Carry Trade Portfolio Optimization using Particle Swarm Optimization

Stuart G. Reid Department of Computer Science University of Pretoria South Africa Email: stuartgordonreid@gmail.com Katherine M. Malan Department of Computer Science University of Pretoria South Africa Email: kmalan@cs.up.ac.za Andries P. Engelbrecht Department of Computer Science University of Pretoria South Africa Email: engel@cs.up.ac.za

Abstract-Portfolio optimization has as its objective to find optimal portfolios, which apportion capital between their constituent assets such that the portfolio's risk adjusted return is maximized. Portfolio optimization becomes more complex as constraints are imposed, multiple sources of return are included, and alternative measures of risk are used. Meta-heuristic portfolio optimization can be used as an alternative to deterministic approaches under increased complexity conditions. This paper uses a particle swarm optimization (PSO) algorithm to optimize a diversified portfolio of carry trades. In a carry trade, investors profit by borrowing low interest rate currencies and lending high interest rate currencies, thereby generating return through the interest rate differential. However, carry trades are risky because of their exposure to foreign exchange losses. Previous studies showed that diversification does significantly mitigate this risk. This paper goes one step further and shows that meta-heuristic portfolio optimization can further improve the risk adjusted returns of diversified carry trade portfolios.

I. INTRODUCTION

Harry Markowitz famously established the basis of portfolio optimization in his seminal article, Portfolio Selection, in 1952 [1]. Subsequently, portfolio optimization has become an activity central to portfolio management. Whilst Markowitz' concept remains in tact, state-of-the-art techniques are required to solve more realistic versions of the classical portfolio optimization problem. These re-envisaged problems incorporate real world constraints such as taxation, transaction costs, market impacts, and liquidity risk. Various computational intelligence paradigms have been used to solve realistic portfolio optimization problems for different asset classes where deterministic methods struggle [2]. Relatively few studies have used swarm intelligence algorithms [3] and those which have, either used the particle swarm optimization (PSO) algorithm [4][5][6][7] or the pareto ant colony optimization algorithm (ACO) [8]. These two swarm intelligence algorithms have been applied to portfolios consisting of shares; however, neither have been applied to portfolios consisting of currencies such as the diversified carry trade portfolio.

Currency carry trades are speculative bets made by investors who borrow a funding currency at a relatively low interest rate and lend an investment currency at a relatively high interest rate [9]. The difference in interest rates between the funding and the investment currency is called the interest rate differential. Carry trades work on the assumption that payments generated by the investment currency will exceed the costs of the funding currency. However, carry trades are

exposed to foreign exchange gains or losses between the two currencies. According to the uncovered interest rate parity condition (UIP), interest rate differentials correspond to expected changes in exchange rates. Therefore, carry trades should return a zero profit because the returns generated through the interest rate differential should be offset by foreign exchange losses. However, evidence of long term carry trade profitability indicates the failure of UIP. This puzzle has been the topic of considerable research [9], [10]. Nevertheless, because foreign exchange markets are extremely volatile, carry trades are risky and have been described as "picking up pennies in front of an oncoming truck". Techniques aimed at mitigating this risk include hedging and diversification [11]. Simple diversification strategies have improved the Sharpe ratio of currency carry trade portfolios by up to 50% [11].

The remainder of this paper describes how the PSO algorithm can be used to optimize diversified carry trade portfolios. Section II defines the portfolio optimization problem in the context of a diversified carry trade portfolio. Section III specifies a set of benchmark portfolios for comparative purposes. Section IV details the approach taken to portfolio optimization and constraint satisfaction. Section V describes the experiments conducted in this study. Section VI presents the results obtained through those experiments and lastly, Section VII concludes and suggests future research topics.

II. PORTFOLIO OPTIMIZATION

A carry trade portfolio, P, consists of a set of n carry trades, C. Each carry trade, c_j , borrows a low interest rate currency called the funding currency and lends a high interest rate currency called an investment currency. A weight, w_j , is associated with each carry trade, c_j , and represents the percentage of the portfolio's capital which is allocated to that carry trade. P is represented as a set of ordered pairs,

$$P = \{(c_j, w_j)\}, \forall c_j \in C$$

The optimality of P with respect to an objective function, f(P), depends on the weights assigned to each carry trade. Portfolio optimization is a process which attempts to find an optimal set of weights such that f(P) is either maximized or minimized. That is,

optimize f(P) subject to a set of constraints, Z

where $f(P) : \Re^n \to \Re$ and f(P) is the Sharpe ratio [4]. The Sharpe ratio was introduced in 1966 by William Sharpe [12] and is a reward to variability ratio. The Sharpe ratio is commonly used in industry as a measure of risk adjusted returns, and is computed as

$$f(P) = \frac{E[R_P] - R_{rf}}{\sigma(E[R_P])} \tag{1}$$

where $E[R_P]$ is a forecast of expected returns for portfolio P, $\sigma(E[R_P])$ is the standard deviation of expected returns, and R_{rf} is the return generated by a risk free portfolio. In this research study the risk free rate was equivalent to the interest rate of the funding currency. This was chosen because if the portfolio's capital were fully invested in the funding currency, there would be no foreign exchange risk.

