Trading The Russell 2000 Index Future Using The Repeated Median Velocity System

In a previous working papers [Ref 4, 5] we examined a trading system that used the velocity of prices fit by a Least Squares straight lines and polynomials through "N" past prices, to determined buy and sell points. The reasoning behind this type of system was to only trade when the price polynomial velocity was above a certain threshold. Many times during the day prices meandering around without a notable trend. At these times we do not wish to trade because of the whipsaws losses that occur from this type of price action. When a price trend finally starts, the velocity of that price trend moves above some minimum threshold value. Thus the velocity system would only issue a trade when certain velocity barriers were crossed.

The Least Squares polynomial is determined by minimizing the sum of the squares of the difference between the N prices and the value of the polynomial line.

$r^2(t)= [Price(t)-(a+b*t+c*t^2+ d*t^3 +\ldots)]^2$ = error squared

$$\text{Minimize}(a,b,c,d,\ldots) \sum_{t=1}^{t=N} r^2(t)$$

This mathematical technique has an exact solution and dates back to Gauss in the 1800's. At the time of it's development there were no computers so this method became a cornerstone of statistics.

Recently (1980's) interest has developed in what is called robust regression and outlier detection techniques, Ref [1]. Regression techniques are now defined by a measure called the "breakdown point". The breakdown point is loosely defined as the smallest amount of bad data points that can cause the regression solutions to take on values arbitrarily far from their true values. Unfortunately the Least Squares technique has a breakdown point of 1/N. In other words only one bad data point can significantly change the computation of the velocity or slope of a straight line. The median of a set of numbers has a breakdown point of 50% . This is because when 50% of the numbers are bad then there is no way of telling which are the bad numbers and which are the good numbers. 50% is the highest breakdown point.

The least absolute deviation (LAD) regression estimator which is

$$\text{Minimize}(a,b) \sum_{i=1}^{i=N} \text{absolute value}[\, r(i)]$$

has a breakdown point of 29.8% (see Ref 1). For the LAD this means around ¼ of the price points can be bad before the computations of a and b become erroneous. Recently Siegel Ref[2], in his paper "Robust regression using repeated medians", introduced a technique for finding the slope that has a 50% breakpoint. The repeated median is also described in Ref [1].

While the repeated median technique may sound complicated it is quite easy to compute. Here's how. For demonstration purposes let's suppose we have 10 data points on an x,y graph such that,

X	1	2	3	4	5	6	7	8	9	10
Y	1	2	3	4	5	15	12	8	9	10

We've added two bad points at X positions 6 and 7. To calculate the repeated median slope we would take the slope of every pair of y values and then find the median of all the pairs of slopes. For this example we would take

Slope 1 = (y(1)-y(2))/(1-2) = 1
Slope 2 = (y(1)-y(3))/(1-3)= 1
Slope 3 = (y(1)-y(4))/(1-4) = 1
Slope 4 = (y(1)-y(5))/(1-5) = 1
Slope 5 = (y(1)-y(6))/(1-6) = 14/5 = 2.8
Slope 6 = (y(1)-y(7))/(1-7) = 11/6 = 1.83
Slope 7 = (y(1)-y(8))/(1-8) = 1
Slope 8 = (y(1)-y(9))/(1-9) = 1
Slope 9 = (y(1)-y(10))/(1-10) = 1

The median slope of the above is 1. The above process is repeated for:
(y(2)-y(i))/(2-i) i=1 to 10 $i \neq 2$,
(y(3)-y(i))/(3-i), i=1 to 10 $i \neq 3$,
…………..
…………..
(y(10)-y(i))/(10-i), i=1 to 9 .

The final slope is then the median of all the medians calculated above. While the repeated median looks redundant because the very first calculation produced the correct slope, price data is not so nicely distributed as our example and the extra calculations are needed to assure that the outliers are eliminated.

