Amsterdam Spice Exchange Simulator 1722

Pritesh M.K.

Abstract

In the early 18th century, trading spices was critical to the economic and political powerhouses of Europe, necessitating efficient global trade mechanisms. And the Dutch were no doubt, right at the forefront of the spice game. But what if we imagined a world where traders could go in and out of positions (spices) and receive real-time live updates on a screener whilst executing trades in 1722. The ASE Simulator 1722 project blends this historical significance with modern technology, offering a dynamic and interactive simulation of the Amsterdam Spice Exchange. The simulator models the prices of various spices using Geometric Brownian Motion (GBM), reflecting realistic market volatility and price fluctuations. Through sophisticated trading strategies such as the Mean Reversion and Bollinger Bands, users can engage in automated and manual trading, experiencing the challenges and opportunities of historical trade. The user-interactive mode, the core of ASE 1722, allows participants to execute trades, manage portfolios, and analyze performance metrics in real-time. This project not only provides an educational tool for understanding historical trade dynamics but also serves as a platform for exploring modern quantitative trading strategies.

Introduction

The ASE Simulator 1722 project envelops a rare endeavor: blending phenomena from vastly different timelines to create a harmonious (or maybe chaotic) coexistence. Such a fusion has no commonality; for it gives us ingredients to explore historical scenarios with the analytical power of modern technology-backed mathematics. The rationale behind this project is to harness the richness of historical trade dynamics, particularly the spice trade, and apply contemporary quantitative techniques to simulate and understand these dynamics in a detailed, interactive environment.

Historically, spices such as cinnamon, pepper, nutmeg, saltpeter, mace, and hyssop played pivotal roles in global trade, driving economic and political powerhouses of Europe to explore and establish intricate trade routes. The Dutch and Portuguese, in particular, were instrumental in discovering and monopolizing these routes, navigating the treacherous seas to bring valuable spices from the East to the West. The importance of these spices went beyond culinary uses; they were integral to medicine, preservation, and even warfare. For instance, saltpeter was a critical ingredient in gunpowder, making it highly sought after during times of war. The 18th century was a period marked by intense geopolitical strife. The Holy Roman Empire, the Dutch Empire, the Danes, Spain, Portugal, Britain, and France were frequently embroiled in conflicts that shaped the political landscape of Europe. These wars had profound impacts on trade, causing significant volatility in the prices of commodities. Traders found it challenging to sell certain spices like cinnamon during turbulent times, leading to drastic price fluctuations. In contrast, wartime necessities like saltpeter saw their prices soar due to the increased demand for ammunition.

While nobody could've possibly had the modern-day convenience of rapidly selling shares on Robinhood whilst lying on the couch after reading discouraging quarterly earnings, the smart traders in 1722 unquestionably deployed ideas that, for the large part, will mimic the strategies of our day. It's the difference between trying to predict when a war would end to buy the dip on cinnamon and unsuccessfully shorting lemongrass before a major new trade route is discovered by Tasman for the Dutch East Company that determined fortunes. So what is stopping us from simulating those times and seeing how we can exploit market sentiment with modern-day computing.

The ASE 1722 emulates such historical scenarios by deploying randomized probabilistic determinations. This project aims to provide insights into how traders might have reacted to news like, "The Holy Roman Empire has sanctioned the sacking of Amsterdam by the fortnight, effective immediately." By incorporating such events into the simulator, we can observe the resultant market sentiment and price adjustments. The backend of the simulator effectively modifies the parameters of the Geometric Brownian Motion (GBM) model to reflect these changes, creating a realistic trading environment.

GBM is a mathematical model typically used to simulate stock prices. It accounts for continuous compounding and random fluctuations, making it ideal for capturing the volatility of spice prices in the 18th century. In addition to GBM, the simulator incorporates advanced trading strategies like Mean Reversion and Momentum. Mean Reversion is based on the idea that prices tend to return to their historical average over time, while Momentum strategies capitalize on the continuation of existing trends.

Simulation and Geometric Brownian Motion

The ASE Simulator 1722 begins on July 27th, 1722, chosen arbitrarily for the sake of consistency. The simulation operates with six-hour trading days, excluding weekends, resulting in 30-hour trading weeks. The unit of price is the guilder, reflecting the historical currency used in the Netherlands during this period. This setup provides a realistic framework for exploring the historical dynamics of the spice trade and the impact of various market conditions.

