Trading 1Min Bar Euro Futures Using
The Nth order Fixed Memory Polynomial Velocity Strategy
August 1 2011 –July 29 2016

Working Paper August 2016

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.

In previous working papers at http://www.meyersanalytics.com/papers.php we showed how the application of a price curve generated by the Nth Order Fixed Memory Polynomial Velocity could be used to develop a strategy to buy and sell futures intraday. The reasoning behind this type of system was to only trade when the price trend velocity was above a certain threshold. Many times prices meander around without any notable trend and this is considered noise. During these times we do not wish to trade because of the cost of whipsaw losses that would occur from this type of price action. When a price trend finally starts, the velocity of that price trend moves above a minimum threshold noise value. Thus the velocity system would only issue a trade when certain velocity thresholds above "noise" levels are crossed.

The velocity system that we will use here to trade the Euro futures contract is called the nth Order Adaptive Polynomial Velocity Strategy. The word "Adaptive" is used because the polynomial inputs change over time, adapting to the changing trading patterns of the Euro futures contract. The nth Order Adaptive Polynomial Velocity Strategy has a number of unknown inputs that we have to determine before we can use this strategy to trade. These unknown inputs are the polynomial order or degree, the optimum number of lookback prices we need to determine the co-efficients of the polynomial and finally the velocity thresholds. Here we will use Walk Forward Optimization and out-of-sample testing to determine the "best" polynomial inputs as well as how these inputs should change over time. We will use the nth Order Adaptive Polynomial Velocity System to trade the Euro futures contract on an intraday basis using one minute bar price data To test this strategy we will use one minute bar prices of the Euro futures contract(EC) traded on the CME/Globex from August 1, 2011 to July 29, 2016.

The nth Order Fixed Memory System Defined

The least squares forecast nth order fixed memory polynomial velocity is constructed by solving for the coefficients $\beta_0, \beta_1, \beta_2, \beta_3 \ldots \beta_n$ for the discrete orthogonal Legendre polynomials each day using the last **N bars** of closing prices and the equation for β_j shown in the **"Math"** appendix at the end of this working paper. Then **Velocity(T+1)** is constructed from the equation shown in the "Math" appendix and plotted under the price chart.

Due to polynomial mathematics, the Velocity of the 2nd , 3rd and 4th order degree polynomial curve changes faster than it's corresponding first order degree polynomial curve velocity. Whether higher order polynomial velocities are an advantage or not, will be determined by the computer when we use a walk forward optimization technique described below.

At each bar we calculate the nth order degree(1st through 4th) fixed memory polynomial velocity from the formulas in the "Math" appendix. As will be shown below, walk forward optimization will determine the degree for the nth order polynomial velocity, the number of lookback prices, N, needed to compute the polynomial coefficients and the threshold amounts vup and vdn. When the nth order degree velocity is greater than the threshold amount ***vup*** we will go long. When the velocity is less than the threshold amount ***-vdn*** we will go short.

Buy Rule:

IF Velocity is greater than the threshold amount ***vup*** then buy at the market.

Sell Rule:

IF Velocity is less than the threshold amount ***-vdn*** then sell at the market.

Intraday Bars Exit Rule:

Close the position at 1500 CST. Trading volume on Globex dies down substantially by 1500 CST. No trades will be carried overnight.

Intraday Bars First Trade of Day Entry Rule:

Ignore all trade signals before **7:00**am . For the Buy and Sell rules above we have included a first trade of the day entry rule. Trading in the EC futures has changed a lot because of 24hr Globex trading. In particular trading starts a lot earlier in the morning right after midnight when Asia and then later Europe opens and then dies down. The EC volume starts to pick up again around 7am CST when US trading kicks in so we will have the strategy resume trading at 7am.

Discussion of Euro Prices

The Euro (EC) is traded on Globex. On Globex the EC is traded on a 23 hour basis. We have restricted our study to only trading the EC during the 7:00 to 1500 hours.

Testing The Polynomial Velocity System Using Walk Forward Optimization

There will be four strategy parameters to determine:

1. **degree**, degree=1 for straight line velocity, degree=2 for 2nd order velocity, etc.
2. ***N***, is the number of lookback bars of prices to calculate the **velocity.**
3. ***vup***, the threshold amount that velocity has to be greater than to issue a buy signal
4. ***vdn***, the threshold amount that velocity has to be less than to issue a sell signal

To test this system we will use one minute bar prices of the Euro (EC) futures contract traded on the CME/Globex and known by the symbol EC from August 1, 2011 to July 29, 2016.

We will test this strategy with the above EC 1 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 the nth Order Polynomial Velocity Strategy on the EC 1min data from August 1, 2011 to July 29, 2016. The PWFO will breakup and create 30 day calendar in-sample sections along with their corresponding one calendar week out-of-sample sections from the 261 weeks of EC (see Walk forward Testing below) creating 261 out-of-sample weeks.

What Is An 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 an ***in-sample*** or ***test 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 in-sample and out-of-sample sections. The PWFO prints out the in-sample sample performance metrics **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 in-sample 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 **http://meyersanalytics.com/Walk-Forward-Optimization.html** for a listing of these performance metrics).

For our computer run we will have the PWFO breakup the 261 weeks of EC one minute bar price data into 261 in-sample and 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 6/30/2011 to 7/29/2011 and the out-of-sample section would be the week following from 8/1/2011 to 8/5/2011.(our in-sample 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 7/7/2011 to 8/5/2011 and the week following out-of-sample section would be from 8/8/2011 to 8/12/2011. Etc.

The PWFO 261 in-sample/out-of-sample section dates are shown in **Table 1** below. We will then use another software product called the Walk Forward Performance Metric Explorer (WFME) on each of the 261 in-sample 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 ***(degree, N, vup, vdn)*** that will be used on the out-of-sample data. Detailed information about the WFME can be found at **http://meyersanalytics.com/wfme.html**

For the in-sample data we will run the TradeStation optimization engine on the 261 weeks of EC 1 min bars with the following ranges for the nth order fading memory polynomial velocity strategy input variables.

1. **pw=degree from 1 to 4**
2. **N from 20 to 70 in steps of 10.**
3. **vup from 0.25 to 3.5 steps of 0.25**
4. **vdn from 0.25 to 3.5 in steps of 0.25**
5. **Mult = 15000, iNorm=1 (See Appendix 3, the Normalization Multiplier)**

This will produce 4704 different cases or combinations of the input parameters for each of the 261 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 in-sample 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 in-sample 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 its 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 and 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 in-sample 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 input 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, neural net optimizations or combinatorial searches with no out-of-sample testing cause loses when traded in real time by mistaking chance fluctuations for genuine effects. Unfortunately it is human nature to extrapolate past performance to project future trading results and thus results from the chance fluctuations of 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 in-sample 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 in-sample/out-of-sample analysis many times. Why not just do the out-of-sample analysis once or just three times? 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 in-sample/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 its path ranges for N weeks in the future.

Finding The Strategy Input Parameters in The Walk Forward Test Sections

The PWFO generates a number of performance metrics in the in-sample section. (Please see **http://meyersanalytics.com/Walk-Forward-Optimization.html** for a listing of these performance metrics). 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 out-of-sample sections. The PWFO produces a total of 32 different performance metrics in the in-sample section. If we have 4704 different input variations or cases then the in-sample section consists of 32 columns of performance metrics for each of the 4704 different strategy inputs 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 a combination ***filter*** that is used in this paper with good out-of-sample results. High profit factors (**PF**) in the in-sample section usually mean poor performance in the out-of-sample-section. This is a kind of reversion to the mean. So in the in-sample section we eliminate all strategy input rows that have a **PF>3** . The PWFO generates the metric **mKr.** This metric is the **Modified K-ratio = Slope of Trade Equity Regression Line/ (The Average of The Absolute Values of The Equity at Each Trade Minus The Equity Regression Trend Line)**. Let us choose the 20 rows in the in-sample section that contain the highest(top) number of **mKr** from the rows that are left from the **PF** elimination. This particular filter will now leave 20 cases or rows in the in-sample section that satisfy the above filter conditions. Suppose for this filter, within the 20 in-sample rows that are left, we want the row that has the minimum metric **tLb** in the in-sample section**. tLb = Total Losing Bars.** This would produce a filter named **t20mKr|p<3|-tLb .** This in-sample 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 **t20mKr|p<3|-tLb** filter finds the strategy inputs parameters in each of the 261 in-sample sections and applies these inputs to each of the 261 out-of-sample sections. Using the filter in-sample strategy inputs on the 261 out-of-sample sections, the average 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 filter 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. A total of 11532 different metric filters were examined. We chose **Row 3** because it had a lower **BE**, **BLW** and **Dev^2** and a higher **eqR2** along with better statistics than the rows below it. More on this below and on how that number of filters combinations effect the probability that the filter chosen was or was not due to chance

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) over all out-of-sample weeks for a given filter like above. A mirror filter is created. However, instead of picking an out-of-sample net profit(OSNP) from a row that the filter picks, the mirror filter picks a ***random*** row's OSNP in each of the 261 PWFO files. Suppose we repeat this random row section 5000 times. Each of the 5000 mirror filters will choose a random row's OSNP of their own in each of the 261 PWFO files. At the end, each of the 5000 mirror filters will have 261 ***random*** OSNP's picked from the rows of the 261 PWFO files. The sum of the 261 random OSNP picks for each of the 5000 mirror filters 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 for each of ***our*** filter's tOnpNet. Thus given the mirror filter's bootstrap random tOnpNet average and standard deviation, we can calculate the probability of obtaining our filter's tOnpNet by pure chance alone. Since for this run we examined 11532(shown in Figure 3) 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 (11532) 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 tOnpNet of $37,400 of Row 7 is 11532 x 2.12 10^{-7} = 0.0024. This is much less than one case so it is improbable that our result was due to pure chance.

Results

Table 1 below presents a table of the 261 in-sample and out-of-sample windows, the selected optimum parameters and the weekly out-of-sample results using the filter described above.

Figure 1 presents a graph of the equity and net equity curves generated by using the filter on the 261 weeks ending 8/5/11 to 7/29/16. The equity curves are plotted from the Equity and Net Equity columns in Table 1. Plotted on the equity curves are 2nd Order Polynomial fits. 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 net equity curve with commissions and the green dots are the new highs in net equity. The grey line is the weekly EC prices superimposed on the equity chart.

