

Trading Crude Light 5min Bars Using The Adaptive Goertzel DFT System

Working Paper January 2014

Copyright © 2014 Dennis Meyers

In the working paper entitled “**Trading Crude Light 5min Bars Using The End Point Fast Fourier Transform**”, <http://www.meyersanalytics.com/publications2/CL5EPFFTXn.pdf>, we demonstrated how to use the EPFFT technique on a walk forward basis to trade 5-minute bars of Crude Light (CL) futures. Here it will be shown how to use the Goertzel DFT(GZ) technique on a walk forward basis to trade 5-minute bars of Crude Light (CL) futures.

In a previous working paper entitled “MESA vs Goertzel DFT”, <http://www.meyersanalytics.com/publications2/MesaVsGDFT.pdf> we demonstrated that the Goertzel Algorithm, a subset of the Discrete Fourier Transform (DFT), has better frequency detection abilities of sine waves imbedded in noise than the MESA algorithm when the noise amplitude is equal to or greater than the signal amplitude. MESA, which stands for Maximum Entropy Spectral Analysis, is more commonly referred to as the Burg AR algorithm in engineering and is a widely used mathematical technique designed to find the frequencies in data. MESA was developed by J.P Burg for his Ph.D dissertation at Stanford University in 1975. The use of the MESA technique for stocks and futures has been written about in many articles.

Previous researchers using MESA constrained themselves to using MESA to find only the cycle with the highest amplitude and called that cycle the dominant cycle. In this paper we will find and use the frequencies with the five or ten highest amplitudes to create a noise filtered signal curve that we will follow to create our strategy buy and sell signals.

The Goertzel Algorithm.

The value of the Discrete Fourier Transform is that for N input data points, the DFT can not only find the frequencies in the data but also the amplitude and phases of the sine waves in the data at the discrete frequency points of $1/N, 2/N, \dots$ to $(N/2)/N$. For instance if N, the number of price bars or closes, were equal to 20 we could find the amplitude and phases for the 20 period cycle (20 bars/ cycle), the 10 period cycle, the 6 2/3 period cycle all the way down to the 2 period cycle. We could then find the ten cycles with the highest amplitudes and reconstruct a new signal with those ten periods, amplitudes and phases. Remember, cycle= 1/period. As traders we are more accustomed to thinking of terms of periods like the 3 day cycle (frequency = 1/3 cycle/day) or the 10 bar cycle (frequency = 1/10 cycles/bar) etc. Unfortunately the DFT can only calculate equally spaced frequencies of $1/N$. In the above example, where $N=20$, using the DFT we could only calculate the amplitudes of the periods (periods=1/frequency) of 20, 10, 6 2/3 etc. What if the true signal period was between 20 and 10? The DFT couldn't find it. That was the advantage of MESA. MESA was not constrained to the $1/N$ spacing. With MESA any grid of frequencies could be examined. Fortunately, as shown in my previous article, a special subset of the discrete Fourier transform called the Goertzel algorithm can be used to find frequencies in between the $1/N$ frequency divisions. The Goertzel algorithm is used extensively in tone detection. We all are familiar with cellular phones. When you press a button have you ever wondered how the telephone company knows what button you pushed? The answer is the Goertzel algorithm. This algorithm is built into tiny integrated circuits and immediately (within 43ms) detects the tone of the button you pushed. Here we will use the Goertzel algorithm to detect the frequencies of the price series we give it.

Despite the advantage of the Goertzel algorithm frequency detection abilities, it has three drawbacks. One drawback of the Goertzel algorithm is that it is much slower than the Fast Fourier transform. If we had 512 data points and we wanted to look at 128 different frequencies, the FFT computation would be proportional to $512 \cdot \log_2(512) = 4608$ operations while the Goertzel computation would be proportional $512 \cdot 128 = 65536$ operations. In other words, in this case, the Goertzel algorithm would take more than 10 times longer than the FFT to compute. However, as a computation time comparison, Goertzel would take about half the time as MESA to compute those 128 frequencies. The second drawback of the Goertzel algorithm is that in order to find a frequency in Goertzel when the noise amplitude is high you need enough data so that your lowest frequency (largest period) is able to complete at least 3

cycles. This means that if we were examining daily closing prices and we wanted to find periods of 75 days and less than we would need at least $3*75=225$ days of prices in order to detect that period in noisy data. The third drawback of the Goertzel algorithm is that while it can detect the frequency within the $1/N$ spacing it cannot detect more than one frequency within that spacing. For instance if $N=20$, and we are looking for a frequency between $1/20$ and $2/20$, the Goertzel can detect the frequency anywhere at or between these two frequencies. However, if there were a second frequency in-between these two frequencies, Goertzel could not find it. If there were two frequencies between these two values, Goertzel would produce one frequency as a weighted average of the two. This is where MESA has a clear advantage. On data with a low noise component, Mesa could detect these two closely spaced frequencies. However, if the data is very noisy than MESA would not be able to detect the closely spaced frequencies either.

The Adaptive n Cycle Goertzel-DFT System.

The nature of intraday price movements are constantly changing due to current economic surprises, events and trader sentiment. Also the time of year changes the nature of intraday markets, such as the seasons, holidays, vacation time, etc. As such, the periods or frequencies found on intraday prices 3 weeks ago may no longer be the same as the frequencies found on today's intraday data. We would expect the frequencies found on intraday data to vary over time.

For this system we will create an indicator that walks forward one bar at a time. The indicator will take a fixed number, N , of closing prices and use the Goertzel Algorithm(GA) to find the N frequencies with the highest amplitudes. Using those frequency's, amplitudes and phases, we will construct a new price that forecasts the price one bar ahead. We will save this next bar forecast value, or forecast point. Next we will move the N closing prices forward one bar adding the next bar and dropping the last bar so that we have exactly N closing prices again. With this new N closing prices window we compute the next bar forecast value and save it. We keep marching the N closing price window forward 1 bar at a time, calculate and save the new forecast point until we reach the end of our data. We will then connect all the generated forecast points to produce a curve that creates the n cycle next bar forecast as the ten frequencies used to create the next bar forecast change over time. Thus this curve adapts to the closing price's changing frequencies and projects one day ahead so that it's lag is minimized when things change.

For example suppose we had daily data from 7/15/13 to 7/20/13. We would calculate and save the next day forecast value of 7/20/13 and slide our data window up one day from 7/16/13 to 7/21/13. We would calculate the next day's forecast value of this new data window and save the forecast value of 7/21/13. We would keep sliding our data window forward one day at a time, calculate the next day's forecast value and save that forecast point. When we reached the end of our data, we would then connect all these saved forecast points to create a new curve which we would follow and use to create the system buy and sell signals.

Adaptive Goertzel DFT System Construction Details

Unfortunately constructing the n cycle DFT of a price data series is not quite as simple as just taking N closing prices, and directly plugging them into a Goertzel algorithm.

The DFT assumes the time domain sample is periodic and repeats. Suppose a price series starts at 400 and wiggles and wags for 512 data samples ending at the value of 600. The DFT assumes that the price series starts at zero, suddenly jumps to 400, goes to 600 and suddenly jumps down to zero again and then repeats. The DFT must create all kinds of different frequencies in the frequency domain to try and match this type of behavior. These false frequencies created to match the jumps and the high average price completely swamp the amplitudes of any real frequencies making them look like noise. Fortunately this effect can be almost eliminated by a simple technique called end point flattening.