The simulation includes a diverse range of spices, each with its own unique characteristics and market behavior. These spices include but are not limited to Cinnamon (CINN), Pepper (PPR), Saltpeter (SPTR), Lemongrass (LEM), Mace (MAC), Nutmeg (NTG), Clove (CLV), Hyssop (HYP), Sugar (SUG), Charcoal (CHAR), Cassia (CAS), Turmeric (TUR), and Ginger (GIN). Each spice is represented by a ticker symbol on the Amsterdam Spice Exchange (ASE), allowing users to track and trade them throughout the simulation.

At the heart of the simulation lies the Geometric Brownian Motion (GBM) model, a widely used mathematical model for simulating stock prices and other financial variables. The GBM formula is defined as:

$$dS_t = \mu S_t d_t + \sigma S_t dW_t$$

Where:

- S_t is the price of the spice at time t
- μ is the drift coefficient, representing the expected return
- σ is the volatility term, representing the standard deviation of returns
- dW_t is a Wiener process, introducing randomness to the price changes

In the ASE Simulator 1722, each spice is assigned initial coefficients for drift and volatility under four different market conditions: normal, war, trade boom, and drought. These conditions reflect the varying economic and political climates that can impact spice prices. For instance, spices like Charcoal (CHAR) and Saltpeter (SPTR) have higher drift coefficients during wartime due to increased demand for fuel and gunpowder.

To introduce further randomness and reflect real market behaviors, the intruder function periodically modifies the drift coefficients. With less than a 15 percent probability, these coefficients may revert back to their static mean, adding complexity to trading strategies. This randomness ensures that using strategies like Bollinger Bands or mean reversion is not straightforward, as traders must account for unexpected market shifts.

Additionally, the simulation assigns small probabilities to exponential growth in the drift coefficients for spices trading below 10 guilders. These spices can experience sudden growth spurts, mimicking the behavior of modern-day penny stocks. However, this potential for rapid growth comes with increased risk, as these spices also exhibit higher volatility. The volatility coefficients can range from 0.2 to 0.9 in wartime, making it challenging to predict whether shorting is the smartest decision unless there is a prolonged war.

Throughout the simulation, the prices of spices are constantly printed on the ticker screener, providing real-time updates on market conditions (see Figure 3). This live data feed allows users to observe and react to market movements, enhancing the interactive experience of the ASE Simulator 1722. By combining historical context with advanced quantitative models, the simulation offers a unique and engaging way to explore the complexities of the spice trade in the 18th century.

```
1722-07-27 14:38 | CINN: 102.39 | PPR: 107.80 | NTG: 58.26 | CLV: 42.92 | LEM: 35.74 | SPTR: 26.46 |
1722-07-27 14:37 | CINN: 102.16 | PPR: 107.88 | NTG: 58.02 | CLV: 42.94 | LEM: 35.73 | SPTR: 26.49 |
1722-07-27 14:36 | CINN: 102.18 | PPR: 107.97 | NTG: 58.21 | CLV: 42.93 | LEM: 35.71 | SPTR: 26.50 |
1722-07-27 14:35 | CINN: 102.44 | PPR: 108.16 | NTG: 58.35 | CLV: 42.93 | LEM: 35.69 | SPTR: 26.49 |
1722-07-27 14:34 | CINN: 102.61 | PPR: 108.02 | NTG: 58.26 | CLV: 42.95 | LEM: 35.70 | SPTR: 26.52 |
1722-07-27 14:33 | CINN: 102.19 | PPR: 107.96 | NTG: 58.38 | CLV: 42.90 | LEM: 35.70 | SPTR: 26.46 |
1722-07-27 14:32 | CINN: 101.75 | PPR: 108.16 | NTG: 58.37 | CLV: 43.01 | LEM: 35.69 | SPTR: 26.50 |
```

Figure 3

Price Stochasticity

The unpredictability of the ASE Simulator 1722 market is a key feature that makes the simulation both realistic and challenging. Despite the hardcoded initial drift coefficients for each spice, these coefficients can change dynamically throughout the simulation. This variability is introduced through the intruder function, which acts as a stochastic modifier of market conditions.