Figure 2 Walk Forward Out-Of-Sample Performance for EC Nth Order Fixed Memory Polynomial Velocity System 1 minute bar chart of EC from 3/10/2016

Figure 3 Partial output of the Walk Forward Metric Performance Explorer (WFME)
Run on the 261 PWFO files of the EC 1min bars Nth Order Fixed Memory Polynomial Velocity

Discussion of System Performance

In Figure 3 Row 3 of the spreadsheet filter output are some statistics that are of interest for our filter. **BE** is the break even weeks. Assuming the trade average and standard deviation for this filter are from a normal distribution, this is how many weeks we need to trade this strategy so that we have a 98% probability that the equity paths after that number of weeks will be greater than zero. BE is 29 weeks for this filter. This means we would have to trade this strategy for at least 29 weeks to have a 98% probability that our equity would be positive. Another interesting statistic is **Blw**. Blw is the maximum number of weeks the OSNP equity curve failed to make a new high. Blw is 38 weeks for this filter. This means that 38 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 ***pw, N, vup and vdn*** take sudden jumps from high to low and back. This is the walk forward process quickly adapting to changing volatility conditions in the in-sample sample. In addition, notice how often ***degree*** changes from a straight line velocity with ***degree=1*** to a 2nd, 3rd and 4th order velocity with ***degree= 2, 3 and 4***. The 3rd and 4th order velocities, due to the higher order components, change much faster than the straight line velocity. When the data gets very noisy with a lot of spurious price movements, it's better to have the velocity change slower filtering out the noisy data. During other times when the noise level is not as much it is better to have the velocity break its ***vup*** and ***vdn*** barriers faster to get onboard a trend faster. This is what the filter is doing. When there is a lot of noise in the in-sample section it switches to the 1st or 2nd order curve velocity. When the noise level is lower in the in-sample section, it switches to the faster changing 3rd or 4th order curve velocity.

Using this filter, the strategy was able to generate $37,004 net equity after commissions and slippage trading one EC contract for 261 weeks. Note $20 roundtrip commission and slippage was subtracted from each trade and no positions were carried overnight. The largest losing week was -$1275 and the largest drawdown was -$3750. The longest time between new equity highs was 29 weeks.

In observing Table 1 we can see that this strategy and filter made trades from a low of no trades/week to a high of 7 trades/week with an average of 2 trades/week on the weeks it did trade. The strategy seemed to wait for really strong trends and then initiate a buy or sell. There were many weeks that had no trades. Out of the 261 out-of-sample weeks the filter only traded 142 of those weeks or 54% of the time with 58% of all trades profitable. In observing the Equity Curve plot in Figure 1 we can see that the equity did quite well in both big up and down moves of the EC. In observing the chart from 5/9/14 to 3/13/15 where EC declined from 1.4027 to 1.0613 we can see the strategy did well on this unusually large Euro decline. After the Brexit vote on 6/23/2016, the strategy did well in the volatile market from Friday 6/24 to Friday 7/1 only losing -$188 on 3 trades that week.

Given 24 hour trading of the Euro, restricting the strategy to trade only from 7am to 3pm CST caused the strategy to miss many profitable trends opportunities when Asia and then Europe opened trading in the early morning. Further research will include the A.M. time zones.

References

1. Efron, B., Tibshirani, R.J., (1993), "An Introduction to the Bootstrap", New York, Chapman & Hall/CRC.
2. Morrison, Norman "Introduction to Sequential Smoothing and Prediction", McGraw-Hill Book Company, New York, 1969.

Figure 1 Graph of Net Equity Curve Applying the Walk Forward Filter Each Week On EC 1min Bar Prices 08/05/11 – 7/29/16

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 grey line is the EC Weekly Closing prices superimposed on the Equity Chart.

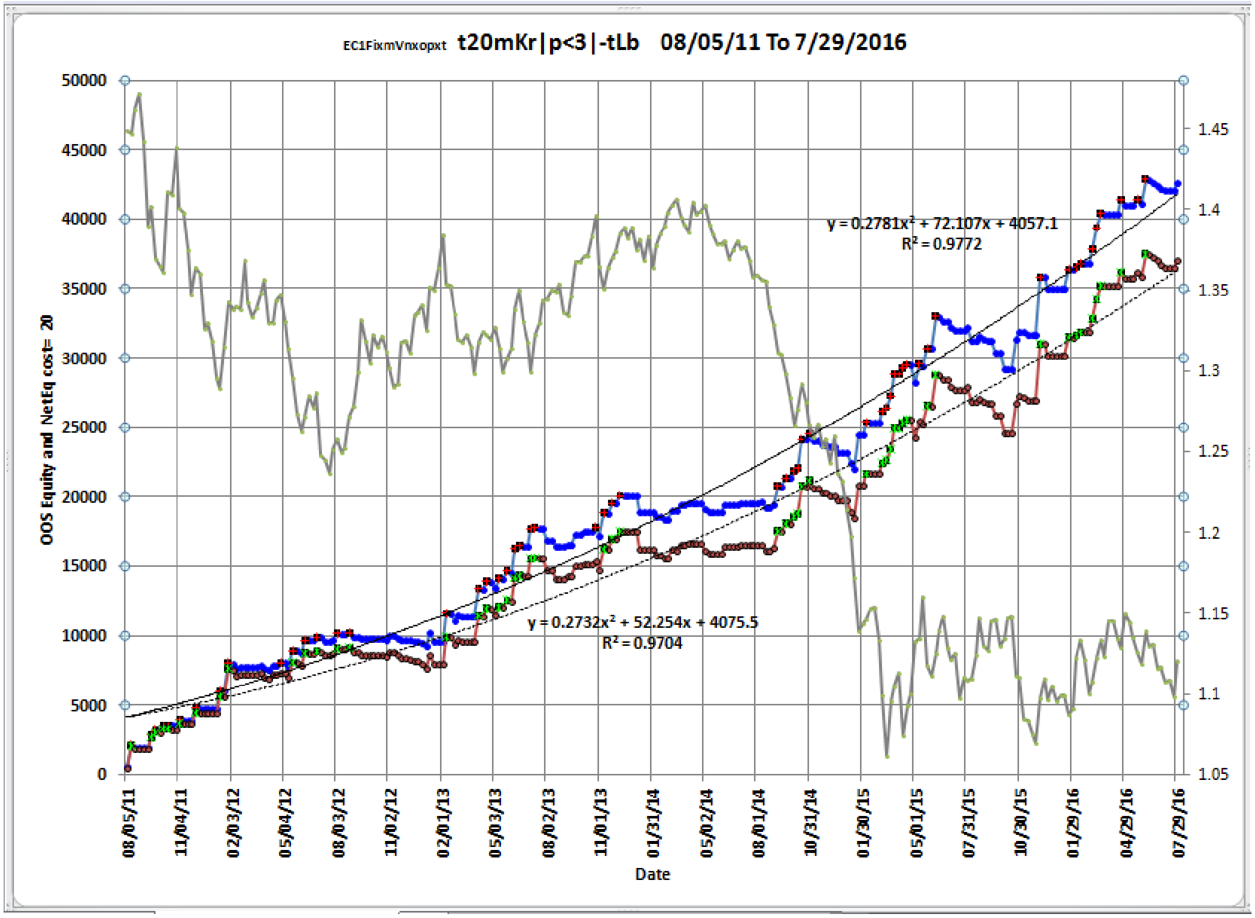


Figure 2 Walk Forward Out-Of-Sample Performance Summary for Nth Order Fixed Memory Polynomial Velocity System EC 1 minute bar chart from 3/10/16

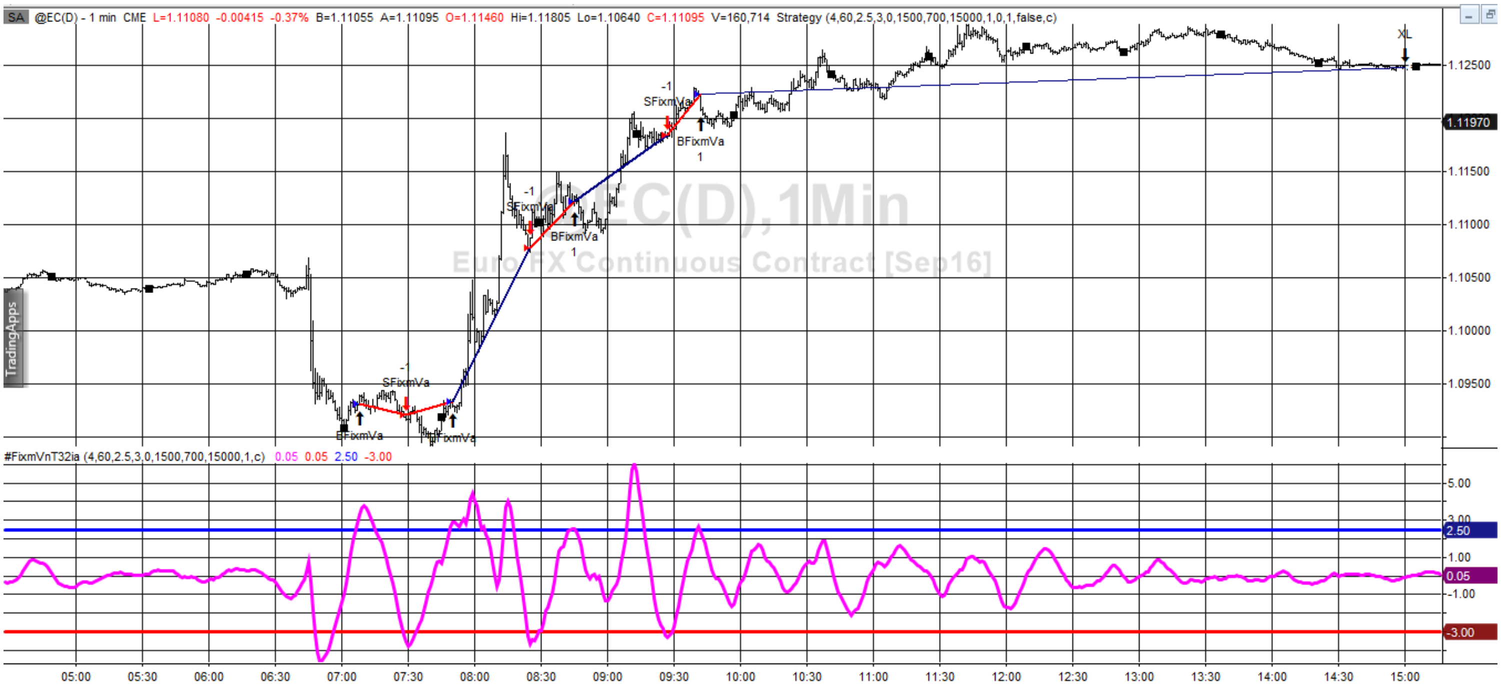
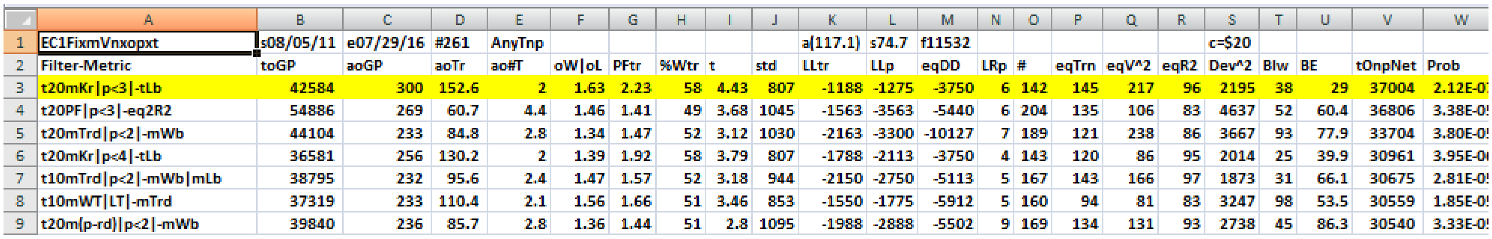


