Evaluation of the PRDE Trading Algorithm

Pragya Gurung
Departmemnt of Computer Science
University of Bristol)
Bristol BS8 1UB, U.K.
wq19451@bristol.ac.uk

***Abstract*—This paper makes use of the Bristol Stock Exchange (BSE) to run a series of experiments to evaluate the performance of the Parameterised Response Differential Evolution (PRDE) trading strategy, to gain some understanding of how the choice its parameters k (number of population) and F (differential weight coefficient) affects its behavior. Results are presented which demonstrate that for a static market session of 10 days trading period, the current default $k = 4$ and $F = 0.8$ configurations may not be the most optimal. We present other settings such as $k = 8$ and $F = 0.8$ that produce overall higher economic performance under the described market condition. Furthermore, PRDE currently uses the most basic form of differential evolution (DE). Replacing this with other DE strategies could improve PRDE's performance. It is demonstrated that PRDE with greedy versions of DE perform just as well or even better than standard PRDE.**

***Index Terms*—Automated Trading, Financial Markets, Adaptive Trader-Agents, Optimization.**

I. Introduction

In recent times, significant advances in trading algorithms have been made. Many present day financial markets make use of progressively more complex adaptive automated trading systems that adjust their trading strategy to be as profitable as can be; replacing once human traders.

Understanding and analyzing real financial markets poses some major difficulties. For instance, the data streams could be huge and need to be handled with high levels of confidentiality. Additionaly, it is often impossible/impracticable to run controlled tests on real exchanges and details about trading algorithms are likely to be closely guarded secrets. Due to these obstacles, researchers seeking to understand the current dynamics of the financial markets utilize computer-simulation models of markets and agent-based modeling (AGM) techniques. Such tools include the Bristol Stock Exchange (BSE) [1] which is an open-source minimal simulated solution of a limited-order-book financial exchange that runs on continuous double auction. It can support simulations involving large numbers of traders using a variety of trader-agent strategies which can be adaptive or non-adaptive. Adaptive traders change their trading behavior in response to market conditions, whereas non-adaptive traders behave according to their internal algorithm that does not alter over time.

Some of the trader-agent strategies included in BSE are ZIC (Zero Intelligence Constrained) and ZIP (Zero Intelligence Plus). Zero Intelligence (ZI) traders were first introduced by Gode & Sunder in their 1993 study [2]. ZIC traders are entirely non-adaptive. They quote random prices for their bids and asks with the constraint that the price should not be one that would lead to a loss-making transaction. These robot traders produced surprisingly human-like market dynamics and so Gode & Sunders study grew traction rapidly. Further developments on ZI traders resulted in the creation of ZIP traders. These were ZI traders with the ability to adapt but it was very minimal, hence more aptly described as minimal-intelligence. ZIP was used by a team of IBM researchers [3] who demonstrated that ZIP continuously outperformed (produced more profit) human traders during experiments.

The decades since IBM's study, more minimal-intelligence trading strategies have been developed. In 2021, two new trader strategies were added to BSE: PRZI (Parameterized-Response Zero-Intelligence) and PRSH (Parameterised Response Stochastic Hill-climber) which are each direct descendants of Gode & Sunder's ZIC.

- PRZI: A PRZI trader's behavior is governed by a single numeric parameter, s, the trader's strategy value. s is a real number between plus and minus 1 $s \in [-1, +1] \in \mathbb{R}$). Depending on its strategy value, a PRZI trader might behave like SHVR, like ZIC, like GVWY, or like some hybrid mix of those strategies. However, an individual PRZI trader is defined to keep the same value of s for as long as it exists; meaning that PRZI does not adapt its strategy over time.
- PRSH: An extension of PRZI that adapts its value of strategy overtime using a very simple stochastic hill-climbing method. This method is very simple but is also very inefficient as it operates on an infinite loop, where in each pass it evaluates a set K containing k different values of s. It does this by trading a market using each s value for a period of time. Afterwards, it will determine which s value generated the most profit and denote this s_0. Using s_0, it will create k-1 mutations and repeats the previous steps.

In 2022, a successor to PRSH, PRDE (PRZI with Differential Evolution) [4] was added to BSE and is the strategy we will be discussing further in this paper.

