

# ViperGrid EA

Viper Trading

Recoded and enhanced by: TradingViper

Base code taken from : FOREXflash (forexfactory)

Original based on: WilliamsA (forexpeacearmy)

Background: This is a Martingale system which will always win given unlimited trade capital/margin. This system likes trending markets and the system "Death Trade" occurs in sideways markets where trades are accumulated, never liquidated, until margin is exhausted. Therefore, the system should be traded with the largest capital possible, the maximum leverage available, AND the absolute smallest position size allowed.

Modifications: I have added a few additional entry filters to try to enter the market when it is trending or most likely to be trending, lessening the chance of hitting the death trade. More importantly, I have added code that tries to determine if we may be caught in the "Death Trade Trap" and enters a "liquidation" mode, in which no further open orders are placed, pending orders are cancelled, and it attempts to liquidate all open positions for a profit. The logic is that if we are whipsawing back and forth, we should be able to close open positions on both sides for a profit, at the right points of the whip. Most likely this will not actually be possible, but even if we can close "most" positions for a profit, we will minimize the losses. Here's to hoping :-)

## **Grid Definition Parameters**

These parameters control the basic Grid structure - from an initial entry or pivot price, a set of orders is placed. All orders are initially placed at the same time in a hedging fashion. More orders are placed as price moves in such a way that regardless of market direction, the trade set will be profitable.

### **Levels**

The number of price levels at which orders placed above the pivot (buy stops) and below the pivot price (sell stops).

### **Increment**

The distance (in pips) between the price levels. The same distance from the highest level is used for both the take profit and the stoploss for the orders in the opposite direction.

### **IncrementAdjustForTpSl**

An adjustment (in pips) to the take profit stoploss levels. To remain profitable, should be  $\geq 0$ .

### **IncrementAdjustFirstLevel**

An adjustment (in pips) to the first level only. Can be positive or negative. All other levels are unaffected and are still spaced by the Increment value.

### **AutoIncrementPercentATR**

Percentage (as integer) of daily ATR used to automatically compute the Increment (if it was equal zero). The increment is computed in such a way that half the grid size will equal the percentage of ATR defined. Using a value 50 will make each half of the grid equal to half the average ATR or in other words the entire grid equal to one ATR.

## **Money Management and Risk Control Parameters**

These parameters control the money management and risk control. They control the position size for the initial trade, which can be multiplied in subsequent positions. This strategy will ALWAYS win given you have enough capital (margin) and a long enough period of time. Therefore, SMALL is BEAUTIFUL. The algorithm will always use the min lot size allowed by your broker unless you choose to risk a percentage of margin or set MinLotSize.

### **ProfitFactor**

This is an integer multiple of the increment used to determine the desired profit from each trade set. The larger the number is, the more aggressive the strategy is and the more likely the "Death Trade" will occur. A value of 1 is pretty conservative, a value of 2 is moderately aggressive (the value used by original developer), a value of 3 is aggressive, and any value greater than 3, and you are simply crazy.

### **PercentToRisk**

This is the percentage of your available free margin to risk on the initial position for each grid level. It is a whole percentage, so 1.0 represent one percent, 1.1 represents 1.1 percent, etc. Use 0 to disable and revert to min lot size allowed.

**MinLotSize**

This is the minimum position size for each position in the grid. Later, as more positions are opened, multiples of this may be used. Setting this to zero will use the min size allowed by the broker (recommended).

**EntireSetTakeProfit**

This is an early profit exit strategy. If set to a positive number (e.g. +200), the algorithm will close / delete all open positions if the intratrade profit in dollars is equal to or greater than the value set.

**EntireSetStopLoss**

This is an early bailout strategy. If set to a negative number (e.g. -200), the algorithm will close / delete all open positions if the intratrade drawdown in dollars is equal to or greater than the value set.

**Low-Water Mark risk mitigation efforts**

This is the first of two "Death Trade" mitigation strategies. If any of the Low-Water parameters get triggered, the software will continue to trade as normal but begin decreasing the ProfitFactor parameter (above). This will make the strategy less aggressive, but still remain profitable. Eventually, if the water rises to high, the strategy will only open enough new positions to prevent a loss. Each can be turned off with a value of 0.

**MaxPercentMarginProfitAdjust**

If our total margin consumed is greater than this parameter value, low-water mitigation is triggered.

**MaxLotsAdjust**

If our total lots (open and pending) is greater than this parameter value, low-water mitigation is triggered.

**MaxOrdersAdjust**

If our total number of orders (open and pending) is greater than this parameter value, low-water mitigation is triggered.

**High-Water Mark risk mitigation efforts**

This, like the Low-Water mark parameter, is a "Death Trade" mitigation strategy. This time, if any of the High-Water parameters get triggered, we take emergency procedures. The strategy stops opening new positions, deletes all pending orders, and begins closing strategic positions which are at least a little profitable (at least X increment of pips). Each can be turned off with a value of 0.