Figure 3_ Partial output of the Walk Forward Metric Performance Explorer (WFME)
EC1 min bars Nth Order Fixed Memory Velocity System



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	EC1FixmVnxopxt	s08/05/11	e07/29/16	#261	AnyTnp						a(117.1)	s74.7	f11532						c=$20				
2	Filter-Metric	toGP	aoGP	aoTr	ao#T	oW|oL	PFtr	%Wtr	t	std	LLtr	LLp	eqDD	LRp	#	eqTrn	eqV^2	eqR2	Dev^2	Blw	BE	tOnpNet	Prob
3	t20mKr|p<3|-tLb	42584	300	152.6	2	1.63	2.23	58	4.43	807	-1188	-1275	-3750	6	142	145	217	96	2195	38	29	37004	2.12E-0
4	t20PF|p<3|-eq2R2	54886	269	60.7	4.4	1.46	1.41	49	3.68	1045	-1563	-3563	-5440	6	204	135	106	83	4637	52	60.4	36806	3.38E-0
5	t20mTrd|p<2|-mWb	44104	233	84.8	2.8	1.34	1.47	52	3.12	1030	-2163	-3300	-10127	7	189	121	238	86	3667	93	77.9	33704	3.80E-0
6	t20mKr|p<4|-tLb	36581	256	130.2	2	1.39	1.92	58	3.79	807	-1788	-2113	-3750	4	143	120	86	95	2014	25	39.9	30961	3.95E-0
7	t10mTrd|p<2|-mWb|mLb	38795	232	95.6	2.4	1.47	1.57	52	3.18	944	-2150	-2750	-5113	5	167	143	166	97	1873	31	66.1	30675	2.81E-0
8	t10mWT|LT|-mTrd	37319	233	110.4	2.1	1.56	1.66	51	3.46	853	-1550	-1775	-5912	5	160	94	81	83	3247	98	53.5	30559	1.85E-0
9	t20m(p-rd)|p<2|-mWb	39840	236	85.7	2.8	1.36	1.44	51	2.8	1095	-1988	-2888	-5502	9	169	134	131	93	2738	45	86.3	30540	3.33E-0

The WFME Filter Output Columns are defined as follows:

Row 1 EC1FixmVnxopxt is the strategy abbreviation, First OOS Week End Date(8/5/11), Last OOS Week End Date(7/29/16), **Number of weeks**(#261) **a**=weekly average of bootstrap random picks. **s**= weekly standard deviation of bootstrap random picks. **f**=number of different filters examined. **c**= slippage and round trip trade cost(c=$20).

Filter = The filter that was run. Row10 filter **t20mKr|p<3|-tLb**

The t20mKr|p<3|-tLb filter produced the following average 261 week statistics on Row 7.

toGP = Total out-of-sample(oos) gross profit for these 261 weeks.

aoGP = Average oos gross profit for the 261 weeks

aoTr = Average oos profit per trade

ao#T = Average number of oos trades per week

oW|oL = The average oos winning trade divided by the average oos losing trade

PFtr = Trade profit factor. Total of oos winning trades / Total of oos loosing trades

%Wtr = percent of all oos trades that were winning trades

t = The student t statistic for the 261 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 261 weekly oos profits

LLtr = The largest losing oos trade in all oos periods,

llp = The largest losing oos period(week)

eqDD = The oos equity drawdown

LRp = 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^2 = The ending velocity of 2nd order polynomial that is fit to the oos equity curve

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

tOnpNet = Total out-of-sample net profit(tOnpNet) minus the total trade cost.
tOnpNet=tOnp – (Number of trade weeks)*aOnT*Cost.

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

Table 1 Walk Forward Out-Of-Sample Performance Summary for **EC Nth Order Fixed Memory Polynomial Velocity System**

EC-1 min bars 8/5/2011 - 07/29/2016. The input values *degree(pw), N, vup, vdn* are the values found from applying the filter to the in-sample section optimization runs.

Filter= t20mKr|P<3|-tLb PF<=3 and top 20 mKr and then minimum tIB

osnp = Weekly Out-of-sample gross profit in $
Equity = Running Sum of weekly out-of-sample gross profits $
NOnp$20 = Weekly Out-Of-Sample Net Profit in $ = **osnp-ont*20**.
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 drawdown in the out-of-sample section in $.
ont = The number of trades in the out-of-sample week.
pw= degree, degree=1 for straight line velocity, degree=2 for 2^{nd} order velocity, etc.
N = N the lookback period
vup, the threshold amount that velocity has to be greater than to issue a buy signal
vdn, the threshold amount that velocity has to be less than to issue a sell signal
Note: Blank rows indicate that no out-of-sample trades were made that week

In-Sample Dates			Out-Of_Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
06/30/11	To	07/29/11	08/01/11	To	08/05/11	538	538	498	498	2	0	0	1	50	3.25	2
07/07/11	To	08/05/11	08/08/11	To	08/12/11	1613	2151	1573	2071	2	0	0	2	60	3.25	2.25
07/14/11	To	08/12/11	08/15/11	To	08/19/11	(225)	1926	(245)	1826	1	-225	-225	3	50	2.75	2.75
07/21/11	To	08/19/11	08/22/11	To	08/26/11		1926		1826				3	50	2.75	2.75
07/28/11	To	08/26/11	08/29/11	To	09/02/11		1926		1826				3	50	3.25	3
08/04/11	To	09/02/11	09/05/11	To	09/09/11		1926		1826				2	40	2.75	3.25
08/11/11	To	09/09/11	09/12/11	To	09/16/11	913	2839	853	2679	3	0	0	4	60	2.75	3
08/18/11	To	09/16/11	09/19/11	To	09/23/11	388	3227	368	3047	1	0	0	3	50	2.5	3.5
08/25/11	To	09/23/11	09/26/11	To	09/30/11		3227		3047				3	50	2.5	3.5
09/01/11	To	09/30/11	10/03/11	To	10/07/11	275	3502	255	3302	1	0	0	2	60	2.5	3.5
09/08/11	To	10/07/11	10/10/11	To	10/14/11	25	3527	5	3307	1	0	0	2	60	2.5	3.5
09/15/11	To	10/14/11	10/17/11	To	10/21/11		3527		3307				1	40	3.5	2.5
09/22/11	To	10/21/11	10/24/11	To	10/28/11		3527		3307				4	30	3.25	2.75
09/29/11	To	10/28/11	10/31/11	To	11/04/11	388	3915	348	3655	2	-125	-125	4	40	3.5	2.5
10/06/11	To	11/04/11	11/07/11	To	11/11/11		3915		3655				2	50	2.75	3.5
10/13/11	To	11/11/11	11/14/11	To	11/18/11		3915		3655				3	20	3.25	3.25
10/20/11	To	11/18/11	11/21/11	To	11/25/11		3915		3655				2	70	2.75	2.5
10/27/11	To	11/25/11	11/28/11	To	12/02/11	888	4803	808	4463	4	-338	-338	4	60	2.75	2.75
11/03/11	To	12/02/11	12/05/11	To	12/09/11		4803		4463				2	20	3	3
11/10/11	To	12/09/11	12/12/11	To	12/16/11		4803		4463				2	60	2	3
11/17/11	To	12/16/11	12/19/11	To	12/23/11		4803		4463				2	60	2	3
11/24/11	To	12/23/11	12/26/11	To	12/30/11		4803		4463				1	50	3.25	2.75
12/01/11	To	12/30/11	01/02/12	To	01/06/12		4803		4463				3	50	2.5	2.25
12/08/11	To	01/06/12	01/09/12	To	01/13/12	1225	6028	1185	5648	2	0	0	4	60	1.75	2
12/15/11	To	01/13/12	01/16/12	To	01/20/12		6028		5648				2	60	1.75	2.5
12/22/11	To	01/20/12	01/23/12	To	01/27/12	2013	8041	1973	7621	2	0	0	3	60	1.25	2.5
12/29/11	To	01/27/12	01/30/12	To	02/03/12	(75)	7966	(115)	7506	2	-525	-525	1	70	2	1.25
01/05/12	To	02/03/12	02/06/12	To	02/10/12	(363)	7603	(423)	7083	3	-150	-363	4	50	2.75	2.25
01/12/12	To	02/10/12	02/13/12	To	02/17/12	175	7778	155	7238	1	0	0	1	70	2	1.25
01/19/12	To	02/17/12	02/20/12	To	02/24/12		7778		7238				4	20	2.25	1.75
01/26/12	To	02/24/12	02/27/12	To	03/02/12		7778		7238				3	70	1.75	3
02/02/12	To	03/02/12	03/05/12	To	03/09/12		7778		7238				1	60	1.75	2.5
02/09/12	To	03/09/12	03/12/12	To	03/16/12		7778		7238				4	50	2.5	2.75

