

Tim Tillson is a software project manager at Hewlett-Packard, with degrees in Mathematics and Computer Science. He has privately traded options and equities for 15 years.

Introduction

"Digital filtering includes the process of smoothing, predicting, differentiating, integrating, separation of signals, and removal of noise from a signal. Thus many people who do such things are actually using digital filters without realizing that they are; being unacquainted with the theory, they neither understand what they have done nor the possibilities of what they might have done."

This quote from R. W. Hamming¹ applies to the vast majority of indicators in technical analysis. Moving averages, be they simple, weighted, or exponential, are *lowpass filters*; low frequency components in the signal pass through with little attenuation, while high frequencies are severely reduced. "Oscillator" type indicators (such as MACD, Momentum, Relative Strength Index) are another type of digital filter called a *differentiator*. Tushar Chande has observed that many popular oscillators are highly correlated, which is sensible because they are trying to measure the rate of change of the underlying time series, i.e., are trying to be the first and second derivatives we all learned about in Calculus.

We use moving averages (lowpass filters) in technical analysis to remove the random noise from a time series, to discern the underlying trend or to determine prices at which we will take action. A perfect moving average would have two attributes:

- It would be smooth, not sensitive to random noise in the underlying time series. Another way of saying this is that its derivative would not spuriously alternate between positive and negative values.
- It would not lag behind the time series it is computed from. Lag, of course, produces late buy or sell signals that kill profits.

The only way one can compute a perfect moving average is to have knowledge of the future, and if we had that, we would buy one lottery ticket a week rather than trade!

(See Appendix II below for more elaboration). Having said this, we can still improve on the conventional simple, weighted, or exponential moving averages. Here's how:

Two Interesting Moving Averages

We will examine two benchmark moving averages based on Linear Regression analysis.

In both cases, a Linear Regression line of length n is fitted to price data.

I call the first moving average ILRS, which stands for *Integral of Linear Regression Slope*. One simply integrates the slope of a linear regression line as it is successively fitted in a moving window of length n across the data, with the constant of integration being a simple moving average of the first n points. Put another way, the derivative of ILRS is the linear regression slope. Note that ILRS is not the same as a SMA (simple moving average) of length n , which is actually the midpoint of the linear regression line as it moves across the data.

We can measure the lag of moving averages with respect to a linear trend by computing how they behave when the input is a line with unit slope. Both SMA(n) and ILRS(n) have lag of $n/2$, but ILRS is much smoother than SMA.

Our second benchmark moving average is well known, called EPMA or *End Point Moving Average*. It is the endpoint of the linear regression line of length n as it is fitted across the data. EPMA hugs the data more closely than a simple or exponential moving average of the same length. The price we pay for this is that it is much noisier (less smooth) than ILRS, and it also has the annoying property that it overshoots the data when linear trends are present, as shown in Figure 1.

However, EPMA has a lag of 0 with respect to linear input! This makes sense because a linear regression line will fit linear input perfectly, and the endpoint of the LR line will be on the input line.

These two moving averages frame the tradeoffs that we are facing. On one extreme we have ILRS, which is very smooth and has considerable phase lag. EPMA has 0 phase lag, but is too noisy and overshoots. We would like to construct a better moving average which is as smooth as ILRS, but runs closer to where EPMA lies, without the overshoot.

A easy way to attempt this is to split the difference, i.e. use $(ILRS(n)+EPMA(n))/2$. This will give us a moving average (call it IE/2) which runs in between the two, has phase lag of $n/4$ but still inherits considerable noise from EPMA. IE/2 is inspirational, however. Can we build something that is comparable, but smoother? Figure 1 shows ILRS, EPMA, and IE/2.

Filter Techniques

¹ Hamming, R.W., *Digital Filters* (Third Edition). Monterrey, CA: Prentice Hall, 1989.

Any thoughtful student of filter theory (or resolute experimenter) will have noticed that you can improve the smoothness of a filter by running it through itself multiple times, at the cost of increasing phase lag.

