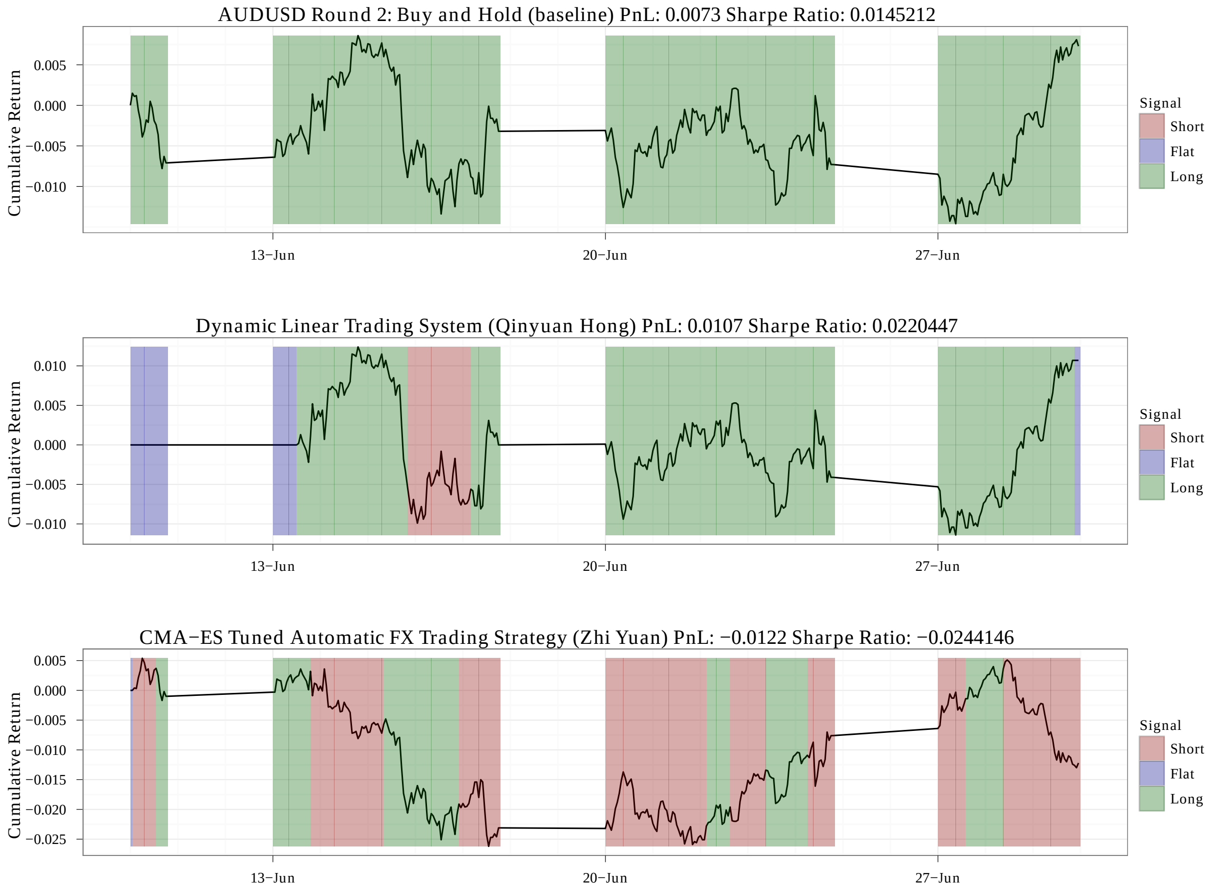

Figure 5: Trading signals and return generated by submitted strategies compared with the Buy and Hold baseline strategy, on the AUDUSD data set of the second (i.e. final) round. Short signals are shown in red, Flat signals in blue, and Long signals in green. Returns are shown in units of counter currency.

is highly dependent on the given market environment. Successful strategies must adapt as the market environment changes. Quaesta Capital GmbH is very interested in the results obtained.

The Industrial Challenge will continue at GECCO 2012, again featuring a highly relevant industry problem provided by an industry partner. More information and the official call for participation will soon be available at `http://gociop.de` and through the official GECCO 2012 channels.

5.1 *Organizing Committee*

- Thomas Bartz-Beielstein, Cologne University of Applied Sciences
- Oliver Flasch, Cologne University of Applied Sciences
- Wolfgang Kantschik, DIP Dortmund Intelligence Project GmbH
- Christian von Strachwitz, Quaesta Capital GmbH
- Wolfgang Konen, Cologne University of Applied Sciences
- Pier Luca Lanzi, Politecnico di Milano
- Jorn Mehnen, Cranfield University

References

M.P. Austin, G. Bates, M.A.H. Dempster, V. Leemans, and S.N. Williams. Adaptive systems for foreign exchange trading. *Quantitative Finance*, 4(4):37–45, August 2004. ISSN 1469-7688.

S. Bhattacharyya, O.V. Pictet, and G. Zumbach. Knowledge-intensive genetic discovery in foreign exchange markets. *IEEE Transactions on Evolutionary Computation*, 6(2):169–181, April 2002.

P.J. Brockwell and R.A. Davis. *Introduction to Time Series and Forecasting*. Springer, New York NY, 2002. ISBN 0387953515.

M.A.H. Dempster and Y.S. Romahi. Intraday FX trading: An evolutionary reinforcement learning approach. In H. Yin, N.M. Allinson, R.T. Freeman, J.A. Keane, and S.J. Hubbard, editors, *Proceedings of Third International Conference on Intelligent Data Engineering and Automated Learning - IDEAL 2002*, volume 2412 of *Lecture Notes in Computer Science*, pages 347–358, Manchester, 2002. Springer. ISBN 3540440259.

J.D. Hamilton. *Time Series Analysis*. Princeton University Press, 1 edition, January 1994. ISBN 0691042896.

T. Weithers. *Foreign Exchange: A Practical Guide to the FX Markets*. Wiley Finance. John Wiley & Sons, Inc., 2006. ISBN 9780470043363.

G. Wilson and W. Banzhaf. Soft memory for stock market analysis using linear and developmental genetic programming. In *Genetic and Evolutionary Computation Conference GECCO '09 (Proceedings)*, pages 1633–1640, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-325-9.

G. Wilson and W. Banzhaf. Interday foreign exchange trading using linear genetic programming. In *Genetic and Evolutionary Computation Conference GECCO '10 (Proceedings)*, pages 1139–1146, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0072-5.