Modeling and Prediction of Financial Time Series Using Linear Recurrent Sequences: An Application to the US Dollar/EURO Exchange Rate

[1] Ahmed S. Abutaleb, [2] Michael G. Papaioannou
[1] Cairo University, School of Engineering, Systems and Bioengineering Department, Giza, Egypt,
E-Mail: ataleb@mcit.gov.eg
[2] IMF, Washington D. C. 20431, USA, E-Mail: mpapaioannou@imf.org

Abstract:
This study introduces the modulus operation as a tool for modeling and forecasting financial time series. The proposed method is based on the assumption that there are signal-determined barriers for each observed signal. Whenever the signal hits such a barrier an abrupt change occurs. The barrier and the change could be described by a modulo operation (similar to the odometer of the car). A finite impulse response (FIR) like filter is introduced. It is based on the concept of recurrent sequences. A nonlinear estimation approach, the Genetic Algorithm (GA), is used to estimate the model's parameters. The linear recurrent sequences model is then applied to the US Dollar / EURO exchange rate. The coefficients of the models are estimated through the GA. The predictive results of the proposed method are shown to be more accurate than that of a conventional AR technique.

Keywords: *Nonstationary Time Series, Modulus operation, Recurrent Sequences, Number Theory, Nonlinear Models, Genetic Algorithms.*

I. Introduction:

In modeling and forecasting time series, the most common applications in finance involve exchange rates, share prices, and commodity prices, and in engineering involve signal analysis, Radar imaging, Ultrasonics. Forecasting exchange rates, through signal models, appears to have attracted the most interest in economics and financial research. A conventional method of signal modeling is to relate, through a constant coefficient model, fundamentals, external inputs and other relevant variables to signal movements. Such a constant coefficient model might be an autoregressive (AR) model, an autoregressive with external inputs (ARX), an autoregressive moving average (ARMA) model, or other enhancements of the latter two [Diebold; 1998, Rahman et al; 1997, Harvey; 1992, Priestley; 1988, Young; 1984]. In addition, time-varying models [Chow; 1987] have also been suggested for the signal analysis, and have shown good performance [Abutaleb and Papaioannou; 1990, Abutaleb and Papaioannou; 2000].

Furthermore, various signaling models have been proposed for modeling the behavior and predicting the movement of stock market share prices. Among them, the most prominent ones are the Capital Asset Pricing Model (CAPM) and the mean reversion model. In the latter tradition, as observed in [Jegadeesh; 1990, and Lehmann; 1990], asset prices seem to obey a reversal effect, i.e. the best performing stocks in one week or month tend to fare poorly in the following period, while the worst performers follow up with good performance. Such an effect, however, could be described by a (modulo p) operation. That is, the share price dynamics could be described by a continual search for profitable trading rules, followed by destruction from overuse of those rules found to be successful, which is then followed by more search for yet-undiscovered rules [Bodie et. al; 1995]. This pattern may also be true for the prices of other financial assets.

The same phenomenon, modulo type operation, seems to come naturally in many applications in bioengineering and in radar. For example, random noise is simulated through what is known as recurrent sequences, which are nothing, but a finite impulse (FIR) filter subjected to modulus operations. Another example is measuring instruments, which usually have a saturation limit. This might be modeled as a threshold device or as a modulus device. In these applications, the phase is an important quantity in the analysis of the echo of many types of signals. The phase, however, is limited between 0 and 360 degrees and thus it is essential to estimate the original phase or signal from the measured modulo 2π signal.

In this study, the exchange rate between the $US and the EURO is modeled as a linear recurrent sequence on a finite field. When the value of the exchange rate hits a maximum or a minimum its value is reversed. This is similar to the (modulo p) operation, where the counting starts from zero, say, to reach the maximum value of (p-1), and then reverses itself to start from zero again. The approach is nonlinear in nature, and is partially based on Number Theory techniques [Abutaleb; 2001, Leveque; 1977]. The estimates of the model parameters are obtained through a nonlinear estimation approach, the simple Genetic Algorithm [Holland; 1969, 1975, IEEE; 1994, Michaelowitz; 1994, and Abutaleb; 1997]. The predictive results of the proposed method are found to be superior when compared to that of a conventional AR technique. Specifically, we show that the forecasting accuracy of the $US/EURO exchange rate improves when we use the proposed nonlinear approach relative to the constant coefficient method. The data used are generated data from the IFS, end of the week $US/EURO exchange rate for the period, January 1999 to March 2001.

In section II, an outline of the conventional AR, MA, and ARMA models is presented. Section III describes the proposed linear recurrent sequence method and the parameter estimation using a Genetic Algorithm. In section IV, we compare the predictive results of the modulo-based methods with that of a conventional AR model. The last section concludes by summarizing the main results and discussing the advantages of the proposed method in modeling and forecasting time series data.