There is a complementary technique (called *twicing* by J.W. Tukey²) which can be used to improve phase lag. If L stands for the operation of running data through a low pass filter, then twicing can be described by:

$$L' = L(\text{time series}) + L(\text{time series} - L(\text{time series}))$$

That is, we add a moving average of the difference between the input and the moving average to the moving average. This is algebraically equivalent to:

$$2L - L(L)$$

This is the *Double Exponential Moving Average* or DEMA, popularized by Patrick Mulloy in TASAC (January/February 1994). In our taxonomy, DEMA has some phase lag (although it exponentially approaches 0) and is somewhat noisy, comparable to IE/2.

We will use these two techniques to construct our better moving average, after we explore the first one a little more closely.

Fixing Overshoot

An n-day EMA has smoothing constant $\alpha = 2/(n+1)$ and a lag of $(n-1)/2$.

Thus EMA(3) has lag 1, and EMA(11) has lag 5. Figure 2 shows that, if I am willing to incur 5 days of lag, I get a smoother moving average if I run EMA(3) through itself 5 times than if I just take EMA(11) once.

This suggests that if EPMA and DEMA have 0 or low lag, why not run fast versions

(eg DEMA(3)) through themselves many times to achieve a smooth result? The problem is that multiple runs through these filters increase their tendency to overshoot the data, giving an unusable result. This is because the amplitude response of DEMA and EPMA is greater than 1 at certain frequencies, giving a gain of much greater than 1 at these frequencies when run through themselves multiple times. Figure 3 shows DEMA(7) and EPMA(7) run through themselves 3 times. $DEMA^3$ has serious overshoot, and $EPMA^3$ is terrible.

The solution to the overshoot problem is to recall what we are doing with twicing:

$$DEMA(n) = EMA(n) + EMA(\text{time series} - EMA(n))$$

The second term is adding, in effect, a smooth version of the derivative to the EMA to achieve DEMA. The derivative term determines how hot the moving average's response to linear trends will be. We need to simply turn down the volume to achieve our basic building block:

$$EMA(n) + EMA(\text{time series} - EMA(n)) * .7;$$

This is algebraically the same as:

$$EMA(n) * 1.7 - EMA(EMA(n)) * .7;$$

I have chosen .7 as my volume factor, but the general formula (which I call "Generalized Dema") is:

$$GD(n,v) = EMA(n) * (1+v) - EMA(EMA(n)) * v,$$

where v ranges between 0 and 1. When v=0, GD is just an EMA, and when v=1, GD is DEMA. In between, GD is a cooler DEMA. By using a value for v less than 1 (I like .7), we cure the multiple DEMA overshoot problem, at the cost of accepting some additional phase delay. Now we can run GD through itself multiple times to define a new, smoother moving average T3 that does not overshoot the data:

$$T3(n) = GD(GD(GD(n)))$$

Figure 4 shows T3(6) plotted with EPMA(15), ILRS(15) and IE/2(15). T3 is very similar to IE/2 and DEMA, but smoother than both, which was our goal.

In filter theory parlance, T3 is a six-pole non-linear Kalman filter. Kalman filters are ones which use the error (in this case (time series - EMA(n))) to correct themselves. In Technical Analysis, these are called Adaptive Moving Averages; they track the time series more aggressively when it is making large moves.

² Tukey, J.W., *Exploratory Data Analysis*. Reading, MA: Addison-Wesley, 1976.

Trading Results

I used Metastock 6.5 to compare 5 moving averages (SMA, ILRS, EMA, DEMA and T3) on the Nasdaq index (NDX) from 7/19/93 to 6/30/97, almost four years of data. I set the interest rate at 4% annualized, and a trading cost of .1% for entry and exit. This is realistic, since I can trade up to 1000 shares through Fidelity Web Express for \$14.95, and a typical trade might be 300 shares of a \$50 stock.

The system used was very simple. A moving average was computed using each of the five above. A derivative was taken (1 period Rate of Change function, ROC). A long position was entered at bottoms, and closed at tops, of the derivative. No shorts were taken. For example, the code for "enter long" for an EMA was:

```
res:=Mov(C,opt1,E);
d1:=ROC(res,1,points);
d1 > Ref(d1,-1) AND Ref(d1,-1) < Ref(d1,-2)
```

I let opt1 vary from 2 to 10. This is a fast system, which executes a lot of trades. The table below summarizes the best results each MA was able to achieve; note that the optimal parameters did not vary widely.