The expected return of a carry trade portfolio is the weighted sum of each carry trade's expected return, $E[R_{c_j}]$, minus a penalty function, g(P), representing transaction costs, that is

$$E[R_P] = \left(\sum_{j=1}^n w_j \cdot E[R_{c_j}]\right) - g(P) \tag{2}$$

where g(P) is a function of the absolute weight change of portfolio, P, from time step t to t + 1. Weight changes are realized through trades which carry transaction costs. The penalty, g(P), assumes that larger weight changes require larger trades which carry higher transaction costs, that is

$$g(P) = \sum_{j=1}^{n} |w_j(t+1) - w_j(t)| \theta, \ \forall (w_j, c_j) \in P, \quad (3)$$

where θ is a parameter used to scale transaction costs higher or lower. A high value for θ causes portfolios to be more penalized for weight changes. Expected returns, $E[R_{c_j}]$, are made up from the interest accumulated through the interest rate differential and the expected foreign exchange returns, $E[\Delta FX]$, computed as

$$E[R_{c_j}] = \left(\sum_{t=1}^{o} (i_{c_j} - i_F)_t\right) - E[\Delta FX]$$
(4)

where i_{c_j} is the interest rate of the investment currency for carry trade c_j , i_F is the interest rate of the funding currency, and o is the optimization frequency in days. If central bank interest rates are used in the carry trades and not floating rates such as LIBOR (The London Interbank Offered Rate) then the returns generated through the interest rate differential can be forecast reliably. This is because central banks do not change interest rates often. However, foreign exchange rates change frequently and expected foreign exchange gains or losses need to be forecast. In this research study, three methods of forecasting $E[\Delta FX]$ were investigated:

1) **Normally distributed** - expected returns are normally distributed around the mean of historical returns, that is

$$E[\Delta FX] \in \mathcal{N}(\mu, \sigma^2) \tag{5}$$

where μ is the mean historical return and σ is the standard deviation of historical returns.

2) **Brownian motion** - expected returns follow a Wiener process, W_t , with a mean expected return of zero

[13]. Foreign exchange prices tend to follow a random walk [14]; a Wiener process is a continuous stochastic process representing a scaled random walk. That is,

$$E[\Delta FX] = \sum_{t=1}^{n} (W_t) \tag{6}$$

3) **Geometric Brownian motion (GBM)** - expected returns follow a Weiner process subject to long term market drift and daily variances. That is,

$$E[\Delta FX] = \prod_{t=1}^{n} \left(\left(1 + \mu \frac{1}{n} \right) + \left(\sigma * \sqrt{\frac{1}{n}} \right) * W_t \right)$$
(7)

where μ is the expected market drift and σ is the expected return variance.

In addition to the penalty function a set, Z, of three constraints is enforced on each portfolio. Firstly, each weight should be positive. Secondly, 100% of the portfolio's capital should be allocated between the carry trades. Thirdly, a lower bound, w_j^{min} , and an upper bound constraint, w_j^{max} , is placed on each individual carry trade weight. The constraints are defined as

$$\sum_{j=1}^{n} w_j = 1.0, \ \forall (w_j, c_j) \in P,$$
(8)

$$w_i \in [w_i^{min}, w_i^{max}], \ \forall j = 1, 2, ..., n$$
 (9)

where $0 \le w_j^{min} < w_j^{max} \le 1.0$.

Further realistic constraints including cardinality constraints (restricting the number of assets in the portfolio that can have a zero weight), transaction costs, and investor behaviours such as loss aversion have been researched in previous studies [15][16][17][18][19].

III. BENCHMARK PORTFOLIOS

This section defines the benchmark portfolios against which the optimized portfolio, P_{PSO} , is compared. Five weighting techniques, two uninformed and three informed, were selected as benchmarks. An uninformed weighting is defined as a technique which does not utilize the model of expected returns and an informed technique as one which does.