**MaxPercentMarginLiquidate**

If our total margin consumed is greater than this parameter value, high-water mitigation is triggered.

**MaxLotsLiquidate**

If our total lots (open and pending) is greater than this parameter value, high-water mitigation is triggered. Setting to zero negates it's use.

**MaxOrdersLiquidate**

If our total number of orders (open and pending) is greater than this parameter value, high-water mitigation is triggered.

**StealthMode**

If true, EA hides the real stoploss values from the Broker. It still places a stoploss with the trade, but a full increment away from the real value. This limits the ability of the broker to 'run' the price to your stops or spike the spread trigger your stops.

**Entry Control Filter Parameters**

These parameters control when a new trade set (for the grid) can be initiated. A trade set includes all orders opened on the grid levels until they are all closed.

**INCLUSIVE FILTERS**

Each one that is turned on MUST pass the filter to start a new trade set.

**MinHourToStart**

This is the min time hour that a new trade set can be started. If the current hour (BROKER TIME) is greater than or equal to the time specified, the trade is allowed to begin. Use a value of -1 to not use a min start time filter.

**MaxHourToStart**

This is the max time hour that a new trade set can be started. If the current hour (BROKER TIME) is less than or equal to the time specified, the trade is allowed to begin. Use a value of -1 to not use a max start time filter. These two parameters create a "tradable time".

**InvertTimeWindow**

The two parameters above create a tradable time window in which trading is allowed. If this parameter is set to true, it changes the definition of the defined time window to one which cannot be traded. Trading will be allowed only outside the time window. This is useful to define a time period in which you want to avoid trading.

### **MaxSetEntryCount**

This is the number of trade sets allowed to start and complete. If a time window is defined, exiting that time window will reset this count. This way a user can set a time window and allow only a set number of trade sets every time we enter, that allowed time window (e.g. 1). If set to zero, there are no counting restrictions. As long as other entry control parameters do not inhibit trading a new trade will be started.

**TradeSunday, TradeMonday, TradeTuesday, TradeWednesday, TradeThursday, TradeFriday**  
Set the days of the week to allow trading - this is BROKER TIME, not local time. The defined time window (if set) will only be active on the days that allow trading.

### **MaxSpreadToOpen**

This filters trade set starting by current spread. The trade will not start until the current spread of the current pair is less than or equal to the current spread. Once established, all future open positions are placed using the same spread value used to initiate the grid so all future orders be placed on the grid with the same take profit and same stop-loss. If the spread is vastly different at the desired time of closure than at the start, all positions will still be closed once one has been closed. Profits may be skewed however.

### **EXCLUSIVE FILTERS**

Any single one of the following filters that are turned on MUST pass the filter to start a new trade set, e.g if all the following filters are set on then either the volatility filter OR the trend filter OR the range filter must pass. It requires only ONE to pass to enter the trade set.

#### **UseVolatilityFilterIRegr**

If set true, trades will only be taken if the current prices of CURRENT CHART TIMEFRAME exceed volatility thresholds - definition of volatility based on iRegr indicator. Price must exceed bounds in the direction.

#### **UseVolatilityFilterBB**

If set true, trades will only be taken if the current prices of CURRENT CHART TIMEFRAME exceed volatility thresholds - definition of volatility based on Bollinger Band width as a ratio to ATR.

#### **UseTrendFilter**

If set true, trades will only be taken if the current prices of CURRENT CHART TIMEFRAME appear to be trending based on some standard indicators.

#### **RangeFilterPeriod**

Number of bars of current chart used to compute the average range. If the RangeFilterPeriod equals zero, the filter is turned off.

#### **RangeFilterThreshold**

The limit for which the average true range (ATR) must be greater than to enter a new set. It is defined here in Pips. If it is set to zero, then logically every possible computed ATR will be greater, effectively turning it off.

### **Visual / Audio Control Parameters**

These variables control the look and feel of the user interface. They do not change the functionality of the software.

#### **Verbose**

If set to false, the software only logs when critical events occur, like new trades. If set to true, the logs will get large quickly.

#### **AlertsEnabled**

If set to true, some events will trigger alerts - starting a new trade set or having illegal inputs for instance.

#### **ShowPriceLevels**

If true, the EA will draw lines on the chart for each price level, including the initial pivot price. It will draw them using the colors defined below.

#### **PivotColor, BuyColor, SellColor**

These are strings but they must be legal single word web colors, two word colors were not implemented (have mercy on my fingers). For example, all the basic colors (e.g. Red, Blue, Purple, Navy...) including many one word exotic colors have been implemented (e.g. Tomato, Cornsilk, Wheat...). Each one must start with a capital letter and contain only one capital letter since it is a single word, e.g. "DarkBlue" was not implemented, spite it being a legal web color.