Futures Trading Algorithm

By: Murad Haider
m_haider3@u.pacific.edu
2023-02-05

System Architecture/Design

The system architecture consists of a single computer with Windows 10 or higher and NinjaTrader 8. The computer will be connected to a prop firm account, which will allow the user to access the NinjaTrader 8 strategy builder and C# environment. The futures trading algorithm will be designed and run in real-time using historical market data.

The Futures Trading Algorithm will be based on an object-oriented design using UML. The software design will consist of a class diagram showing all classes and their associations, including public and protected members. The main entry/exit condition of the algorithm will be based on the EMA and VMA crosses. These crosses will help determine the direction of the trend and provide signals for entering and exiting trades.

In addition to the EMA and VMA crosses, the algorithm will also incorporate the use of different candle types to reduce the number of trades taken in chop and filter price action. The use of stochastics and the dynamic nature of these indicators will also help to identify potential trades with a higher probability of success.

Furthermore, the algorithm will be designed to run in real-time using historical market data. This allows the trading strategy to be backtested and optimized prior to live trading. By using NinjaTrader 8 and the Rithmic Trader software, the user will have access to a variety of tools for analyzing and testing the performance of the algorithm. The software will be designed to be user-friendly and easy to navigate, making it accessible to traders of all skill levels.

Hardware, Software and System Requirements

The hardware requirements include a computer with Windows 10 or higher. No specific CPU, RAM, or GPU is required. The software requirements include the NinjaTrader 8 64-bit software, which includes the Rithmic Trader software for monitoring the prop firm account. No special connections are necessary for this project besides the connection of the prop firm account to NinjaTrader8 which is simply based on logging in using your Prop Firm credentials.

External Interfaces

No external interfaces are required for this project.

System Design Diagram

Preparation

This is the first step in the process of developing the Futures Trading Algorithm. In this phase, the necessary programs will be lined up and organized on the Windows operating system. The user will download and install NinjaTrader 8, the prop firm account, and external NinjaTrader 8 candles such as Ninzarenkos and Unirenkos. These candles are a free download from the NinjaTrader library and will be used to help filter prices better than regular candles.

Development

In the second phase of the project, the user will work on the algorithm. This phase will include front testing and backtesting the algorithm along different months in the past year. The

use of multiple months will help avoid form fitting and ensure that the results are accurate. The user will use the NinjaTrader 8 strategy builder and C# environment to design and run the algorithm.
After Code Production