MA	Optimal Parameter	Best APR
ILRS	7	19.02
SMA	8	20.20
EMA	8	28.17
DEMA	7	28.81
T3	5 (volume = .7)	33.59

No single set of backtesting results is conclusive, of course. But these numbers confirm that T3 has merit - it not only looks good to the eye on the chart, but can be a powerful building block in other indicators and trading systems.

Appendix I - Metastock Implementations

Metastock 6.5 code for ILRS:

```
{input number of lookback periods, default is 11}
periods:=Input("Periods? ",2,63,11);
{determine how many points are in the time series}
size:=LastValue(Cum(1));
{determine the constant of integration by taking the simple moving average of the
first periods points in the time series}
start:=LastValue(Ref(Mov(P,periods,S),periods-size));
{value is the integral of linear regression slope plus the constant of integration}
Cum(LinRegSlope(P,periods))+start;
Metastock 6.5 code for T3:
```

If x stands for the action of running a time series through an EMA, f is our formula for Generalized Dema with 'volume' a:

$$f := (1 + a)x - ax^2$$

Running the filter through itself three times is equivalent to cubing f:

$$-a^3 x^6 + (3a^2 + 3a^3)x^5 + (-6a^2 - 3a - 3a^3)x^4 + (1 + 3a + a^3 + 3a^2)x^3$$

Thus the Metastock 6.5 code for T3 is:

```
periods:=Input("Periods? ",1,63,5);
a:=Input("Hot? ",0,2,.7);
e1:=Mov(P,periods,E);
e2:=Mov(e1,periods,E);
e3:=Mov(e2,periods,E);
e4:=Mov(e3,periods,E);
e5:=Mov(e4,periods,E);
e6:=Mov(e5,periods,E);
c1:=-a*a*a;
```

```

c2:=3*a*a+3*a*a*a;
c3:=-6*a*a-3*a-3*a*a*a;
c4:=1+3*a+a*a*a+3*a*a;
c1*e6+c2*e5+c3*e4+c4*e3;

```

Appendix II - Perfect Moving Averages and Oscillators

If one has knowledge of the future, perfect moving averages and oscillators can be constructed. Here's how it is done computationally:

- A perfect moving average can be constructed by adding an Exponential Moving Average which moves *backward* in time to one which moves *forward* in time, then dividing by two. The phase lead of the backward EMA cancels the phase lag of the forward EMA producing a moving average which is both smooth and in phase (but only for historical data, not the data on the right hand side of the chart, where we want to trade).
- A perfect oscillator can be had by subtracting the forward EMA from the backward EMA, or by taking the derivative of the perfect moving average. These calculations are analogous to the two-sided derivative of Calculus, which uses the future $f(x+h)$ term in the definition:

$$\lim_{h \rightarrow 0} \frac{f(x+h) - f(x-h)}{2h}$$

In the real world, oscillators are constrained to be like the noisier left sided derivative:

$$\lim_{h \rightarrow 0} \frac{f(x) - f(x-h)}{h}$$

The limitations of technical analysis become very apparent when one compares left sided oscillators with perfect oscillators above on historical data! Using about 600 days worth of HWP daily closes, the two sided derivative PERFECT(20) is plotted as a dotted line, while the 20 day Linear Regression Line slope is the black line. Note that PERFECT(20) has about 10 days of phase lead over LRS, about what one would expect.

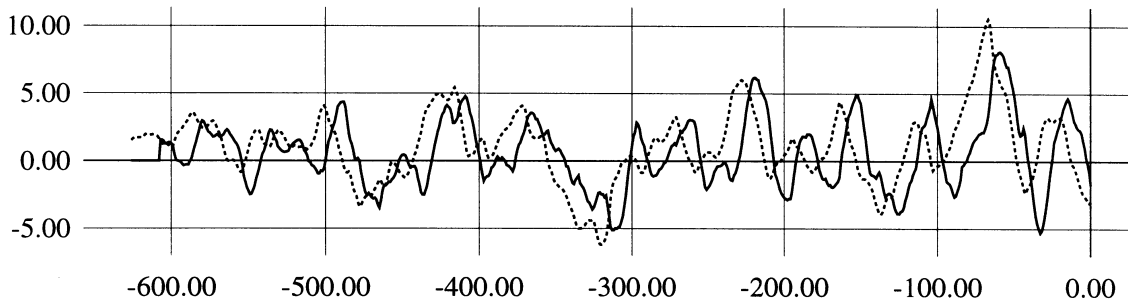


Figure 1 - EPMA(15) - Red, IE/2(15) - Blue, ILRS(15) - Green



Figure 2 - EMA(11) - Red vs EMA(3)^5 - Green



Figure 3 - Overshoot of Dema(7)^3 (Red) and EPMA(7)^3 (Blue)