The intruder function mimics a roulette wheel, dialed at each time interval. While most of the spins result in no change, certain outcomes can cause significant modifications. The log odds of these changes follow a predefined distribution. For example, the probabilities might be configured as follows under a certain setting: the odds of halving the drift coefficient are 3/e, doubling it might occur with odds of 2/e, and there might be a 1/e chance of a systemic drop by 0.1 or 0.2. These changes can show significant movement even on a daily basis and can happen even during normal market conditions. This mechanism replicates short-term overbought and oversold conditions. The Iberians don't have to launch a large-scale naval invasion for Pepper or Clove to significantly drop in price. Regular events such as a sudden distaste for a particular spice or major pepper farmers being taxed more heavily can be incorporated through these stochastic adjustments. Look at Figure 4.

```python
def intruder(ticker, data, market_condition, harddisk, drift):
    probabilities = {
            "Negate": 4,
            "Double": 4,
            "Halve": 5,
            "StaySame": 160,
            "DownBy0.1": 2,
            "UpBy0.1": 4,
            "Revert": 22,
            "BlowUp": 0.004,
            "Crash": 0.002
    }
    if spice_prices[ticker][-1] < 7:
        probabilities['BlowUp'] = 0.013
    if market_condition == "war":
        probabilities["DownBy0.1"] = 5
    if ticker == "CLV":
        probabilities["DownBy0.1"] = 3
    actions = list(probabilities.keys())
    weights = list(probabilities.values())
    chosen_action = random.choices(actions, weights=weights, k=1)[0]

    if chosen_action == "Negate":
        data.loc[data['ticker_symbol'] == ticker, f'{market_condition}_drift'] = -data.loc[data['tick
        #print(f"Negated for {ticker}")
```

Figure 4

There are 3 types of reversibility with respect to changing drift coefficients.

- Type I is Bommel's reversibility. Bommel's reversibility is relatively commoner of the three and is characterized as follows: Upon a coefficient modification, the modified value is in permanent use *until* the wheel is spun to “Revert” at which point the factory drift settings are restored for the ticker involved. The restoration is done through temporary storage which is a copy of all coefficients read in prior to start of simulation. The *intruder* function has access to temporary storage, so there is a 9.56% chance of reverting to its original setting at every time interval. This form of reversibility is critical for replicating precise market behavior. Assume a short oversold period for a certain ticker. Let’s say some Dutch fishermen have begun spreading a false rumor about large imports of sugar from Southern Wales. This can lead to overselling (due to inferred competition) and is captured in the Python code through the wheel settling on “Negate” or “DownBy0.1”. The part where the SGR’s market remains at a negative drift is the duration until which “Revert” is triggered. Upon “Revert”, there is a correction and normal conditions for sugar’s price growth is restored. This clearly introduces entry opportunities during certain dips, however it’s still very unclear how long the dips last due to the randomness.