The mathematical formula for the above is

$$\text{Slope}(t) = \underset{i=1\text{ to }N}{\text{median}_i} \{ \underset{j=1\text{ to }N}{\text{median}_{i\neq j}} [\text{price}(t)-\text{price}(t-i))/(i-j)] \}$$

Figure 1 below shows a plot of the x,y numbers above with the repeated median line and the least squares line on the graph. Notice how the bad points draw the least squares line towards them while the repeated median line is completely unaffected by the outliers. The least Squares line is given by the formula y=0.6 + 1.1455*x . The true line is given by the formula y=x . From this simple example we can observe how noise has distorted the least squares estimates of **a** and **b**.

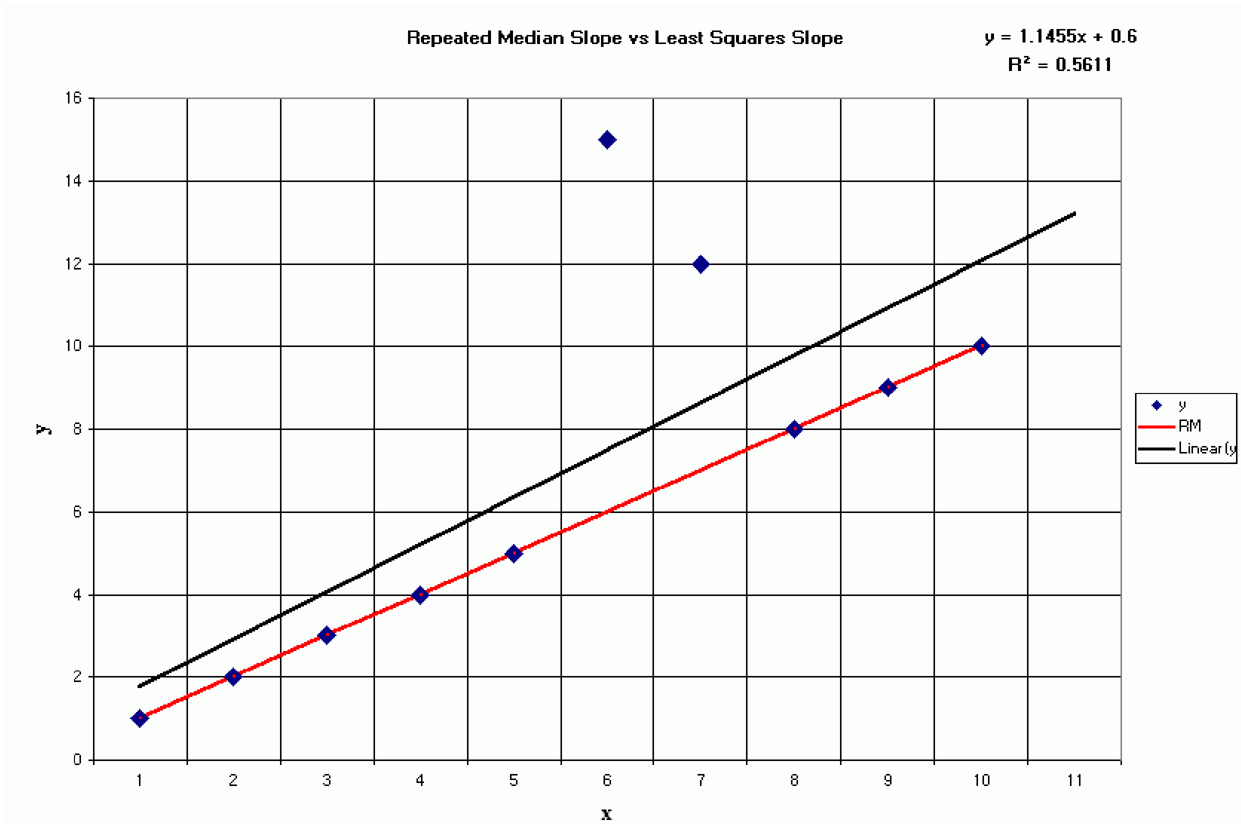

Figure 1 Repeated Median Slope vs Least Squares Slope.