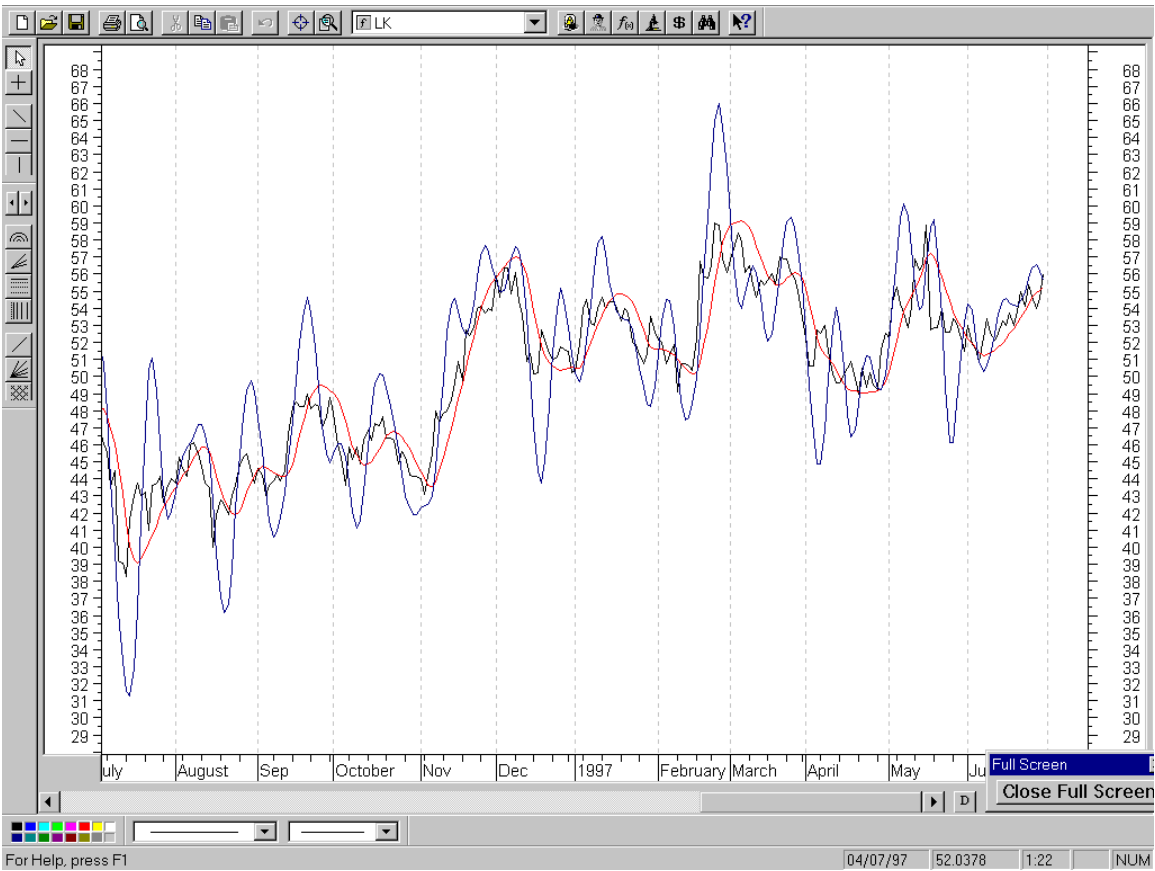
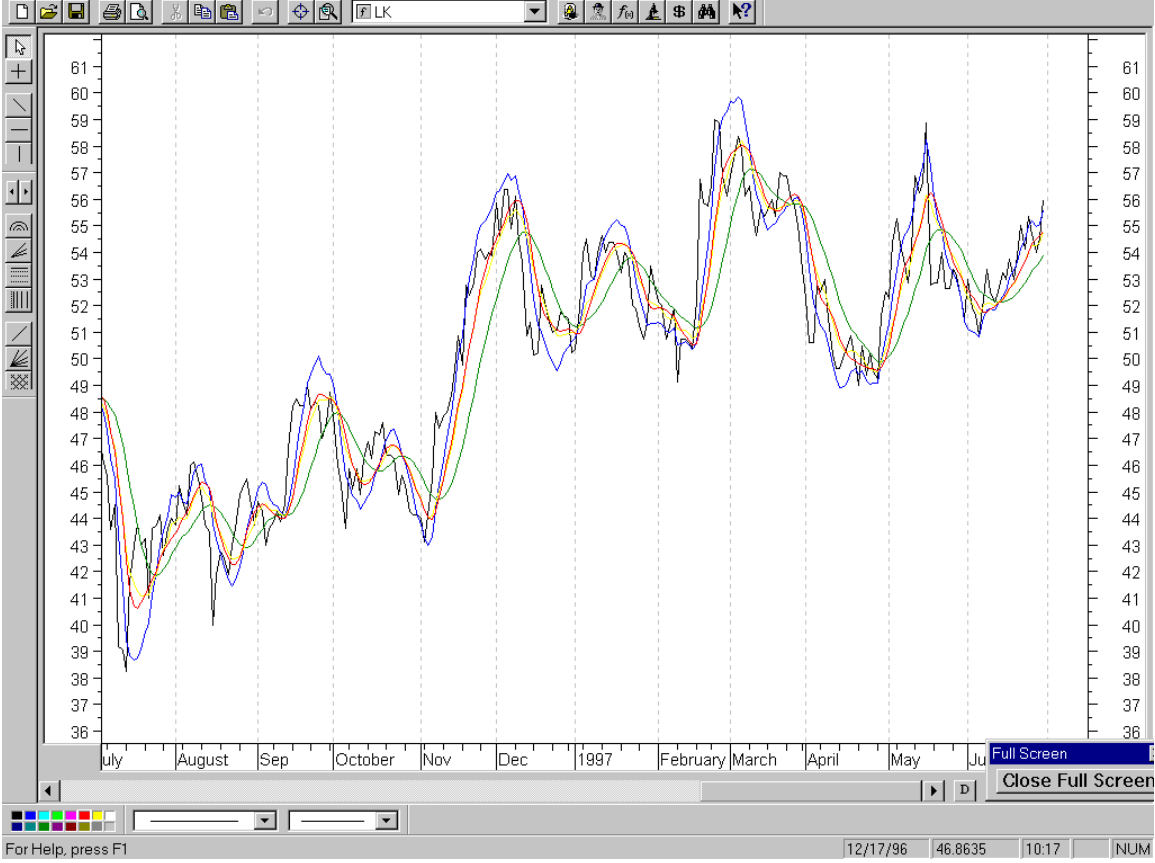


