Exploring Behavioural Changes when Modifying Control Parameters Using Differential Evolution and Comparing Performance when Implementing Automatic Adaptation of the Parameters

Anton Wallstedt
Department of Computer Science
Bristol, United Kingdom
su19941@bristol.ac.uk

***Abstract*—This paper reports results from exploring different control parameters for a trading algorithm based on differential evolution (DE), *Parameterised-Response Differential Evolution* (PRDE). This type of trader attempts to adapt its strategy over time using DE to maximise its profit, meaning its strategy in use will depend on the activity of other traders in the market. Each trader maintains a local, private population of strategy values, and uses DE to improve its performance over time. The type of DE used in PRDE is minimal, and requires pre-determined control parameters, i.e. size of the population and the mutation factor. This study aims to further extend Dave Cliff's study: "Metapopulation Differential Co-Evolution of Trading Strategies in a Model Financial Market", which reported a 100% increase in profitability when replacing the original adaptation algorithm: stochastic hillclimber, with DE. The paper only reported results from one set of control parameters, which is further explored in this paper. PRDE is then extended to use JADE, a more complex form of differential evolution, to see if this has an effect on its performance. The trading algorithm was tested in two markets: perfectly elastic market, and unit elastic market with a shock. The results displayed that PRDE using larger populations and higher mutation factors handled the market shock better than the baseline implementation. Implementing JADE did not improve performance. The study concludes that smaller populations are more prone to premature convergence, and larger populations take longer to converge. Lower mutation factors further increase the risk of premature convergence.**

***Index Terms*—PRDE, differential evolution, JADE, control parameters, market shock**

I. Introduction

The Continuous Double Auction (CDA) is a type of auction that underpins many global financial exchanges, due to traders operating in a CDA rapidly converging on the theoretical equilibrium price. In 1962, Smith produced a fundamental study in the field of experimental economics, in which he demonstrated that surprisingly small groups of humans, operating in a lab-setting CDA, successfully converged on the equilibrium [1]. This work inspired Gode and Sunder, who in 1993 produced a seminal study that explored how much intelligence is actually required of traders, for them to successfully converge on the market equilibrium. In [2], Gode and Sunder proposed that the amount of intelligence required is zero, by simulating a CDA populated by automated trading agents of two types: *zero-intelligence unconstrained* (ZIU) and *zero-intelligence constrained* (ZIC). Both of these traders generate random quote prices, however ZIC was constrained to never trade at loss. Their results demonstrated that this simple non-adaptive mechanism can give performance similar to human traders, and concluded that a trader's intelligence has little effect on how efficiently they behave in a CDA, proposing that Smith's invisible hand may have more agency in financial markets than expected.

However, in 1997, Cliff and Bruten extended Gode and Sunder's work and proved that zero intelligence is not enough [3]. They identified that Gode and Sunder's results were artefactual, and found market conditions under which the ZIC trader would fail to equilibrate. Thus, they proposed a new trading algorithm, *zero-intelligence plus* (ZIP), which relies on a basic machine learning algorithm to adapt its behaviour over time, a type of trader commonly referred to as *minimal-intelligence*. Another notable minimal-intelligence trader was created by Gjerstad and Dickhaut in 1998 [4], which at the time remained unnamed, however is often referred to as "GD". GD relies on a probabilistic model to adapt its behaviour over time.

In 2001, a team of researchers at IBM demonstrated that these two minimal-intelligence traders could repeatedly outperform human traders in a market populated by both automated traders and humans [5]. They proposed that efficient trading algorithms could give rise to a business opportunity worth billions of dollars annually. Many refer to their study as initiating the rise of algorithmic trading in real financial markets, and since then have spawned the creation of numerous new algorithmic traders, each with the aim of improving from the other (see e.g. [6]–[9], and [10] for a detailed explanation of some of these algorithms). One of these traders, known as *Adaptive-Aggressive* (AA), was widely thought to be the best trading algorithm, however its supposed dominance was disputed in a series of publications [11]–[15]. In studies [14], [15], AA was tested against two non-adaptive, zero-intelligence trading algorithms, known as *Give-*

away (GVWY) and *Shaver* (SHVR) [16], [17], both of which outperformed AA, among many other machine-learning based adaptive algorithms. As such, Cliff had "an appetite for further zero-intelligence market simulations" and proposed a new trading algorithm: *Parameterised-Response Zero-Intelligence* (PRZI, pronounced "prezzy"), which is based on three zero-intelligence traders: GVWY, SHVR and ZIC [10].

PRZI does not adapt over time, however can act as any one of these algorithms, or a hybrid mix between them all, given a strategy value. Consequently, he implemented a simple stochastic hillclimbing algorithm such that PRZI could adapt which strategy to employ over time, giving rise to *Parameterised-Response Stochastic Hillclimber* (PRSH, pronounced "pursh"). In 2022, he replaced the stochastic hillclimbing algorithm with a minimal form of differential evolution (DE), creating *Parameterised-Response Differential Evolution* (PRDE, pronounced "purdy") [18]. In his results, he reported that using DE is a promising method of adaptation, with a $\sim$100% increase in profit compared to PRSH, courtesy of the DE algorithm. However, his paper explored only one set of control parameters for the differential evolution, which this paper aims to further explore. The line of Parameterised-Response traders, along with a brief description of the DE mechanism, is detailed in Section II.

