

Sample Report

Executive Summary of the proposed strategy

The proposed strategy makes use of the MACD indicator for HSI futures trading. It takes all the ticks into consideration. The 30-second EMA of HSI futures prices and the 60-second EMA of HSI futures prices are calculated to construct the MACD. The 45-second EMA of the MACD will be the signal line. Several measures are adopted to avoid losses due to short term fluctuation. Cutloss is also implemented to limit the maximum loss in each trade.

Description of the trading idea

This trading strategy is written based on the concept of "moving average convergence divergence". The moving average convergence divergence (MACD) is a momentum indicator that shows the relationship between two exponential moving averages (EMA) of prices. It is calculated by subtracting the "slower" EMA from a "faster" EMA. A "signal line", which is the EMA of the MACD, is then plotted on top of the MACD, functioning as a trigger for buy and sell signals.

If the MACD rises above the signal line, it gives a bullish signal and this triggers a long signal for the asset. On the other hand, if the MACD drops below the signal line, the indicator gives a bearish signal and triggers a short signal for the asset.

Methodologies and trading logics

For this strategy, the MACD is going to be calculated by the 30-second EMA of HSI futures prices and the 60-second EMA of HSI futures prices. The signal line will be the 45-second EMA of the MACD. Every tick with tick time in the trading hours will be analyzed. When there is a signal from the MACD (either long or short), trades will be performed if the net position is 0 or if the signal has been reversed. The absolute position will be limited to 1.

The MACD makes use of the fact that different EMA has different sensitivity towards price change. When the price is going up, since the shorter period EMA is more sensitive to price change than the longer period EMA, the shorter period EMA will rise more quickly than the longer period EMA. The MACD line will then turn up. Conversely, when the price is going down, the shorter period EMA will drop more quickly than the longer period EMA, and therefore the MACD line will tend to go down. In conclusion, the MACD line could possibly reflect the recent market trend of the price of an asset.

The signal line can be considered as a confirmation of the price trend. Because the MACD itself is more sensitivity to changes than its EMA, if the MACD crosses the signal line from below, this means that the trend of the MACD is going upward. Therefore, the different between the two EMAs is getting larger, which means the price of the asset is likely to have an upward trend. Thus, a long signal is confirmed. Similarly, if the MACD line crosses the signal line from above, the trend of the MACD is going downward. Thus, the different between the two EMAs is getting smaller, which means the price of the asset is likely to have a downward trend and a short signal is confirmed.

Implementation details

The core of the strategy is the determination of the MACD signal. To check the MACD signal, the EMAs of the HSI futures need to be calculated first. The price of the first snapshot of each second is used to calculate the EMA. The recursive formula for finding the N -second EMA in period t is:

$$EMA_t = P_t \times \frac{2}{N+1} + EMA_{t-1} \times \left(1 - \frac{2}{N+1}\right),$$

where t are positive integers and the initial EMA, EMA_0 , is the simple average of the first N prices.

The formula for the MACD in period t is:

$$MACD_t = EMA_{t,30} - EMA_{t,60},$$

where $EMA_{t,N}$ is the N -second EMA of the futures prices in period t . The confirmation signal will then be calculated using the recursive formula for the EMA and 45 past data point of the MACD.

The program will analyze every tick of the data. First of all, some parameters will be set before the tick data enter the program. Three parameters are needed, namely *time_frame*, *threshold* and *cutloss*. *time_frame* is a time object while *threshold* and *cutloss* are positive constants. Their uses will be discussed later.

After setting the parameters, the next step is to handle the tick data. For every tick, the program will decide if the signal is the first tick of the day. If yes, then three variables called *futures_code*, *last_trade* and *last_check_dt* are going to be set. *futures_code* is the code of the futures that the program will trade. *last_trade* is the date and time of the futures' expiration. The expiration date is scheduled by the HKEX while the time is set before the settlement to indicate the last analysis (and perhaps the last trade) of the spot month futures contract. *last_check_dt* is the time for the program to

perform a “check”, which will be described later. *last_check_dt* could be set to 0 (in term of time object) at the beginning. If the tick is not the first one of the day, the program will directly go to the next step. Next, the program will check if the tick time is within the trading hours and whether the product code of the tick matches with the *futures_code*. If either the tick time is before the trading hours, or the product code does not match, the program will go back and get the next tick. If the tick time is after the trading hours, the trading program will be stopped as the market has closed. Only if the tick time is within the trading hours and the product code matches the *futures_code*, the program can proceed to the next stage.

The next step is to make sure a sufficient time has been passed since the last MACD signal checking. To reduce the possibility of getting reversing signal just because of some short term fluctuations instead of the price trend, a minimum “waiting time” will be implemented. Recall that there is a variable called *last_check_dt* and a parameter called *time_frame*. If the tick time is larger than or equal to the sum of *last_check_dt* and *time_frame*, then the program can go on to check the MACD signal. Otherwise, the program should go back and get another tick.

