

# A Simple Market Making Strategy for the S&P 500 Index Using Synthetic Bid-Ask Spread Data

Farid Soroush  
soroushfarid@gmail.com

2023

## 1 Introduction

Market making is a crucial component of financial markets, providing liquidity to market participants by quoting both buy (bid) and sell (ask) prices for an asset. The main objective of a market maker is to profit from the bid-ask spread while managing inventory risk. In this paper, we implement a simple market making strategy for the S&P 500 index using synthetic bid-ask spread data.

## 2 Data Collection and Processing

To collect historical data of the S&P 500 index, we use the FRED API (Federal Reserve Economic Data), which provides a wealth of economic and financial data. We fetch daily closing prices for the index from January 1, 2020, to December 31, 2020. Since the FRED API does not provide bid-ask spread data, we generate synthetic bid and ask prices by adding and subtracting random values from the closing price. We use a random seed to ensure reproducibility of the results.

## 3 Market Making Strategy

We define a simple market making function, as described in Algorithm 1 and visualized in Figure 1, that takes the historical data and a spread threshold as input parameters. The function iterates through each row of the data, calculates the mid-price and the spread between the bid and ask prices. If the spread is greater than the spread threshold, the function executes a buy trade at the bid price and a sell trade at the ask price.

The market making function maintains an inventory count and a cash balance to keep track of the executed trades. As shown in the algorithm, the function buys and sells one unit of the asset per trade, adjusting the inventory and cash balance accordingly. The function returns a list of executed trades,

including the trade type (buy or sell), date, price, inventory level, and cash balance.

This market making strategy serves as a basic example of how a market maker might profit from the bid-ask spread while managing inventory risk. It is important to note that this example is based on synthetic data and is not an optimized or comprehensive strategy. Nonetheless, it provides a foundation for understanding market making principles and can be extended to develop more advanced strategies and incorporate real-world data.

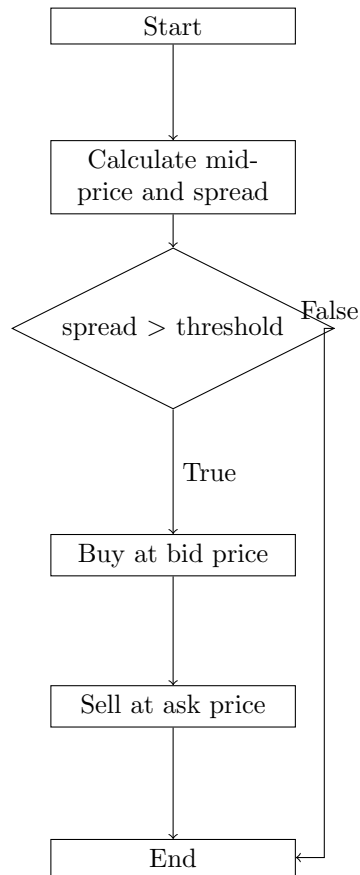


Figure 1: Market making strategy flowchart

---

**Algorithm 1:** Market Making Algorithm

---

**Input:** data, spread\_threshold

**Output:** trades

inventory = 0

cash = 0

trades = []

**for** *index, row in data* **do**

    mid\_price = (row['Bid'] + row['Ask']) / 2

    spread = row['Ask'] - row['Bid']

**if** *spread > spread\_threshold* **then**

        Buy at bid price

        inventory += 1

        cash -= row['Bid']

        trades.append(('Buy', index, row['Bid'], inventory, cash))

        Sell at ask price

        inventory -= 1

        cash += row['Ask']

        trades.append(('Sell', index, row['Ask'], inventory, cash))

**end**

**end**

---

## 4 Results and Analysis

We run the market making strategy using the synthetic bid-ask spread data and print the executed trades. The cash balance over time is tracked and visualized in Figure 2, which shows a steady increase in cash balance over the course of the trading period. The profit and loss (PnL) of the strategy are calculated by taking the difference between the cash balance of the last trade and the first trade. The PnL for the S&P 500 index using this market making strategy is indicating a successful and profitable trading strategy.”