A. Uninformed Weighting Techniques

An equally weighted and a randomly weighted portfolio were used as uninformed benchmarks. The equally weighted portfolio, P_E , assigned an equal weight to each carry trade, that is

$$w_j = \frac{1}{n}, \ \forall (w_j, c_j) \in P_E \tag{10}$$

The randomly weighted portfolio, P_r , assigned a random weight to each carry trade, that is

$$w_j \sim U(0,1), \ \forall (w_j,c_j) \in P_r$$
 (11)

B. Informed Weighting Techniques

For informed weighting techniques, the carry trades in portfolio, P, must be ordered in descending order of expected return, $E[R_{c_j}]$. A scale-weighted and two return-weighted carry trade portfolios were used as informed benchmarks. The scale-weighted portfolio, P_S , assigned an exponentially decaying weight to each carry trade recursively, that is

$$w_j = 0.5^j, \ \forall (w_j, c_j) \in P_S \tag{12}$$

The first return-weighted portfolio, P_R , weighted each carry trade proportionately to its expected return, that is

$$w_j = \frac{1}{E[R_{c_j}]}, \ \forall (w_j, c_j) \in P_R$$
(13)

The second return-weighted portfolio, P_{R_S} , weighted each carry trade proportionately to its scaled expected return, that is

$$w_j = (2 + E[R_{c_j}])^{n-j}, \ \forall (w_j, c_j) \in P_{R_S}$$
 (14)

For all weighting techniques the portfolio was repaired using a set of constraint satisfaction rules described in Section IV. Also, after the weighting technique was applied, the transaction cost penalty, g(P), was calculated and applied.

IV. OPTIMIZED PORTFOLIO

This section describes how an optimal set of weights for a carry trade portfolio, P_{PSO} , can be found using a particle swarm optimization (PSO) algorithm. The section begins by describing the basic PSO and the adaptations applied to it in this study. Furthermore, this section provides a pseudo-code representation of the adapted PSO algorithm and elaborates on the constraint handling approaches used in this study.

A. Basic PSO for portfolio optimization

A PSO is a population based search algorithm originally inspired by the social behaviour of flocking birds [20]. In a PSO, a swarm, S, of m particles is randomly initialized. Each particle in the swarm, \mathbf{x}_i , is encoded as an n-dimensional weight vector representing a candidate solution to the portfolio optimization problem, that is

$$\mathbf{x}_i = (x_{ij}), \ \forall j = 1, 2, ..., n \text{ and } x_{ij} = w_j \text{ for } c_j$$
 (15)

Each particle maintains a reference to its personal best historical weight vector, \mathbf{y}_i . For each iteration of the algorithm, the global best weight vector is found over all of the particles' personal bests, $\hat{\mathbf{y}}$. For each particle, the vector of weights is updated by adding a velocity term, \mathbf{v}_i , to each weight, that is

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \tag{16}$$

where, $\mathbf{x}_i(t)$ is the weight vector associated with the portfolio at time t, and $\mathbf{v}_i(t+1)$ is the velocity vector update required for \mathbf{x}_i at time t + 1, computed as,

$$v_{ij}(t+1) = w.v_{ij}(t) + c_1.r_{1j}(t).[y_{ij}(t) - x_{ij}(t)] + c_2.r_{2j}(t).[\hat{y}_j(t) - x_{ij}(t)]$$
(17)

where w is the inertia weight; c_1 and c_2 are positive acceleration coefficients to scale the contribution from the cognitive component, $r_{1j}(t).[y_{ij}(t) - x_{ij}(t)]$, and the social component, $r_{2j}(t).[\hat{y}_j(t) - x_{ij}(t)]$, respectively; $v_{ij}(t)$ is the j^{th} component of the previous velocity calculated for \mathbf{x}_i ; and lastly $r_{1j}(t)$ and $r_{2j}(t)$ are uniformly distributed random numbers, $r_{1j}(t)$, $r_{2j}(t) \sim U(0, 1)$.

B. PSO Adaptations

PSO performance is affected by the explorationexploitation trade off [21]. Exploration describes a PSO's ability to explore different regions of the search space and exploitation describes the PSO's ability to concentrate the search in a promising region of the search space. The PSO used in this research study was adapted in the following ways:

1) To improve the exploration ability of the swarm, particles that have converged on the global best particle are re-initialized. Kennedy and Eberhart were the first to study re-initialization of particles in the swarm [22]. Convergence was measured using a similarity function, $d(\mathbf{x}_i, \hat{\mathbf{y}})$ where the average similarity of each weight was calculated as

$$d(\mathbf{x}_i, \hat{\mathbf{y}}) = \frac{\left(\sum_{j=1}^n \frac{x_{ij}}{\hat{y}_j}\right)}{n}$$
(18)

If $d(\mathbf{x}_i, \hat{\mathbf{y}}) = 1.0$ (to within a 10^{-5} threshold) then on average the weights in particle \mathbf{x}_i and $\hat{\mathbf{y}}$ were the same. If this condition was met, the position of particle \mathbf{x}_i was randomly reinitialized and the velocity of \mathbf{x}_i was set to zero. Because \mathbf{x}_i converged on the global best position, the personal best position of \mathbf{x}_i was set to the new randomly initialized position.