In-Sample Dates			Out-Of_Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
02/16/12	To	03/16/12	03/19/12	To	03/23/12	63	7841	43	7281	1	0	0	4	60	2.25	1.75
02/23/12	To	03/23/12	03/26/12	To	03/30/12	(213)	7628	(233)	7048	1	-213	-213	4	60	2.25	1.75
03/01/12	To	03/30/12	04/02/12	To	04/06/12	(113)	7515	(193)	6855	4	-438	-775	4	20	2.25	2
03/08/12	To	04/06/12	04/09/12	To	04/13/12	400	7915	380	7235	1	0	0	2	70	1.5	1.75
03/15/12	To	04/13/12	04/16/12	To	04/20/12		7915		7235				1	60	1.75	1.75
03/22/12	To	04/20/12	04/23/12	To	04/27/12	138	8053	118	7353	1	0	0	4	20	1.75	2.5
03/29/12	To	04/27/12	04/30/12	To	05/04/12		8053		7353				4	20	2	2.5
04/05/12	To	05/04/12	05/07/12	To	05/11/12	(313)	7740	(353)	7000	2	-163	-313	1	50	1.75	1
04/12/12	To	05/11/12	05/14/12	To	05/18/12	1188	8928	1088	8088	5	-88	-138	2	20	1.5	1.5
04/19/12	To	05/18/12	05/21/12	To	05/25/12		8928		8088				4	70	2.25	1.75
04/26/12	To	05/25/12	05/28/12	To	06/01/12	(188)	8740	(208)	7880	1	-188	-188	1	60	1.75	1.75
05/03/12	To	06/01/12	06/04/12	To	06/08/12	925	9665	885	8765	2	0	0	1	60	1.25	2.25
05/10/12	To	06/08/12	06/11/12	To	06/15/12		9665		8765				4	20	2.5	3.25
05/17/12	To	06/15/12	06/18/12	To	06/22/12	(25)	9640	(85)	8680	3	-475	-475	3	30	2.25	1.75
05/24/12	To	06/22/12	06/25/12	To	06/29/12	225	9865	205	8885	1	0	0	1	60	1.5	1.5
05/31/12	To	06/29/12	07/02/12	To	07/06/12		9865		8885				4	70	2.5	2.5
06/07/12	To	07/06/12	07/09/12	To	07/13/12	(238)	9627	(258)	8627	1	-238	-238	4	40	2.5	2
06/14/12	To	07/13/12	07/16/12	To	07/20/12		9627		8627				1	50	1.75	2.75
06/21/12	To	07/20/12	07/23/12	To	07/27/12	75	9702	55	8682	1	0	0	1	40	2.25	2.5
06/28/12	To	07/27/12	07/30/12	To	08/03/12	450	10152	410	9092	2	0	0	1	50	2.5	2
07/05/12	To	08/03/12	08/06/12	To	08/10/12		10152		9092				3	30	2.25	2.75
07/12/12	To	08/10/12	08/13/12	To	08/17/12		10152		9092				3	30	2.25	2.75
07/19/12	To	08/17/12	08/20/12	To	08/24/12	63	10215	43	9135	1	0	0	3	30	2.25	2.75
07/26/12	To	08/24/12	08/27/12	To	08/31/12	(263)	9952	(303)	8832	2	-388	-388	3	30	2.25	3.5
08/02/12	To	08/31/12	09/03/12	To	09/07/12		9952		8832				1	70	1.75	1.75
08/09/12	To	09/07/12	09/10/12	To	09/14/12	(163)	9789	(203)	8629	2	-513	-513	1	50	1.5	2
08/16/12	To	09/14/12	09/17/12	To	09/21/12		9789		8629				4	40	3	2.25
08/23/12	To	09/21/12	09/24/12	To	09/28/12		9789		8629				4	40	3	2.25
08/30/12	To	09/28/12	10/01/12	To	10/05/12		9789		8629				4	40	3	2.25
09/06/12	To	10/05/12	10/08/12	To	10/12/12		9789		8629				1	60	1.75	1.5
09/13/12	To	10/12/12	10/15/12	To	10/19/12		9789		8629				4	70	2.25	1.5
09/20/12	To	10/19/12	10/22/12	To	10/26/12	(75)	9714	(95)	8534	1	-75	-75	4	60	1.75	1.5
09/27/12	To	10/26/12	10/29/12	To	11/02/12	288	10002	268	8802	1	0	0	4	30	2	1.25
10/04/12	To	11/02/12	11/05/12	To	11/09/12		10002		8802				4	40	1.75	1.25
10/11/12	To	11/09/12	11/12/12	To	11/16/12	(163)	9839	(203)	8599	2	-325	-325	2	30	1.5	1.25
10/18/12	To	11/16/12	11/19/12	To	11/23/12	(138)	9701	(178)	8421	2	-88	-138	4	20	1.25	1.25
10/25/12	To	11/23/12	11/26/12	To	11/30/12	25	9726	5	8426	1	0	0	2	30	1.5	1.25
11/01/12	To	11/30/12	12/03/12	To	12/07/12	(88)	9638	(108)	8318	1	-88	-88	1	30	1	1.75
11/08/12	To	12/07/12	12/10/12	To	12/14/12	(38)	9600	(78)	8240	2	-250	-250	1	50	1	1.25
11/15/12	To	12/14/12	12/17/12	To	12/21/12	(63)	9537	(83)	8157	1	-63	-63	4	60	1.75	1.25
11/22/12	To	12/21/12	12/24/12	To	12/28/12	(100)	9437	(160)	7997	3	-150	-250	4	70	1.75	1.25
11/29/12	To	12/28/12	12/31/12	To	01/04/13	(200)	9237	(300)	7697	5	-488	-838	4	40	1.25	1.75
12/06/12	To	01/04/13	01/07/13	To	01/11/13	975	10212	935	8632	2	0	0	1	40	1.75	1.5
12/13/12	To	01/11/13	01/14/13	To	01/18/13	(638)	9574	(698)	7934	3	-325	-638	1	70	2.25	1
12/20/12	To	01/18/13	01/21/13	To	01/25/13		9574		7934				1	60	1.25	2
12/27/12	To	01/25/13	01/28/13	To	02/01/13	50	9624	(10)	7924	3	-163	-163	1	50	1	2.25
01/03/13	To	02/01/13	02/04/13	To	02/08/13	2000	11624	1960	9884	2	0	0	3	40	1.25	2
01/10/13	To	02/08/13	02/11/13	To	02/15/13		11624		9884				2	50	3	2.25
01/17/13	To	02/15/13	02/18/13	To	02/22/13	(488)	11136	(528)	9356	2	-300	-488	4	30	2	2.25
01/24/13	To	02/22/13	02/25/13	To	03/01/13	425	11561	385	9741	2	0	0	3	60	2.25	1.75
01/31/13	To	03/01/13	03/04/13	To	03/08/13	(163)	11398	(203)	9538	2	-313	-313	4	30	2.25	2.25
02/07/13	To	03/08/13	03/11/13	To	03/15/13		11398		9538				4	70	2.25	3
02/14/13	To	03/15/13	03/18/13	To	03/22/13		11398		9538				4	70	2.25	3
02/21/13	To	03/22/13	03/25/13	To	03/29/13		11398		9538				4	70	2.25	2.25

In-Sample Dates			Out-Of_Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
02/28/13	To	03/29/13	04/01/13	To	04/05/13	1975	13373	1915	11453	3	-75	-75	3	40	2	2.5
03/07/13	To	04/05/13	04/08/13	To	04/12/13		13373		11453				1	50	1.75	1
03/14/13	To	04/12/13	04/15/13	To	04/19/13	538	13911	518	11971	1	0	0	1	60	1.75	1.5
03/21/13	To	04/19/13	04/22/13	To	04/26/13		13911		11971				1	60	2	1.5
03/28/13	To	04/26/13	04/29/13	To	05/03/13	(438)	13473	(478)	11493	2	-850	-850	1	60	2.25	1.5
04/04/13	To	05/03/13	05/06/13	To	05/10/13	625	14098	565	12058	3	-213	-213	2	70	2	1.25
04/11/13	To	05/10/13	05/13/13	To	05/17/13		14098		12058				4	40	2	3
04/18/13	To	05/17/13	05/20/13	To	05/24/13	538	14636	518	12576	1	0	0	1	30	1.75	3.25
04/25/13	To	05/24/13	05/27/13	To	05/31/13	(113)	14523	(133)	12443	1	-113	-113	1	30	1.75	3.25
05/02/13	To	05/31/13	06/03/13	To	06/07/13	1738	16261	1698	14141	2	0	0	2	40	1.75	3.25
05/09/13	To	06/07/13	06/10/13	To	06/14/13	225	16486	205	14346	1	0	0	2	70	1.5	2
05/16/13	To	06/14/13	06/17/13	To	06/21/13		16486		14346				2	60	1.5	2.5
05/23/13	To	06/21/13	06/24/13	To	06/28/13		16486		14346				1	30	2	2.5
05/30/13	To	06/28/13	07/01/13	To	07/05/13	1238	17724	1218	15564	1	0	0	3	30	3	2.25
06/06/13	To	07/05/13	07/08/13	To	07/12/13	38	17762	18	15582	1	0	0	1	60	2.25	1.75
06/13/13	To	07/12/13	07/15/13	To	07/19/13		17762		15582				2	60	2	2.5
06/20/13	To	07/19/13	07/22/13	To	07/26/13		17762		15582				2	70	1.75	2
06/27/13	To	07/26/13	07/29/13	To	08/02/13	(838)	16924	(878)	14704	2	-488	-838	2	70	1.75	2
07/04/13	To	08/02/13	08/05/13	To	08/09/13		16924		14704				4	60	2.25	3.5
07/11/13	To	08/09/13	08/12/13	To	08/16/13	(525)	16399	(565)	14139	2	-1063	-1063	2	30	2.25	2.75
07/18/13	To	08/16/13	08/19/13	To	08/23/13		16399		14139				2	50	1.75	2.75
07/25/13	To	08/23/13	08/26/13	To	08/30/13		16399		14139				2	50	1.75	2.75
08/01/13	To	08/30/13	09/02/13	To	09/06/13	200	16599	180	14319	1	0	0	3	50	1.75	2.75
08/08/13	To	09/06/13	09/09/13	To	09/13/13		16599		14319				2	20	2.25	2.75
08/15/13	To	09/13/13	09/16/13	To	09/20/13	700	17299	680	14999	1	0	0	1	30	1.75	2.25
08/22/13	To	09/20/13	09/23/13	To	09/27/13		17299		14999				1	70	1.5	1.25
08/29/13	To	09/27/13	09/30/13	To	10/04/13	238	17537	198	15197	2	0	0	2	70	1	2.25
09/05/13	To	10/04/13	10/07/13	To	10/11/13		17537		15197				2	70	1.5	2.25
09/12/13	To	10/11/13	10/14/13	To	10/18/13		17537		15197				4	20	1.75	2
09/19/13	To	10/18/13	10/21/13	To	10/25/13	275	17812	195	15392	4	-238	-263	3	50	1.25	1.75
09/26/13	To	10/25/13	10/28/13	To	11/01/13	(600)	17212	(660)	14732	3	-388	-600	4	20	2	2
10/03/13	To	11/01/13	11/04/13	To	11/08/13	1613	18825	1533	16265	4	-50	-50	4	50	1.75	2.25
10/10/13	To	11/08/13	11/11/13	To	11/15/13		18825		16265				3	20	2	3.25
10/17/13	To	11/15/13	11/18/13	To	11/22/13	725	19550	705	16970	1	0	0	1	50	1	1
10/24/13	To	11/22/13	11/25/13	To	11/29/13		19550		16970				1	50	1	2.5
10/31/13	To	11/29/13	12/02/13	To	12/06/13	538	20088	518	17488	1	0	0	2	40	3.5	2.25
11/07/13	To	12/06/13	12/09/13	To	12/13/13		20088		17488				1	20	3	3
11/14/13	To	12/13/13	12/16/13	To	12/20/13		20088		17488				2	70	1.25	2.75
11/21/13	To	12/20/13	12/23/13	To	12/27/13		20088		17488				1	30	1.5	1.5
11/28/13	To	12/27/13	12/30/13	To	01/03/14		20088		17488				1	30	1.5	1.5
12/05/13	To	01/03/14	01/06/14	To	01/10/14	(1225)	18863	(1285)	16203	3	-475	-1225	2	50	1.75	1.5
12/12/13	To	01/10/14	01/13/14	To	01/17/14		18863		16203				2	70	1.25	2
12/19/13	To	01/17/14	01/20/14	To	01/24/14		18863		16203				3	70	1.75	2.25
12/26/13	To	01/24/14	01/27/14	To	01/31/14		18863		16203				3	20	2.5	3.25
01/02/14	To	01/31/14	02/03/14	To	02/07/14	(300)	18563	(380)	15823	4	-513	-513	3	20	2.5	3.25
01/09/14	To	02/07/14	02/10/14	To	02/14/14		18563		15823				1	40	2	2.25
01/16/14	To	02/14/14	02/17/14	To	02/21/14	(188)	18375	(208)	15615	1	-188	-188	1	60	1	1.25
01/23/14	To	02/21/14	02/24/14	To	02/28/14		18375		15615				3	20	1.75	3.5
01/30/14	To	02/28/14	03/03/14	To	03/07/14	638	19013	598	16213	2	-50	-50	3	20	1.75	3.5
02/06/14	To	03/07/14	03/10/14	To	03/14/14	0	19013	(40)	16173	2	-25	-25	3	20	2	2.75
02/13/14	To	03/14/14	03/17/14	To	03/21/14	425	19438	405	16578	1	0	0	2	60	2	1.5
02/20/14	To	03/21/14	03/24/14	To	03/28/14		19438		16578				2	60	2.75	1.5
02/27/14	To	03/28/14	03/31/14	To	04/04/14	138	19576	98	16676	2	-238	-238	4	30	3.5	1.75
03/06/14	To	04/04/14	04/07/14	To	04/11/14		19576		16676				1	70	2	1.25