The experiments will be run in a simulated financial market, known as *Bristol Stock Exchange* (BSE) [16], [17], which is an open-source, minimal simulation of a financial exchange which uses a limit order book (LOB). In BSE, traders can issue new orders at any time, replacing previous orders on the LOB, meaning any one trader can have at most one order on the LOB. It simulates the market at sub-second time-resolutions, meaning experiments spanning hundreds of days can be simulated in a much shorter time. BSE also allows for custom order schedules, custom supply and demand configurations, and supports a very large amount of traders. It also includes multiple types of trading algorithms, some of which are adaptive and change their behaviour in response to other traders, other are non-adaptive, issuing orders set by an internal algorithm that does not adapt over time.

The trading algorithms used in BSE will be detailed in Section II, with details about the form of DE used in PRDE, and a more complex mechanism known as JADE [19] which is used in this study to extend PRDE further. The market and trader configurations are detailed in Section III, with additional details about the technical set up used to produce the results. The results are observed in Section IV, and concluded in Section V. Finally, shortcomings of the methodology and hence suggestions for future work, are detailed in Section VI.

II. Background

A. Parameterised-Response Zero Intelligence (PRZI)

PRZI uses a strategy parameter $s = [-1, +1]$ that dictates to what extent it employs SHVR, ZIC and GVWY. When $s = 0$, PRZI behaves identically to ZIC, when $s \approx \pm 1$ the algorithm becomes more or less "urgent" in its price generation [10], acting like SHVR or GVWY.

B. Parameterised-Response Stochastic Hillclimber (PRSH)

PRSH uses a *k*-point stochastic hillclimber to adapt the strategy parameter, s, in PRZI over time. It consists of one control parameter, *k*, which denotes how many different values of *s* are considered on each adaptive step. It has a "current" trial strategy s_0, and candidate strategies $[s_1, \ldots, s_{k-1}]$, all of which are evaluated in sequence. PRSH evaluates its strategies by trading some number of orders using each strategy, subject to a timeout constraint which is triggered if the current strategy does not lead to enough trades in the given time-window. After a full evaluation of all *k* strategies, it picks the best strategy and copies it into the "current" trial strategy s_0, from which another set of $[s_1, \ldots, s_{k-1}]$ mutants are generated, by applying a mutation function.

C. Parameterised-Response Differential Evolution (PRDE)

PRDE replaces the stochastic hillclimber algorithm in PRSH, with a minimal form of DE [18]. The type of DE used is called DE/rand/1, *rand* denoting the method used to generate new strategy values. The formula consists of two control parameters, *NP* which denotes the population size, and *F*, known as the *mutation factor*. *NP* is essentially the same as *k* in PRSH, it determines how many strategy values to create and evaluate. For the sake of consistency, *k* will be used in place of *NP* in this paper.

The implementation of DE in PRDE uses a *steady-state* evolutionary algorithm, contrary to the original DE algorithm as proposed by Storn and Price in [20] which employs a *generational* evolutionary algorithm. The generational implementation maintains two populations of strategies simultaneously. It iterates through the first population and copies the current strategy, or its mutated equivalent, into the second population subject to their performance. This means, at the end of iterating through the first population, the second population is complete, representing the next *generation* in the evolutionary algorithm.

The steady-state implementation, as used in [18], a single trial strategy from the population is selected at random for each iteration, from which a mutated strategy is created, against which the trial strategy is tested. It writes the trial strategy, or the mutated strategy, back into the population after both of their performances have been evaluated and compared. Whichever has a better fitness is copied back into the population. A steady-state implementation of DE/rand/1 works as follows:

$$s_{new} = s_0 + F \cdot (s_1 - s_2) \tag{1}$$

where s_{new} denotes the mutated strategy value, generated from three distinct randomly chosen strategy values: $s_0,\ s_1,\ s_2$. The trial strategy is s_0, against which s_{new} is evaluated. To mutate the trial strategy, a difference vector, $s_1 - s_2$, is scaled by the *mutation factor*, F, and added to s_0. This form of DE is called DE/rand/1, because the strategies it chooses in the mutation step is random. Before running this algorithm, two control parameters have to be chosen: $k,\ F$, i.e. the population size and the mutation factor respectively, F is usually in the range of $F = [0, 2.0]$ [20]. Finding the best selection of these usually consists of a lengthy trial-and-error process, which is why Zhang and Sanderson [19] proposed a new form of DE, called JADE.

D. JADE: Adaptive Differential Evolution with Optional Archive

JADE is a more complex form of DE, proposed by Zhang and Sanderson in 2009 [19]. It employs a greedier approach than DE/rand/1, with methods in place to prevent premature convergence. The mutation algorithm is known as DE/current-to-*p*best/1, and differs from (1) as follows:

$$s_{new} = s_i + F_i \cdot (s_{best}^p - s_i) + F_i \cdot (s_1 - s_2) \tag{2}$$

The mutation formula, creates a mutated strategy from the trial strategy s_i, called s_{new}, using three strategy values: $s_{best}^p,\ s_1,\ s_2$ and itself (hence *current* in current-to-*p*best). We also have $s_1 \neq s_2 \neq s_i$. The trial strategy is denoted s_i here, as opposed to s_1 in equation (1). Current-to-*p*best picks $s_i,\ s_1$ and s_2 randomly, similar to (1), however also chooses s_{best}^p randomly from a $100p\%$ subset of the best strategies in the population, where p usually ranges over $p = [5\%, 20\%]$ [19]. The mutation factor, F_i is unique for every trial strategy s_i. Moreover, Zhang and Sanderson include an extension of JADE which uses an archive, A, of strategy values that were replaced by their equivalent mutations, to help drive progress in more promising directions [19]. The implementation of JADE using an archive differs slightly from equation (2):