The calculation of end point flattening coefficients is simple. If $x(1)$ represents the first price in the sampled data series, $x(n)$ represent the last point in the data series and $y(i)$ equal to the new endpoint flattened series then:

$$a = x(1) \quad b = (x(n) - x(1)) / (n - 1)$$

$$y(i) = x(i) - [a + b * (i - 1)] \quad \text{for } i = 1 \text{ to } n \quad (1)$$

We can see that when $i=1$ then $y(1)=0$ and when $i=n$ then $y(n)=0$. What we've done is subtract the beginning value of the time series to make the first value equal to zero and then rotate the rest of the time series such that the end point is now zero. This technique reduces the endpoint distortion but introduces a low frequency artifact into the Fourier Frequency spectrum. Fortunately we won't be looking for frequencies in that range so this distortion will have minimal impact.

10 Cycle Goertzel-DFT Curve Construction

Before we start, we have to determine what is the largest period we will be looking to include in the 10 cycle Goertzel DFT construction. For intraday data a 1-day cycle is very important. Higher periods are also important but they don't impact on intraday trading as much. For this article we will use 5-minute bars of the Crude Light(CL) Futures. The CL trades on the CME Globex from 6pm EST to 5:15pm EST on the next day with a 45min break six days a week from Sunday at 6pm EST to Friday at 5:15 EST. Each day consists of 279 5min bars a day. Thus the 1-day cycle would need 279 bars. In order to detect the 1-day cycle we need the cycle to repeat at least three times. This means we need at least $3*279 = 837$ data points in order to detect the 1-day cycle using 5-minute bars. We will use 837 five min bars of CL data.

For this article a sliding time-bar window of 837 five-minute bars of the CL futures from 12/01/11 to 1/03/14 will be used.

We will use the Goertzel algorithm described in our previous article to determine the amplitude of the frequencies. We will also use the Goertzel algorithm to determine the phases of the frequencies with the 10 highest amplitudes.

Step 1 End flatten the 837 prices using equation (1) above.

Step 2 Use the Goertzel algorithm to calculate the amplitudes for the frequencies of $1/279$ down to $1/6$. Frequency = $1/\text{period}$. We are scanning for periods because as traders we think in terms of periods not frequencies. That is, the 1 day cycle, the 20 bar cycle etc. Thus, here we are scanning for frequencies of 279 bars/cycle to 6 bars/cycle. The frequencies of 5bars/cycle down to 2 bars/cycle move to fast with 5min bars to take advantage trading these cycles. The slippage and commissions would eat up any profits made.

Step 3 Find the ten frequencies with the highest amplitudes, calculate the phases of these frequencies and save these amplitudes ($a[i]$), phases ($\phi[i]$) and frequencies ($f[i]$). Where $[i]$ is one of the ten highest amplitudes found.

Step 4 Calculate the forecast next bar value and the end point bar value using the above ten frequencies, amplitudes and phases. The forecast(fp) = $\sum_{i=1}^{10} a[i] \cdot \cos(2 \cdot \pi \cdot f[i] \cdot 838 + \phi[i])$ $i=1$ to 10 and the Endpoint(ep) = $\sum_{i=1}^{10} a[i] \cdot \cos(2 \cdot \pi \cdot f[i] \cdot 837 + \phi[i])$ $i=1$ to 10 where $a[i]$, $f[i]$ and $\phi[i]$ are the ten frequency, amplitude and phases found.

Step 5 Save the calculated forecast next bar point and the end point values. Call the forecast next point $fp(k)$ and the end point $ep(k)$ where k denotes the order of the sliding window. That is, the first sliding window $k=1$, the second, $k=2$, etc. Slide the 837 bar data window forward one bar, and repeat steps 1 through 4.

Why do we need fp and ep ? When the data window is moved forward one bar at a time a new data sample is added to the end and the data sample at the beginning is subtracted. This adding and subtracting causes the end point flattening coefficients and the power in the frequency spectrum to jump around creating distortion and jitter in the calculation of the forecast next bar point. This random jumping as the data window slides forward in time adds a small random jump to the forecast next bar point curve. Fortunately this jumping can be minimized by creating a curve from the two saved end points, $fp(k)$ and $ep(k)$, above in step 5. Since turning points are of interest rather than magnitude then in **step 5** a new variable will be created called $sumv$ where

$$sumv(k) = sumv(k-1) + fp(k) - ep(k)$$

$fp(k)-ep(k)$ is like a one bar ahead momentum or velocity. This new curve **sumv(k)** is the sum of all the changes in the next bar's ten cycle forecast value $fp(k)$ from the end point ten cycle value. This change series minimizes the magnitude jump problem creating a fairly smooth momentum sum curve.

The Adaptive 10 Cycle Goertzel-DFT System Defined

Even though **sumv** is a fairly smooth curve it still has a number of short term wiggles preventing us from simply going long when the curve turns up and going short when the curve turns down. To create a system, we will use a simple curve following technique on CL 5-minute bars.

Buy Rule:

- **IF sumv** has moved up by more than the point amount of *ptup* from the lowest low recorded in **sumv** while short then buy the CL futures at the market..

Sell Rule:

- **IF sumv** has moved down by more than the point amount *pntdn* from the highest high recorded in **sumv** while long then sell the CL futures at the market.

Intraday Bars Exit Rule:

Close the position at 1420 ET before the CL open outcry close at 1430 ET (no trades will be carried overnight).

Intraday Bars First Trade of Day Entry Rule:

Ignore all trade signals before **8:30am ET**. For the Buy and Sell rules above we have included a first trade of the day entry rule. Trading in the CL futures has changed a lot in the last 5 years because of 24hr Globex trading. In particular trading starts a lot earlier in the morning when Asia and then Europe opens and then dies down. The CL volume starts to pick up again around 830am ET before the open outcry floor opens at 9am ET so we will have the strategy resume trading at 830am ET

Discussion of Crude Light Future Prices

Crude Light futures (CL) are traded on Globex and on the trading floor at the NYMEX. On Globex the CL is traded on a 23hour basis. The NYMEX hours for open outcry floor trading (RTH) are 9:00 ET to 14:30 ET. Over 60% of the volume in the CL is done on Globex during the NYMEX open outcry hours. We have restricted our study to only trading the CL during the 8:30 to 14:20 ET hours.

Testing The Adaptive N Cycle Goertzel-DFT (GZ) System Using Walk Forward Optimization

There will be three strategy parameters to determine:

1. *ncy*, Number of Cycles (for this study we will look at both 10 and 5 cycles).
2. *ptup*, if **sumv** has moved up by more than the point amount of *ptup* from the lowest low recorded in **sumv** while short then issue a buy signal
3. *pntdn*, if **sumv** has moved down by more than the point amount *pntdn* from the highest high recorded in **sumv** while long then sell

To test this system we will use five minute bar prices of Crude Light (CL) futures contract traded on the NYMEX/Globex and known by the symbol CL for the 109 weeks from December 1, 2011 to January 3, 2014.

We will test this strategy with the above CL 5 min bars on a walk forward basis, as will be described below. To create our walk forward files we will use the *add-in* software product called the Power Walk Forward Optimizer (PWFO). In TradeStation (TS), we will run the PWFO strategy *add-in* along with GZ Strategy on the CL 5min bar data from December 1, 2011 to January 3, 2014..

What Is A In-sample section and Out-Of-Sample Section?

Whenever we do a TS optimization on a number of different strategy inputs, TS generates a report of performance metrics (total net profits, number of losing trades, etc) vs these different inputs. If the report is sorted on say the total net profits(*tnp*) performance metric column then the highest *tnp* would correspond to a certain set of inputs. This is called a *in-sample section*. If we choose a set of strategy inputs from this report based upon some performance

metric we have no idea whether these strategy inputs will produce the same results on future price data or data they have not been tested on. Price data that is not in the in-sample section is defined as **out-of-sample data**. Since the performance metrics generated in the in-sample section are mostly due to “curve fitting” (see Walk Forward Out-of-Sample Testing section below) it is important to see how the strategy inputs chosen from the in-sample section perform on out-of-sample data.