2) To improve the exploitation ability of the swarm, fitter random neighbours replace the global best particle. A mutation process creates a neighbour, $\hat{\mathbf{y}}_N$, close to the global best particle. This mutation involved multiplying each weight in the neighbour by a normally distributed random number between -0.01 and 0.01 with a zero mean, that is

$$\hat{\mathbf{y}}_N = \hat{\mathbf{y}} + (\hat{\mathbf{y}}.\mathbf{r}) \tag{19}$$

where $r_j \sim \mathcal{N}(0.0, 0.01) \quad \forall j = 1, 2, ..., n$ in **r**. If the neighbour was fitter than the global best particle, it replaced the global best particle. A similar technique was used by Miranda and Fonseca [23].

Additionally, the execution time of the PSO was improved by updating each particle's position in a concurrent thread. Where applicable, memoization was also used [24]. Memoization is an optimization technique used to improve performance by having functions avoid recalculation of previously processed inputs. Hash map data structures were used to store input-output pairs from frequently called functions to improve performance. An example is the function which fetched interest rates and exchange rates for a particular date. After the first call of this function for a particular date, the hash value of that date is calculated. Then the hash value and rates are stored in the hash map. The hash map is then queried if the rates for that date are requested by any other particle.

C. PSO Portfolio Optimizer Pseudo-code

The pseudo-code in Algorithm 1 describes how the adapted PSO was used to optimize a carry trade portfolio.

Data: Carry Trade Portfolio, $P_{PSO}(t)$ **Result**: Optimized Carry Trade Portfolio, $P_{PSO}(t+1)$ Initialize swarm, S, of size mfor *iteration* $\leftarrow 0$ to maxiterations do $\hat{\mathbf{y}} \leftarrow \mathbf{y}_1$ for $i \leftarrow 1$ to m do if $f(\mathbf{y}_i) > f(\hat{\mathbf{y}})$ then $\hat{\mathbf{y}} \leftarrow \mathbf{y}_i$ //Get global best particle end end Initialize thread pool, $q = \{q_1, q_2, ..., q_m\};$ for $i \leftarrow 1$ to m do if $x_i = \hat{y}$ then create global best neighbour $\hat{\mathbf{y}}_N$ using equation (19); replace global best if neighbour is fitter; else $q_i \rightarrow$ update \mathbf{x}_i velocity using equation (17); $q_i \rightarrow$ update \mathbf{x}_i position using equation (16); end end Wait for threads q_1 to q_m to finish... for $i \leftarrow 1$ to m do if $f(\mathbf{x}_i) > f(\mathbf{y}_i)$ then $| \mathbf{y}_i \leftarrow \mathbf{x}_i // \mathbf{U}$ pdate personal best positions end if $d(\mathbf{x}_i, \hat{\mathbf{y}}) = 1.0$ then reinitialize \mathbf{x}_i as per equation (18) end end end Algorithm 1: Adapted Particle Swarm Optimization

D. Constraint Satisfaction Rules

This section describes how portfolio optimization constraints were handled by the adapted PSO algorithm. One constraint handling approach for PSO lets only feasible solutions be selected as the personal bests or the global best. Other approaches randomly reinitialize infeasible solutions [25] or penalize infeasible solutions using penalty functions for each constraint. In the context of portfolio optimization Chang et al [26] used search heuristics to find solutions which satisfied the cardinality constraint and Deng et al [27] used repair methods to satisfy boundary and cardinality constraints. This paper uses the repair approach. Three repair operators were defined to convert an infeasible solution to a feasible solution:

The lower bound constraint requires that each weight 1) in the portfolio be greater than w_i^{min} . The repair method added the absolute value of the weight, $|w_i|$, and w_i^{min} to the infeasible weight value thereby ensuring that the repaired weight is positive and greater than or equal to the lower bound; that is,

$$w_j = |w_j| + w_j^{min}, \ \forall w_j < w_j^{min} \tag{20}$$

The equality constraint requires that 100% of the 2) portfolio's capital be allocated. The repair method re-based each weight as its percentage of the sum of weights, T; that is,

$$w_j = \frac{w_j}{T}, \ \forall w_j \tag{21}$$

where $T = \sum_{j=1}^{n} w_j$. The upper bound constraint requires that each weight 3) in the portfolio be less than w_i^{max} . The repair method added a portion of the difference between w_i^{max} and the infeasible weight to other weights in the portfolio ensuring that the total weight remained the same and no weights exceeded the upper bound; that is,

$$w_j = w_j + \frac{w_j^{max} - w_j}{n}, \ \forall w_j \ge w_j^{max}, j \ne k$$
(22)

These three repair methods were applied in order to each particle in the swarm after updating their positions.

V. EXPERIMENT DESIGN

The historic performance of each portfolio defined in Sections III and IV were determined through back-tests. Back-tests are experiments which simulate the historical performance of a portfolio or trading strategy. This section provides detail about how the back-tests were designed and executed. The results of these back-tests are then presented in Section VI.