$$s_{new} = s_i + F_i \cdot (s_{best}^p - s_i) + F_i \cdot (s_1 - \tilde{s}_2) \tag{3}$$

the only difference being that s_2 is now denoted $\tilde{s}_2$, indicating that it was selected from the union of the population, P, and archive: $\tilde{s}_2 \in P \bigcup A$. If a mutated strategy replaces its equivalent trial strategy in the evaluation step, the replaced strategy is copied into A. If left to its own, A could grow without bound. Hence, strategies in A are randomly removed if $|A| > k$ until $|A| \leq k$.

Each mutation factor, F_i for $i \in \{1, 2, \dots, k\}$, is uniquely generated for each trial strategy using a Cauchy distribution, with mean μ_F and scale parameter 0.1: $F_i = \text{randc}_i(\mu_F, 0.1)$. Moreover, the implementation of JADE in [19] is of the generational form, where μ_F is updated at the end of each generation as follows:

$$\mu_F = (1 - c) \cdot \mu_F + c \cdot \text{mean}_L(S_F) \tag{4}$$

However, since our implementation of DE uses a steady-state evolution, a counter is kept that is incremented once a trial strategy has been evaluated against its mutation. Once the counter equals the size of the population k, μ_F is updated according to equation (4). The formula uses c, a new control parameter, dictating the rate of adaptation, usually in the range $c = [1/20, 1/5]$ [19]. Moreover, the mean is updated using a Lehmer mean:

$$\text{mean}_L(S_F) = \frac{\sum_{F \in S_F} F^2}{\sum_{F \in S_F} F} \tag{5}$$

where S_F denotes a set of successful F_i values, added to S_F at the evaluation step if the mutated strategy replaced its equivalent trial strategy. Zhang and Sanderson motivated the use of a Cauchy distribution as aiding in the process of diversifying the mutation factors F_i, minimising the risk of premature convergence which is often the case with greedy mutation strategies (e.g. current-to-best). The Lehmer mean is used to propagate larger mutation factors, to improve progress rate [19].

This form of DE now adapts μ_F over time resulting in new mutation factors being generated, still keeping k fixed. This removes the need for a trial-and-error process for finding the most suitable $k,\ F$ pair. However, the adaptation of μ_F and generation of F_i depends on two new control parameters: $c,\ p$. In [19], Zhang and Sanderson ran a few experiments and proposed that the best settings for these parameters are $1/c = [5, 20]$ and $p = [5\%, 20\%]$. They propose that these two parameters are insensitive to different problems, as opposed to a fixed mutation factor. Control parameter c dictates the rate of parameter adaptation, while p determines how greedy the mutation strategy is.

While JADE implements DE/current-to-*p*best/1, one must note the relationship between p and the population size, k. If $p = 20\%$, and we have a population size of $k = 4$, it will consider the 20% best strategies from our population, meaning it will always pick the best strategy, effectively implementing current-to-best: $\lceil 0.2 \times 4 \rceil = 1$. Zhang and Sanderson claim that existing greedy strategies, such as current-to-best, have fast but less reliable performance. As such, it is important to pick $k,\ p$ such that JADE is actually running current-to-*p*best, and not current-to-best, to have more reliable performance.

III. Methodology

The set up of the experiment is now detailed, including trader configuration, supply and demand schedule, the motivation for these choices, and the technical set up used for runing the experiments. Since this paper attempts to reproduce and extend the results from Cliff's 2022 PRDE study [18], a similar configuration has been used.

A. Market Configuration

Two types of markets were used in this study: (I) a perfectly elastic market, and (II) a dynamic market with a predetermined market shock. Market (I) was set up as identically as possible to the market used in [18], to create similar baseline results that varying $k,\ F$ values could be tested against. The reason for implementing market (II) was to further examine how different mutation parameters could affect the response of the traders when faced with a sudden market shock. Market