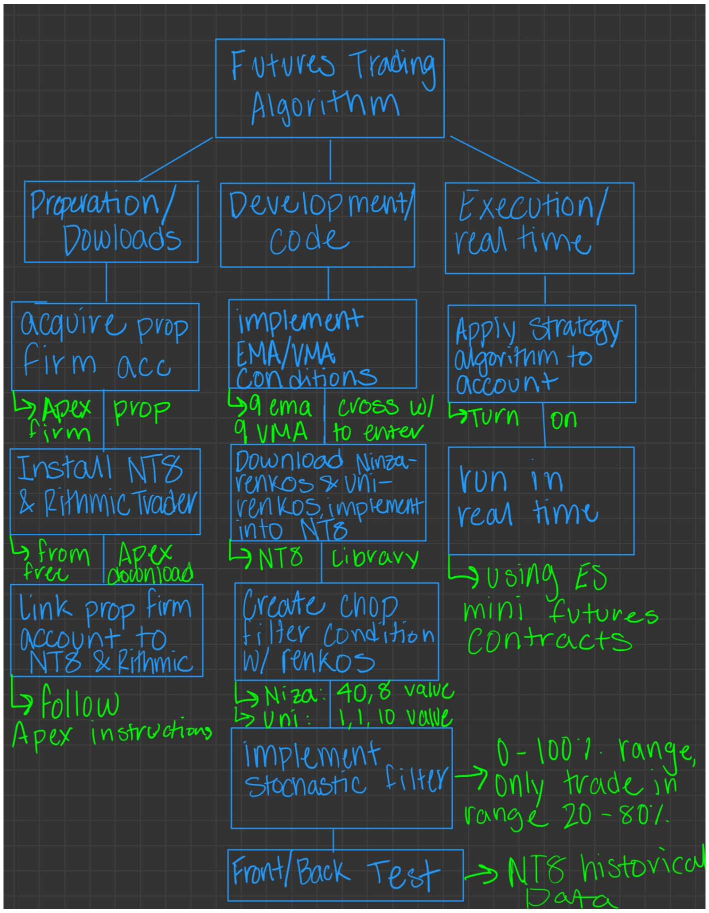
Execution

The final phase of the project will involve running the algorithm in real-time on the market. The user will watch the performance of the algorithm based on the conditions and code provided. The user will monitor the prop firm account to keep track of the trades and adjust the algorithm as needed.

Design Considerations

In designing the classes and their interactions, the focus was on creating a robust and flexible algorithm that could effectively filter out chop and catch high-frequency trends. To achieve this, the EMA and VMA crosses were used as the main entry/exit conditions, while the stochastics and dynamicness of the indicators were meant to reduce the number of trades taken in chop and filter price action. For example we will only be trading when the stochastic filter is between a range of 20-80%.