In-Sample Dates			Out-Of_Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
03/13/14	To	04/11/14	04/14/14	To	04/18/14		19576		16676				1	70	1.75	1.25
03/20/14	To	04/18/14	04/21/14	To	04/25/14		19576		16676				2	40	1.25	1.25
03/27/14	To	04/25/14	04/28/14	To	05/02/14	(475)	19101	(535)	16141	3	-638	-638	1	60	1	1
04/03/14	To	05/02/14	05/05/14	To	05/09/14	(175)	18926	(215)	15926	2	-938	-938	1	30	1	2.25
04/10/14	To	05/09/14	05/12/14	To	05/16/14		18926		15926				1	70	1	1.25
04/17/14	To	05/16/14	05/19/14	To	05/23/14		18926		15926				1	70	1	1.25
04/24/14	To	05/23/14	05/26/14	To	05/30/14		18926		15926				1	70	1	1.25
05/01/14	To	05/30/14	06/02/14	To	06/06/14	550	19476	510	16436	2	-288	-288	1	60	1.25	1.25
05/08/14	To	06/06/14	06/09/14	To	06/13/14		19476		16436				1	30	1.5	2.75
05/15/14	To	06/13/14	06/16/14	To	06/20/14		19476		16436				3	60	1	1.5
05/22/14	To	06/20/14	06/23/14	To	06/27/14		19476		16436				1	40	1.25	1
05/29/14	To	06/27/14	06/30/14	To	07/04/14	125	19601	105	16541	1	0	0	1	40	1.25	1
06/05/14	To	07/04/14	07/07/14	To	07/11/14		19601		16541				4	20	1.75	2.75
06/12/14	To	07/11/14	07/14/14	To	07/18/14		19601		16541				3	20	1.75	1.5
06/19/14	To	07/18/14	07/21/14	To	07/25/14		19601		16541				3	70	1	1
06/26/14	To	07/25/14	07/28/14	To	08/01/14	0	19601	(20)	16521	1	0	0	1	70	0.5	1.5
07/03/14	To	08/01/14	08/04/14	To	08/08/14	38	19639	(2)	16519	2	0	0	1	70	0.5	1.25
07/10/14	To	08/08/14	08/11/14	To	08/15/14	(375)	19264	(415)	16104	2	-350	-375	3	40	1	2
07/17/14	To	08/15/14	08/18/14	To	08/22/14	0	19264	(20)	16084	1	0	0	2	70	1.75	1
07/24/14	To	08/22/14	08/25/14	To	08/29/14	225	19489	205	16289	1	0	0	1	60	1.75	0.75
07/31/14	To	08/29/14	09/01/14	To	09/05/14	1263	20752	1243	17532	1	0	0	1	20	2	1
08/07/14	To	09/05/14	09/08/14	To	09/12/14		20752		17532				1	30	1.5	1.25
08/14/14	To	09/12/14	09/15/14	To	09/19/14	600	21352	560	18092	2	0	0	4	40	2.5	1.25
08/21/14	To	09/19/14	09/22/14	To	09/26/14		21352		18092				3	30	2.75	2
08/28/14	To	09/26/14	09/29/14	To	10/03/14	475	21827	455	18547	1	0	0	3	30	2.75	2
09/04/14	To	10/03/14	10/06/14	To	10/10/14	238	22065	218	18765	1	0	0	3	30	2.75	3
09/11/14	To	10/10/14	10/13/14	To	10/17/14	2075	24140	1995	20760	4	0	0	2	30	1.75	3.25
09/18/14	To	10/17/14	10/20/14	To	10/24/14		24140		20760				4	70	3	3.5
09/25/14	To	10/24/14	10/27/14	To	10/31/14	438	24578	418	21178	1	0	0	4	70	3	3.5
10/02/14	To	10/31/14	11/03/14	To	11/07/14	(513)	24065	(553)	20625	2	-525	-525	4	70	3	3.5
10/09/14	To	11/07/14	11/10/14	To	11/14/14		24065		20625				1	50	3	1.5
10/16/14	To	11/14/14	11/17/14	To	11/21/14	(288)	23777	(328)	20297	2	-313	-313	2	70	2.25	0.75
10/23/14	To	11/21/14	11/24/14	To	11/28/14		23777		20297				2	20	2.5	1.5
10/30/14	To	11/28/14	12/01/14	To	12/05/14	(188)	23589	(208)	20089	1	-188	-188	1	70	1.25	3.5
11/06/14	To	12/05/14	12/08/14	To	12/12/14		23589		20089				4	30	3.5	2.25
11/13/14	To	12/12/14	12/15/14	To	12/19/14	(350)	23239	(370)	19719	1	-350	-350	4	30	2.75	2.75
11/20/14	To	12/19/14	12/22/14	To	12/26/14		23239		19719				3	20	3	2.75
11/27/14	To	12/26/14	12/29/14	To	01/02/15		23239		19719				3	20	3	2.75
12/04/14	To	01/02/15	01/05/15	To	01/09/15	(813)	22426	(833)	18886	1	-813	-813	3	20	3	2.75
12/11/14	To	01/09/15	01/12/15	To	01/16/15	(363)	22063	(403)	18483	2	-375	-375	3	50	1.5	2
12/18/14	To	01/16/15	01/19/15	To	01/23/15	2400	24463	2340	20823	3	-188	-188	2	70	1.75	1.5
12/25/14	To	01/23/15	01/26/15	To	01/30/15		24463		20823				4	40	2.75	2.75
01/01/15	To	01/30/15	02/02/15	To	02/06/15	888	25351	828	21651	3	0	0	4	40	2.75	2.75
01/08/15	To	02/06/15	02/09/15	To	02/13/15		25351		21651				3	60	3	3
01/15/15	To	02/13/15	02/16/15	To	02/20/15		25351		21651				3	60	3	3
01/22/15	To	02/20/15	02/23/15	To	02/27/15		25351		21651				3	60	3	3
01/29/15	To	02/27/15	03/02/15	To	03/06/15	800	26151	760	22411	2	0	0	4	50	3	2.75
02/05/15	To	03/06/15	03/09/15	To	03/13/15	263	26414	243	22654	1	0	0	3	40	3	2.5
02/12/15	To	03/13/15	03/16/15	To	03/20/15	863	27277	763	23417	5	-400	-400	3	70	2.5	1.75
02/19/15	To	03/20/15	03/23/15	To	03/27/15	1550	28827	1510	24927	2	0	0	2	40	2.5	2.5
02/26/15	To	03/27/15	03/30/15	To	04/03/15	50	28877	30	24957	1	0	0	4	70	3.25	3.25
03/05/15	To	04/03/15	04/06/15	To	04/10/15	400	29277	360	25317	2	-188	-188	2	20	3.5	2.25
03/12/15	To	04/10/15	04/13/15	To	04/17/15	250	29527	230	25547	1	0	0	2	20	3.5	2.5
03/19/15	To	04/17/15	04/20/15	To	04/24/15		29527		25547				1	50	2	2.5

