

# Backtesting with 99% modelling quality in MT4

## Why backtest with 99% modelling quality?

Because it is far more accurate than the widely employed 90% modelling quality approach that uses 1M data and the MT4 strategy tester fractal interpolation option.

It is unfortunately quite easy to create EAs that inadvertently exploit the fractal interpolation algorithm to generate grossly unrealistic profits in backtests. They are typically scalpers that close large numbers of trades with less than 20 pips of profit or loss. Most perform far worse, or are not profitable at all, in live trading.

Many commercial EAs are sold on the basis of these 90% modelling backtests, and the forex forums are littered with disappointed comments from people who did not see the same results when forward testing or trading live.

Because most MT4 users know that something is seriously wrong with 90% modelling quality, a huge amount of energy goes into forward testing EAs that were never destined to be profitable. Forward testing is usually still a necessary final step before committing an EA to live trading, but it is my hope that, by describing this method, traders will be able to focus more productively on truly profitable strategies.

## How to get 99% modelling quality in the MT4 strategy tester

### Obtain tick data

#### Logging tick data

The best quality tick data comes from your broker's live server, but unfortunately most brokers don't supply it. The reason you need this is that each broker has different data feeds and applies different tick filtering.

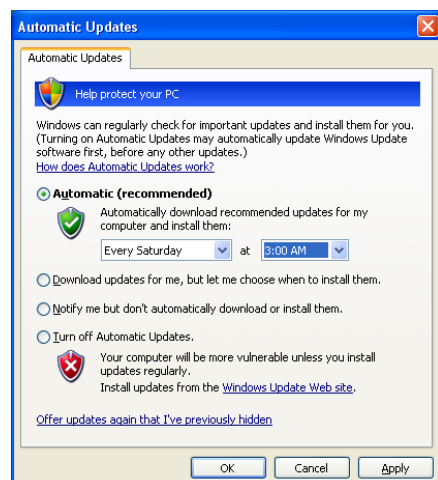
This logging EA will record ticks against whatever currency pair you attach it to. The timeframe doesn't matter. [http://www.lightpatch.com/forex/mt\\_yahoo/TickLoggerForFXT.mq4](http://www.lightpatch.com/forex/mt_yahoo/TickLoggerForFXT.mq4)

It stores log files in experts\files\AAABBB\_yyyyymmdd\_ticks.csv, where AAABBB is the currency pair that you are logging and yyyyymmdd is the date you started.

Over time you will get a number of tick log files with different dates. They can be merged into one file using a Windows XP DOS command like this:

```
copy AAABBB_yyyyymmdd_ticks.csv + AAABBB_yyyyymmdd_ticks.csv + AAABBB_yyyyymmdd_ticks.csv  
AAABBB_ticks.csv
```

Note that if you're serious about logging ticks (and more importantly, live trading) then you need to schedule windows updates for only on the weekend.



## You want to backtest using tick data now, but you have no data?

A less accurate alternative is to obtain tick data from a supplier, but the advantage is that an existing large historical record exists.

Here is a free source of tick data <http://ratedata.gaincapital.com/>. The quality may be suspect.

This script will convert the Gain files into FXT format

[http://www.lightpatch.com/forex/mt\\_yahoo/GainTicks2fxt.mq4](http://www.lightpatch.com/forex/mt_yahoo/GainTicks2fxt.mq4)

Drop it on the chart timeframe that you want the FXT file to be generated in, then copy the AAABBBnn\_0.fxt file from experts\files to tester\history

Here you can pay for tick data: [www.disktrading.com](http://www.disktrading.com). Presumably this would be good quality but I haven't tried them.

## Converting tick data into the strategy tester FXT history file format

Store this Metaquotes converter (simple\_csv2fxt.mq4) in experts\scripts and compile: <http://codebase.mql4.com/516>. It requires this header file (FXTheader.mqh) to be stored in experts\include: <http://codebase.mql4.com/515>

You will get a "Function ReadAndCheckHeader is not referenced" warning, which can be ignored because the function exists in the header file but is not used.

To run the script, first you open the chart for the timeframe and currency pair that you want to test. You then drop the script onto the chart and enter a filename of the logged ticks. This will look like AAABBB\_yyyymmdd\_ticks.csv. The script produce a filename like AAABBB30\_0.fxt in experts\files, The 30 is for 30M timeframe, and the 0 is for tick level. This file then needs to be moved to tester\history. Copy over the existing file if it's there.

## Running the backtest on real ticks

Select a currency pair and timeframe for which you have generated an FXT file, uncheck "recalculate" and observe the results. Likewise for optimisation. The report should show a modelling accuracy of 99%.

## Which EAs should be backtested using this method

- Any EA which frequently closes trades with less than 20 pips of profit or loss is likely to show a very different result using the 90% modelling quality that comes from fractal interpolation.
- Any EA which could enter and exit within the same 1M candle.

An explanation of why this is so is further down in this document.