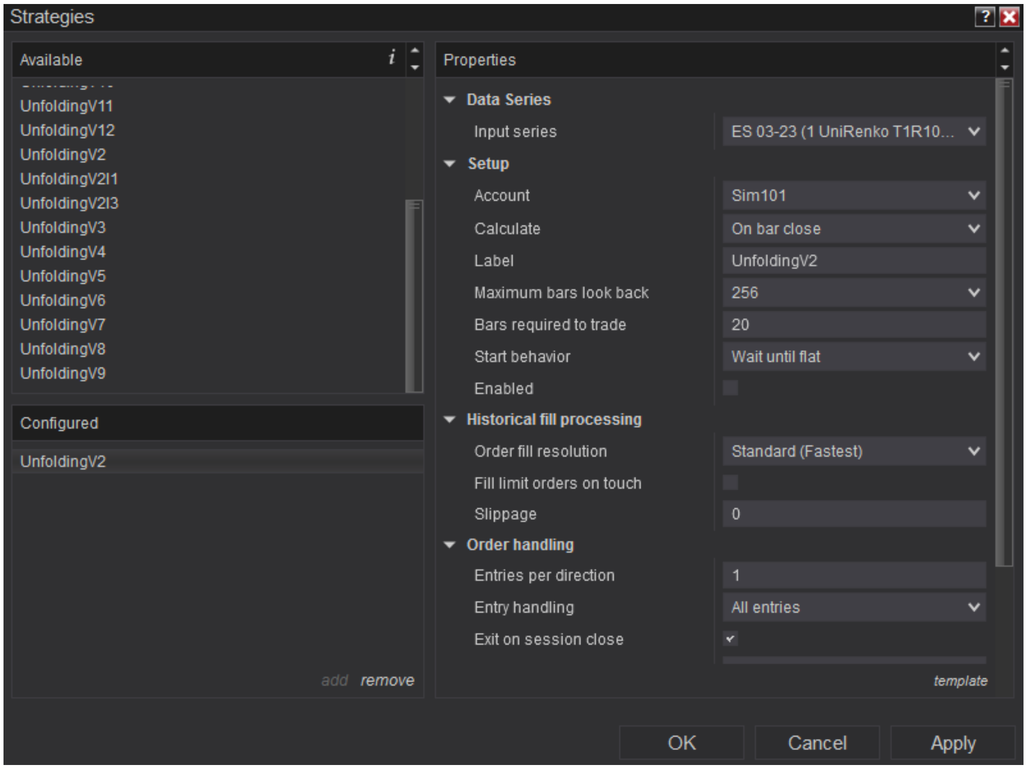
Diagram



Futures Trading Algorithm
Preparation/ Dowloads
Development/ code
Execution/ real time
acquire prop firm acc
↳ Apex firm | prop
Install NT8 & Rithmic Trader
↳ from free | Apex download
Link prop firm account to NT8 & Rithmic
↳ follow Apex instructions
implement EMA/VMA Conditions
↳ 9 ema 9 VMA | cross w/ to enter
Download Ninza-renkos & Uni-renkos, implement into NT8
↳ NT8 | Library
Create chop filter condition w/ renkos
↳ Niza: | 40, 8 value
↳ Uni: | 1, 1, 10 value
implement Stochastic filter
→ 0 - 100% range, only trade in range 20 - 80%.
Front/Back Test
→ NT8 historical Data
Apply Strategy algorithm to account
↳ Turn | on
run in real time
↳ using ES mini futures contracts

Conclusion

The Futures Trading Algorithm project will utilize an object-oriented design using UML and a computer with Windows 10 or higher and NinjaTrader 8. The algorithm will be based on different EMA crosses and stochastic conditions to attempt to filter out chop and catch high-frequency trends. The project will consist of three phases: Preparation, During Process, and After Code Production. The goal of this project is to create a robust and reliable futures trading There is no interface design creation necessary. Users will only need to log in and turn on the algorithmic strategy and it will do the rest. This is done very easily through NinjaTrader8, the image below shows how you can apply a strategy to your charts. By double clicking on one of the available strategies on the left it will show all the details on the right. This is where the user selects the input series in our case it will be ES Futures and selects the correct account. All other parameters come from the development stages of the algorithm. Then once the algorithm is selected the user simply presses apply and can watch the algorithm work.



Available:
UnfoldingV11
UnfoldingV12
UnfoldingV2
UnfoldingV2I1
UnfoldingV2I3
UnfoldingV3
UnfoldingV4
UnfoldingV5
UnfoldingV6
UnfoldingV7
UnfoldingV8
UnfoldingV9

Configured:
UnfoldingV2

Properties

Data Series
Input series: ES 03-23 (1 UniRenko T1R10...

Setup
Account: Sim101
Calculate: On bar close
Label: UnfoldingV2
Maximum bars look back: 256
Bars required to trade: 20
Start behavior: Wait until flat
Enabled: □

Historical fill processing
Order fill resolution: Standard (Fastest)
Fill limit orders on touch: □
Slippage: 0

Order handling
Entries per direction: 1
Entry handling: All entries
Exit on session close: ☑

Glossary of Terms:

EMA - Exponential Moving Average: A type of moving average that gives more weight to recent price data, making it more responsive to price changes.

VMA (Volume-Weighted Moving Average) - A type of moving average that gives more weight to prices with higher trading volume.

Stochastic: A technical analysis indicator that compares a stock's closing price to its price range over a set period of time.

Renko: A type of charting technique used in technical analysis that filters out price noise and focuses on price trends.
NinjaTrader: A trading software platform used to trade stocks, futures, forex, and options.
Prop firm: A firm that provides traders with access to a trading account and proprietary software to trade in the financial markets, typically with the goal of generating profits for both the trader and the firm.
Front testing: The process of testing a trading strategy on historical data to determine its potential profitability.
Back testing: The process of testing a trading strategy by examining its performance on historical market data.
Chop: Refers to the short-term fluctuations in market prices that can make it difficult to identify meaningful trends.
High-frequency trends: Refers to rapid price movements in a market over a short period of time.

References

NinjaTrader 8. (n.d.). Retrieved February 5, 2023, from https://ninjatrader.com/support/helpGuides/nt8/NT%20HelpGuide%20English.html?stochastics.htm

Unirenko Universal Renko BarType 8. NinjaTrader Ecosystem. (2022, February 14). Retrieved February 5, 2023, from https://ninjatraderecosystem.com/user-app-share-download/unirenko-universal-renko-bartype-8/

Apex trader funding. Apex Trader Funding - Funded Trader Program. (n.d.). Retrieved February 5, 2023, from https://apextraderfunding.com/

Home. LizardIndicators. (2022, December 7). Retrieved February 5, 2023, from https://www.lizardindicators.com/