- PRDE (PRZI with Differential Evolution): PRDE is a better adaptive version of PRZI. It uses a basic form of optimisation technique called Differential Evolution

(DE), first introduced by Storn & Price [5]. The form of DE used in PRDE is $DE/rand/1$ and it replaces the simple stochastic hill-climber of PRSH. DE is a form of evolutionary computation that maintains and updates a population of candidate solutions which is usually referred to as NP - for "Number of Population" (k in BSE). Candidate solutions in PRDE are single real values $s \in [-1, +1]$ meaning it is one-dimensional. As a result, the DE parameter CR (the crossover parameter) is undefined as there is no crossover operator in PRDE making the implementation of DE simpler. The algorithm chooses one particular strategy s_i whose results have been evaluated. Next three distinct strategies are chosen from the population at random: s_{r1}, s_{r2}, s_{r3}, where $a \neq b \neq c$. These are then combined to create a new strategy v_i using the equation below. The fitness of v_i is compared with the s_i and if the former performs better, it will replace s_i. This is repeated for the next strategy $s_{i,x+1}$.

$$v_i = s_{r1} + F(s_{r2} - s_{r3})$$

Key parameters in PRDE are: k - the number of candidate solutions, F - differential weight coefficient. F controls the extent to which the differences between two candidate solutions merge in with another candidate solution drawn from the same population.

This paper will explore how the choice of values for k and F affects the performance of PRDE and if there is a specific combination of those values that generates better performance than the current default values in BSE $k = 4$ and $F = 0.8$. We will then propose, implement and test possible improved versions of PRDE by making use of contemporary forms of machine learning algorithms where $De/rand/1$ is replaced by more sophisticated instances of DE such as $De/best/1$ and JADE.

A. The Trade Off

For adaptive trading algorithms, they need to be able to adapt quickly to changes in market conditions, but it is equally important that they converge fast and settle to an optimum configuration. This means an algorithm that is designed to adapt quickly may not converge as quickly and vice versa. The performance of PRDE will rely on both its ability to adapt and converge; therefore, this should be taken into consideration when designing and analyzing the experiments.

II. EVALUATING THE IMPACT OF CHANGING K

A. Hypotheses

Null Hypothesis: Given a static market, symmetric supply and demand schedules, a homogeneous market populated by PRDE traders and a differential coefficient $F = 0.8$, a choice of $k = 4$ produces the most profit per trading second (PPS).

Alternative Hypothesis: Given a static market, symmetric supply and demand schedules and a homogeneous market populated by PRDE traders and a differential coefficient $F = 0.8$, a choice of $k \neq 4$ produces the most profit per trading second (PPS).

B. Experiment Design

The PRDE algorithm may be used by adopters in real markets with the intention of generating profit. As a result, we aimed to keep the setup of the market for our experiments somewhat realistic but also simple due to technical constraints. We take inspiration from this [4] paper introducing PRDE by Dave Cliff - 60 PRDE traders, 30 buyers and 30 sellers, populating a static market. Although a dynamic market with shocks would have provided a more realistic environment, the system would possibly need more time to adapt to the sudden changes in the environment and settle on an optimum. Therefore, due to time efficiency, a static market was the more ideal choice. This is good as we are focusing on the impact of k on the performance of PRDE, not on the performance of PRDE in difficult market conditions.

However, unlike Dave Cliff's experiment, we made use of unitary elasticity (supply/demand = (60,140)) instead of perfect. Perfect elasticity is an extreme case in which the quantity demanded increases by an infinite amount in response to any decrease in price and vice versa. It's difficult to relate it to a real world example, therefore is less realistic in comparison. As we are aiming to find the effects of changing k and thereby a best k value, allowing the market to be somewhat realistic is ideal as results may vary if perfect elasticity were to be implemented instead.

For this experiment, multiple market sessions with different values of $k = 4, \ldots, 11$ were executed. As we only wanted to view the effects of changing k, we needed to ensure other aspects of the market were kept constant. For example, the differential coefficient F was set to a value 0.8 and never changed. This value was also used in Dave Cliff's paper [4] and is considered a typical setting [6] for the variable. In addition, the number of hours spent trading per market session was also constant. In any system where you wish to illustrate or explore its response to changes in its inputs, then you need to run it for a long enough time period to properly show its response to that change. As the per-strategy evaluation time is 7200s or two hours, we decided to run the market session for 10 days to give the market enough time to converge and settle. Fig.1 shows an example of profit over seconds for the described market condition.