The Repeated Median Velocity System Defined

Here we will use the repeated median slope to create a trading system. For a straight line the velocity is equal to the slope. The repeated median velocity, also called the **robust velocity,** has the advantage that it is a natural random price noise inhibitor. We can create a system such that unless the repeated median velocity using N past price bars is greater than some threshold we will not buy or sell. A large percentage of price noise generates a lot of back and forth movements of small magnitudes over a short period of time. With a lot of systems this back and forth movement creates many false buy and sell signals. However using the repeated median velocity over N past prices, we can filter out many of the small price noise movements by requiring that the velocity to be greater than some threshold before we act.

At each price bar we calculate the repeated median velocity (**RMV**) from the formula above. When the 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.

The Repeated Median Velocity Trading Strategy

Buy Rule:

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

Sell Rule:

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

Intraday Bars Exit Rule:
Close all positions 5 minutes before the close (no trades will be carried overnight).

Intraday Bars First Trade of Day Entry Rule:
Ignore all trade signals before 10:00 EST (30 minutes after the open). Buy and Sell rules above we have included a first trade of the day entry rule. We've included this rule because often there are gaps in the open creating immediate system buys and sells. Many times these gaps are closed creating a losing whipsaw trade. In order to avoid the opening gap whipsaw trade problem we've delayed the first trade of the day for 30 minutes until after10:00 EST

Discussion of Russell 2000 Index E-Mini Future Prices

The **Russell 2000 Index E-Mini Future** (ER2) is traded on Globex and on the trading floor at the CME. On Globex the ER2 is traded on a 23hour basis . The CME hours for floor trading (RTH) are 8:30 to 15:15 CST. Over 70% of the volume in the ER2 is done on Globex during the CME RTH hours. We have restricted our study to only trading the ER2 during the CME RTH 8:30 to 11515 hours. To test this system we will use 1 minute bar prices of the ER2 E-Mini futures contract.

Testing The Repeated Median Velocity System Using Walk Forward Optimization

There are three system parameters to determine:
1. ***N***, is the lookback period to calculate the **RMV.**
2. ***vup***, the threshold amount that RMV has to be greater than to issue a buy signal
3. ***vdn***, the threshold amount that RMV has to be less than to issue a sell signal

In order to use the walk forward testing method, we will break the test data up into fifty-three 30 calendar day test sections and run an optimization on each of those test sections. We will then use an Excel filter described below on the TradeStation optimization output on each of those ten test sections to find the system input parameters ***N, vup,*** and ***vdn***. We will then use the three input parameters, N,vup,vdn, found in the test section by our Excel filter on the out-of-sample section on the week directly following the test section

For each test section we will run a TradeStation optimization on:
1. vup from 1 to 10 steps of 0.5
2. vdn from 1 to 10 in steps of 0.5
3. N from 20 to 70 in steps of 10.

This will produce 2166 different cases or combinations of the input parameters. On each of the fifty-three runs of test data we will export the TradeStation optimization run by saving it in an Excel file. ***Please Note***: this is the way one would perform walk forward runs with the TradeStation(TS) Platform. To shortcut this process we used our TradeStation add-on strategy product called the Power Walk Forward Optimizer (PWFO). This add-on product generated all fifty-three test files for us in one TS run using the 1minute ER2 bars from October 27, 2004 through December 2, 2005.

A technical note. The standard deviation of the RMV is proportional to the square root of the lookback period N. All RMV values were multiplied by the square root of N to normalize the RMV for vup and vdn.

Why use walk forward analysis? Why not just optimize over the total amount of data once and choose the inputs from the best performance parameters?