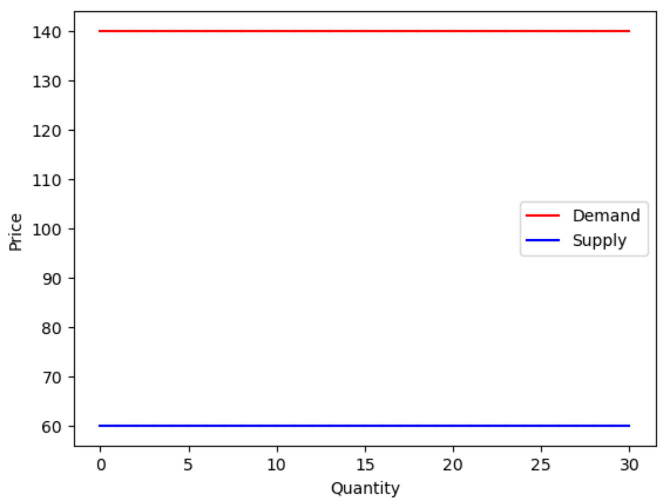
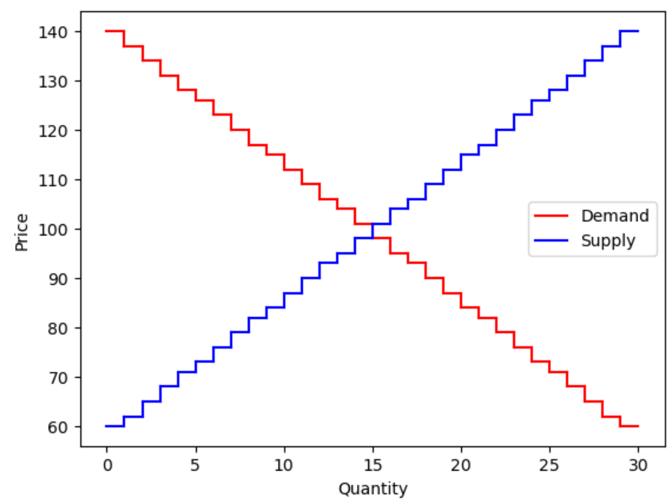
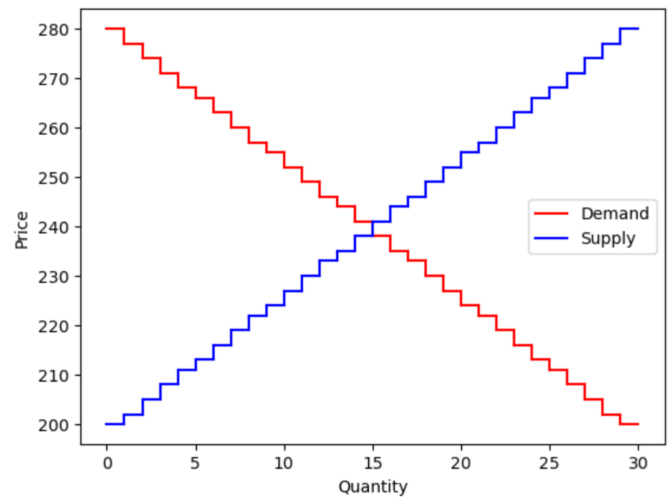


Fig. 1: Perfectly elastic supply and demand schedule used in market (I).



(a) Initial supply and demand ranges used in market (II), swapped out for (b) to simulate a market shock.



(b) Second supply and demand ranges used in market (II) to simulate a shock.

Fig. 2: The supply and demand schedules used in market (II). (a) is used for the first 10 days, after which the schedule is switched to (b). (b) is used for another 10 days, after which the schedule is reset to (a) again, for another 10 days. As displayed, both schedules have unit elasticity, however note the different ranges in (b), i.e. $[200, 280]$ as opposed to $[60, 140]$ in (a).

(II) could also be used with the implementation of JADE, since it automatically adapts the mutation factor, to observe if this had an effect on its ability to deal with a shock. In both markets, all inactive traders were resupplied every 5 seconds. All experiments were run for 30 simulated trading days, with traders occurring around the clock every day.

1) Market I: Perfectly Elastic Market: The first market configuration used a perfectly elastic supply and demand schedule. As in [18], buyers were instructed to pay no more than \$140 per unit, while sellers were instructed to sell for no less than \$60 per unit. A market of this type has been used in other studies, e.g. [21], and ensures that every seller can find a buyer (and vice versa), such that no traders are given extra-marginal limit prices that could prevent them from finding a counter party. See Fig. 1 for the supply and demand curves.

2) Market II: Unit Elastic Market with Shock: The second market configuration used a unit elastic supply and demand schedule, with supply and demand ranges $[60, 140]$ and $[140, 60]$ respectively. After 10 days, the supply and demand ranges were switched to $[200, 280]$ and $[280, 200]$, lasting for another 10 days after which the ranges were reset to the original ranges. See Fig. 2 for the supply and demand curves.

B. Trader Configuration

1) PRDE: The market was populated with 30 PRDE buyers and 30 PRDE sellers, homogeneously using the same control parameters k, F, the same as in [18]. Three different mutation factors were used: $F \in \{0.4, 0.8, 1.6\}$, and for each mutation factor, four different population sizes were tested: $k \in \{4, 8, 12, 16\}$. This means, for each trial, a total of 12 experiments were run, for each market. The k, F values used in [18] were 4, 0.8 respectively.

2) Extending PRDE with JADE: PRDE was extended to implement JADE. The new control parameters, c, p were set to 4, 20% respectively, on population sizes $k \in \{4, 10\}$. As mentioned in Section II.D, the trading algorithm removes the need to test different mutation factors F, since these are adapted over time. Moreover, to see the difference between JADE effectively running current-to-best (i.e. $p = 20\%$ with $k = 4$) and current-to-*p*best, population size $k = 10$ was selected such that at each evolution step, it picks s^p_{best} in equation (3) from a set of strategy values of size 2: $0.2 \times 10 = 2$.

C. Technical Set Up

The experiments were run on multiple lab machines provided at our faculty, each running a 12th generation Intel

Core i7-12700K at 3.60GHz, with 12 cores in total. Since Python runs single-threaded, it allowed for multiple trials to be run in parallel on one machine. As such, 10 trials were run at once for each experiment. There were 12 experiments per market, each experiment run 10 times simultaneously, resulting in $12 \times 2 \times 10 = 240$ output files produced. Sweeping the k parameter over $k \in \{4, 8, 12, 16\}$ for one F-value took roughly 20 hours. Running PRDE using JADE for one k, took roughly ∼4 hours. At times, multiple lab machines were used to parallelise the generation of results even further, due to time constraints.