C. Analysis of Results

Through the use of whisker-plots, we were able to visualize the profit per second (PPS) when changing the value of k. Whisker plots allow us to visualise and compare values such as the medians. Medians are unaffected by extreme outliers or non-symmetric distributions of data, whereas means can vary in skewed distributions. Which is the reason why the results were compared based on the medians.

Observing the plot in Fig.2, we can see that $k = 6$ produces the highest median PPS. It's median seems to be high enough to be above some other k's Interquartile Ranges (IQR) such as

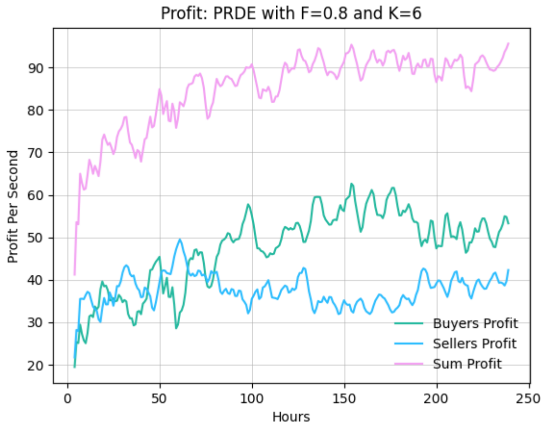

Fig. 1. Plot of profit per time one 10-day experiment in a market populated entirely by PRDE traders. Horizontal axis is time, measured in hours and vertical axis is simple moving average of profit per second (PPS) over each preceding hour. The line labelled ”Buyer Profit” shows the total PPS generated by the population of buyers. Line ”Seller Profit” is the total PPS generated by the the sellers and the line labelled ”Sum Profit” shows the total PPS from by all traders.

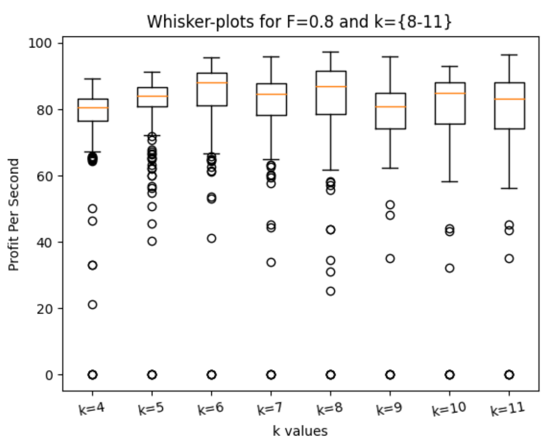

Fig. 2. Whisker plot of data over a 10 day trading period for each k. The horizontal axis contains labels about the k values and the vertical axis is a simple moving average of PPS. The read lines are the medians, the boxes the Interquartile Range and the circles are outliers.

$k = 4$ and $k = 9$. This indicates it is likely that there is some difference between choosing $k = 6$ verses choosing those k other values. However, for k's where there are more overlaps in IQR, further testing is required to come to a conclusion.

As we are comparing multiple groups of data, rather than running a sequence of pairwise A/B t-tests, we will use a method that compares all of the groups at once - A/B/C/D/E. Such tests include the ANOVA-style test (parametric) and the Kruskal-Wallis test (non-parametric). Non-parametric tests do not assume anything about the underlying distribution of the groups, whereas parametric tests assume normal distribution. To decide between the two, we need to determine if the groups underlying distribution was normal and this was achieved in two ways - visually by using Q-Q plots and statistically using

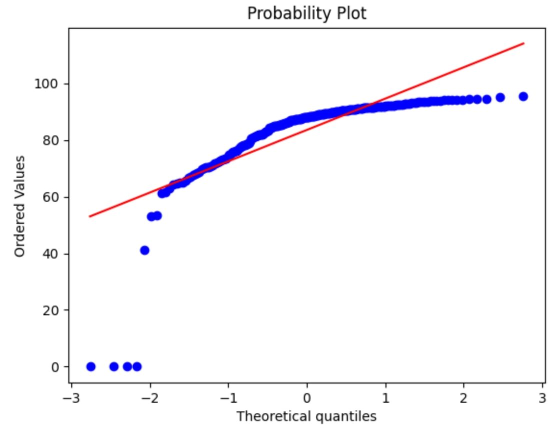

Fig. 3. Q-Q plot for $k = 6$ $F = 0.8$ to show if the data is distributed normally. If the blue data points lie on the red line, then we can assume the data is normally distributed.