A. Experiment Data

Back-tests for the benchmark portfolios and P_{PSO} were simulated using real data for 22 currencies over a 12 year time period starting on 2000-07-03 and ending on 2012-10-08. This data was sourced from Quandl.com, an online data store. The funding currency was the Japanese Yen (JPY) as it has maintained the lowest interest rate for the better part of two decades (0.346% in the given time period). Table I shows the set of investment currencies.

TABLE I. SET OF INVESTMENT CURRENCIES, THEIR CURRENCY CODES, AND AVERAGE INTEREST RATES OVER THE 12 YEAR PERIOD

Currency Name	Code	Interest Rate
Australian Dollar	AUD	1 080
Australian Dollar	DDI	14.009
Brazillan Keal	DKL	14.27
Canadian Dollar	CAD	1.223
Swiss Franc	CHF	0.763
Chinese Yuan	CNY	2.605
Danish Krone	DKK	2.527
Euro	EUR	2.325
Great British Pound	GBP	3.439
Indonesian Rupiah	IDR	9.741
Israeli New Sheqel	ILS	4.087
Indian Rupee	INR	11.369
Mexican Peso	MXN	3.243
Malaysian Ringgit	MYR	2.989
Norwegian Krone	NOK	3.643
New Zealand Dollar	NZD	5.680
Philippine Peso	PHP	4.987
Russian Ruble	RUB	5.155
Swedish Krona	SEK	1.634
Thai Baht	THB	2.187
Turkish Lira	TRY	29.47
United Stated Dollar	USD	1.994

B. Experiment Configuration

The results from the PSO algorithm are averages across 30 independent runs and were configured to use the values given in Table II for each of the free parameters. The PSO parameters (inertia weight and acceleration coefficients) were set to the popular values published by Eberhart and Shi [28].

TABLE II.	PARAMETER VALUES	USED FOR THE PSO	ALGORITHM AND
	THE $E[FX]$	FORECASTER	

Parameter	Value
Samples	30 samples
Optimization frequency	20 trading days
Objective function	Sharpe Ratio
E[FX] technique	GBM
Forecast data for $E[FX]$	20 trading days
Historical days for $E[FX]$	120 trading days
GBM market drift	0.05
GBM daily variance	0.30
Particle swarm size	30 particles
Iterations per period	3000 iterations
Inertia value, w	0.729844
Social acceleration coefficient, c_2	1.496180
Cognitive acceleration coefficient, c_1	1.496180
Multiple starts	true
Mutate global best particle	true
Weight lower bound, w_i^{min}	0.0
Weight upper bound, w_i^{max}	0.7
Portfolio staring capital	\$100,000

VI. RESULTS

This section first defines the key performance indicators for the portfolios and then presents the averaged back-test results achieved by the portfolios. The portfolios include P_E , P_r , P_S , P_R , P_{R_S} , and P_{PSO} .

A. Key Performance Indicators

For each portfolio, the following key performance indicators were calculated over the 12 year back-test:

- 1) Hit rate (%) the percentage of months which achieved positive returns during the back-test.
- 2) Total return (%) the total return achieved from the beginning to the end of the back-test.
- 3) Average return (%) the average monthly return generated by the portfolio during the back-test.
- Variance the variance of monthly returns. Variance measures portfolio risk, so this should be minimized.
- Standard deviation (stdev) the standard deviation of monthly returns. Standard deviation measures portfolio risk, so this should be minimized.
- Negative standard deviation (Negative stdev) the standard deviation of months which achieved negative returns.
- Maximum draw down the maximal drop of the portfolio value from its running maximum value, that is

minimize
$$D_t = M_t - S_t$$
 (23)

where $M_t = max_{u \in [0,t]} S_u$ and M_t is the running maximum value.

 Maximum draw up - the maximum increase of the portfolio value from its running minimum value, that is

maximize
$$U_t = S_t - m_t$$
 (24)

where $m_t = min_{u \in [0,t]} S_u$ and m_t is the running minimum value.

- 9) Sharpe ratio (Sharpe) a measure of risk adjusted return where risk is the standard deviation of returns.
- 10) Sharpe ratio variance (var(Sharpe)) Sharpe ratio variance on a six month sliding window.

- 11) Average Sharpe ratio (avg(Sharpe)) Mean Sharpe ratio on a six month sliding window.
- 12) Sortino ratio (Sortino) a measure of risk adjusted return where risk is the standard deviation of negative returns.
- 13) Compounded annual growth rate (CAGR%) The year-over-year portfolio growth rate computed as

maximize
$$CAGR = \left(\frac{\text{Ending capital}}{\text{Starting capital}}\right)^{\frac{1}{12}} - 1$$
(25)

B. Tabular Results

Table III contains the results achieved for each of the performance indicators achieved by the portfolios over the 12 year back-test.