What Does The Power Walk Forward Optimizer (PWFO) Do?

The PWFO is a TS **add-in** that breaks up the TS optimization run into a number of user selectable test and out-of-sample sections. The PWFO prints out the test sample performance **and the out-of-sample performance results**, on one line, for each case or input variable combination that is run by the TradeStation(TS) optimization module to a user selected spreadsheet comma delimited file. The PWFO can generate up to 500 different test and out-of-sample date optimization files in one TS run, saving the user from having to generate optimization runs one at a time. The PWFO output allows you to quickly determine whether your procedure for selecting input parameters for your strategy just curve fits the price and noise, or produces statistically valid out-of-sample results. In addition to the out-of-sample performance results presented for each case, 30+ superior and robust performance metrics (many are new and never presented before) are added to each case line in the in-sample section and printed out to the comma delimited file. These 30+ performance metrics allow for a superior and robust selection of input variables from the in-sample section that have a higher probability of performing well on out-of-sample data (Please see Appendix 2 for a listing of these performance metrics).

For our computer run we will have the PWFO breakup the 109 weeks of CL five-minute bar price data into 105 test/out-of sample files. The in-sample sections will be 30 calendar days and the out-of-sample(oos) section will be the one week following the in-sample section. The oos week will always end on a Friday as will the 30-day calendar in-sample section. As an example the first in-sample section would be from 12/1/2011 to 12/30/2011 and the out-of-sample section would be from 1/2/2012 to 1/6/2012.(our test and out-of-sample sections always end on a Friday). We would then move everything ahead a week and the 2nd in-sample section would be from 12/8/2011 to 1/6/2012 and the out-of-sample section would be from 1/9/2012 to 1/13/2012. Etc.

The PWFO 105 test/out-of-sample section dates are shown in **Table 1** on page 8 below. We will then use another software product called the Walk Forward Performance Metric Explorer (WFME) on each of the 105 test and out-of-sample(oos) sections generated by the PWFO to find the best in-sample section performance **filter** that determines the system input parameters (**ncy, pntup, pntdn**) that will be used on the out-of-sample data. Detailed information about the PWFO and the WFME can be found at www.meyersanalytics.com

For the test data we will run the TradeStation combinatorial optimization engine on the 105 weeks of CL 5 min bars with the following ranges for the Goertzel(GZ) strategy input variables.

1. ncy 5 and 10
2. pntup from 0.2 to 3 steps of 0.2
3. pntdn from 0.2 to 3 in steps of 0.2
4. myStartTime = 830
5. Xtime = 1420

This will produce 450 different cases or combinations of the input parameters for each of the 105 PWFO output files.

Walk Forward Out-of-Sample Testing

Walk forward analysis attempts to minimize the curve fitting of price noise by using the law of averages from the Central Limit Theorem on the out-of-sample performance. In walk forward analysis the data is broken up into many test and out-of-sample sections. Usually for any system, one has some performance metric selection procedure, which we will call a **filter**, used to select the input parameters from the optimization run. For instance, a **filter** might be all cases that have a profit factor (PF) greater than 1 and less than 3. For the number of cases left, we might select the cases that had the best percent profit. This procedure would leave you with one case in the in-sample section output and it's associated strategy input parameters. Now suppose we ran our optimization on each of our many in-sample sections and applied our filter to each in-sample section output. We would then use the strategy input

parameters found by the **filter** in each in-sample section on the out-of-sample section immediately following that in-sample section. The input parameters found in each in-sample section and applied to each out-of-sample section would produce independent net profits or losses for each of the out-of-sample sections. Using this method we now have "x" number of independent out-of-sample section profit and losses from our filter. If we take the average of these out-of-sample section net profits and losses, then we will have an estimate of how our system will perform on average. Due to the Central Limit Theorem, as our sample size increases, the spurious noise results in the out-of-sample section performance tend to average out to zero in the limit leaving us with what to expect from our system and filter. **Mathematical note:** *This assumption assumes that the out-of-sample returns are from probability distributions that have a finite variance.*

Why use the walk forward technique? Why not just perform an optimization on the whole price series and choose the input parameters that give the best total net profits or profit factor? Surely the price noise cancels itself out with such a large number of test prices and trades. Unfortunately, nothing could be farther from the truth! Optimization is a misnomer and should really be called combinatorial search. As stated above, whenever we run a combinatorial search over many different combinations of input parameters on noisy data on a fixed number of prices, **no matter how many**, the best performance parameters found are guaranteed to be due to **"curve fitting"** the noise and signal. What do we mean by **"curve fitting"**? The price series that we trade consists of random spurious price movements, which we call noise, and repeatable price patterns (*if they exist*). When we run, for example, 5000 different inputs parameter combinations, the best performance parameters will be from those system input variables that are able to produce profits from the price pattern **and** the random spurious movements. While the price patterns will repeat, the same spurious price movements will not. If the spurious movements that were captured by a certain set of input parameters were a large part of the total net profits, then choosing these input parameters will produce losses when traded on future data. These losses occur because the spurious movements will not be repeated in the same way. This is why system optimization or combinatorial searches with no out-of-sample testing cause losses when traded in real time from something that looked great in the in-sample section. Unfortunately it is human nature to extrapolate past performance to project future trading results and thus results from curve fitting give the illusion, a modern "siren call" so to speak, of future trading profits.

In order to gain confidence that our input parameter selection method using the optimization output of the test data will produce profits, we must test the input parameters we found in the in-sample section on out-of-sample data. In addition, we must perform the test/out-of-sample analysis many times. Why not just do the out-of-sample analysis once? Well just as in Poker or any card game, where there is considerable variation in luck from hand to hand, walk forward out-of-sample analysis give considerable variation in week-to-week out-of-sample profit "luck". That is, by pure chance we may have chosen some input parameter set that did well in the in-sample section data **and** the out-of-sample section data. In order to minimize this type of "luck", statistically we must repeat the walk forward out-of-sample (oos) analysis over many test/oos sections and take an average of our weekly results over all out-of-sample sections. This average gives us an expected weekly return and a standard deviation of weekly returns which allows us to statistically estimate the expected equity and it's range for N weeks in the future.

Finding The Strategy Input Parameters in The Walk Forward In-sample sections

The PWFO generates a number of performance metrics in the in-sample section. The question we are attempting to answer statistically, is which performance metric or combination of performance metrics (which we will call a **filter**) in the in-sample section will produce strategy inputs that produce statistically valid profits in the out-of-sample section. In other words we wish to find a metric **filter** that we can apply to the in-sample section that can give us strategy inputs that will produce, on average, good trading results in the future. The PWFO produces a total of 32 different performance metrics in the in-sample section. If we have 450 different input variations or cases then the in-sample section consists of 32 columns of performance metrics for each of the 450 input cases or rows.

An example of a simple filter would be to choose the row in the in-sample section that had the highest net profit or perhaps a row that had one the best performance metric from one of the other 32 PWFO metrics. Unfortunately it was found that this type of simple filter very rarely produces good out-of-sample results. More complicated metric filters can produce good out-of-sample results minimizing spurious price movement biases in the selection of strategy inputs.