TABLE I
SHAPIRO-WILK RESULT FOR CHANGING K

k value	Statistics	pvalue
k=4	0.608	1.451e-12
k=5	0.645	6.693e-12
k=6	0.586	6.266e-13
k=7	0.612	1.738e-12
k=8	0.665	1.564e-11
k=9	0.594	8.378e-13
k=10	0.504	3.296e-14
k=11	0.594	8.514e-13

the Shapiro-Wilk test. For a Q-Q plot if the data is normally distributed, the plots would lie in a straight line. Fig.3 is a Q-Q plot for $k = 6$ and it is clear the data points do not follow a straight line. Similar Q-Q plots were produced for the other k values. This conclusion was further supported by the Shapiro-Wilk test which produced *pvalue* ¡ 0.05 for all k's as shown in Table 1, meaning we reject the null hypothesis and accept that the data is not normally distributed.

For the Kruskal-Wallis test we describe another set of null and alternative hypotheses. **Null Hypothesis:** the medians for all groups are similar, hence come from the same population. **Alternative Hypothesis:** at least one of the medians is different to the other group's.

The *pvalue* after performing the test was $1.33e-13$ which is < 0.05, so we can reject the null hypothesis and state that at least one of the groups do not come from the same population. As a result, we are able to reject the overall Null Hypothesis described in **[A. Hypotheses]** and accept the alternative. Meaning we have enough statistical evidence to say that changing k would result in a statistically significant difference in PRDE's performance. Additionally, we've found evidence that $k = 4$ is not the best choice of k for the market setup described in **[A. Hypotheses]**. Instead either $k = 6$ or

$k = 8$ would be more suitable choices.

III. EVALUATING THE IMPACT OF CHANGING F

A. Hypotheses

Similar to the experiment for changing k, we'd like to determine if any other F values outperforms the BSE's default value $F = 0.8$.

Null Hypothesis: Given a static market, symmetric supply and demand schedules and a homogeneous market populated by PRDE traders, a choice of $F = 0.8$ produces the most profit per trading second.
Alternative Hypothesis: Given a static market, symmetric supply and demand schedules and a homogeneous market populated by PRDE traders, a choice of $F \neq 0.8$ produces the most profit per trading second.

B. Experiment Design

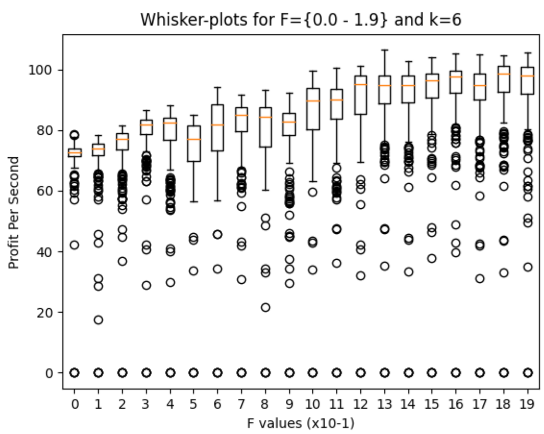

Fig. 4. Whisker plot of data over a 10 day trading period for each F value (0-2). The horizontal axis contains labels about the F values and the vertical axis is a simple moving average of PPS. The read lines are the medians, the boxes the Interquartile Range and the circles are outliers.

For this experiment, multiple market sessions with different values of F were executed. As we are only interested in the effects of changing F, we needed to ensure other aspects of the market were kept constant. Therefore, we made the decision to fix the value of k. Two k values were chosen to run two sets of experiments. Both were taken as the top two k values with the highest median from the previous experiment - $k = 6$ and $k = 8$. Selecting values of $k = 6$ and $k = 8$ gives a good balance between the number of adaptive steps and the time taken to run each simulation. Finally, many of the same design decisions from the previous experiment were implemented in this experiment, such as a static market, 60 PRDE traders in the market and unitary elastic supply and demand. Fig. 5 shows a plot of the profits per second for PRDE traders with $k = 6$ and $F = 0.8$. based on the market setup described previously. Compared to Fig.1 where $F = 0.8$, the "Sum Profit" for Fig.5 reaches significantly higher PPS values.