IV. Results

We now turn to the results collected from varying the population size and mutation factor k, F in Market I and Market II. After observing the change in behaviour for PRDE, we turn to the behaviour when using JADE for differential evolution. The results are discussed and concluded in Section V.

A. PRDE

1) Market I: Perfectly Elastic Supply and Demand: Results from this set of experiments can be seen in Fig. 3. In Fig. 3 (a), where F was fixed to $F = 0.4$, using PRDE with a population size of $k = 4$ resulted in premature convergence, with the PPS-curve displaying a short initial adaptive transient over days 0 to ∼5, converging on a PPS-value of ∼335. The remaining population sizes, $k \in \{8, 12, 16\}$ had a longer initial adaptive transient over days 0 to ∼20-25. As F is doubled to $F = 0.8$, seen in Fig. 3 (b), the lowest population $k = 4$ does not prematurely converge contrary to (a) and seems to near convergence around day ∼25. As the population size is increased to $k \in \{8, 12, 16\}$, it is not clear whether or not the curves converged by the end of the experiment, as all of them display an upwards trend for the full length of the trading period. When F is further increased to $F = 1.6$, as seen in (c), the lowest population size $k = 4$ does display premature convergence again, with an initial adaptive transient over days 0 to ∼7. Increasing the population size displays an increased adaptation period, where $k = 8$ exhibits a convergence after ∼20 days, while the remaining population sizes $k \in \{12, 16\}$ are still showing upwards trends even on the last trading day (i.e. day 30). Analysing the box plots, one can observe a clear increase in cumulative profit between $F = 0.4$ and $F = 1.6$. When using $F = 0.4$, the cumulative profits for all population sizes are clustered around ∼250,000, whereas for $F = 1.6$ they are more tightly clustered around ∼270,000, an ∼8% increase.

2) Market II: Unit Elastic Supply and Demand with Shock: Results gathered from these experiments can be seen in Fig. 4. There is a similar trend in this market for the lowest and highest mutation factors, $F = 0.4$ and $F = 1.6$, in the smallest population size $k = 4$. In these experiments, the PPS-curves display a short initial adaptive transient of around ∼5 days, converging on a pps value of around ∼70 for $F = 0.4$ and ∼90 for $F = 1.6$, as seen in Fig. 4 (a) and (c). More interestingly, in (a), a lower mutation factor of $F = 0.4$ appears to cause a considerable negative shift in PPS. As the mutation factor is increased to $F = 0.8$ in (c), the negative impact seems less pronounced, and for the largest $F = 1.6$, the impact is hardly noticeable, especially for the larger population sizes $k \in \{8, 12, 16\}$. Observing the box plots in Fig. 4, there is no considerable difference in cumulative profit when going from $F = 0.4$ to $F = 0.8$. However, when using $F = 1.6$, with a larger population $k \in \{8, 12, 16\}$, there is a considerable increase in cumulative profit, going from values clustered around ∼58,000 to ∼66,000, a ∼14% increase.

B. Extending PRDE with JADE: APE

The results from running PRDE with JADE using two different population sizes: $k \in \{4, 10\}$ in both markets are now presented, compared against the baseline implementation of PRDE using $k = 4$ and $F = 0.8$ as in [18].

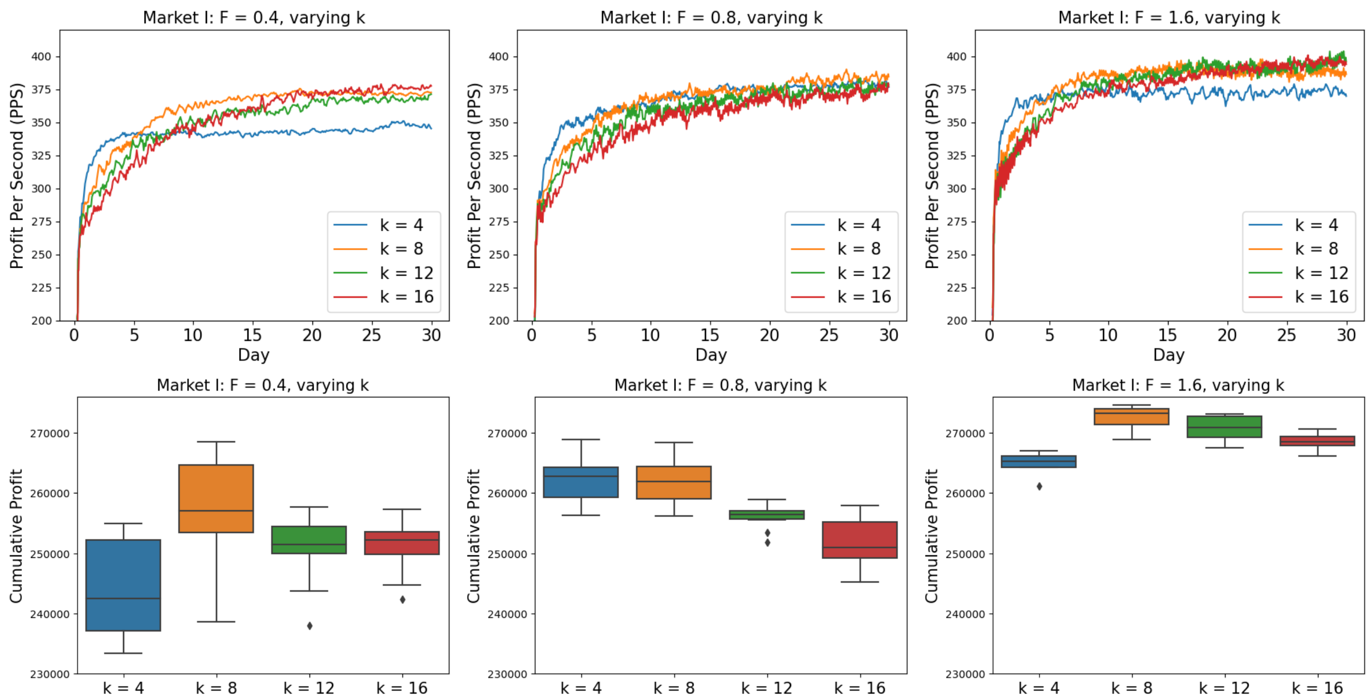
1) Market I: Perfectly Elastic Supply and Demand: In a perfect market, the implementation of JADE did not perform better than PRDE. In Fig. 6 the PPS-curve of the baseline PRDE implementation, with $k = 4$ and $F = 0.8$, produced a PPS-curve higher than that of both configurations of JADE. However, when observing the box plots displaying cumulative profit, there is notable overlap of error bars, so this difference is not conclusive.