## Why is it only 99% modelling accuracy?

The figure of 99% is a notional one, but even a backtest done with live data from your own broker is certainly not 100% accurate. Here are some factors that affect the accuracy of backtests:

- The EA start() function only receives a tick if it has completed processing the last one. It will therefore skip over ticks where the broker's server is responding slowly, or the internet connection or PC is too slow.
- Backtesting assumes perfect fills at the Bid or Ask, and this is not always the case in live trading.

## How does the MT4 strategy tester perform backtests?

### The mechanism

A backtest should as much as possible try to simulate what would have happened if you traded live in the past. When you trade live, the start() function in your EA is executed every time a tick arrives, regardless of the timeframe to which your EA is attached. So the most accurate backtest will read historical ticks from a file and submit them one by one to the start() function.

The strategy tester has three methods of repeatedly supplying tick data to the start() function, each with different levels of backtesting accuracy. The main reason for this is that the most accurate is also the slowest.

We will focus on the “fractal interpolation of ticks” method, which is the slowest and most accurate model. A procedure for backtesting from 1M data using this method is here:

<http://www.forex-tsd.com/attachments/expert-advisors-metatrader-4/16102-phoenix-2007-new-thread-modellingquality.pdf?d=1160306256>

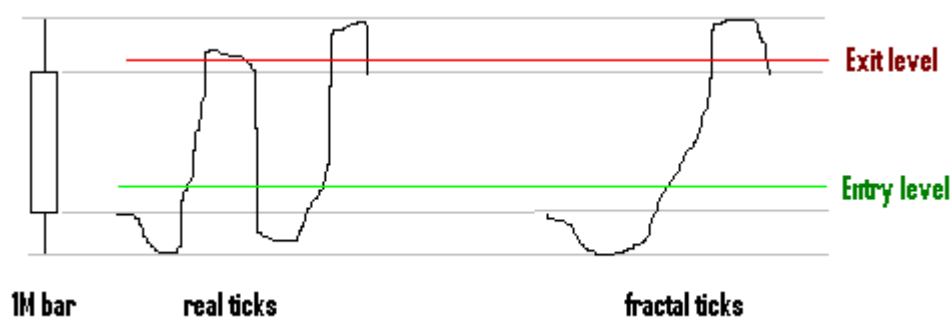
Here's what happens when you perform a 90% modelling backtest as described above, which occurs when you check “recalculate” in the strategy tester window and have sufficient data in the 1M timeframe and the timeframe required by your EA.

- The MT4 strategy tester reads the 1M data file stored in the history folder
- It applies an algorithm called fractal interpolation to generate ticks within each 1M bar. If we are testing EURUSD on a 30M timeframe it will store these in tester\history\EURUSD30\_0.fxt, where 30 is the timeframe, and 0 means fractal interpolation.
- It then submits ticks from this file to the start() function in your EA

### Flaws in the 90% modelling fractal interpolation approach

#### Anything can happen in a 1M candle

The fractal interpolation algorithm attempts to guess what happened in a 1M candle by simply using the 1M Open/High/Low/Close data, and clearly this may not be anything like what really happened. The figure below gives just one situation



If your EA has entry and exit levels at the point shown, then you will get two trades on live data, but only one in the simulation. So if your EA has any possibility of entering and exiting a trade within 1 minute, then fractal interpolation will give highly inaccurate results. And remember that 1M candles frequently have a range of 50 pips during news announcements.

#### Fractal interpolation supplies ticks with larger jumps than reality

The following two graphs show the difference in distribution of tick jumps between live and fractal interpolation

[http://www.lightpatch.com/forex/mt\\_yahoo/USDJPY%20strategy%20tester%20tick%20movements%2027.7.2006%20to%207.8.2006.GIF](http://www.lightpatch.com/forex/mt_yahoo/USDJPY%20strategy%20tester%20tick%20movements%2027.7.2006%20to%207.8.2006.GIF)

[http://www.lightpatch.com/forex/mt\\_yahoo/USDJPY%20tick%20movements%2027.7.2006%20to%207.8.2006.GIF](http://www.lightpatch.com/forex/mt_yahoo/USDJPY%20tick%20movements%2027.7.2006%20to%207.8.2006.GIF)

About 90% of tick movements on a 1 week record of live USDJPY data (see above) were 1 pip. This means that the takeprofit level in an EA would have been the exact value used in the order exit 90% of the time.

In contrast, the fractal interpolation of ticks from 1M data delivered only 68% of ticks with a 1 pip movement. This means that the price could have exceeded the takeprofit value 32% of the time, delivering a higher profit. In a scalper that may take hundreds or thousands of profits of only a few pips, this grossly inflates the estimates of total profit.

### **Bypassing fractal interpolation to supply historical ticks for 99% modelling**

The MT4 strategy tester reads tick data from files in the tester\history folder of the format AAABBBnn\_0.fxt, where AAABBB is the currency pair, nn is the timeframe, and 0 means that the file was created using fractal interpolation. This 0 = fractal interpolation code is only correct if you generated it by checking "recalculate" in the strategy tester window. If you created it from your own tick data then it contains real tick data, and you should backtest using "recalculate" unchecked.