FABLE III.	BACK-TEST RESULTS FOR EACH PORTFOLIO ON THE 13
	KEY PERFORMANCE INDICATORS

Indicator	P_E	P_r	P_S	P_R	P_{R_S}	P_{PSO}
Hit rate (%)	63.4	62.1	71.4	69.6	71.4	74.5
Total return (%)	133	118	1563	512	2004	2429
Average return (%)	0.57	0.5	1.82	1.18	1.98	2.12
Variance	.0006	.0003	.001	.0006	.001	.002
Stdev	.024	.018	.035	.026	.038	.041
Negative Stdev	.019	.01	.014	.011	.015	.016
Maximum draw down	-10.8	-4.1	-4.9	-4.5	-5.0	-5.6
Maximum draw up	8.21	6.18	18.3	14.5	21.6	23.0
Sharpe	.24	.28	.52	.45	.52	.51
avg(Sharpe)	.34	.32	.63	.53	.63	.61
var(Sharpe)	.29	.24	.37	.35	.34	.27
Sortino	.29	.50	1.28	1.05	1.36	1.35
CAGR %	6.7	6.2	24.1	15.	26.4	28.2

The following observations are made from Table III:

- 1) Each diversified carry trade portfolio made long term profits over the 12 year period.
- 2) Informed weighting techniques $(P_S, P_R, P_{R_S}, \text{ and } P_{PSO})$ outperformed uninformed weighting techniques $(P_E \text{ and } P_r)$.
- 3) P_r was the worst performing portfolio in terms of hit rate, total return, average return, maximum draw up, average Sharpe ratio, and compounded annual growth.
- 4) P_E had the highest maximum draw down and had the worst Sharpe ratio and Sortino ratio.
- 5) P_{PSO} outperformed the benchmark portfolios in terms of the hit rate, total return, average return, maximum draw up, and compounded annual growth rate.
- 6) P_{PSO} had a higher standard deviation of returns than P_{R_S} did. This might indicate greater risk taking, however because the standard deviation of negative returns for the two portfolios were similar, the higher standard deviation of returns are as a result of the P_{PSO} having higher positive returns. This is supported by the fact that P_{PSO} and P_{R_S} have similar Sortino ratios.
- 7) The maximum drawn down of P_{PSO} was larger than all the portfolios except P_E . However the maximum draw up of P_{PSO} was the largest of all the portfolios.
- 8) The average Sharpe ratio was higher than the Sharpe ratio for all of the portfolios. This implies that the

carry trade portfolio has become less attractive over time.

Table IV presents the average performance improvements realized by P_{PSO} on selected key performance indicators as compared to the set of uninformed (P_E and P_r) and the set of informed portfolios (P_S , P_R , and P_{R_S}).

TABLE IV. Average performance improvements of P_{PSO} against the uninformed and informed benchmark portfolios

Key Performance Indicator	Uninformed	Informed
Hit rate improvement	18.82%	5.28%
Total return improvement	1839.85%	150.30%
Average return improvement	296.14%	34.79%
Sharpe ratio improvement	102.14%	4.07%
Sortino ratio improvement	259.65%	10.93%
CAGR improvement	337.9%	37.3%

Table IV shows that the average P_{PSO} key performance values were much higher than the average values for the uninformed and informed portfolios. The percentage increase in the key performance values of P_{PSO} over the uninformed portfolios was higher than the increase over the informed portfolios.

C. Return Results

Fig. 1 graphs the compounded returns of each portfolio, the dark line on top represents the compounded returns generated by the P_{PSO} which are mirrored by P_{R_S} . Fig. 1 shows that:

- 1) The P_{PSO} performed consistently better than the other portfolios over the given time period.
- 2) Portfolio's which scaled weights based on ranked $E[R_{c_j}]$ (P_{R_S} and P_S) performed better than portfolios which did not scale weights ($P_R, P_E, and R_r$).
- 3) P_{PSO} , P_S , P_R , and P_{R_S} follow the same trend because they used the same E[FX] forecaster.
- 4) P_E performed similarly to P_r .



Fig. 1. Compounded 20 day returns generated by each of the five benchmark portfolios and the P_{PSO} portfolio during the period 2000-07-03 to 2012-10-08. Each portfolio began with \$100,000 worth of capital.

Fig. 2 graphs the average Sharpe ratios for each portfolio over time. This graph was smoothed across 50 optimization periods in order to visualize long term trends. Fig. 3 graphs the returns generated by each portfolio over time. This graph was also smoothed over 50 optimization periods. These figures confirm that the carry trade portfolio had become less profitable over time.



Fig. 2. Sharpe ratio change over time for each of the portfolios from 2000-07-03 to 2012-10-08.



Fig. 3. Average 20 day returns generated by each of the benchmark portfolios and the P_{PSO} from 2000-07-03 to 2012-10-08.

From Figures 2 and 3 it is observed that:

- 1) The returns generated by each portfolio, except P_E , experienced the same general ups and downs.
- 2) The returns generated by P_{PSO} were just slightly larger than P_{R_S} and P_S , but when compounded over the entire time period, this equated to more than 400%.