2) Market II: Unit Elastic Supply and Demand with Shock: Observing the same implementations of JADE and PRDE in a dynamic market with a market shock, all traders exhibit a negative shift in PPS from day 10 through 20, JADE did not handle the market shock any better than PRDE did, and had a slightly slower rate of convergence, only reaching the same PPS-value to PRDE on the last day of the experiment.

V. Discussion and Conclusion

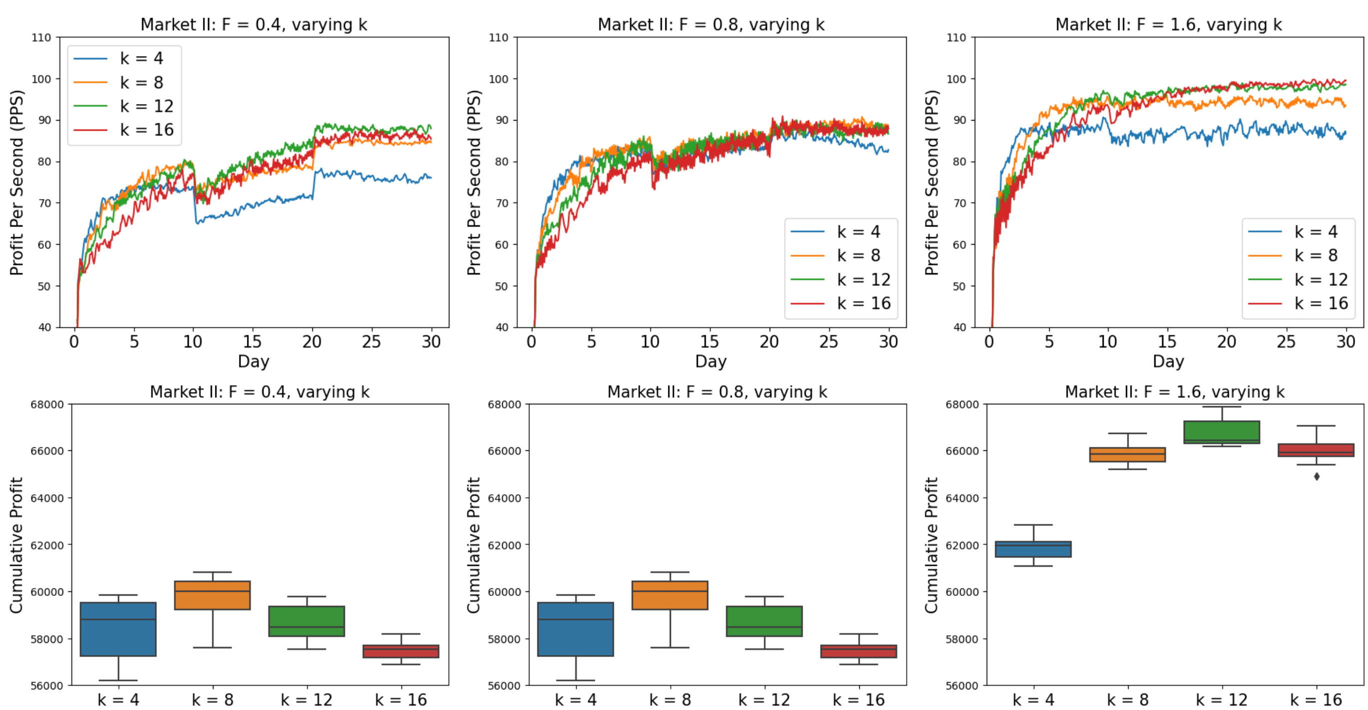
Observing the results from Fig. 3, it can be concluded that increasing the population size results in a longer initial adaptive transient. Moreover, for smaller population sizes, using a lower F-value, may result in premature convergence. The impact of using an F-value somewhere in the middle of the range $[0, 2.0]$ as suggested by Storn and Price in [20], premature convergence is seemingly less of an issue, as could be seen in Fig. 3 (b), where all the PPS-curves follow a similar trend. Moreover, it can also be concluded that using a smaller population size will result in quicker convergence, which could be due to the entire population more quickly being populated with newer mutated strategies, which naturally would take longer in a larger population.