Out-of-sample testing is the norm in all mathematical economic disciplines ***except*** technical analysis. The basis of walk forward out-of-sample testing is to take a test sample of data, perform a combinatorial parameter search (dubbed optimization by current trading platforms) on the test data, choose the input parameters from the combinatorial search based upon some performance statistic(s) (net profit, profit factor, etc) in the test sample and then apply those chosen input parameters to data they haven't been tested on to see if they work
.
Whenever you run an combinatorial search over many different combinations of input parameters on noisy data on a fixed time interval, the best performance parameters found are guaranteed be due to ***"curve fitting"*** the noise and signal. If you look hard enough you find patterns in the .price movements. Unfortunately it is human nature to extrapolate past performance to project future trading results and so results from curve fitting give the ***illusion*** of future trading profits. A fixed number of prices on a fixed time interval has many spurious and irregular movements which is also called noise. When we run, say, 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 or signal dynamics ***and*** the spurious movements . While the signal dynamics, *if there*, 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 net profit, then choosing these input parameters would 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 optimization or combinatorial searches with no out-of-sample testing cause loses in the out-of-sample section from something that looked great in the test section.

Why perform the walk forward out-of-sample analysis so many times? Why not just do the analysis once or twice? Well just as in poker, there is considerable vagaries in hand to hand luck. In our case there is considerable vagaries in week to week out-of-sample profit luck. That is, by pure chance we may have chosen some input parameters that did well in the test section data ***and*** the out-of-sample section data. In order to minimize this type of "luck" we repeat the walk forward out-of-sample (oos) analysis over many test/oos sections and take an average of our weekly results over all test/oos sections. This average gives us an expected weekly return and a standard deviation of weekly returns which allows us to estimate the expected equity and it's range for N weeks in the future. More on this later.

One thing we do know is that the curve fitting of the signal and noise will always produce the best performance results. Thus if we eliminate the optimization cases with the very best performance results we are sure to eliminate many of the data mining system input parameters that fitted the past spurious movements. We know from experience that very few good intraday systems can sustain profit factors above 2 over time. If we eliminate, from our optimization, all cases that have profit factors greater than 2 we will probably eliminate most of the cases that are due to curve fitting the noise. In addition, since in real time it is tough to sustain more that five losses in a row and still keep trading, we will take the extra precaution and eliminate all those cases that have more than five losses in a row.

To find the system input parameters we will use the Excel Autofilter on each of the fifty-three test section optimization runs. What is an Excel Autofilter? Our TradeStation output has 2166 rows of the different combinations of the input values and a number of performance statistic columns. For instance, one column would be Total Net Profits, another column would be Profit Factor, Percent Profitable and so on. These columns are user selectable in the TradeStation optimization module. When we export the optimization output into an excel spreadsheet we have the columns we selected and the rows which, for our example, would be the 2166 cases. Excel has a feature called the Autofilter. The Autofilter allows us to tell Excel to only show those rows that satisfy some criteria on a column or columns. For instance if one of my columns of data was the Profit Factor(PF) , I could tell Excel that I only wanted to show those cases that had PF's of greater than or equal to 1.0 and less than or equal to 2.0. Excel would then hide all other rows and only display those rows that satisfied my PF criteria.

Using Excel's auto filter we will apply the following screen or filter to each test section run.

Filter: PF>=1 and PF<=2 and LR<=5 and #Trds>=12.

Where:
PF = Profit Factor in Test optimization section
LR=Maximum consecutive loses in a row in test optimization section
#Trds = The number of trades in the test run. We want our system to trade almost every day. There are 20 to 22 trading days a month, so we only want to look at input parameters that are able to generate at least 12 trades in the one month test sample.

This filter in excel will leave anywhere from 10 to 200 cases that satisfy the filter conditions. Out of the cases that are left we will choose the case that has the lowest number of total losing bars. This selection procedure on the test optimization run will leave only one choice for the system input values of ***N, vup,*** and ***vdn***. We will then use these input values on the next week following the test section.

Results

Figure 2 presents a table of the fifty-three test and out-of-sample windows and the selected input parameters from the filter and the out-of-sample results using the Excel filter described above.