Fig. 4 contrasts the starting interest rate differentials against ending interest rate differentials on a log graph. Interest rate differentials have mostly decreased which contributes to the declining profitability of carry trades over time.

Observations from Fig. 4 are that:

1) Emerging nations have higher interest rate differentials, e.g. Turkey, Brazil, Indonesia, and India.



Fig. 4. Interest rate differentials for each one of the countries at the beginning of the period, 2000-07-03, against the interest rate differentials at the end of the period, 2012-10-08

 Interest rates in Switzerland and the United States dropped below that of Japan. This means that the USD and CHF became less expensive funding currencies than JPY towards the end of the given time period.

D. Carry Trade Weights

Fig. 5 is a stacked logarithmic line graph showing the P_{PSO} weights of the top and bottom five currencies over time.



Fig. 5. A stacked logarithmically scaled line graph showing the weights selected over the period from 2000-07-03 to 2012-10-08 by a P_{PSO}

Fig. 5 shows that:

- Countries with high interest rate differentials made up the majority of the portfolio and countries with low interest rate differentials made up the minority.
- 2) Over time the portfolio converged on a set of optimal weights with very little variance after that. This may present a problem if a market correction were to occur.

E. Penalty Function

Fig. 6 graphs the compounded returns of P_{PSO} with and without the penalty function, g(P). The portfolio with no penalty performed slightly worse, implying that frequent weight changes do not always result in greater returns.



Fig. 6. Comparison of the average compounded returned generated by the P_{PSO} with and without the penalty function, g(P)

F. E[FX] Forecaster Analysis

Fig. 7 graphs the compounded returns of P_{R_S} when using each of the three different foreign exchange forecasting forecasting techniques, E[FX], discussed in Section II.



Fig. 7. The 20 day returns generated by P_{R_S} when using the three E[FX] forecasting techniques discussed in Section II

Fig. 7 illustrates the impact that poor heuristics have on long term performance. The quality of a forecaster can be measured by calculating the correlation co-efficient, $\Gamma \in [-1, 1]$, between the expected returns generated by the forecaster and the actual returns realized by the portfolio. This is shown in Table V.

Values for Γ of 1, 0 and -1 are indicative of a perfect correlation, no correlation, and perfect anti-correlation, respectively, between the forecaster and the actual returns. The values in Table V show that Geometric Brownian Motion was the only predictor to show a positive correlation to actual returns. Optimizing E[FX] such that Γ is maximized is a worthwhile sub-optimization problem.

TABLE V. Correlation co-efficient values between the expected returns forecast by each E[FX] technique and actual returns generated by the portfolio

E[FX] technique	Г
Normally distributed	0.426
Normany distributed	-0.430
Brownian Motion	-0.007
Geometric Brownian Motion	0.4431

VII. CONCLUSIONS AND FUTURE WORK

Using particle swarm optimization (PSO) to find optimal carry trade portfolios resulted in significant long term performance improvements. Portfolios optimized by the PSO had higher returns, higher compounded annual growth rates, larger draw ups, and better risk adjusted returns as measured by the Sharpe and Sortino ratios.

Despite constraining the search space using an equality and two boundary constraints, the PSO found quality solutions to the carry trade portfolio optimization problem. Using a penalty function to simulate the impact of transaction costs on the portfolio resulted in a slight performance improvement of the portfolio's returns over the long term.

The performance of the portfolio optimizer depends on the reliability of expected returns. Three foreign exchange return forecasters were investigated in this study and it was found that Geometric Brownian Motion produced the most reliable expected returns. Furthermore the correlation coefficient between expected returns and actual returns was observed to be a good heuristic for forecaster reliability.

In conclusion, the particle swarm optimization algorithm is able to find quality solutions to the constrained carry trade portfolio optimization problem. It is expected that algorithmic tuning would result in further performance improvements. Avenues for further research include the use of neural networks to forecast expected returns, enhancing the PSO to self adapt control parameters, and formulating the portfolio optimization problem as a multi-objective problem.

REFERENCES

- H. Markowitz, "Portfolio Selection", *Journal of Finance*, vol. 7, pp. 77-91, 1952.
- [2] F. Busetti, "Metaheuristic Approaches to Realistic Portfolio Optimisation", Masters thesis, Operations Research, University of South Africa, 2000.
- [3] T. Cura, "Particle swarm optimization approach to portfolio optimization", *Nonlinear Analysis: Real World Applications*, vol. 10, pp. 2396-2406, 2009.
- [4] H. Zhu, "Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem", *Expert Systems with Applications*, vol. 38, no. 8, pp. 10161-10169, Aug. 2011.
- [5] F. Xu, J. Jinan, W. Chen and L. Yang, "Improved Particle Swarm Optimization for Realistic Portfolio Selection", in *Proceedings of Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, pp. 185-190, 2007.
- [6] G. Kendall and Y. Su, "A Particle Swarm Optimization Approach in the construction of optimal risky portfolios", in *Proceedings of the* 23rd IASTED International Multi-Conference Artificial Intelligence and Applications, February 14-16, pp. 140-145, 2005.
- [7] G. H. Reza and M. Fazel, "Constrained portfolio selection using particle swarm optimization", *Expert Systems with Applications*, vol. 38, no. 7, pp. 8327-8335, 2011.