However, the most notable difference with varying k and F can be seen in Fig. 4, in which the traders had to adapt to a market shock. In these experiments, a lower mutation factor meant the traders were negatively affected by the shock, due to having to re-adapt to the new market environment. Using a low mutation factor in this case could mean that it would take longer for the strategies in the population to change such that they are more suited to the new environment. Larger populations are less affected by the shock, which could



(a) Fixed $F = 0.4$ (b) Fixed $F = 0.8$ (c) Fixed $F = 1.6$

Fig. 3: Three graphs for each F-value displaying 6-hour SMA PPS curves over 30 days in a perfectly elastic market, when keeping F fixed to: $F \in \{0.4, 0.8, 1.6\}$ and varying k for $k \in \{4, 8, 12, 16\}$]. Also displays three box plots below each PPS-graph displaying cumulative profit. All graphs display results from 10 i.i.d. trials. Y-ranges for PPS-curves and box plots have been kept constant.



(a) Fixed $F = 0.4$ (b) Fixed $F = 0.8$ (c) Fixed $F = 1.6$

Fig. 4: Same as Fig. 3, but in a dynamic market with unit elastic supply and demand, with a market shock at day 10 lasting until day 20. All graphs display results from 10 i.i.d. trials. Y-ranges for PPS-curves and box plots have been kept constant.

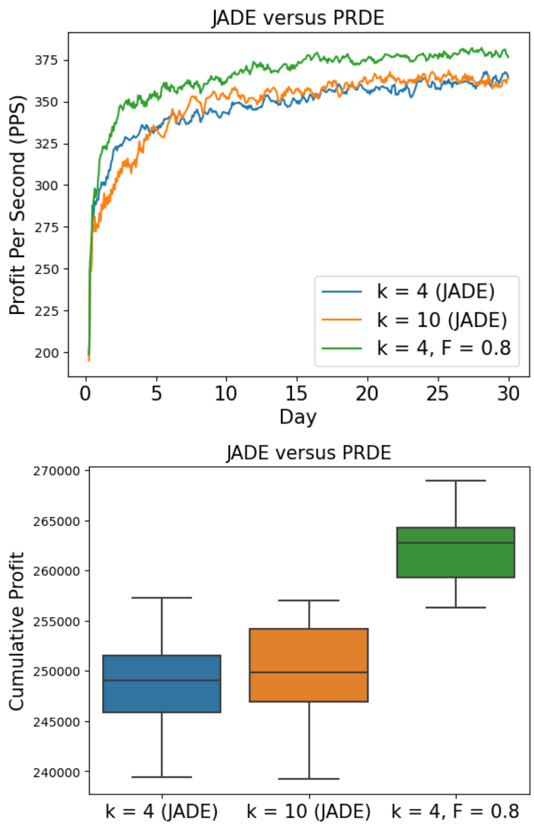


Fig. 5: PRDE with $k = 4,\ F = 0.8$ plotted against JADE with population sizes $k \in \{4, 10\}$ in the perfectly elastic market I.

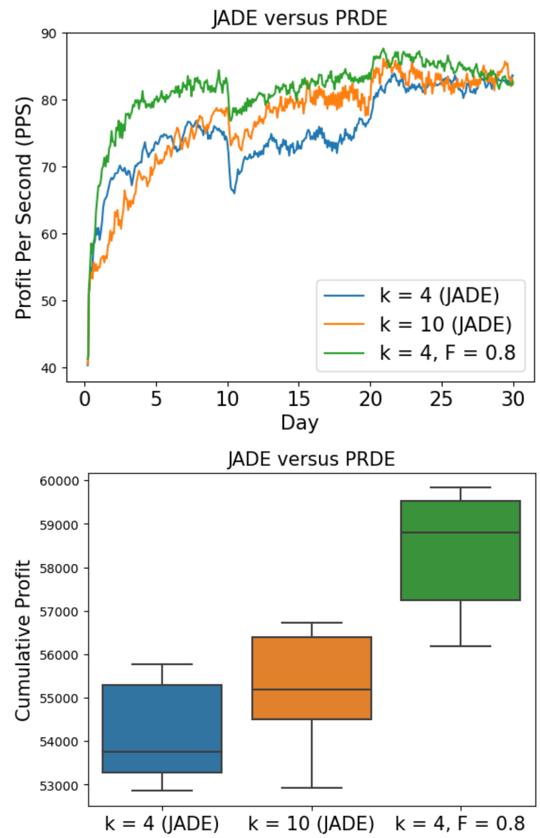


Fig. 6: PRDE with $k = 4,\ F = 0.8$ plotted against JADE with population sizes $k \in \{4, 10\}$ in the unit elastic market II with a shock.

be due to a larger variety of strategy values than in smaller populations. More interestingly, as F is increased, the negative shift in PPS from the market shock decreases, as seen in Fig. 4 (b) and (c). The same trend, with larger populations handling the shock better, are exhibited in these plots. The negative shift is almost unintelligible for the largest population sizes, using the largest mutation factor. Since a larger population means a wider range of different mutation factors, coupling this with a more aggressive mutation factor means the population is more diverse, and perhaps better suited to handle a market shock.