Figure 3 presents the out-of-sample 5 minute bar chart of ES from 10/10/05 to 10/14/05. with the Repeated Median Velocity Indicator and all the buy and sell signals for those dates indicated on the charts.

Discussion of System Performance

This strategy trades an average of 3.4 times a week. Sixty-seven percent of the weeks were profitable. There were 176 trades during the fifty three weeks for a total profit was $15550. this works out to an average trade profit of $88.35 before slippage and commissions. A $88 per trade average is excellent for an index.

At the bottom of Figure 2 are some statistics that are of interest. The first statistic ave=299 is the average weekly net profit for the fifty-three OOS weeks. Std=843 is the standard deviation of the weekly return. Given a process that generates the above average and standard deviation, it would be interesting to know statistically the probabilistic outcome from trading this system for 13 weeks (one quarter). There is a statistical technique call bootstrap Ref [3]. Using the bootstrap technique we would randomly choose 13 of the fifty-three weekly profits in figure 2, with replacement and take the sum of those 13 randomly chosen oos net profits. "With replacement" means we don't eliminate the randomly chosen weekly profit from being chosen again. Let us suppose that we repeat this random choosing of 13 weeks and summing the results 200 times. We would then have 200 different 13 week net profit summations. If we take the average and standard deviation of those 200 different 13 week net profit summations we would have an estimate of what to expect from this system by trading it for 13 weeks.

For this system the 200 bootstrap average of 13 week net profits is $4070 and the standard deviation is $2978. This means that at two standard deviations we can expect our 13 week return from this system to be between ($1886) and $10026, 95% of the time. This mean that we need to trade this strategy for more than 13 weeks to statistically be assured of being on the plus side. Please note that slippage and commissions were not taken into account so these numbers would be higher that could be attained.

To see the effect of walk forward analysis take a look at Figure 2. Notice how the input parameter N takes sudden jumps from high to low and back . This is the walk forward process adapting to changing volatility conditions.

In observing the chart in Figure 3, we can see that the system did very well in catching most major intraday trend of the E-Mini. On the day when the noon up trend changed later in the day (10/11/2005) the RMV system changed direction quickly allowing for the capture of the profits from the trend change to the close of trading. Overall the RMV system did a good job in minimizing the losses due to the inevitable whipsaws that will occur in any trading system and maximizing the profits from the major intraday trend moves of the ER2.

The strategy is not an active trader, but when the RMV strategy does issue a signal it's well worth trading.

References

1. Rousseeuw, P.J., Leroy, A.M., (1987) "Robust Regression and Outlier Detection", New York, John Wiley & Sons.
2. Siegel, A.F. (1982), "Robust Regression using Repeated Medians." *Biometrika*. 69, pp242-244.
3. Efron, B., Tibshirani, R.J., (1993), "An Introduction to the Bootstrap", New York, Chapman & Hall/CRC.
4. "The Polynomial Velocity System Applied To E-Mini 1min Bars", Working Paper Sep/2005, www.meyersanalytics.com/articles.php
5. "The Polynomial Velocity System Applied To Japanese Yen 1min Bars", Working Paper Dec/2005, www.meyersanalytics.com/articles.php

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 algorithmic trading software.

Figure 2 Walk Forward Out-Of-Sample Performance Summary
ER2-Mini 1 min bars Repeated Median Velocity System

ER2-1 min bars 10/27/2004 - 12/02/2005 using the below filter on each test segment. The input values N, vup, and vdn are the values found from applying the filter to the test sample's optimization run.

Filter= PF>=1; PF<=2; LR<=3, #Trds>=12, smallest LBR(total number of losing bars).