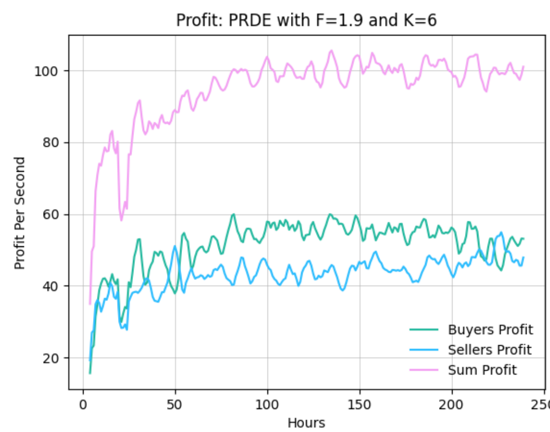

Fig. 5. Plot of profit per time one 10-day experiment in a market populated entirely by PRDE traders where $k = 6$ and $F = 0.8$. The setup is similar to Fig.1

C. Analysis of Results

Looking at the whisker plots, we can see that $F > 1$ produces higher PPS values for both $k = 6$ and $k = 8$. Most of the IQR's for $F > 1$ plots are above the IQR's for $F < 1$ plots, therefore, it is likely there is significant difference between choosing $F > 1$ versus $F < 1$. Further testing is required to come to a clear conclusion as many of the medians are overlapping with IQR's of other plots.

To conduct further testing, we once again need to decide on the appropriate type of test. As we are working with multiple groups of data, ANOVA (parametric) or Kruskal-Wallis (non-parametric) are once again the most appropriate tests. Fig. 6 shows the Q-Q plot for $k = 8$ and $F = 0.8$, which gives us some useful information in deciding on an appropriate test. We subsequently came to the conclusion that the Kruskal-Wallis was a better test for this scenario. This finding was further supported by a Shapiro-Wilk test as the *pvalues* produced for $k = 6$ and $k = 8$ where were both around $e - 20$ in value and so were < 0.05. As a result, we reject the null hypothesis and accept the data is not normally distributed, so we proceed with conducting a Kruskal-Wallis test.

The test was performed on both $k = 6$ and $k = 8$ for which produced $pvalue < 0.05$, meaning we can reject the

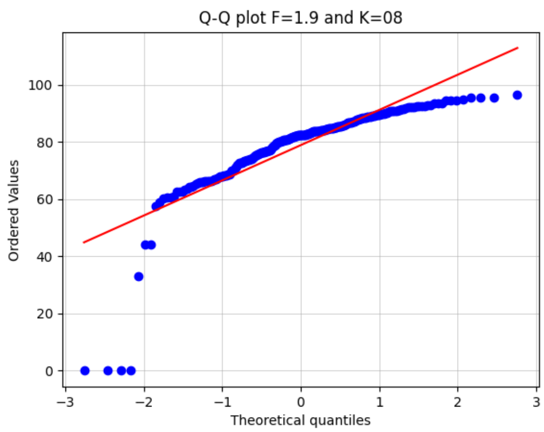

Fig. 6. Q-Q plot for $k = 8$ $F = 0.8$ to show if the data is distributed normally. If the blue data points lie on the red line, then we can assume the data is normally distributed.

null hypothesis and accept that at least one of the medians is not from the same population. This is good as it tells us there might be some F values that could improve PRDE's current performance. Unfortunately, the Kruskal-Wallis test only indicates if there are significant differences in results among groups of data, but does not tell which groups are different from each other. To know which are significantly different from each other, we will perform the Dunn's test [7] as post-hoc test for the significant Kruskal-Wallis test.

The Dunn test will show which combinations of F values produce statistically significant different profit medians. A section of the results can be seen in Table II. The cells containing FALSE showcase the F values that produce statistically different results. We can observe that for $F = 0.8$ verses $F = 0.8$, the cell is FALSE. This along with the knowledge that the median for the latter is higher than the former (Fig.4), we can say that $F = 0.8$ performs better than $F = 0.8$. For that reason, we could reject the null hypothesis described in **A. Hypotheses** and accept that $F = 0.8$ does not produce the most profit per trading second.

TABLE II
DUNN TEST FOR CHANGING F

F value	0.8	1.0	1.9
0.8	TRUE	TRUE	FALSE
1.0	TRUE	TRUE	FALSE
1.9	FALSE	FALSE	TRUE

These results are very interesting as typically the most suitable F value for DE/rand/1 is around 0.6 [8]. There could be a number of reasons for the stark difference. Firstly, unlike the typical DE, PRDE does not have a crossover operator because the candidate solutions are all one-dimensional. Typically, $F = 0.6$ is a good initial choice for populations with dimensions 3D and up. Another reason we could have achieved this unexpected optimum F value could be because the market session are not running for long enough. A trading period of 10 days might not have been sufficient enough for the market to settle to an optimum. Possibly running the market for 100+ days could have resulted in different results.