As for the comparative experiment between PRDE using JADE and the baseline implementation of PRDE in Market I, using JADE results in convergence on slightly lower PPS-values. However, observing the box plots in Fig. 5, there is overlap of the error bars, as such more experiments should be run to state this with confidence. In Market II, running JADE with $k = 4$ produced the worst results, where the baseline implementation of PRDE produced significantly better results. However, for both population sizes $k = 4,\ k = 10$, JADE did converge on a similar PPS-value to baseline PRDE towards the end of the experiment (on day 30).

To conclude, smaller population sizes when using differential evolution may result in premature convergence when using very low, or very high mutation factors. Larger populations will take longer to converge, but may offer more reliance in a dynamic market when encountered with a shock. PRDE handled a market shock the best with a large population, e.g. $k \in \{12, 16\}$, when using a large mutation factor. This could be due to a larger population, with a strong mutation factor, consisting of a larger variety of strategies that can be employed when the trader encounters a shock. Implementing PRDE with JADE did not result in better performance in neither a perfectly elastic market or unit elastic market with a shock, however future work in this regard is detailed in Section VI.

VI. Future Work

The experiments run in this study only tested homogeneous parameter sweeps, thus a future line of enquiry is to test heterogeneous sweeps of $k,\ F$, i.e. different traders using different control parameters. While the implementation of JADE did not prove better than the original PRDE implementation, JADE requires two new control parameters to be tested. Hence, these control parameters could be further explored to see if a slower or higher rate of adaptation (i.e. varying c) has an effect on profit extraction. Moreover, the highest recommended value for p in [19] is 20%, and using this value with larger populations, resulting in a larger set of the p-“best” strategy values is also worth exploring. If a large enough population is used, varying p is also a possible line of study. Another line of enquiry could be to study PRDE's behaviour under more market conditions, such as volatile markets, perhaps simulating geometric Brownian motion (drunkard's walk) with jump diffusion, to see if the same results observed in Market II (dynamic market with a shock) translate to more realistic market, where the equilibrium is constantly changing.

REFERENCES

[1] Vernon L. Smith. "An experimental study of competitive market behavior." In "Journal of Political Economy", p.111-137. 1962.

[2] Dhananjay K. Gode, and Shyam Sunder. "Allocative Efficiency of Markets with Zero-Intelligence Traders: Markets as a Partial Substitute for Individual Rationality". In "Journal of Political Economy", vol. 101, no. 1, pp. 119-137. 1993.

[3] Dave Cliff, and Janet Bruten. "Zero is Not Enough: On The Lower Limit of Agent Intelligence For Continuous Double Markets". 1997.

[4] Steven Gjerstad, and John Dickhaut. "Price Formation in Double Auctions". 1998.

[5] Rajarshi Das, James E. Hanson, Jeffrey O. Kephart, and Gerald Tesauro. "Agent-Human Interactions in the Coutninuous Double Auction". 2001

[6] Gerald Tesauro, and Jonathan L. Bredin. "Strategic sequential bidding in auctions using dynamic programming". 2002.

[7] Steven Gjerstad. "The impact of pace in double auction bargaining." 2003.

[8] Perukrishnen Vytelingum, Dave Cliff, and Nicholas R. Jennings. "Strategic bidding in continuous double auctions". 2008.

[9] Jasmina Arifovic, and John Ledyard. "A behavioral model for mechanism design: Individual evolutionary learning. In "Journal of Economic Behavior & Organization", vol. 78, pp. 374-395.

[10] Dave Cliff. "Parameterised-Response Zero-Intelligence Traders". 2022.

[11] Daniel Vach. "Comparison of double auction bidding strategies for automated trading agents". 2015

[12] Dave Cliff. "Exhaustive testing of trader-agents in realistically dynamic continuous double auction markets: AA does not dominate". In "Proceedings of the 11th International Conference on Agents and Artificial Intelligence (ICAART)", pp. 224-236. 2019.

[13] Daniel Snashall, and Dave Cliff. "Adaptive-Aggressive Traders Don't Dominate". 2019.

[14] Michael Rollins, and Dave Cliff. "Which Trading Agent is Best? Using a Threaded Parallel Simulation of a Financial Market Changes the Pecking-Order". In "Proceedings of the 32nd European Modelling and Simulation Symposium (EMSS2020)". 2020.

[15] Dave Cliff, and Michael Rollins. "Methods matter: A trading algorithm with no intelligence routinely outperforms AI-based traders". In "Proceedings of IEEE Symposium on Computational Intelligence in Financial Engineering (CIFEr2020). 2020.

[16] Dave Cliff. "Bristol Stock Exchange: open-source financial exchange simulator". https//github.com/davecliff/BristolStockExchange. 2012.

[17] Dave Cliff. "BSE: A Minimal Simulation of a Limit-Order Book Stock Exchange. In "Proc. 30th Euro. Modeling and Simulation Symposium (EMSS2018)", pp. 194-203. 2018.

[18] Dave Cliff. "Metapopulation Differential Co-Evolution of Trading Strategies in a Model Financial Market". 2022.

[19] Jingqiao Zhang, and Arthur C. Sanderson. "JADE: Adaptive Differential Evolution with Optional External Archive". 2009.

[20] Rainer Storn, and Kenneth Price. "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces". 1997.

[21] Vernon Smith. "Experimental auction markets and the Walrasian Hypothesis". In "Journal of Political Economy", vol. 73, no. 4, pp. 387-393. 1965.