Here is an *example* of a better more complicated *filter* that was used in this here. Not many traders can stay with a strategy that has a large number of losers in a row (**LR**). For this filter we will choose **LR**≤**3**. This choice of LR is completely arbitrary and is what I feel comfortable with. After using the **LR** filter, as described, there can still be 100's of rows left in the PWFO file in-sample section. The PWFO generates the metric **mWb**. **Wb** is the number of bars in each of the winning trades. This metric calculates the median of all the **Wb**. We use the median for this metric, because we do not want the statistic distorted by an outlier trades. Thus we would want the median to be as large as possible. Let us choose the 5 rows that contain the **largest(top) mWb** values from the rows that are left from the **LR** screen. In other words we sort **mWb** from high to low, eliminate the rows that have LR>3 then choose the top 5 Rows of whatever is left. This particular filter will now leave 5 cases or rows in the PWFO file that satisfy the above filter conditions. We call this filter **t5mWb|lr3** where **t5mWb** means the Top 5 **mWb** rows left *after* the LR filter.. Suppose for this filter, within the 5 PWFO rows that are left, we want the row that has the minimum PWFO **mLTr** metric in the in-sample section. The metric **mLTr** is the median of all the losing trades in the in-sample section. Again we use the median rather than the average because we do not want this statistic distorted by outlier trades. This would produce a filter named **t5mWb|lr3-mLTr**. This filter leaves only one row in the PWFO in-sample section **with its associated strategy inputs** and out-of-sample net profit in the out-of-sample section. This particular **t5mWb|lr3-mLTr** filter is then calculated for each of the 105 PWFO files and the average 105 out-of-sample performance is calculated. In addition many other important out-of-sample performance statistics for this filter are calculated and summarized. **Figure 3** shows such a computer run along with a small sample of other filter combinations that are constructed in a similar manner. Row 3 of the sample output in Figure 3 shows the results of the filter discussed above.

Bootstrap Probability of Filter Results. Using modern "Bootstrap" techniques, we can calculate the probability of obtaining each filter's total out-of-sample *net* profits by chance. By *net* we mean subtracting the cost and slippage of all round trip trades from the total out-of-sample profits. Here is how the bootstrap technique is applied. Suppose as an example, we calculate the total out-of-sample net profits(tOnpNet) of 5000 different **TopN-Metric-LR** filters. A mirror filter is created for each of the 5000 filters. However, instead of picking an out-of-sample net profit(OSNP) from a filter row as above, the mirror filter picks a *random* row's OSNP in each of the 105 PWFO files. Each of the 5000 mirror filters will choose a random row's OSNP of their own in each of the 105 PWFO files.. At the end, each of the 5000 mirror filters will have 105 *random* OSNP's picked from the rows of the 105 PWFO files. The sum of the 105 random OSNP picks for each mirror filter will generate a random total out-of-sample net profit(tOnpNet). The average and standard deviation of the 5000 mirror filter's different random tOnpNets will allow us to calculate the chance probability of each **TopN-Metric-LR** filter's tOnpNet. Thus given the mirror filter's bootstrap random tOnpNet average and standard deviation, we can calculate the probability of obtaining the TopN-Metric-LR- filter's tOnpNet by pure chance alone. Since for this run there are 3844 different filters, we can calculate the expected number of cases that we could obtain by pure chance that would match or exceed the tOnpNet of the filter we have chosen or (3844) X (tOnpNet Probability). For our filter, in row 3 in Figure 3, the expected number of cases that we could obtain by pure chance that would match or exceed the \$59260 is $3844 \times 6.13 \times 10^{-5} = 0.236$. This is much less than one case so it is improbable that our result was due to pure chance.

The partial run shown in Figure 3 reveals that the following filter will produce the most consistent and reliable out-of-sample results.

Filter: LR≤3 and Top 5 mWb then minimum mLTr

Where:

- **LR** = Maximum losing trades in a row for a given set of inputs in the in-sample section
- **mWb= Wb** is the number of bars in each of the winning trades. This metric calculates the median of all the **Wb**
- **mLTr** = The metric **mLTr** is the median of all the losing trades in the in-sample section

The first part of the filter chooses those rows or cases out of the 450 rows in each PWFO file test(in-sample) section that satisfy the criteria **LR**≤**3**. After using the LR filter, there can still be 100's of rows left in the PWFO file. The PWFO generates the metric **mWb**, which is the median of the number of bars in each of the winning trades. Let us

choose the 5 rows that contain the **Top 5 mWb** values from the rows that are left from the **LR** screen.. This particular filter will now leave 5 cases or rows in the PWFO file that satisfy these filter conditions. We call this filter **t5mWb|lr3** . Within the 5 PWFO rows that are left, we want the row that has the minimum PWFO metric, **mLTr**, in the in-sample section. This **Filter** or selection procedure will leave only one choice for the strategy input values of **ncy, pntup, pntdn**. We then use these input values found in the in-sample section by the **Filter** on the **next week** of five minute CL **out-of-sample** price bars **following** the in-sample section.

Results

Table 1 below presents a table of the 105 test and out-of-sample windows , the selected optimum parameters and the weekly out-of-sample profit/loss results using the filter described above.

Figure 1 presents a graph of the equity curve generated by using the filter on the 105 weeks of 1/06/12 – 1/03/14. The equity curve is plotted from NetEq in Table 1. Plotted on the equity curve is the least squares straight line. The weekly CL prices are also included on the plot so that we can compare how the equity curve changes vs CL price changes.

Figure 2 presents the out-of-sample 5 minute bar chart of CL for 12/30/13 to 1/3/14 with the GZ Indicator and all the buy and sell signals for those dates.

Figure 3 Partial output of the Walk Forward Metric Performance Explorer

Discussion of System Performance

In Figure 3 Row 2 of the spreadsheet filter output are some statistics that are of interest for our filter. An interesting statistic is **Blw**. Blw is the maximum number of weeks the OSNP equity curve failed to make a new high. Blw is 12 weeks for this filter. This means that 12 weeks was the longest time that the equity for this strategy failed to make a new equity high.

To see the effect of walk forward analysis, take a look at **Table 1**. Notice how the input parameters **ncy, pntup and pntdn** take sudden jumps from high to low and back . This is the walk forward process quickly adapting to changing volatility conditions in the test sample. In addition, notice how often **ncy** changes from 5 to 10.

In Figure 1, which presents a graph of the equity curve using the filter on the 105 weeks of 1/06/12 – 1/03/14, notice how the equity curve follows the trend line with an R^2 of 0.96. The weekly plot of CL prices is superimposed on the equity curve plot. Notice how fast the equity curve is able to adapt with changing trends in CL. From 2/24/12 to 6/22/12 CL fell fast from 110 to 80. From 6/22/12 to 9/14/12 CL climbed fast from 80 to 99. Yet despite these changing price trends the GZ strategy was able to adapt and continue to generate net profits.

Using this filter, the strategy was able to generate \$51050, net equity after commissions and slippage trading one CL contract for 105 weeks. This period of time from 1/06/12 – 1/03/14 was at volatile period for crude oil. Yet the GZ strategy was able to adapt quite well. Note \$30 roundtrip commission and slippage was subtracted from each trade and no positions were carried over night. From Table 1, the largest net losing week was -\$3880 on 5/10/13 a very whipsaw market week. The largest net equity drawdown was -\$9970 during whipsaw CL price movements from 12/28/12-3/15/13. The longest time between new equity highs was 12 weeks.

In observing Table 1 we can see that this strategy and filter made trades from a low of no trades per week to a high of 9 trades/week with an average of 4.5 trades/week. Note that the GZ strategy did not trade in 14 of the 105 weeks.

Given 24 hour trading of the Crude Light, restricting the strategy to trade only from 830am to 2:20pm ET caused the strategy to miss many profitable trends opportunities when Asia and then Europe opened trading in the early morning. As can be seen from the CL 5min bar chart in Figure 2, while the strategy indicated to remain short restricting trading from 830am to 2:20pm missed the main overnight trend and the strategy profits were less then they could have been if overnight trading was allowed. Further research will include the A.M. time zones.