IV. IMPROVING PRDE

A. Differential Evolution and DE/rand/1

Differential Evolution (DE), originally proposed by Storn & Price [5] is an optimization meta-heuristic. Like other evolutionary algorithms it is population-based using a stochastic search technique using mutation, crossover and selection operators to move the population towards the global optimum. An advantage of DE is that it only has a few parameters to adjust: population size k, the mutation factor (or differential weight) $F = [0, 2]$and the crossover probability (or crossover control parameter) $CR = [0, 1]$.

DE optimizes a problem by maintaining a population of candidate solutions and creating solutions by combining existing ones, then only keeping whichever candidate performs better. This process is repeated until termination. DE uses the mutation operator F to produce a new strategy v_i for each individual x_i in the current population. Some of the most frequently used mutation strategies in DE are shown below.

$$DE/rand/1 v_i = x_{r1} + F(x_{r2} - x_{r3}) \quad (1)$$

$$DE/best/1 v_i = x_{best} + F(x_{r1} - x_{r2}) \quad (2)$$

$$DE/rand/2 v_i = x_{r1} + F(x_{r2} - x_{r3}) + F(x_{r4} - x_{r5}) \quad (3)$$

$$DE/best/2 v_i = x_{best} + F(x_{r1} - x_{r2}) + F(x_{r3} - x_{r4}) \quad (4)$$

$$\begin{aligned} &DE/current-to-best/1 \\ &v_i = x_i + F(x_{best} - x_i) + F(x_{r1} - x_{r2}) \end{aligned} \quad (5)$$

$$\begin{aligned} &DE/current-to-rand/1 \\ &v_i = x_i + rand(x_{r1} - xi) + F(x_{r2} - x_{r3}) \end{aligned} \quad (6)$$

The indices r1,r2,r3,r4 and r5 are mutually exclusive and are randomly sampled integers from range(1, k). DE/rand/1 is the mutation strategy used in BSE's PRDE and is arguably the most simplest form of DE. This formula samples all of its strategy values randomly from the population of candidate solutions. In the following sections we will explore the co-evolutionary dynamics in BSE when DE/rand/1 is replaced by more sophisticated variations of DE: DE/best/1 and JADE.

B. DE/best/1

A paper [9] by Uwe Pahner and Kay Hameyer suggests DE/best/1 is the most favorable mutation strategy for most technical problems and usually performs better than DE/rand/1. This along with the knowledge that implementing the DE/best/1 with the current PRDE code would be a simple task were the reasons why it was the first strategy we chose to implement. To do this, strategies x_{r1} and x_{r2} were chosen at random from the population of possible candidate solutions and the current best performing strategy value is stored in x_{best}. Incorporating best solution information in the evolutionary search means greedy strategies such as DE/best/1 benefit from their fast convergence in comparison to non-greedy strategies like DE/rand/1.

Fig. 7. Plot of profit per time one 10-day experiment in a market populated half by PRDE traders and the other half by PRDE_BEST traders. The setup is similar to Fig.1

To compare DE/best/1 with DE/rand/1 we took inspiration from IBM's Trading Contest [3] where they performed balanced-group tests - buyers and sellers being evenly split between two types of trader algorithms. They deemed this test to be the fairest way to test two different algorithms against each other. For the experiment setup we populated the static market with 15 PRDE and 15 PRDE_BEST traders for both sellers and buyers. The k value was set to 8 and F to 0.8. Fig. 7 showcases the profits per time for both types of traders for 10 days of trading. Both graphs are very similar in shape and even their medians are close ~ 40. However, PRDE_BEST appears to converge sooner and appears to be settling to an optimum faster. This is expected behavior due to the greedy nature of DE/best/1.

To determine if PRDE_BEST is better than PRDE we will need to perform a Mann-Whitney test on the two groups. Mann-Whitney is a non-parametric test that does not assume anything about the underlying distribution of the data. It is a test of the null hypothesis that the distribution underlying sample A is the same as the distribution underlying sample B. The resulting *pvalue* was 0.3097 which is > 0.05 therefore, we can not reject the null hypothesis. This is an expected outcome due to the medians of both being very similar. Although using DE/best/1 over DE/rand/1 did not produce statistically significant differences in profit per second, it allowed the market to reach the same optimum in a shorter amount of time.