In-Sample Dates			Out-Of_Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
03/26/15	To	04/24/15	04/27/15	To	05/01/15	(1275)	28252	(1315)	24232	2	-863	-1275	1	50	2	2
04/02/15	To	05/01/15	05/04/15	To	05/08/15	1288	29540	1248	25480	2	0	0	2	60	1.75	3.25
04/09/15	To	05/08/15	05/11/15	To	05/15/15	(163)	29377	(203)	25277	2	-450	-450	3	70	2.5	2.25
04/16/15	To	05/15/15	05/18/15	To	05/22/15	1288	30665	1268	26545	1	0	0	3	60	2.5	2.5
04/23/15	To	05/22/15	05/25/15	To	05/29/15		30665		26545				1	60	2.25	3
04/30/15	To	05/29/15	06/01/15	To	06/05/15	2313	32978	2233	28778	4	0	0	2	40	2.5	3.25
05/07/15	To	06/05/15	06/08/15	To	06/12/15		32978		28778				4	70	3	3.5
05/14/15	To	06/12/15	06/15/15	To	06/19/15	(338)	32640	(358)	28420	1	-338	-338	3	30	2	2.25
05/21/15	To	06/19/15	06/22/15	To	06/26/15		32640		28420				2	70	1.75	3.25
05/28/15	To	06/26/15	06/29/15	To	07/03/15	(475)	32165	(515)	27905	2	-275	-475	4	30	2.75	3
06/04/15	To	07/03/15	07/06/15	To	07/10/15	(175)	31990	(195)	27710	1	-175	-175	2	70	1.5	2
06/11/15	To	07/10/15	07/13/15	To	07/17/15		31990		27710				2	70	1.5	2
06/18/15	To	07/17/15	07/20/15	To	07/24/15		31990		27710				4	20	2.75	2.5
06/25/15	To	07/24/15	07/27/15	To	07/31/15	238	32228	158	27868	4	-475	-600	4	20	2.75	2
07/02/15	To	07/31/15	08/03/15	To	08/07/15	(975)	31253	(1035)	26833	3	-738	-1263	1	30	2.25	1.5
07/09/15	To	08/07/15	08/10/15	To	08/14/15		31253		26833				2	20	3.5	2.25
07/16/15	To	08/14/15	08/17/15	To	08/21/15	313	31566	273	27106	2	0	0	3	70	1.5	3
07/23/15	To	08/21/15	08/24/15	To	08/28/15	(250)	31316	(310)	26796	3	-350	-388	1	70	1.25	2
07/30/15	To	08/28/15	08/31/15	To	09/04/15	(50)	31266	(70)	26726	1	-50	-50	1	60	2	2.75
08/06/15	To	09/04/15	09/07/15	To	09/11/15		31266		26726				3	70	3.5	3
08/13/15	To	09/11/15	09/14/15	To	09/18/15	(850)	30416	(870)	25856	1	-850	-850	3	60	3.5	2
08/20/15	To	09/18/15	09/21/15	To	09/25/15		30416		25856				3	60	3.5	2.25
08/27/15	To	09/25/15	09/28/15	To	10/02/15	(1188)	29228	(1208)	24648	1	-1188	-1188	1	60	1.75	1.75
09/03/15	To	10/02/15	10/05/15	To	10/09/15		29228		24648				1	60	1.25	2.25
09/10/15	To	10/09/15	10/12/15	To	10/16/15		29228		24648				3	50	2.75	2.5
09/17/15	To	10/16/15	10/19/15	To	10/23/15	2163	31391	2123	26771	2	0	0	2	50	3	1.25
09/24/15	To	10/23/15	10/26/15	To	10/30/15	550	31941	530	27301	1	0	0	1	40	3.25	2
10/01/15	To	10/30/15	11/02/15	To	11/06/15	(75)	31866	(95)	27206	1	-75	-75	2	60	3.5	3.25
10/08/15	To	11/06/15	11/09/15	To	11/13/15	(225)	31641	(285)	26921	3	-450	-450	4	20	2	1.75
10/15/15	To	11/13/15	11/16/15	To	11/20/15		31641		26921				4	30	2.5	2.5
10/22/15	To	11/20/15	11/23/15	To	11/27/15		31641		26921				4	30	2.5	2.5
10/29/15	To	11/27/15	11/30/15	To	12/04/15	4163	35804	4083	31004	4	-263	-263	3	40	1.5	2.75
11/05/15	To	12/04/15	12/07/15	To	12/11/15		35804		31004				2	30	2.25	3.25
11/12/15	To	12/11/15	12/14/15	To	12/18/15	(788)	35016	(828)	30176	2	-475	-788	4	70	3.5	2.5
11/19/15	To	12/18/15	12/21/15	To	12/25/15		35016		30176				3	70	3	3
11/26/15	To	12/25/15	12/28/15	To	01/01/16		35016		30176				3	70	3	3
12/03/15	To	01/01/16	01/04/16	To	01/08/16		35016		30176				3	70	3	3
12/10/15	To	01/08/16	01/11/16	To	01/15/16		35016		30176				2	70	2.25	2.25
12/17/15	To	01/15/16	01/18/16	To	01/22/16	1338	36354	1298	31474	2	0	0	4	60	2.25	2.25
12/24/15	To	01/22/16	01/25/16	To	01/29/16		36354		31474				4	40	3.5	3
12/31/15	To	01/29/16	02/01/16	To	02/05/16	169	36523	149	31623	1	0	0	2	70	2.5	1.75
01/07/16	To	02/05/16	02/08/16	To	02/12/16	256	36779	216	31839	2	0	0	1	20	2	3
01/14/16	To	02/12/16	02/15/16	To	02/19/16		36779		31839				1	20	2	3
01/21/16	To	02/19/16	02/22/16	To	02/26/16		36779		31839				1	20	2	3
01/28/16	To	02/26/16	02/29/16	To	03/04/16	1056	37835	976	32815	4	-138	-138	4	50	1.25	2.5
02/04/16	To	03/04/16	03/07/16	To	03/11/16	1556	39391	1416	34231	7	-556	-556	4	60	2.5	3
02/11/16	To	03/11/16	03/14/16	To	03/18/16	988	40379	968	35199	1	0	0	3	30	2.5	3.25
02/18/16	To	03/18/16	03/21/16	To	03/25/16		40379		35199				1	70	2.25	3.5
02/25/16	To	03/25/16	03/28/16	To	04/01/16		40379		35199				1	70	2.25	3.5
03/03/16	To	04/01/16	04/04/16	To	04/08/16		40379		35199				1	70	2.25	3.5
03/10/16	To	04/08/16	04/11/16	To	04/15/16		40379		35199				1	70	2.25	3.5
03/17/16	To	04/15/16	04/18/16	To	04/22/16	1000	41379	980	36179	1	0	0	4	30	3.25	1.5
03/24/16	To	04/22/16	04/25/16	To	04/29/16	(388)	40991	(408)	35771	1	-388	-388	3	40	2.75	1.5
03/31/16	To	04/29/16	05/02/16	To	05/06/16		40991		35771				1	70	1.5	1

In-Sample Dates			Out-Of_Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
04/07/16	To	05/06/16	05/09/16	To	05/13/16		40991		35771				3	30	3.25	2
04/14/16	To	05/13/16	05/16/16	To	05/20/16	425	41416	405	36176	1	0	0	3	30	3.25	2
04/21/16	To	05/20/16	05/23/16	To	05/27/16	(281)	41135	(301)	35875	1	-281	-281	1	40	1.25	2.25
04/28/16	To	05/27/16	05/30/16	To	06/03/16	1725	42860	1625	37500	5	-188	-188	3	20	2	2
05/05/16	To	06/03/16	06/06/16	To	06/10/16		42860		37500				2	20	2.25	3.25
05/12/16	To	06/10/16	06/13/16	To	06/17/16	(213)	42647	(253)	37247	2	-500	-500	3	20	1.5	3
05/19/16	To	06/17/16	06/20/16	To	06/24/16	(225)	42422	(245)	37002	1	-225	-225	3	20	3	3.5
05/26/16	To	06/24/16	06/27/16	To	07/01/16	(188)	42234	(248)	36754	3	-906	-925	4	20	1.75	2.75
06/02/16	To	07/01/16	07/04/16	To	07/08/16	(163)	42071	(243)	36511	4	-688	-688	4	70	2.25	1.75
06/09/16	To	07/08/16	07/11/16	To	07/15/16		42071		36511				4	60	2.75	2.75
06/16/16	To	07/15/16	07/18/16	To	07/22/16		42071		36511				2	40	1.5	3.25
06/23/16	To	07/22/16	07/25/16	To	07/29/16	513	42584	493	37004	1	0	0	2	40	1.5	3.25

Appendix I: n^{th} Order Polynomial Next Bar's Forecast Math

What is the n^{th} Order Polynomial ?

The n^{th} Order Polynomial, also called the n^{th} Order Fixed Memory Polynomial, is simply the least square fit of a polynomial of the form $\mathbf{b_0+b_1*t+b_2*t^2+b_3*t^3+\ldots b_n*t^n}$ to a ***fixed*** number of past data points. Where **t** is discrete time bars. Time could be daily bars or one minute bars. We use the term "Fixed Memory" to designate that only a fixed number of data points are used to calculate the polynomial coefficients. It is assumed that the time bars occur at fixed intervals of time so tic bars would not be appropriate for this analysis. Least squares is a mathematical technique where the squared vertical distance between the data and the curve that is being fit to the data is minimized. When the net squared distance (also called the sum of the squared errors) is minimized, a unique set of coefficients $\mathbf{b_0,b_1,b_2,\ldots,b_n}$ for the polynomial is determined. This type of error minimization is mathematically solvable and is widely used in science and mathematics.

For a 4th order polynomial equation, the least squares coefficients are obtained from the solution of the following matrix equation.

$$\begin{vmatrix} T & \Sigma t & \Sigma t^2 & \Sigma t^3 & \Sigma t^4 \\ \Sigma t & \Sigma t^2 & \Sigma t^3 & \Sigma t^4 & \Sigma t^5 \\ \Sigma t^2 & \Sigma t^3 & \Sigma t^4 & \Sigma t^5 & \Sigma t^6 \\ \Sigma t^3 & \Sigma t^4 & \Sigma t^5 & \Sigma t^6 & \Sigma t^7 \\ \Sigma t^4 & \Sigma t^5 & \Sigma t^6 & \Sigma t^7 & \Sigma t^8 \end{vmatrix} \begin{vmatrix} a_0 \\ b_0 \\ c_0 \\ d_0 \\ e_0 \end{vmatrix} = \begin{vmatrix} \Sigma p(t) \\ \Sigma(p(t)*t) \\ \Sigma(p(t)*t^2) \\ \Sigma(p(t)*t^3) \\ \Sigma(p(t)*t^4) \end{vmatrix}$$

where

p(T) is the current bar's price, **p(T-1)** is the previous bar's price and **p(1)** is the price **T** bars ago.
T is the number of Bars in the Least Squares estimation
∑p(t) is the summation of prices from **t=1 to T** bars
∑p(t)*t is the summation of prices times t from **t=1 to T** bars
∑t is the summation of the integer **t** from **t=1 to T** bars
$\mathbf{\Sigma t^2}$ is the summation of the integer **t** squared from **t=1 to T** bars
etc.

Once the coefficients to the polynomial have been solved for we generate the forecast for the next bar's price which is given for the equation by:

$$\mathbf{P_f = a_0 + b_0*(T+1) + c_0*(T+1)^2 + d_0*(T+1)^3 + e_0*(T+1)^4}$$

Where $\mathbf{P_f}$ stands for price forecast.

With these coefficients, we can also generate the forecast for the next bar's ***velocity*** and ***velocity*** by the equations:

$$\mathbf{Velocity(T+1) = dP_f /dt = b_0 + 2c_0*(T+1) + 3d_0*(T+1)^2 + 4e_0*(T+1)^3}$$

$$\mathbf{Velocity(t+1) = d^2P_f /d^2t = 2\ c_0 + 6d_0*(T+1) + 12e_0*(T+1)^2}$$

We use the next bar forecast because changes in the trend are more quickly reflected in the forecast price, velocity and velocity than in the end point price, velocity and velocity.

Programs that solve for the solution to matrix equations can be found in the book "Numerical Recipes" by W. Press, et. al. However these type of matrix equation solvers are very slow and for these type of problems are unstable. They cause numerical errors and floating point overflows due to matrix inversion ill conditioning which produces false results.