Disclaimer

The strategies, methods and indicators presented here are given for educational purposes only and should not be construed as investment advice. Be aware that the profitable performance presented here is based upon hypothetical trading with the benefit of hindsight and can in no way be assumed nor can it be claimed that the strategy and methods presented here will be profitable in the future or that they will not result in losses.

References:

1. Burg, J. P., "Maximum Entropy Spectral Analysis", Ph.D. dissertation, Stanford University, Stanford, CA. May 1975.
2. Goertzel, G., "An Algorithm for the evaluation of finite trigonometric series" American Math Month, Vol 65, 1958 pp34-35.
3. Kay, Steven M., "Modern Spectral Estimation", Prentice Hall, 1988
4. Marple, Lawrence S. Jr., "Digital Spectral Analysis With Applications", Prentice Hall, 1987
5. Meyers, Dennis, "MESA vs Goertzel DFT", *Working Paper*, <http://meyersanalytics.com/articles.htm>
6. Press, William H., et al, "Numerical Recipes in C++: the Art of Scientific Computing", Cambridge Press, 2002.
7. Oppenheim, A, Schafer, R. and Buck, J., "Discrete Time Signal Processing", Prentice Hall, 1996, pp663-634
8. Proakis, J. and Manolakis, D. "Digital Signal Processing-Principles, Algorithms and Applications", Prentice Hall, 1996., pp480-481

Info on Dennis Meyers

Dennis Meyers (info@MeyersAnalytics.com) has a doctorate in applied mathematics in engineering. He is a private trader, and president of Meyers Analytics (www.MeyersAnalytics.com). His firm specializes in Financial Engineering consulting for financial institutions and developing publicly available analytical software for traders.

Table 1 Walk Forward Out-Of-Sample Performance Summary for CL End Point Fast Fourier Transform System

CL-5 min bars 11/30/2011 - 1/03/2014. The input values *ncy*, *pntup*, *pntdn* are the values found from applying the filter to the in-sample section optimization runs.

In-Sample Dates		Out-Of Sample Dates		osnp	NOnp\$30	NetEq	ollt	odd	ont	avosnp	ncy	pup	pdn	sTime	xt	CL CLS
11/30/11	to 12/30/11	01/02/12	to 01/06/12	130	10	10	-1280	-1850	4	32.5	5	1	2.4	830	1420	101.93
12/07/11	to 01/06/12	01/09/12	to 01/13/12	(3760)	(3880)	(3870)	-2360	-3760	4	-940	10	1.2	2.4	830	1420	99.03
12/14/11	to 01/13/12	01/16/12	to 01/20/12	3470	3320	(550)	-360	-360	5	694	5	2	1	830	1420	98.75
12/21/11	to 01/20/12	01/23/12	to 01/27/12	1230	1110	560	-70	-70	4	307.5	10	1.4	1.6	830	1420	100.12
12/28/11	to 01/27/12	01/30/12	to 02/03/12	(2650)	(2740)	(2180)	-2700	-3780	3	-883.3	10	1.6	1.6	830	1420	98.13
01/04/12	to 02/03/12	02/06/12	to 02/10/12			(2180)					5	2.2	3	830	1420	99.41
01/11/12	to 02/10/12	02/13/12	to 02/17/12			(2180)					5	2.2	2.8	830	1420	104.42
01/18/12	to 02/17/12	02/20/12	to 02/24/12	360	240	(1940)	-1040	-1040	4	90	10	1.2	1	830	1420	109.62
01/25/12	to 02/24/12	02/27/12	to 03/02/12			(1940)					5	2.2	2.8	830	1420	106.5
02/01/12	to 03/02/12	03/05/12	to 03/09/12	(1480)	(1660)	(3600)	-1300	-1750	6	-246.7	5	0.8	1.4	830	1420	107.94
02/08/12	to 03/09/12	03/12/12	to 03/16/12			(3600)					5	1.8	3	830	1420	107.67
02/15/12	to 03/16/12	03/19/12	to 03/23/12	(420)	(450)	(4050)	0	0	1	-420	5	1.8	3	830	1420	106.75
02/22/12	to 03/23/12	03/26/12	to 03/30/12	(330)	(420)	(4470)	-420	-420	3	-110	5	1	2.4	830	1420	102.93
02/29/12	to 03/30/12	04/02/12	to 04/06/12	2890	2800	(1670)	0	0	3	963.3	5	0.8	2.6	830	1420	103.25
03/07/12	to 04/06/12	04/09/12	to 04/13/12	(230)	(380)	(2050)	-1320	-1320	5	-46	5	0.8	1.6	830	1420	102.84
03/14/12	to 04/13/12	04/16/12	to 04/20/12	1320	1170	(880)	-360	-360	5	264	10	0.6	1.2	830	1420	104.04
03/21/12	to 04/20/12	04/23/12	to 04/27/12	840	690	(190)	-270	-490	5	168	10	0.6	1.2	830	1420	104.81
03/28/12	to 04/27/12	04/30/12	to 05/04/12	2900	2720	2530	-1740	-2400	6	483.3	10	1.2	1.2	830	1420	98.59
04/04/12	to 05/04/12	05/07/12	to 05/11/12	(1030)	(1300)	1230	-1100	-2490	9	-114.4	10	0.4	1	830	1420	95.57
04/11/12	to 05/11/12	05/14/12	to 05/18/12	1350	1230	2460	0	0	4	337.5	5	1.2	0.6	830	1420	91.65
04/18/12	to 05/18/12	05/21/12	to 05/25/12	1360	1240	3700	-200	-200	4	340	10	1.4	0.6	830	1420	90.72
04/25/12	to 05/25/12	05/28/12	to 06/01/12	3600	3450	7150	-200	-200	5	720	10	1.4	0.8	830	1420	83.26
05/02/12	to 06/01/12	06/04/12	to 06/08/12	800	590	7740	-730	-1400	7	114.3	5	1	0.6	830	1420	84.34
05/09/12	to 06/08/12	06/11/12	to 06/15/12	760	610	8350	-910	-1440	5	152	10	2	0.8	830	1420	84.84
05/16/12	to 06/15/12	06/18/12	to 06/22/12	1570	1390	9740	-1500	-1500	6	261.7	5	1	0.2	830	1420	80.15
05/23/12	to 06/22/12	06/25/12	to 06/29/12	6310	6160	15900	-380	-520	5	1262	5	1	0.6	830	1420	84.87
05/30/12	to 06/29/12	07/02/12	to 07/06/12	10	(140)	15760	-1160	-1840	5	2	10	0.6	2	830	1420	84.12
06/06/12	to 07/06/12	07/09/12	to 07/13/12			15760					5	2.8	2.6	830	1420	87.37
06/13/12	to 07/13/12	07/16/12	to 07/20/12			15760					5	2.8	2.6	830	1420	91.7
06/20/12	to 07/20/12	07/23/12	to 07/27/12	(1490)	(1520)	14240	0	0	1	-1490	5	2.8	2.2	830	1420	90.26
06/27/12	to 07/27/12	07/30/12	to 08/03/12	1540	1480	15720	0	0	2	770	5	3	2.4	830	1420	91.39
07/04/12	to 08/03/12	08/06/12	to 08/10/12	(460)	(520)	15200	-570	-570	2	-230	10	2	1.8	830	1420	93.7
07/11/12	to 08/10/12	08/13/12	to 08/17/12	(1370)	(1400)	13800	0	0	1	-1370	5	1.4	2.4	830	1420	96.52
07/18/12	to 08/17/12	08/20/12	to 08/24/12	1790	1640	15440	-540	-540	5	358	5	1.4	2.4	830	1420	96.12
07/25/12	to 08/24/12	08/27/12	to 08/31/12	1950	1830	17270	-380	-620	4	487.5	5	1.4	2.4	830	1420	96.56
08/01/12	to 08/31/12	09/03/12	to 09/07/12	2270	2150	19420	-530	-530	4	567.5	5	1.6	2.4	830	1420	96.29
08/08/12	to 09/07/12	09/10/12	to 09/14/12	10	(140)	19280	-560	-560	5	2	10	2.2	2.8	830	1420	99.06
08/15/12	to 09/14/12	09/17/12	to 09/21/12	2030	1850	21130	-980	-1060	6	338.3	5	0.8	1	830	1420	93.39
08/22/12	to 09/21/12	09/24/12	to 09/28/12	(720)	(900)	20230	-1300	-1300	6	-120	10	1.2	0.6	830	1420	92.53
08/29/12	to 09/28/12	10/01/12	to 10/05/12	(310)	(460)	19770	-2530	-2880	5	-62	5	1.8	3	830	1420	90.35
09/05/12	to 10/05/12	10/08/12	to 10/12/12	2660	2540	22310	0	0	4	665	10	2.2	1	830	1420	92.07
09/12/12	to 10/12/12	10/15/12	to 10/19/12	1130	980	23290	-780	-780	5	226	10	2	0.8	830	1420	90.6
09/19/12	to 10/19/12	10/22/12	to 10/26/12	3980	3860	27150	0	0	4	995	10	2	0.8	830	1420	86.24
09/26/12	to 10/26/12	10/29/12	to 11/02/12	810	660	27810	-920	-1180	5	162	10	3	0.8	830	1420	84.79
10/03/12	to 11/02/12	11/05/12	to 11/09/12	1690	1600	29410	-1670	-1670	3	563.3	10	2.2	0.8	830	1420	86.17
10/10/12	to 11/09/12	11/12/12	to 11/16/12	(210)	(360)	29050	-980	-1410	5	-42	10	2.8	0.8	830	1420	86.99
10/17/12	to 11/16/12	11/19/12	to 11/23/12	660	540	29590	-1090	-1090	4	165	10	2.2	1.8	830	1420	88.26