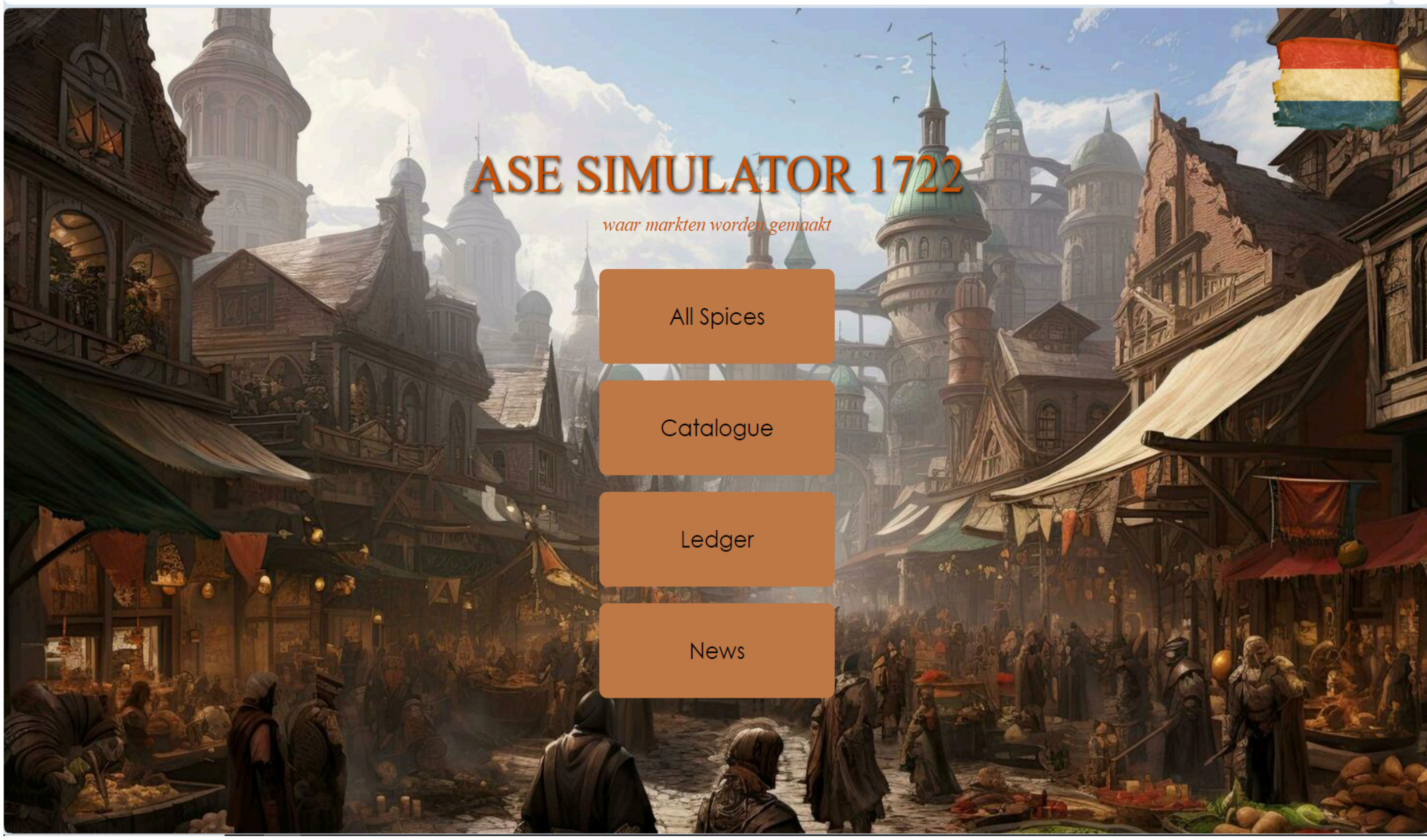
- Type II is Claasen’s reversibility. Whilst Claasen’s is far less frequent than Bommel’s, it is more significant. It is characterized as follows: Upon Type II reversibility’s activation, temporary storage is overwritten with current values. This implies that the “Revert” spin on the wheel cannot restore factory settings anymore. The 9.56% chance of returning to default drifts is taken away. However, Claasen’s is not completely irreversible. It maintains a stochastic possibility of returning to its initial coefficient. A random variable is sampled from mean 1723 and standard deviation 115. We’ll call the sampled value “modular target” or in short, “modutar”. Once t (the time interval, which begins at 0 and increments by 1 for each minute) mod “modutar” equals 0, temporary storage is rewritten with factory settings. This is possible because a second copy called “harddisk” is made with the starter coefficients of all tickers and market conditions. The contents of “harddisk” are written into temporary storage when the clock hits `t%modutar == 0`. It’s crucial to note here that the drifts do not change at these values of t. Rather, only the temporary storage is restored to its original version, meaning a “Revert” would now restore drift to its factory setting. This encapsulates longer periods of anomalous behavior, which is often seen in modern markets where lack of liquidity can cause illogical trends. As noticed these can take longer than a week to correct itself, and can be major trading opportunities.
- Type III is Visser’s reversibility. This is irreversible and any change made through this is permanent for the rest of the simulation. The contents of the harddisk itself are rewritten with the modified drift, so it is virtually impossible to bring back the initial drift coefficient since we never re-access the csv file after beginning simulation. This can mirror events like major trade routes getting permanently disrupted for a certain spice like nutmeg (NTG) due to the end of a political allegiance. The algorithms we write should be capable enough to know that these aren’t temporary oversold or overbought conditions but rather long-term redirection in trends.