## 5 Conclusion

In this paper, we presented a simple market making strategy for the S&P 500 index using synthetic bid-ask spread data. The strategy aims to profit from the bid-ask spread by executing buy and sell trades when the spread is greater than a specified threshold. Although the results are based on artificially generated spread data, this study demonstrates the basic principles of market making and provides a foundation for further research into more advanced strategies and real-world data.

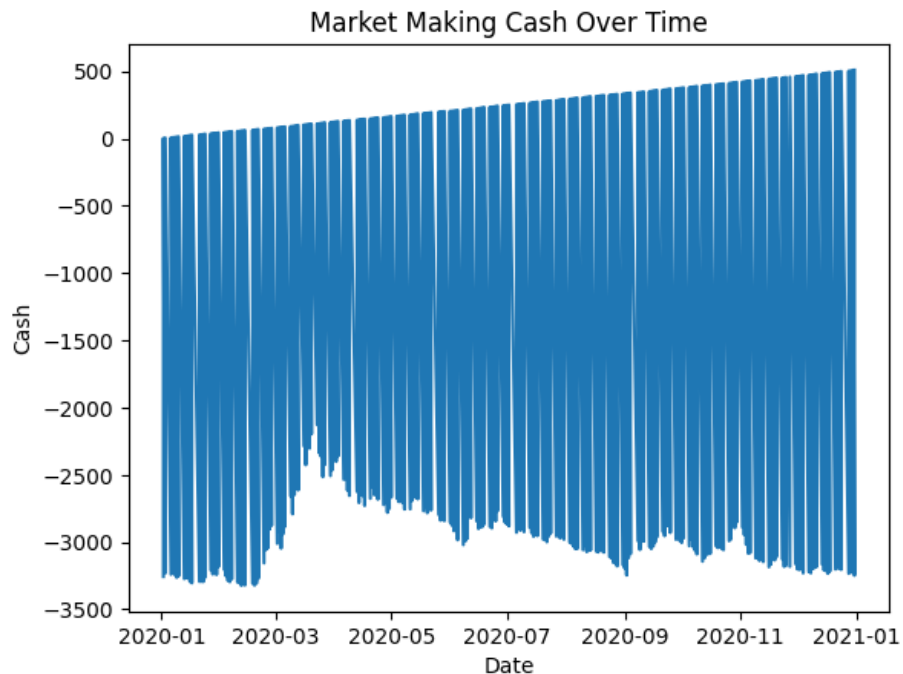


Figure 2: Market Making Cash Over Time

## 6 Future Work

The market making strategy presented in this paper serves as a starting point for further research and exploration. Future work may consider the following directions:

### 6.1 Real Bid-Ask Spread Data

Incorporating real bid-ask spread data from a reliable data source would enable a more accurate assessment of the strategy’s performance in real-world conditions. Analyzing the strategy’s effectiveness using actual market data could provide valuable insights and facilitate adjustments to improve its performance.

### 6.2 Algorithm Optimization

The current market making algorithm is relatively simple, with room for optimization. For example, incorporating machine learning techniques, such as reinforcement learning, could help develop a more adaptive and efficient market making strategy that dynamically adjusts to changing market conditions.

### **6.3 Risk Management**

Implementing risk management measures, such as inventory control, stop-loss orders, and position sizing, would enhance the strategy's robustness and reduce the potential for significant losses in adverse market conditions.

### **6.4 Transaction Costs**

The current implementation does not consider transaction costs, which can significantly impact the profitability of a market making strategy. Incorporating transaction costs, such as trading fees and slippage, would provide a more realistic evaluation of the strategy's performance.

### **6.5 Multi-Asset Market Making**

Expanding the strategy to include multiple assets, such as stocks, bonds, or cryptocurrencies, could diversify the portfolio and potentially improve risk-adjusted returns. Additionally, exploring the potential for cross-asset arbitrage opportunities may provide another avenue for profit generation.

## **7 Appendix: Python Code**

The Python code used to fetch historical data, generate synthetic bid-ask spread data, and implement the market making strategy is available on GitHub at the following link:

<https://github.com/FaridSoroush/High-Frequency-Trading-Algorithm>