In-Sample Dates		Out-Of_Sample Dates		osnp	NOnp\$30	NetEq	ollt	odd	ont	avosnp	ncy	pup	pdn	sTime	xt	CL CLS
10/24/12	to 11/23/12	11/26/12	to 11/30/12	(70)	(100)	29490	0	0	1	-70	10	2.2	2.2	830	1420	88.94
10/31/12	to 11/30/12	12/03/12	to 12/07/12	1710	1560	31050	-450	-590	5	342	10	2.2	2.2	830	1420	85.98
11/07/12	to 12/07/12	12/10/12	to 12/14/12	1430	1280	32330	-460	-460	5	286	10	2.2	2.2	830	1420	87.28
11/14/12	to 12/14/12	12/17/12	to 12/21/12	(110)	(140)	32190	0	0	1	-110	5	2.2	2.6	830	1420	88.96
11/21/12	to 12/21/12	12/24/12	to 12/28/12			32190					5	1.4	2	830	1420	90.62
11/28/12	to 12/28/12	12/31/12	to 01/04/13	(30)	(60)	32130	0	0	1	-30	5	1.4	2.2	830	1420	93.19
12/05/12	to 01/04/13	01/07/13	to 01/11/13	(380)	(530)	31600	-560	-1110	5	-76	10	1.2	1.4	830	1420	93.77
12/12/12	to 01/11/13	01/14/13	to 01/18/13			31600					5	1.6	2.6	830	1420	95.79
12/19/12	to 01/18/13	01/21/13	to 01/25/13	680	530	32130	-850	-980	5	136	10	2	1.4	830	1420	96.09
12/26/12	to 01/25/13	01/28/13	to 02/01/13	(620)	(710)	31420	-480	-620	3	-206.7	5	0.8	1.2	830	1420	97.61
01/02/13	to 02/01/13	02/04/13	to 02/08/13	(3010)	(3160)	28260	-1640	-3160	5	-602	5	0.4	1	830	1420	95.76
01/09/13	to 02/08/13	02/11/13	to 02/15/13	120	(60)	28200	-1020	-1470	6	20	5	0.2	1.4	830	1420	97
01/16/13	to 02/15/13	02/18/13	to 02/22/13	(2630)	(2780)	25420	-1900	-2670	5	-526	5	0.4	1.4	830	1420	93.73
01/23/13	to 02/22/13	02/25/13	to 03/01/13	(1880)	(2030)	23390	-1250	-1880	5	-376	5	1	3	830	1420	91.39
01/30/13	to 03/01/13	03/04/13	to 03/08/13	(530)	(710)	22680	-750	-1270	6	-88.3	10	0.8	0.2	830	1420	92.24
02/06/13	to 03/08/13	03/11/13	to 03/15/13	(310)	(460)	22220	-750	-750	5	-62	5	0.8	0.2	830	1420	93.82
02/13/13	to 03/15/13	03/18/13	to 03/22/13	2300	2150	24370	-90	-90	5	460	10	0.6	1	830	1420	93.8
02/20/13	to 03/22/13	03/25/13	to 03/29/13			24370					5	3	3	830	1420	97.23
02/27/13	to 03/29/13	04/01/13	to 04/05/13	1440	1320	25690	-280	-480	4	360	5	0.8	1	830	1420	93.05
03/06/13	to 04/05/13	04/08/13	to 04/12/13	(70)	(100)	25590	0	0	1	-70	5	1.2	1.6	830	1420	90.66
03/13/13	to 04/12/13	04/15/13	to 04/19/13	1310	1160	26750	-620	-620	5	262	5	1.8	2.8	830	1420	88.27
03/20/13	to 04/19/13	04/22/13	to 04/26/13	4210	4060	30810	-240	-240	5	842	5	1.8	2.8	830	1420	92.78
03/27/13	to 04/26/13	04/29/13	to 05/03/13	4050	3930	34740	-760	-760	4	1012.5	5	1.8	2.8	830	1420	95.61
04/03/13	to 05/03/13	05/06/13	to 05/10/13	(3130)	(3280)	31460	-980	-3130	5	-626	5	3	1.8	830	1420	96.26
04/10/13	to 05/10/13	05/13/13	to 05/17/13	1560	1470	32930	0	0	3	520	10	1.6	3	830	1420	96.27
04/17/13	to 05/17/13	05/20/13	to 05/24/13	1930	1810	34740	-220	-220	4	482.5	10	1.8	3	830	1420	93.87
04/24/13	to 05/24/13	05/27/13	to 05/31/13	(1980)	(2100)	32640	-1640	-2000	4	-495	5	1.8	2.6	830	1420	91.61
05/01/13	to 05/31/13	06/03/13	to 06/07/13	2080	1930	34570	-160	-160	5	416	5	0.2	1.4	830	1420	96.26
05/08/13	to 06/07/13	06/10/13	to 06/14/13	3000	2850	37420	0	0	5	600	10	0.8	1.4	830	1420	97.87
05/15/13	to 06/14/13	06/17/13	to 06/21/13	(3670)	(3820)	33600	-1820	-3670	5	-734	10	2.4	1.6	830	1420	93.68
05/22/13	to 06/21/13	06/24/13	to 06/28/13	2550	2370	35970	-420	-420	6	425	10	0.4	0.8	830	1420	96.22
05/29/13	to 06/28/13	07/01/13	to 07/05/13	3480	3300	39270	-270	-270	6	580	5	0.6	0.6	830	1420	103.36
06/05/13	to 07/05/13	07/08/13	to 07/12/13	4070	3890	43160	-80	-80	6	678.3	10	0.2	0.8	830	1420	105.98
06/12/13	to 07/12/13	07/15/13	to 07/19/13	(600)	(840)	42320	-1180	-1180	8	-75	10	0.2	0.8	830	1420	108.14
06/19/13	to 07/19/13	07/22/13	to 07/26/13	2880	2640	44960	-600	-600	8	360	5	0.4	0.6	830	1420	104.35
06/26/13	to 07/26/13	07/29/13	to 08/02/13	(110)	(290)	44670	-1160	-1160	6	-18.3	10	0.2	0.6	830	1420	106.74
07/03/13	to 08/02/13	08/05/13	to 08/09/13	630	360	45030	-1300	-2420	9	70	10	0.2	0.8	830	1420	105.89
07/10/13	to 08/09/13	08/12/13	to 08/16/13			45030					5	3	2.8	830	1420	107.5
07/17/13	to 08/16/13	08/19/13	to 08/23/13			45030					5	3	2.8	830	1420	106.32
07/24/13	to 08/23/13	08/26/13	to 08/30/13	(190)	(280)	44750	-320	-520	3	-63.3	5	3	2.8	830	1420	107.76
07/31/13	to 08/30/13	09/02/13	to 09/06/13	1970	1820	46570	-1060	-1060	5	394	5	0.6	1	830	1420	110.23
08/07/13	to 09/06/13	09/09/13	to 09/13/13	(850)	(1000)	45570	-600	-940	5	-170	10	2	2.8	830	1420	108
08/14/13	to 09/13/13	09/16/13	to 09/20/13	1100	950	46520	-1800	-1800	5	220	10	1.4	0.4	830	1420	104.81
08/21/13	to 09/20/13	09/23/13	to 09/27/13	70	(50)	46470	-280	-550	4	17.5	5	0.8	0.6	830	1420	102.81
08/28/13	to 09/27/13	09/30/13	to 10/04/13	1620	1470	47940	-410	-410	5	324	5	1.4	0.6	830	1420	103.7
09/04/13	to 10/04/13	10/07/13	to 10/11/13	(1040)	(1190)	46750	-1560	-1600	5	-208	5	1.2	1.6	830	1420	101.94
09/11/13	to 10/11/13	10/14/13	to 10/18/13			46750					10	1.6	1.6	830	1420	101.15
09/18/13	to 10/18/13	10/21/13	to 10/25/13	(1780)	(1840)	44910	-1670	-1780	2	-890	10	1.2	1.6	830	1420	97.9
09/25/13	to 10/25/13	10/28/13	to 11/01/13	1010	920	45830	-490	-490	3	336.7	10	1.2	1.6	830	1420	94.6
10/02/13	to 11/01/13	11/04/13	to 11/08/13	(760)	(910)	44920	-850	-990	5	-152	5	1.6	3	830	1420	95
10/09/13	to 11/08/13	11/11/13	to 11/15/13	(40)	(220)	44700	-1400	-1400	6	-6.7	5	0.2	1	830	1420	94.39
10/16/13	to 11/15/13	11/18/13	to 11/22/13	980	800	45500	-550	-550	6	163.3	10	0.8	0.2	830	1420	94.72
10/23/13	to 11/22/13	11/25/13	to 11/29/13			45500					10	3	2.6	830	1420	92.78