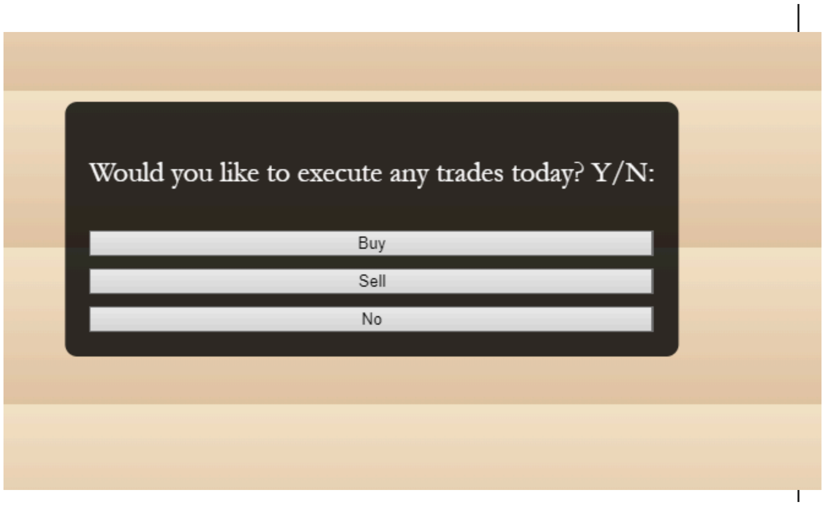
User Interaction and Trading

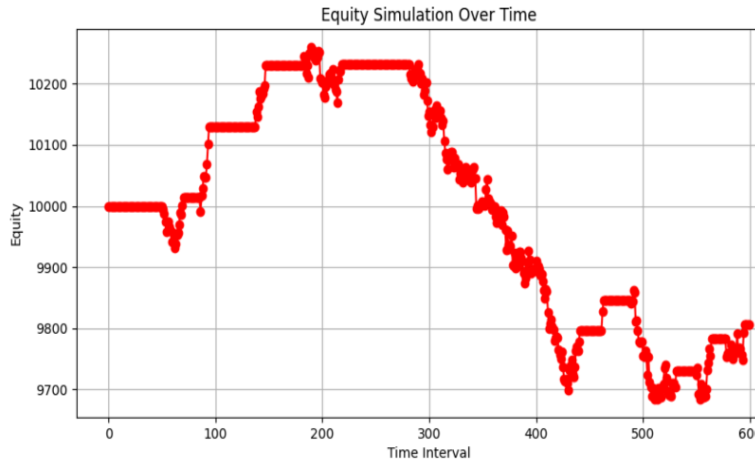
In the simulator, users begin with a bankroll totaling 10,000 guilders. The simulator prompts users to execute trades at the beginning of the simulation and at the end of each trading day, which occurs every 1,500 hours. This allows users to make informed decisions based on the day's market activities. On the interactive web page front end, users can access detailed news and data about each spice, including historical price movements and current odds of war or trade disruptions, before committing to any investment decisions.

When the user navigates to the simulation page, they are prompted to buy shares of various spices. Shares can only be purchased in whole numbers and users can buy as many shares of as many different spices as their bankroll allows. Throughout the simulation, shares can be bought and sold through market orders, allowing for dynamic trading strategies. The default simulation period is set to 12 weeks, from July 22nd, 1722, to September 14th, 1722, providing ample time for users to test and refine their trading strategies. However, this period can be adjusted for longer or shorter simulation runs; so can the speed of the clock.

The frontend web page also features occasional pop-ups predicting the possibility of droughts, wars, or new trade routes in the upcoming weeks. These predictions are not guaranteed and come with a variance element, adding to the realism and unpredictability of the market. At the end of the simulation, users are presented with comprehensive graphs displaying their equity history, along with detailed performance reviews, including percent gains and losses on each spice.







Automated Strategies

Mean Reversion: Bollinger Bands Strategy

The mean reversion strategy revolves around the fundamental principle that, despite short-term volatilities, prices tend to revert to their long-term mean. This concept is rooted in the belief that extreme price movements are often temporary and that prices will eventually return to their average levels. In the simulator, I implemented this strategy using Bollinger Bands, a popular technical analysis tool that consists of a middle band (SMA-20) and two outer bands set at two standard deviations above and below the middle band.

The Bollinger Bands are calculated as follows: the middle band is the 20-day simple moving average (SMA) of the spice prices. The upper and lower bands are set two standard deviations away from the SMA, capturing the expected range of price movements. The buy and sell signals are triggered based on the price crossing these bands. When the price crosses below the lower band, it is considered oversold, and a buy order is executed. Conversely, when the price crosses above the upper band, it is considered overbought, and a sell order is executed. The amount of the bankroll dedicated to each trade is calculated as a fraction of the total bankroll, typically around 75%, ensuring significant but not overwhelming exposure to each trade.