C. JADE - DE/current-to-bestp/1

Selecting the optimal control parameters, k and F, is not a trivial task. Typically trial-and-error attempts for fine-tuning them is possible, but it is time consuming and the results are usually problem specific. In an attempt to dynamically update the control parameters, JADE was developed. JADE implements a new mutation strategy denoted as DE/current-to-pbest. This strategy updates the control parameters in an adaptive way and can be implemented with an optional external archive denoted as A which contains the set of inferior solutions.

The mutation factor F_i for each individual x_i is independently generated according to a Cauchy distribution (7) with location parameter μ_f and scale parameter 0.1. $F \geqslant 1$ and $F < 0$. Using a Cauchy distribution allows the system to diversify the mutation factors, thereby avoiding premature convergence which can often occur in greedy mutation strategies such as DE/best. The variable μ_f is set to 0.5 initially and is updated every generation using formula (8), where $mean_L(S_F)$ refers to the set of successful mutation factors in generation g and $mean_L$ is the Lehmer mean (9). By using the Lehmer mean, the adaption of μ_f places more weight on larger successful mutation factors.

$$F_i = randc_i(\mu_F, 0.1) \tag{7}$$

$$\mu_F = (1-c)\mu_F + c * mean_L(S_F) \tag{8}$$

$$mean_L(S_F) = \frac{\sum_{F \in S_F} F^2}{\sum_{F \in S_F} F} \tag{9}$$

The algorithm (10) does not use just the best solution, but randomly chooses one from the top 100p%, $p \in (0, 1]$, solutions. This is because recently explored solutions that were not the best value can still provide useful information about the desired search direction. x_i and x_{r1} are drawn from the population P of possible candidate solutions. x_{r2} however can be drawn two ways. If an archive is not implemented, then x_{r2} is drawn randomly from P, otherwise it is randomly drawn from the union of A and P. An observation made in [10] claims that JADE without archive works well for low dimensional problems (D=30), while JADE with archive performs the best with high dimensional problems (D=100). However, the values of k used in that article were 30, 100, 400 which is much larger than value of k being tested with PRDE. Our population size of 8 may not be sufficient enough, so implementing archive can help introduce more diversity in the mutation operation

$$v_{i,g} = x_{i,g} + F_i(x_{pbest,g} - x_{i,g}) + Fi(x_{r1,g} - x_{r2,g}) \quad (10)$$

To compare PRDE_JADE with PRDE, we once again performed a balanced-group test. A static market was populated with 15 PRDE and 15 PRDE_JADE traders for both sellers and buyers. The k value was set to 8 and the amount trading time was 20 days. A longer time period allows us to view where the graph starts to converge as it takes 4 hours to evaluate a mutation and solutions for each k.

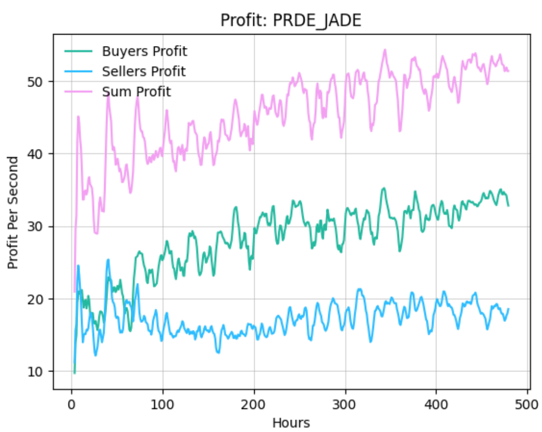

Fig. 8. Plot of profit per time one 20-day experiment in a market populated half by PRDE traders and the other half by PRDE_JADE traders. The setup is similar to Fig.1

Fig. 8 contains the profit graphs produced by both types of traders. From looking at the y-axis, PRDE_JADE does seem to be producing higher PPS than PRDE. Suggesting it could perform better than PRDE in a market filled with only both types of traders.

Fig.9 is a whisker plot for both traders profits. From comparing the medians we can see that PRDE_JADE median is slightly higher than PRDE. However, as the medians overlap with each others IQR boxes, we can not say for sure if there is a significant difference between the two trading strategies. Further testing is required.