In-Sample Dates			Out-Of_Sample Dates			osnp	NOnp\$30	NetEq	ollt	odd	ont	avosnp	ncy	pup	pdn	sTime	xt	CL CLS
10/30/13	to	11/29/13	12/02/13	to	12/06/13	2800	2650	48150	0	0	5	560	10	2.8	2.2	830	1420	97.82
11/06/13	to	12/06/13	12/09/13	to	12/13/13	(1210)	(1360)	46790	-750	-1210	5	-242	5	2.8	2.6	830	1420	96.76
11/13/13	to	12/13/13	12/16/13	to	12/20/13	470	350	47140	-80	-80	4	117.5	5	1	0.8	830	1420	99.1
11/20/13	to	12/20/13	12/23/13	to	12/27/13			47140					5	1	0.8	830	1420	100.15
11/27/13	to	12/27/13	12/30/13	to	01/03/14	4030	3910	51050	0	0	4	1007.5	5	1.8	0.8	830	1420	94.31

Filter, LR<=3 and Top 5 mWb, Minimum mLTr

osnp = Weekly Out-of-sample gross profit in \$

NOnp\$30 = Weekly Out-Of-Sample Net Profit in \$ = **osnp-ont*30**.

NetEq = running sum of the weekly out-of-sample net profits in \$

ollt = The largest losing trade in the out-of-sample section in \$.

odd = The close drawdown in the out-of-sample section in \$.

ont = The number of trades in the out-of-sample week.

avosnp = The average out-of-sample profit per trade for that week \$



**Figure 1 Graph of Net Equity Applying the Walk Forward Filter Each Week
On CL 5min Bar Prices 1/06/2012 – 1/03/2014**

Note: The blue line is the equity curve without commissions and the red dots on the blue line are new highs in equity. The brown line is the equity curve with commissions and the green dots are the new highs in net equity. The green line is the CL close price