Week #	Test Dates			OOS Dates			osnp	Equity	ollt	odd	ont	N	vup	vdn
1	10/27/04	to	11/26/04	11/29/04	to	12/03/04	(950)	(950)	(370)	(1160)	8	20	7	7.5
2	11/03/04	to	12/03/04	12/06/04	to	12/10/04	1090	140	(350)	(410)	6	30	7	7
3	11/10/04	to	12/10/04	12/13/04	to	12/17/04	(740)	(600)	(450)	(740)	2	20	7.5	7.5
4	11/17/04	to	12/17/04	12/20/04	to	12/24/04	(130)	(730)	(610)	(610)	2	70	5	8.5
5	11/24/04	to	12/24/04	12/27/04	to	12/31/04	360	(370)	(20)	(20)	2	20	6.5	9
6	12/01/04	to	12/31/04	01/03/05	to	01/07/05	1050	680	(360)	(530)	12	20	6.5	8.5
7	12/08/04	to	01/07/05	01/10/05	to	01/14/05	(820)	(140)	(500)	(890)	6	20	9	8.5
8	12/15/04	to	01/14/05	01/17/05	to	01/21/05	210	70	0	0	3	60	10	6
9	12/22/04	to	01/21/05	01/24/05	to	01/28/05	(670)	(600)	(350)	(670)	2	60	10	8.5
10	12/29/04	to	01/28/05	01/31/05	to	02/04/05	320	(280)	0	0	1	40	10	9.5
11	01/05/05	to	02/04/05	02/07/05	to	02/11/05	1280	1000	0	0	2	40	10	8.5
12	01/12/05	to	02/11/05	02/14/05	to	02/18/05	420	1420	0	0	1	30	7.5	9.5
13	01/19/05	to	02/18/05	02/21/05	to	02/25/05	(140)	1280	(650)	(810)	5	30	7.5	9
14	01/26/05	to	02/25/05	02/28/05	to	03/04/05	(520)	760	(390)	(520)	2	30	8	9
15	02/02/05	to	03/04/05	03/07/05	to	03/11/05	(790)	(30)	(610)	(800)	3	20	9	9
16	02/09/05	to	03/11/05	03/14/05	to	03/18/05	(40)	(70)	0	0	1	20	9	9
17	02/16/05	to	03/18/05	03/21/05	to	03/25/05	320	250	0	0	1	20	9	9
18	02/23/05	to	03/25/05	03/28/05	to	04/01/05	1320	1570	0	0	2	20	9	7
19	03/02/05	to	04/01/05	04/04/05	to	04/08/05	(20)	1550	(620)	(620)	2	20	8	7
20	03/09/05	to	04/08/05	04/11/05	to	04/15/05	2200	3750	(430)	(430)	7	20	8	6.5
21	03/16/05	to	04/15/05	04/18/05	to	04/22/05	1360	5110	(700)	(700)	5	70	10	6
22	03/23/05	to	04/22/05	04/25/05	to	04/29/05	2490	7600	0	0	6	20	8.5	9
23	03/30/05	to	04/29/05	05/02/05	to	05/06/05	(1100)	6500	(700)	(1110)	3	20	8.5	10
24	04/06/05	to	05/06/05	05/09/05	to	05/13/05	(30)	6470	(440)	(1150)	7	20	8.5	8
25	04/13/05	to	05/13/05	05/16/05	to	05/20/05	70	6540	0	0	1	40	9	9.5
26	04/20/05	to	05/20/05	05/23/05	to	05/27/05	90	6630	0	0	1	20	8	10
27	04/27/05	to	05/27/05	05/30/05	to	06/03/05	50	6680	0	0	1	40	8	9.5
28	05/04/05	to	06/03/05	06/06/05	to	06/10/05	900	7580	0	0	3	20	8	7.5
29	05/11/05	to	06/10/05	06/13/05	to	06/17/05	90	7670	0	0	1	30	8	8
30	05/18/05	to	06/17/05	06/20/05	to	06/24/05	620	8290	(170)	(170)	4	20	8	7
31	05/25/05	to	06/24/05	06/27/05	to	07/01/05	250	8540	0	0	1	20	8	7.5
32	06/01/05	to	07/01/05	07/04/05	to	07/08/05	2300	10840	0	0	4	40	5.5	7
33	06/08/05	to	07/08/05	07/11/05	to	07/15/05	390	11230	(630)	(820)	7	70	5.5	6
34	06/15/05	to	07/15/05	07/18/05	to	07/22/05	980	12210	(530)	(900)	5	60	5	8
35	06/22/05	to	07/22/05	07/25/05	to	07/29/05	(130)	12080	0	0	1	60	6	10
36	06/29/05	to	07/29/05	08/01/05	to	08/05/05	420	12500	(320)	(320)	3	60	6	10
37	07/06/05	to	08/05/05	08/08/05	to	08/12/05	(560)	11940	(600)	(600)	3	70	6	10
38	07/13/05	to	08/12/05	08/15/05	to	08/19/05	320	12260	0	0	1	30	8	9.5
39	07/20/05	to	08/19/05	08/22/05	to	08/26/05	170	12430	(240)	(240)	2	40	9	8.5
40	07/27/05	to	08/26/05	08/29/05	to	09/02/05	290	12720	0	0	2	40	8	8
41	08/03/05	to	09/02/05	09/05/05	to	09/09/05	630	13350	0	0	1	40	8	8
42	08/10/05	to	09/09/05	09/12/05	to	09/16/05		13350				30	8	7