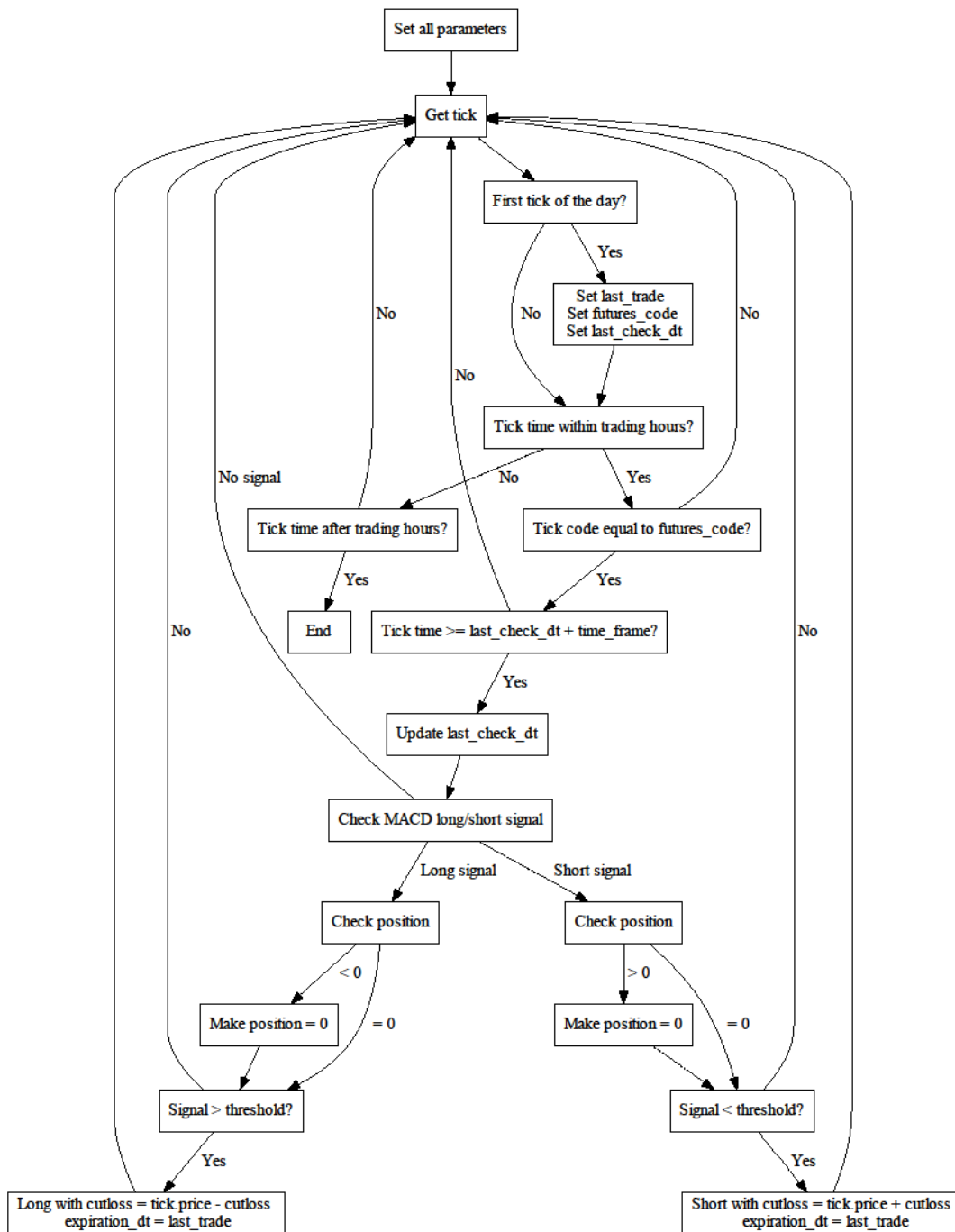
After that, the program will get the MACD long/short signal and then check the position of the account. If there is a long signal and the account is currently taking no position, the program will go on directly. However, if there is a long signal but the account is in short position, then a trade will be initiated to close the position before continuing to the next step. The case is similar for short signal. A trade will be performed to close the position if the account is initially in long position.

Recall that two parameters called *threshold* and *cutloss* have been set. For the case of a long signal, *threshold* is the minimum limit of the signal such that the signal can initiate a long position. If the signal is smaller than or equal to *threshold* (i.e. the result of the MACD minus the signal line is smaller than or equal to *threshold*), the program will not take a long position but go back to get the next tick. For the case of a short signal, the negative of *threshold* is the maximum limit such that the signal can initiate a short position. If the signal is larger than or equal to *threshold* (i.e. the result of the MACD minus the signal line is larger than or equal to the negative of *threshold*), the program will go back to get the next tick without performing any trade action too. The “*threshold* requirement” is important because it could reduce the chance of taking a position while the signal is just a short term fluctuation and does not reflect the market trend.

If the signal can satisfy the “*threshold* requirement”, a trade will occur to take the position as the signal suggested. To prevent a huge loss due to sudden and rapid movement of the futures price, *exit price* will be set. The formula for the *exit price* is:

$$exit\ price = \begin{cases} tick_price - cutloss & \text{if long} \\ tick_price + cutloss & \text{if short} \end{cases}$$

where $tick_price$ is the price of the tick when the trade performed. Lastly, the program will need to update the expiration date with $last_trade$ to make sure that the position is closed before the expiration date and time of the futures. After performing the trade and the updates, the program will go back to the initial stage and take the next tick for the analysis. The whole process will be repeated until market close.



Team background

1. Chan Siu Ming

I am a student who majored in quantitative finance. I have 1 year of algo trading experience and I have taken some courses about programming before. I am good at data analysis and financial mathematics.

2. Wong Tai Man

I am a student from the statistics department. I have not tried algo trading before. However, I am good at data analysis and statistical modelling. Moreover, I have studied about technical analysis too.

3. Lee Ka Yan

I am a year 3 student who is studying BBA. I am experienced in traditional trading but green in algo trading. I have studied lots of business and finance related subjects, including accounting, business economics, international finance, etc.

4. Lau Wai Shan

I am a year 3 physics student. I am not very familiar with algo trading but I am interested in financial mathematics and financial engineering. I have some basic knowledge in programming and mathematical modelling.