Figure 4 - EPMA(15) - Blue, T3(6) - Red, IE/2(15) - Yellow, ILRS(15) - Green





From CCT Steve:

For anyone who hasn't snagged a copy of Tim Tillson's T3 paper...I've attached a word file. It's the article that was published in TASC in '98. Tim has given me *permission* to pass the paper to traders who are interested in "a better moving average".

I would like to offer a trading system, using the indicator described in the Tim's article. I met with Tim on MLK day and I showed him my work ...using his indicators. Instead of Tim's .7 "Hot" designation, I substitute: .618. I like to look at the interplay of the T3 set to: 3, 5, 8, 13, 21, and 34.

Entry: Buy the opening the period after the $T3(3) > T3(5)$.

Exit: Sell the opening the period after the $T3(3) < T3(5)$.

Stops: Closes below the $T3(5)$ or $T3(8)$.

Trend: I.e., put on trades in the direction of the 144 sma or other qualifiers.

The $T3(3)$ and the $T3(5)$ are sensitive to market changes and keep you in most of the "head fakes" that many moving averages create. The big added bonus is that I have never seen a dynamic, intraday indicator that can define support and resistance as do the $T3(3)$ and the $T3(5)$.

There are a billion mix and match opportunities. I've always hated the slippage that is inherent in a cross-over, trend-following system. To eliminate this (and unfortunately, eliminating a percentage of trades), I like to wait for the cross-over and then place an order to buy or sell on a retracement to the $T3(5)$. See gif ABOVE.