This strategy leverages the assumption that extreme deviations from the mean are rare and that prices will revert to their average levels. However, one of the main weaknesses of this strategy is its vulnerability to prolonged trends. If a price continues to move in one direction for an extended period, the mean reversion strategy can incur significant losses, as the prices may not revert to the mean in the short term. Additionally, the stochastic elements introduced in the simulation, such as the intruder function and varying drift coefficients, add layers of unpredictability, making it challenging to rely solely on this strategy. Figure 7 shows an example of PPR’s price movements that defeats the purpose of the Bollinger Bands mean reversion strategy.

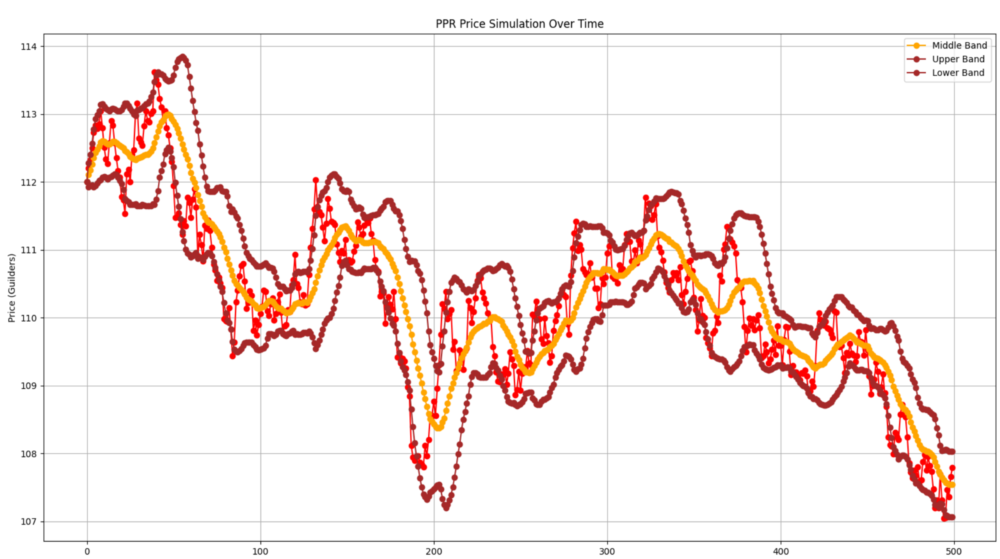
MACD: Momentum Strategy

The Moving Average Convergence Divergence (MACD) strategy is another automated trading strategy implemented in the simulator. The MACD is a momentum indicator that shows the relationship between two moving averages of a spice's price. It consists of the MACD line, which is the difference between the 12-day and 26-day exponential moving averages (EMAs), and the signal line, which is the 9-day EMA of the MACD line. The MACD strategy aims to identify changes in the strength, direction, momentum, and duration of a trend in a spice's price.

In the initial approach, the MACD strategy executed buy signals when the MACD line crossed above the signal line, indicating a bullish trend, and sell signals when the MACD line crossed below the signal line, indicating a bearish trend. This naive approach worked well in identifying short-term momentum shifts but lacked robustness against false signals and market noise.

To enhance the strategy, I incorporated additional filters such as the 200-day EMA and the Relative Strength Index (RSI). The 200-day EMA serves as a long-term trend filter, ensuring that trades are only executed in the direction of the primary trend. For example, buy signals are only acted upon if the price is above the 200-day EMA, indicating an overall uptrend. The RSI, on the other hand, is used to identify overbought and oversold conditions. The enhanced strategy only executes buy signals if the RSI is below 30 (indicating oversold conditions) and sell signals if the RSI is above 70 (indicating overbought conditions).

This enhanced MACD strategy aims to reduce false signals and improve the accuracy of trade entries and exits. However, like any trading strategy, it is not without its weaknesses. The stochastic elements in the ASE Simulator, such as the dynamic drift coefficients and intruder function, can introduce unexpected price movements that may lead to premature exits or missed opportunities. Additionally, the reliance on historical price data means that the strategy may not always adapt quickly to sudden market changes or news events.



Statistics

To evaluate the effectiveness of said automated trading strategies, I conducted extensive testing under various market conditions. Below are the summarized results and insights from these tests.