43	08/17/05	to	09/16/05	09/19/05	to	09/23/05	**580**	**13930**	(220)	(220)	**4**	**40**	**6**	**8**
44	08/24/05	to	09/23/05	09/26/05	to	09/30/05	**1110**	**15040**	(40)	(40)	**4**	**70**	**5.5**	**9.5**
45	08/31/05	to	09/30/05	10/03/05	to	10/07/05	**(1110)**	**13930**	(690)	(1110)	**10**	**30**	**6**	**7**
46	09/07/05	to	10/07/05	10/10/05	to	10/14/05	**1450**	**15380**	(440)	(440)	**6**	**20**	**8.5**	**9**
47	09/14/05	to	10/14/05	10/17/05	to	10/21/05	**740**	**16120**	(550)	(550)	**4**	**40**	**10**	**9.5**
48	09/21/05	to	10/21/05	10/24/05	to	10/28/05	**190**	**16310**	(600)	(600)	**5**	**30**	**9**	**8**
49	09/28/05	to	10/28/05	10/31/05	to	11/04/05	**590**	**16900**	(450)	(450)	**4**	**30**	**9**	**8.5**
50	10/05/05	to	11/04/05	11/07/05	to	11/11/05	**(1060)**	**15840**	(910)	(1140)	**3**	**30**	**9**	**8**
51	10/12/05	to	11/11/05	11/14/05	to	11/18/05	**420**	**16260**	0	0	**1**	**30**	**10**	**8.5**
52	10/19/05	to	11/18/05	11/21/05	to	11/25/05	**(1030)**	**15230**	0	0	**1**	**30**	**9**	**8**
53	10/26/05	to	11/25/05	11/28/05	to	12/02/05	**320**	**15550**	(100)	(100)	**2**	**30**	**8.5**	**8**
				OSNP AVE =			**299**			Ave NT=	**3.4**			
				OSNP STD			**843**							
				200 Boot 13Wk sum AVE=			**4070**							
				200 Boot 13Wk sum STD=			**2978**							

Where
PF = Profit Factor in Test optimization section
LR= Consecutive loses in a row in test optimization section
LBR = Total number of losing bars in the test section.
osnp = Weekly Out-of-sample net profit
ollt = out-of-sample largest losing trade for that week.
odd = Out-of-Sample closing trade drawdown for that week
ont = The number of trades in the out-of-sample week.
Equity = running sum of the weekly out-of-sample net profits

Figure 3 ER2 1min Repeated Median Velocity System 10/10/05 to 10/14/05