Fortunately these type of problems can be solved by a fast, efficient and accurate algorithm using Discrete

Appendix I: *n*th Order Polynomial Next Bar's Forecast Math

Orthogona1 Legendre Polynomials. This method is explained in detail in Norman Morrison' book entitled "Introduction to Sequential Smoothing and Prediction", Chapter 7 page 223., referenced at the end of this section.

Without going into detail here (see Morrison reference), the polynomial filter can now be represented by:

$$P_e(t) = \sum_{j=0}^{n} \beta_j * \phi_j(t) \quad j=0 \text{ to } n$$

Where **n** is the polynomial order, **T** is the total number of Bars of data used in the Least Squares fit and

$$\beta_j = \sum_{k=0}^{T-1} p(t-T+k) * \phi_j(k)$$

$\phi_j(t)$ = the *normalized* discrete Legendre polynomial. t = an integer from 0 to T

The coefficients, $\beta_0, \beta_1, \beta_2, \beta_3, , , ,\beta_n$ for a **n**th order polynomial can now be solved for by the equation above and we can generate the forecast for the next bar's close, velocity and velocity which are given by the equations

$$P_F(T+1) = \beta_0*\phi_0(T+1) + \beta_1*\phi_1(T+1) + \beta_2*\phi_2(T+1) + \beta_3*\phi_3(T+1) + \ldots + \beta_n*\phi_n(T+1)$$

$$\text{Velocity} = (dP_F/dt)_{(T+1)} = \beta_1*(d\phi_1/dt)_{(T+1)} + \beta_2*(d\phi_2/dt)_{(T+1)} + \beta_3*(d\phi_3/dt)_{(T+1)} ++ \beta_n*(d\phi_n/dt)_{(T+1)}$$

$$\text{Velocity} = (d^2P_F/d^2t)_{(T+1)} = \beta_2*(d^2\phi_2/d^2t)_{(T+1)} + \beta_3*(d^2\phi_3/d^2t)_{(T+1)} +\ldots+ \beta_n*(d^2\phi_n/d^2t)_{(T+1)}$$

The nth Order Fixed Memory Forecast Next Bar's Velocity System Defined

The least squares forecast is constructed by solving for the least squares coefficients $\beta_1, \beta_2, , , \beta_n$ at each bar using the last **T** bars of closing prices and the Discrete Orthogonal Legendre Polynomial equations for β_j above. Then **Velocity** = **$d^2P_F(T+1)/d^2t$** is constructed from the velocity equation above and plotted under the price chart. In general what we will be doing is following the plotted curve of **Velocity** which is calculated at each bar from the previous T bars. When the velocity is greater than a threshold amount ***vup*** we will go long. When the velocity is less than a threshold amount ***-vdn*** we will go short.

Buy Rule:
IF Velocity is greater than the threshold amount ***vup*** then buy at the market.

Sell Rule:
IF Velocity is less than the threshold amount ***-vdn*** then sell at the market.

References

1. Morrison, Norman "Introduction to Sequential Smoothing and Prediction", McGraw-Hill Book Company, New York, 1969.

The *Normalization Multiplier*

What is the Multiplier ?

The n^{th} Order Fixed Memory Polynomial, also called an n^{th} Order Polynomial, is the least square fit of a polynomial of the form $\mathbf{b_0+b_1*t+b_2*t^2 +b_3*t^3+\ldots b_n*t^n}$ to a ***fixed*** number of past data points. Where **t** is discrete time bars. Time could be daily bars or one minute bars. We use the term "Fixed Memory" to designate that only a fixed number of data points are used to calculate the polynomial coefficients. It is assumed that the time bars occur at fixed intervals of time so tic bars would not be appropriate for this analysis. Least squares is a mathematical technique where the squared vertical distance between the data and the curve that is being fit to the data is minimized. When the net squared distance (also called the sum of the squared errors) is minimized, a unique set of coefficients $\mathbf{b_0,b_1,b_2,\ldots,b_n}$ for the polynomial is determined. This type of error minimization is mathematically solvable and is widely used in science and mathematics. Once the $\mathbf{b_n}$ coefficients are found using a lookback period of **T** bars to calculate the $\mathbf{b_n}$ coefficients, then the next bar's estimate **(T+1)** of the n^{th} order polynomial velocity and acceleration can be easily found by the equations below.

$$\mathbf{Velocity(T+1) = dP_f /dt = b_0 + 2c_0*(T+1) + 3d_0*(T+1)^2 + 4e_0*(T+1)^3 +\ldots.+n*b_n*(T+1)^{n-1}}$$

$$\mathbf{Acceleration(t+1) = d^2P_f /d^2t = 2\ c_0 + 9d_0*(T+1) + 12e_0*(T+1)^3+\ldots\ldots.+ n*(n-1)*b_n*(T+1)^{n-2}}$$

Please see the **n^{th} Order Fixed Memory Polynomial Next Bar's Forecast Math** section for a more detailed explanation.

For any tradable, the inputs to the polynomial are the **polynomial degree(Order)** and the number or lookback bars **N** (denoted by T in equations above). When we plot the velocity or acceleration we notice that the amplitude, and the maximum and minimum values of the velocity or acceleration vary quite significantly with different degree and N inputs.

Below is a table of the standard deviation(Std) of the 56340 calculated Velocity values for different **degree** and **N** inputs. We used 1min bars of the E-Mini from 8/1/2014 to 2/20/2015 to generate this table. As one can see the standard deviation of the velocity for each degree and N vary greatly. For instance for degree=4, N=20 the Std is 6.8 times the Std for degree=1, N=20. This creates problems when trying to determine the correct ranges for vup/vdn and aup/adn during optimization.

@ES.D 1 min bars Date Range 1140801 to 1150220
Total Number of Bars=56340 Norm=0
FixmVn Multiplier= 1/Std to Scale Velocity pw and N Range to One Std

Degree	N	Std	1/Std
1	20	0.1902	5.2565
1	30	0.1540	6.4916
1	40	0.1328	7.5279
1	50	0.1183	8.4502
1	60	0.1077	9.2880
1	70	0.0996	10.0440
avg		0.1338	7.8430
2	20	0.4351	2.2982
2	30	0.3443	2.9046
2	40	0.2936	3.4060
2	50	0.2613	3.8275
2	60	0.2371	4.2180
2	70	0.2173	4.6010
avg		0.2981	3.5425
3	20	0.7854	1.2732

3	30	0.5933	1.6855
3	40	0.4973	2.0111
3	50	0.4347	2.3005
3	60	0.3949	2.5324
3	70	0.3656	2.7352
avg		0.5119	2.0897
4	20	1.2924	0.7738
4	30	0.9279	1.0777
4	40	0.7582	1.3189
4	5	0.6542	1.5285
4	60	0.5804	1.7228
4	70	0.5314	1.8818
avg		0.7908	1.3839

The problem may get worse when we want to find good inputs for other tradables. Other tradables, because of their scales and tick size have much different Velocity ranges then the E-Mini for the same degree and N. Thus the NS search ranges have to be different for each different tradable.

To solve this problem and to have a standard search space for each tradable, I created a **Mult** input for each FixmXVA Velocity and Acceleration strategy and indicator .. If each tradable's Velocity is multiplied by a number such that the standard deviation of that tradable's Velocity is around one then the search space for vup and vdn for each tradable would be 0 to 3.5 SDs and we wouldn't have to change the TS search space every time we wanted to examine a new stock or future. The complicated equations that I use to normalize the ranges to one standard deviation were derived using the software TableCurve 3D, automated surface and equation discovery. .

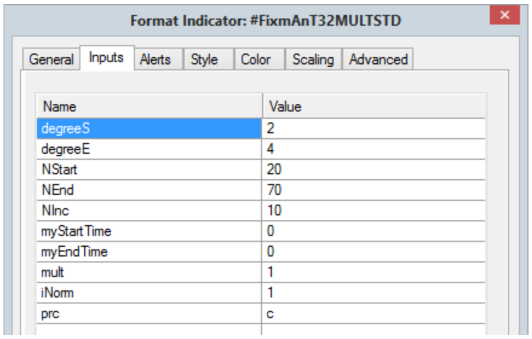
How to Find The Mult.

The program supplies two indicators that you can apply to your chart that will give you the **Mult** to use.

The indicator for the Velocity is named #**FixmVnTxxMultStd** and the indicator for the acceleration is named #**FixmAnTxxMultStd(where xx=32 or 64)**. The #FixmVnTxxMultStd indicator inputs are described in the FixmVn Indicator & Strategy Input Parameters chapter of this manual. The #FixmAnTxxMultStd indicator inputs are described in the FixmAn Indicator & Strategy Input Parameters chapter of this manual.

Here is how to use the #**FixmAnTxxMultStd**. The same method applies to the indicator **#FixmVnTxxMultStd**.

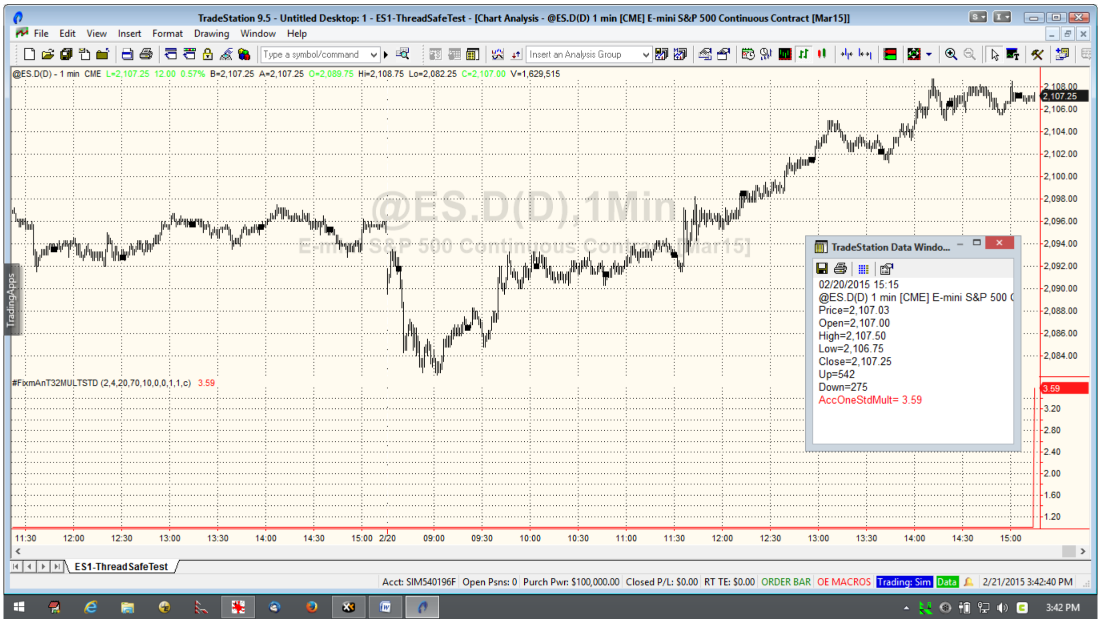
First load your chart with as much data as you are going to use. *Click* on **Insert-> Indicator**. Select **#FixmAnTxxMultStd(where xx=32 or 64).** Modify the inputs to the indicator or just leave them at their default values as shown below.