- [8] K. Doerner, W. J. Gutjahr, R. F. Hartl, C. Strauss and C. Stummer, "Pareto Ant Colony Optimization: A metaheuristic approach to multiobjective portfolio selection", *Annals of operations research*, vol. 131, pp. 79-99, Oct. 2004.
- [9] C. Burnside, M. Eichenbaum, I. Kleshchelski and S. Rebelo, "Do Peso Problems Explain the Returns to the Carry Trade?", *The Review of Financial Studies*, vol. 24, pp. 853-891, 2010.
- [10] M. Brunnermeier, S. Nagel and L. Pedersen, "Carry Trades and Currency Crashes", *National Bureau of Economic Research Macroeconomics Annual 2008*, vol. 23, University of Chicago Press, 2009.
- [11] C. Burnside, M. Eichenbaum and S. Rebelo, "Carry Trade: The Gains of Diversification", *Journal of the European Economic Association*, vol. 6, Issue 2-3, pp. 581-588, 2008.
- [12] W. F. Sharpe, "The sharpe ratio", in *Streetwise The best of the Journal of Portfolio Management*, Princeton University Press, Princeton, New Jersey, pp. 169-185, 1998.
- [13] E. Vanden-Eijnden, "Wiener Process", in 2005 Program of Studies : Fast Times and Fine Scales, Technical Report WHOI-2006-12, Woods Hole Oceanographic Institution, Lecture 6, pp. 50-56, 2006.
- [14] J. W. McFarland, R. R. Pettit and S. K. Sung, "The distribution of foreign exchange price changes: trading day effects and risk measurement", *The Journal of Finance*, vol. 37, no.3, pp. 693-715, 1982.
- [15] J. E. Fieldsend, J. Matatko and M. Peng, "Cardinality constrained portfolio optimisation", *Intelligent Data Engineering and Automated Learning*, Lecture Notes in Computer Science, vol. 3177, pp. 788-793, 2004.
- [16] A. B. Berkelaar, R. Kouwenberg and T. Post, "Optimal portfolio choice under loss aversion", *Review of Economics and Statistics*, vol. 86, no. 4, pp. 973-987, 2004.
- [17] J. Kallsen and J. Muhle-Karbe, "On using shadow prices in portfolio optimization with transaction costs", *The Annuls of Applied Probability*, vol. 20, no. 4, pp. 1341-1358, 2010.
- [18] F. J. Fabozzi, P. N. Kolm, D. A. Pachamanova and S. M. Focardi, "Robust portfolio optimization", *The Journal of Portfolio Management*, vol. 33, no. 3, pp.40-48, 2007.
- [19] D. Maringer, "Constrained index tracking under loss aversion using differential evolution", in *Natural Computing in Computational Finance*, Springer Berlin Heidelberg, pp. 7-24, 2008.
- [20] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory", in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1995.
- [21] I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection", *Information processing letters*, vol. 85, no. 6, pp. 317-325, 2003.
- [22] J. Kennedy and R. C. Eberhard, "Particle Swarm Optimization", Proceedings of the IEEE International Joint Conference on Neural Networks, pp. 1942-1948, 1995.
- [23] V. Miranda and N. Fonseca, "EPSO Best of two worlds meta-heuristic applied to Power System Problems", *Proceedings of the IEEE Congress* on Evolutionary Computation, pp. 1080-1085, 2002.
- [24] U. A. Acar, G. E. Blelloch and R. Harper, "Selective memoization", ACM SIGPLAN Notices, vol. 38. no. 1, pp. 14-25, 2003.
- [25] A. P. Engelbrecht, Computational intelligence: an introduction, Wiley, Chapter 16, pp. 342-343, 2007.
- [26] T. J. Chang, N. Meade, J. E. Beasley and Y. M. Sharaiha, "Heuristics for cardinality constrained portfolio optimisation", *Computers & Operations Research*, vol. 27, no. 13, pp. 1271-1302, 2000.
- [27] G-F. Deng, W-T. Lin, and C-C. Lo, "Markowitz-based portfolio selection with cardinality constraints using improved particle swarm optimization", *Expert Systems with Applications*, vol. 39, no. 4, pp. 4558-4566, 2012.
- [28] R. Eberhart and Y. Shi. "Comparing inertia weights and constriction factors in particle swarm pptimization", in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 8488, 2000.