Bollinger Bands Strategy

Non-Volatile Spices (CINN, CLV, NTG):

- Average equity after 12 weeks: 13,500 guilders
- Number of trades executed: 45 buys, 40 sells
- Average percent increase in equity: 35%
- Success rate: 70%

In these tests, when the Bollinger Bands strategy was applied to non-volatile spices like Cinnamon (CINN), Clove (CLV), and Nutmeg (NTG), the results were favorable. The equity, starting at 10,000 guilders, grew to an average of 13,500 guilders over 12 weeks, with a 35% increase in equity on average. The strategy performed 45 buy orders and 40 sell orders, with a success rate of 70%.

Volatile Spices (LEM, HYS, TUR):

- Average equity after 12 weeks: 9,200 guilders
- Number of trades executed: 55 buys, 50 sells
- Average percent decrease in equity: 8%
- Success rate: 45%

However, when more volatile spices like Lemongrass (LEM), Hyssop (HYS), and Turmeric (TUR) were incorporated, the Bollinger Bands strategy struggled, often ending in a loss. The equity, starting at 10,000 guilders, dropped to an average of 9,200 guilders, showing an 8% decrease. There were 55 buy orders and 50 sell orders, with a lower success rate of 45%.

Wartime Conditions:

- Average equity after 12 weeks: 8,000 guilders
- Number of trades executed: 60 buys, 55 sells
- Average percent decrease in equity: 20%
- Success rate: 40%

During wartime, the Bollinger Bands strategy performed poorly across almost all spices except Saltpeter (SPTR), which saw a minor average equity increase to 10,500 guilders. For other spices, the average equity dropped to 8,000 guilders, a 20% decrease, with 60 buy orders and 55 sell orders, and a success rate of 40%.

Drought Conditions:

- Average equity after 12 weeks: 11,500 guilders
- Number of trades executed: 50 buys, 45 sells
- Average percent increase in equity: 15%
- Success rate: 60%

Surprisingly, during drought conditions, the Bollinger Bands strategy yielded better results due to the predictability and reduced volatility. The average equity rose to 11,500 guilders, a 15% increase. There were 50 buy orders and 45 sell orders, with a 60% success rate.

MACD Strategy

Non-Volatile Spices (CINN, CLV, NTG):

- Average equity after 12 weeks: 14,200 guilders
- Number of trades executed: 50 buys, 45 sells
- Average percent increase in equity: 42%
- Success rate: 75%

For the MACD strategy, non-volatile spices like Cinnamon, Clove, and Nutmeg performed well. The equity grew to an average of 14,200 guilders, a 42% increase, with 50 buy orders and 45 sell orders, and a success rate of 75%.

Volatile Spices (LEM, HYS, TUR):

- Average equity after 12 weeks: 9,800 guilders
- Number of trades executed: 60 buys, 55 sells
- Average percent decrease in equity: 2%
- Success rate: 50%

When applied to volatile spices, the MACD strategy showed mixed results. The average equity decreased slightly to 9,800 guilders, a 2% decrease. There were 60 buy orders and 55 sell orders, with a 50% success rate.

Wartime Conditions:

- Average equity after 12 weeks: 7,500 guilders
- Number of trades executed: 70 buys, 65 sells
- Average percent decrease in equity: 25%

- Success rate: 35%

During wartime, the MACD strategy, like the Bollinger Bands strategy, faced significant challenges, with average equity dropping to 7,500 guilders, a 25% decrease. There were 70 buy orders and 65 sell orders, with a success rate of 35%.

Drought Conditions:

- Average equity after 12 weeks: 10,500 guilders
- Number of trades executed: 55 buys, 50 sells
- Average percent increase in equity: 20%
- Success rate: 65%

In drought conditions, the MACD strategy also performed reasonably well, with average equity increasing to 10,500 guilders, a 20% increase. The strategy executed 55 buy orders and 50 sell orders, achieving a 65% success rate.

Conclusively, it’s quite disheartening to see that the majority of good results in the automated trades occur non-volatile spices under normal market conditions. These spices also have positive initial drift coefficients which actually makes it hard to lose considerable equity on them, unless Visser’s reversibility is activated. The goal moving forward would be to improve user experience by adding more features like statistics on 52-week highs and lows, beta values and other financial metrics, providing the option to trade during days as opposed to only between days and introducing the option to short sell and perhaps even trade options. And simultaneously, more regressive time series analysis would be required to get the numbers up during wartime conditions, in particular for volatile stocks.