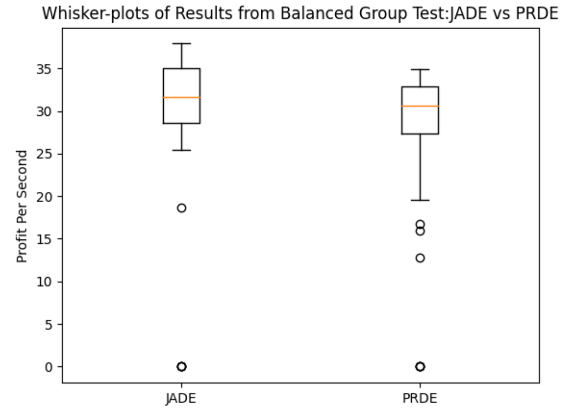

Fig. 9. Whisker plot of data from a balanced group test over a 20 day trading period. The horizontal axis contains labels of the two types of strategies and the vertical axis is a simple moving average of PPS. The read lines are the medians, the boxes the Interquartile Range and the circles are outliers.

Similar to DE/best/1, we perform another Mann-Whitney test to determine if the results for PRDE and PRDE_JADE come from the same population. The *pvalue* generated from this test was 0.000190 which is less than 0.05. This means we can reject the null hypothesis and accept that there is enough evidence to suggest PRDE_JADE and PRDE produce statistically significant differences in PPS. In addition, as the median for PRDE_JADE (Fig.9) is higher than PRDE's, we can claim that JADE performs better under these market conditions.

V. Conclusion

Overall we have seen evidence that changing the values of k and F does produce statistically significant differences in the performances of a PRDE trader. We also found some evidence suggesting that the PRDE's default of $k = 4$ and $F = 0.8$ may not the best configuration (under certain market conditions). Instead $k = 8$ and $F = 0.8$ seem to perform better.

However, given the constraints of the experimental setting, it is impossible to conclude with certainty that there is one k and F value that performs the best in all scenarios. In fact, further research [11] [12] suggests that finding the perfect combination is a trail-and-error process and the values would be problem specific.

We also explored other DE algorithms and analysed how well they perform in comparison to BSE's default DE/rand/1. Both DE/best/1 and JADE performed just as well or better than DE/rand/1. As both are variations of a greedy algorithm, it suggests that BSE's PRDE might benefit from a more greedy approach.

The main limitation for all the experiments detailed in this report, is that they were all recorded under specific market conditions. Additional testing with dynamic and/or extremely volatile market conditions could have lead to different interesting results about the optimal k and F values and DE algorithm.

References

[1] D. Cliff, "davecliff/bristolstockexchange," GitHub, 01 2022. [Online]. Available: https://github.com/davecliff/BristolStockExchange

[2] D. K. Gode and S. Sunder, "Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality," *Journal of Political Economy*, vol. 101, no. 1, pp. 119–37, 1993. [Online]. Available: https://EconPapers.repec.org/RePEc:ucp:jpolec:v:101:y:1993:i:1:p:119-37

[3] R. Das, J. Hanson, J. Kephart, and G. Tesauro, "Agent-human interactions in the continuous double auction," IJCAI International Joint Conference on Artificial Intelligence, 05 2001.

[4] D. Cliff, "Metapopulation differential co-evolution of trading strategies in a model financial market," SSRN, 06 23. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract$_id = 4153519$

[5] R. Storn and K. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341–359, 01 1997.

[6] W. Contributors, "Differential evolution," Wikipedia, 11 2019. [Online]. Available: https://en.wikipedia.org/wiki/Differential$_evolution$

[7] "How to perform dunn's test in python," GeeksforGeeks, 02 2022. [Online]. Available: https://www.geeksforgeeks.org/how-to-perform-dunns-test-in-python/

[8] E. Zulueta, "Differential evolution optimal parameters tuning with artificial neural network," 02 21.

[9] U. Pahner and K. Hameyer, "Adaptive coupling of differential evolution and multiquadrics approximation for the tuning of the optimization process," *IEEE Transactions on Magnetics*, vol. 36, no. 4, pp. 1047–1051, 2000.

[10] J. Zhang and A. C. Sanderson, "Jade: Adaptive differential evolution with optional external archive," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 945–958, 2009.

[11] M. Georgioudakis and V. Plevris, "On the performance of differential evolution variants in constrained structural optimization," *Procedia Manufacturing*, vol. 44, pp. 371–378, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2351978920308684

[12] G. M and P. V, "A comparative study of differential evolution variants in constrained structural optimization," frontiers, 07 9. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fbuil.2020.00102/full