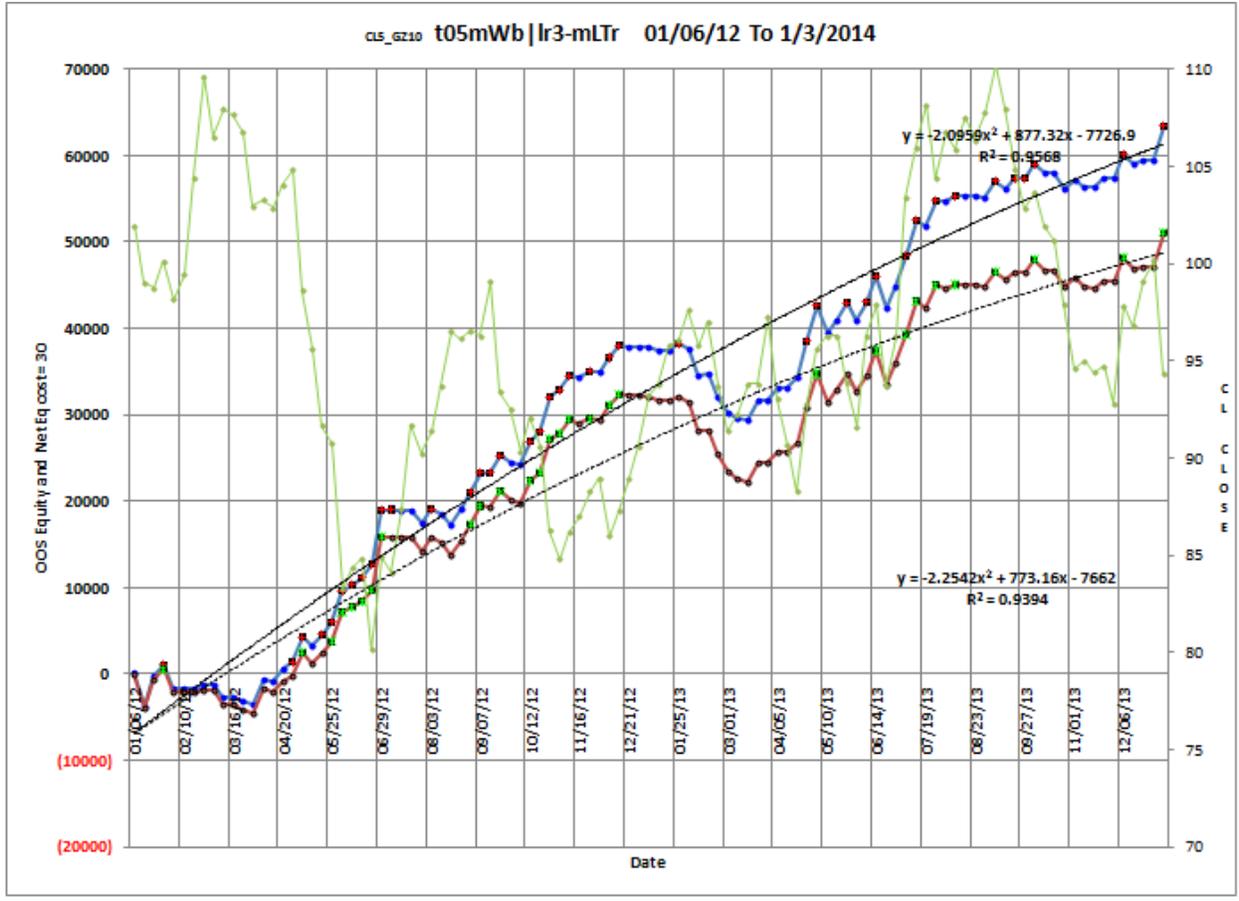


Figure 2 CL GZ System 5 minute bar chart from 12/30/13-1/03/2014

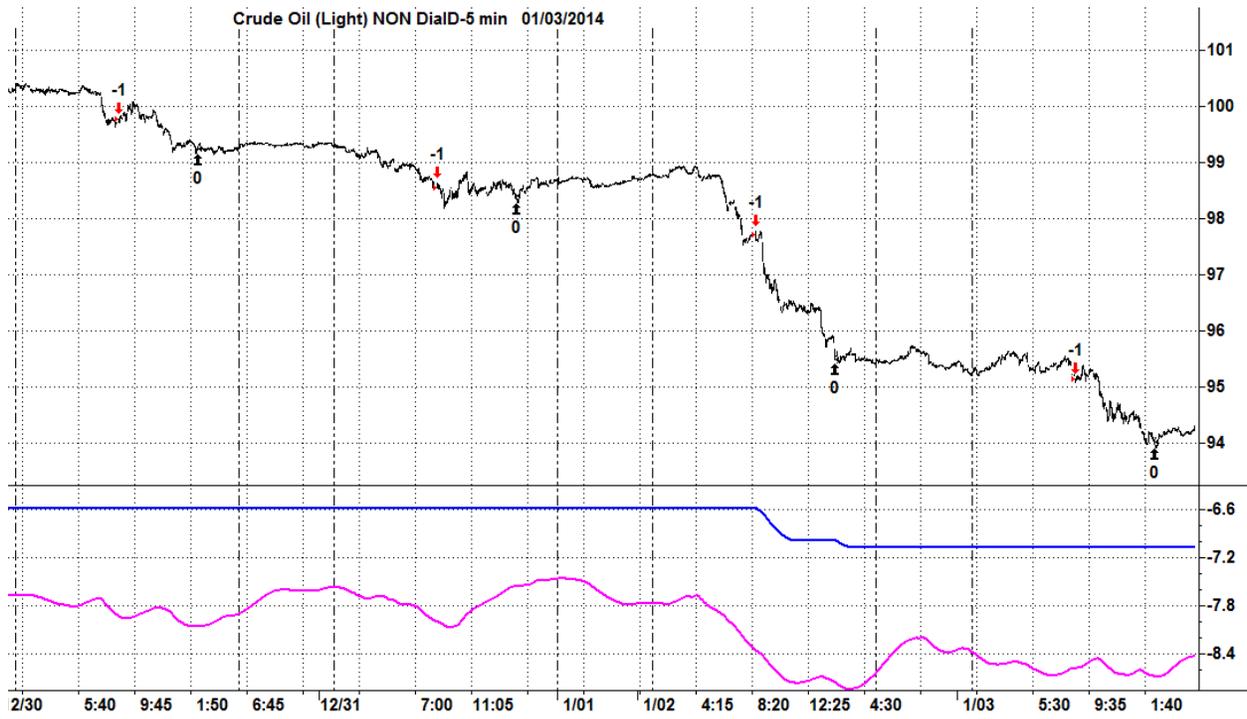


Figure 3 Partial output of the Walk Forward Metric Explorer (WFME)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	CLS-GZ10	s01/06/12	e01/03/14	#105	AnyTnp				a(8366)	s17509	f1922										c=\$ 30	
2	Filter-Metric	tOnp	aOnp	aOTrd	aOHf	B0	%P	t	std	lLp	eqDD	lr	#	eqTrn	eqV^3	eqR2	Dev^2	Blw	BE	eff	tOnpNet	Prob
3	t05mWb-mLTr	65710	714	153.2	4.7	(2.0)	61	3.43	2000	-4390	-9270	7	92	705	5	92	6106	17	31.3	0.74	52840	2.36E-04
4	t05mWb lr3-mLTr	63440	697	153.6	4.5	1.4	60	3.49	1904	-3760	-8860	4	91	655	555	95	4574	12	29.8	0.71	51050	3.45E-04
5	t05mLTr-mLTr	49940	595	144.3	4.1	(9.2)	57	3.13	1741	-3130	-9350	9	84	521	-88	85	6662	21	34.3	0.82	39560	3.10E-03
6	t05mWT LT-e-3	47800	498	111.4	4.5	(0.4)	64	2.83	1724	-3480	-6610	3	96	490	-412	92	4286	14	48	0.53	34930	6.70E-03

The WFME Filter Output Columns are defined as follows:

Row 1 EC1FadmV is the strategy abbreviation, First OOS Week End Date(1/6/12), Last OOS Week End Date(01/03/14), **Number of weeks(#105)** a=average of bootstrap random picks. s= standard deviation of bootstrap random picks. f=number of different filters examined. c= slippage and round trip trade cost(c=\$30).

Filter = The filter that was run. Row 3 filter **t5 mWb|lr3-mLTr**

The t5 mWb|lr3-mLTr filter produced the following average 105 week statistics on row 3.

tOnp = Total out-of-sample(oos) net profit for these 105 weeks.

aOsp = Average oos net profit for the 105 weeks

aOTrd = Average oos profit per trade

aO#T = Average number of oos trades per week

B0 = The 105 week trend of the out-of-sample weekly profits

%P = The percentage of oos weeks that were profitable

t = The student t statistic for the 105 weekly oos profits. The higher the t statistic the higher the probability that this result was not due to pure chance

std = The standard deviation of the 105 weekly oos profits

lLp = The largest losing oos period(week)

eqDD = The oos equity drawdown

lr = The largest number of losing oos weeks in a row

= The number of weeks this filter produced a weekly result. Note for some weeks there can be no strategy inputs that satisfy a given filter's criteria.

eqTrn = The straight line trend of the oos gross profit equity curve in \$/week.

eqV^3 = The ending velocity of 3rd order polynomial that is fit to the equity curve

eqR2 = The correlation coefficient(r^2) of a straight line fit to the equity curve

Dev^2 = A measure of equity curve smoothness. The square root of the average [(equity curve minus a straight line)²]

Blw = The maximum number of weeks the oos equity curve failed to make a new high.

BE = Break even weeks. Assuming the average and standard deviation are from a normal distribution, this is the number of weeks you would have to trade to have a 98% probability that your oos equity is above zero.

eff = Efficiency. The average daily out-of-sample profit divided by the average daily in-sample profit.

tOnpNet = Total out-of-sample net profit(tOnpNet) minus the total trade cost.
$$tOnpNet = tOnp - (\text{Number of trade weeks}) * aOnT * \text{Cost}.$$

Prob = the probability that the filter's tOnpNet was due to pure chance.