Name	Value
degreeS	2
degreeE	4
NStart	20
NEnd	70
NInc	10
myStartTime	0
myEndTime	0
mult	1
iNorm	1
prc	c

The *Normalization Multiplier*

Click on **OK**. If you have a lot of bars of data, ***TS*** may take a minute or so to calculate the Multiplier and plot. The chart below shows the indicator output for the E-Mini 1 min bars example above. We **Clicked** on **Data-Window** to get the multiplier value.


TradeStation 9.5 - Untitled Desktop: 1 - ES1-ThreadSafeTest - [Chart Analysis - @ES.D(D) 1 min [CME] E-mini S&P 500 Continuous Contract [Mar15]]
File Edit View Insert Format Drawing Window Help
Type a symbol/command
Insert an Analysis Group
@ES.D(D) - 1 min CME L=2,107.25 12.00 0.57% B=2,107.25 A=2,107.25 O=2,089.75 Hi=2,108.75 Lo=2,082.25 C=2,107.00 V=1,629,515
@ES.D(D), 1Min
E-mini S&P 500 Continuous Contract [Mar15]
#FixmAnT32MULTSTD (2,4,20,70,10,0,0,1,1,c) 3.59
TradingApps
ES1-ThreadSafeTest
Acct: SIM540196F Open Psns: 0 Purch Pwr: $100,000.00 Closed P/L: $0.00 RT TE: $0.00 ORDER BAR OE MACROS Trading: Sim Data 2/21/2015 3:42:40 PM
3:42 PM
TradeStation Data Windo...

02/20/2015 15:15
@ES.D(D) 1 min [CME] E-mini S&P 500 C
Price=2,107.03
Open=2,107.00
High=2,107.50
Low=2,106.75
Close=2,107.25
Up=542
Down=275
AccOneStdMult= 3.59

After you record the Multiplier delete the indicator from you chart. The indicator does not change and you do not want TS to calculate it every time you load the chart or do an optimization.

In addition to the multiplier value displayed by the indicator, the DLL prints the following table to the file **C:\@ES.D1FixmAnxMultStdNorm1.txt** on the C:\ directory on your hard drive, where the first part of the file name is the symbol and time bar.

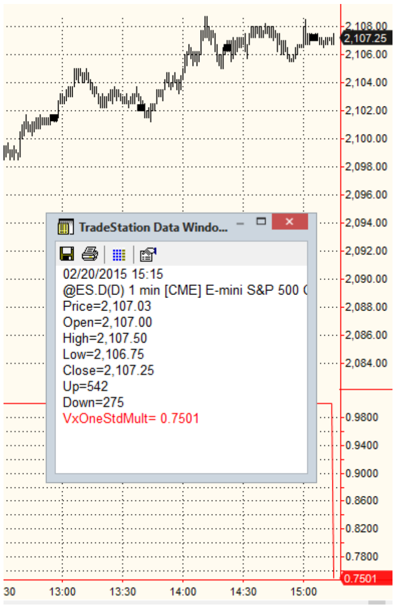
```
@ES.D 1 min bars Date Range 1140801 to 1150220
Total Number of Bars=56340 Norm=1
Trading Times Constraint Start Time=0 EndTime=0
FixmAn Multiplier to Scale Velocity pw and N Range to One Std
pw N
2 20 Std=0.275702 1/Std Mult=3.62711
2 30 Std=0.276678 1/Std Mult=3.6143
2 40 Std=0.278718 1/Std Mult=3.58785
2 50 Std=0.282359 1/Std Mult=3.54159
2 60 Std=0.284526 1/Std Mult=3.51462
2 70 Std=0.284405 1/Std Mult=3.51611
degree=2 1/Std Mult=3.56693

3 20 Std=0.282649 1/Std Mult=3.53796
3 30 Std=0.275064 1/Std Mult=3.63551
3 40 Std=0.275119 1/Std Mult=3.6348
3 50 Std=0.275343 1/Std Mult=3.63183
3 60 Std=0.279536 1/Std Mult=3.57735
3 70 Std=0.284174 1/Std Mult=3.51897
degree=3 1/Std Mult=3.5894

4 20 Std=0.292656 1/Std Mult=3.41698
4 30 Std=0.276035 1/Std Mult=3.62273
4 40 Std=0.272496 1/Std Mult=3.66978
4 50 Std=0.271404 1/Std Mult=3.68455
4 60 Std=0.270641 1/Std Mult=3.69493
4 70 Std=0.273555 1/Std Mult=3.65558
degree=4 1/Std Mult=3.62409

1/Std Mult Ave=3.59348
```

Applying the same method but inserting the **#FixmVnTxxMultStd (where xx=32 or 64)** indicator the velocity multiplier is shown below.:



02/20/2015 15:15
@ES.D(D) 1 min [CME] E-mini S&P 500 (
Price=2,107.03
Open=2,107.00
High=2,107.50
Low=2,106.75
Close=2,107.25
Up=542
Down=275
VxOneStdMult= 0.7501

```
@ES.D 1 min bars Date Range 1140801 to 1150220
Total Number of Bars=56340 Norm=1
Trading Times Constraint Start Time=0 EndTime=0
FixmVn Multiplier to Scale Velocity pw and N Range to One Std
pw N
1 20 Std=1.23911 1/Std Mult=0.807028
1 30 Std=1.29301 1/Std Mult=0.773391
1 40 Std=1.33485 1/Std Mult=0.749148
1 50 Std=1.36727 1/Std Mult=0.731383
1 60 Std=1.39422 1/Std Mult=0.717248
1 70 Std=1.41978 1/Std Mult=0.704335
degree=1 1/Std Mult=0.747089

2 20 Std=1.28656 1/Std Mult=0.777264
2 30 Std=1.2948 1/Std Mult=0.77232
2 40 Std=1.30967 1/Std Mult=0.763551
2 50 Std=1.3304 1/Std Mult=0.751652
2 60 Std=1.34514 1/Std Mult=0.743415
2 70 Std=1.35128 1/Std Mult=0.740039
degree=2 1/Std Mult=0.75804

3 20 Std=1.40562 1/Std Mult=0.711432
3 30 Std=1.33297 1/Std Mult=0.750206
3 40 Std=1.31288 1/Std Mult=0.761682
3 50 Std=1.30074 1/Std Mult=0.768795
3 60 Std=1.30892 1/Std Mult=0.763991
3 70 Std=1.3213 1/Std Mult=0.756829
degree=3 1/Std Mult=0.752156

4 20 Std=1.57493 1/Std Mult=0.634949
4 30 Std=1.40116 1/Std Mult=0.713696
4 40 Std=1.33307 1/Std Mult=0.750151
4 50 Std=1.29424 1/Std Mult=0.772655
4 60 Std=1.26449 1/Std Mult=0.790835
4 70 Std=1.256 1/Std Mult=0.796177
degree=4 1/Std Mult=0.743077

1/Std Mult Ave=0.750091
```

In conclusion, for each different tradable chart:

1. Apply the **#FixmVnTxxMultStd** or **#FixmAnTxxMultStd** indicators to your chart.
2. Find the Multiplier as shown above.
3. Record the multiplier value and delete the indicator from your chart.
4. **Apply that multiplier to the Mult input** in the strategy.
5. Let ***TS*** do it's parameter search for vup or aup in the default range 0.25 to 3.5 by 0.25 and vdn or adn in it's default range 0.25 to 3.5 by 0.25.

With Normalization you never have to look for the proper TS parameter **vup, vdn**, or **aup, adn** search ranges.

Added Note:

The complicated equations that I use to normalize the ranges to one standard deviation were derived using the software TableCurve 3D, automated surface and equation discovery. These equations are ***independen***t of the tradable and only depend on the polynomial degree and N. If you do not wish to use these equations then set inorm=0. Then when you run **#FixmVnTxxMultStd** or **#FixmAnTxxMultStd with inorm =0** you will get a multiplier that gives an average of all the ranges. With iNorm=0 you will have to determine the vup/aup-vdn/adn ranges for your optimization. They will be different for different tradables and time bars. Below is the output for iNorm=0 for **#FixmVnTxxMultStd.** As you can see the variation for degree(pw)=1 is 6 times the variation for degree=4. And the variation for n-70 is twice the variation for N=20. so for this example in order to cover all degree and N ranges at 3 standard deviations maximum you would have to go from vup/vdn from 0.25 to 30 by 0.25. For pw=1 this would be too fine a cut and for pw=4 this would be out of range past 6. Of course the best solution for iNorm=0 would be to modify the strategy multiplier for each degree and N which is essentially what my equation does but not as exact.

```
@ES.D 1 min bars Date Range 1140801 to 1150220
Total Number of Bars=56340 Norm=0
Trading Times Constraint Start Time=0 EndTime=0
FixmVn Multiplier to Scale Velocity pw and N Range to One Std
pw N
1 20 Std=0.190241 1/Std Mult=5.25649
1 30 Std=0.154045 1/Std Mult=6.49162
1 40 Std=0.13284 1/Std Mult=7.52787
1 50 Std=0.11834 1/Std Mult=8.45021
1 60 Std=0.107666 1/Std Mult=9.28796
1 70 Std=0.0995619 1/Std Mult=10.044
degree=1 1/Std Mult=7.84303

2 20 Std=0.435123 1/Std Mult=2.2982
2 30 Std=0.344283 1/Std Mult=2.90459
2 40 Std=0.293599 1/Std Mult=3.40601
2 50 Std=0.261266 1/Std Mult=3.82752
2 60 Std=0.237079 1/Std Mult=4.21801
2 70 Std=0.217346 1/Std Mult=4.60095
degree=2 1/Std Mult=3.54255

3 20 Std=0.785411 1/Std Mult=1.27322
3 30 Std=0.593282 1/Std Mult=1.68554
3 40 Std=0.497251 1/Std Mult=2.01106
3 50 Std=0.434682 1/Std Mult=2.30053
3 60 Std=0.394887 1/Std Mult=2.53237
3 70 Std=0.365599 1/Std Mult=2.73524
degree=3 1/Std Mult=2.08966

4 20 Std=1.29238 1/Std Mult=0.773769
4 30 Std=0.927906 1/Std Mult=1.0777
4 40 Std=0.758239 1/Std Mult=1.31885
4 50 Std=0.654224 1/Std Mult=1.52853
4 60 Std=0.580442 1/Std Mult=1.72283
4 70 Std=0.531418 1/Std Mult=1.88176
degree=4 1/Std Mult=1.3839

1/Std Mult Ave=3.71478
```