II. The conventional Approach: AR Process

For expositional purposes, we use the exchange rate paradigm, which could also be applied to any type of signals. When analysts are confronted with the task of short-term forecasting a certain exchange rate or signal, they often use a set of past measurements, and a set of past relevant variables. Hence, the objective is to predict the change in the exchange rate or signal (m+1) steps in the future, given only information until the present time, n. Using a predictive autoregressive with external inputs (ARX) model, one might formulate the problem as follows:

$$\phi(n+m)=\sum_{i=1}^{i=I}\alpha_i\phi(n-i)+\sum_{j=1}^{j=J}\beta_{1j}u_1(n-j)+\sum_{j=1}^{j=J}\beta_{2j}u_2(n-j)+...+\varepsilon(n) \quad \text{(II. 1)}$$

which can be written in compact form as:

$$\phi(n+m)=\sum_{i=1}^{i=I}\alpha_i\phi(n-i)+\sum_{j=1}^{j=J}\sum_{l=1}^{l=L}\beta_{lj}u_l(n-j)+\varepsilon(n) \quad \text{(II. 2)}$$

where

n	the present time
m	the prediction step
α_i	the unknown coefficient of the change in the exchange rate or signal at the time (n-i)
I	the total number of lags for the change in the exchange rate
$\phi(n+m)$	the predicted value of the change in the signal (m+1) steps in the future
$\phi(n-i)$	the change in the signal at the time (n-i)
β_{lj}	the unknown coefficient of the l^{th} variable delayed j steps

$u_l(n-j)$	the lth exogenous variable at time (n-j)
$\varepsilon(n)$	the residual or error term
L	the number of independent variables
J	the number of lags

Note that $u_l(n-j)$ might represent a nonlinear function of the signal.

There are many techniques to estimate the unknown parameters. A numerically stable and accurate method is to use the regression analysis via the method of Singular Value Decomposition (SVD) [Press et. al.; 1988]. Accordingly, one might rewrite equation (II. 2) in the following regression format:

$$A\underline{X} = \underline{b} \tag{II. 3}$$

where

$$\underline{X}^T = [\alpha_1\ \alpha_2 \ldots \alpha_I\ \beta_{11}\ \beta_{21} \ldots \beta_{L1}\ \beta_{L2} \ldots \beta_{LJ}]$$

$$\underline{b}^T = [\phi(I+m)\ \phi(I+m+1) \ldots \phi(N)]$$

with "T" being the transpose operation, and N the total number of sample periods. If matrix A has the corresponding values, equation (II. 3) has the following solution:

$$\underline{\hat{X}} = \left(A^T A\right)^{-1} A^T \underline{b} = A^{\#}\underline{b} \tag{II. 4}$$

Where

$A^{\#}$	the pseudo inverse of the matrix A.

This constant coefficient method is efficient when the relation is linear and time-invariant. If there is a nonlinearity or time variation in the data, this method is known not to yield the best results [Harvey; 1992].

III. Recurrent Sequences for Data Analysis

In this section, we introduce the notion of recurrent sequences and how it could be used to model time series data. We shall first introduce the familiar pseudorandom sequences and show that they are applications of the notions of recurrent sequences. We then give some theoretical analysis of recurrent sequences and the concept of a period. Finally, we introduce the simple Genetic Algorithm as a tool to estimate the parameters of the recurrent sequence. In this analysis we will be dealing with fields and polynomials over fields [Denning; 1982, Lidl and Niederreiter; 1997]

III. 1Pseudorandom Sequences

The notion of a random sequence of events is basic in probability theory, statisitcs, econometrics, and signal processing. Random sequences of bits are used frequently for simulation purposes, for various applications in electrical engineering, and cryptography. Pseudorandom sequences of bits are deterministic sequences of bits that pass various tests for randomness. A commonly employed method of generating psuedorandom sequences of bits is based on the use of suitable linear recurrence relations in the finite field F_2. The sequences that one generates are the maximal period sequences. One could also generate pseudorandom sequences that have elements in the field F_q, where q is an integer not necessarily a prime number. This last field, F_q, is the one that we will be dealing with.

III. 1a Maximal Period Sequence

A kth order maximal period sequence in F_q is a sequence s(0), s(1), … of elements of F_q generated by a linear recurrence relation:

$$s(n+k) = a_{k-1}s(n+k-1) + a_{k-2}s(n+k-2) + ... + a_0 s(n) \qquad \text{for n=0, 1, …} \tag{III. 1}$$

for which the characterisitc polynomial, f(x), is defined as:

$$f(x) = x^k - a_{k-1}x^{k-1} - a_{k-2}x^{k-2} - ... - a_0 \tag{III. 2}$$

is a primitive polynomial over F_q and not all initial values , s(0), …, s(k-1) are zero.

Definition of a primitive polynomial: A polynomial $f \in F_q[x]$ of degree m>0 is called a primitive polynomial over F_q if and only if it is monic, f(0)$\neq$ 0, and the order of the polynomial equals $(q^m - 1)$. The following is the definition of the order of the polynomial.

Definition of the order of a polynomial: Let $f \in F_q[x]$ be a nonzero polynomial. If $f(0) \neq 0$, then the least positive integer e for which $f(x)$ divides $(x^e - 1)$, i.e. $f(x)$ is a factor of $(x^e - 1)$, is called the order of $f(x)$ and is denoted $ord(f)$.

A kth order maximal period sequence is periodic with least period r = $q^k - 1$. A requirement we have to impose is that r be very large, at least as large as the total number of pseudorandom elements of F_q to be used in the specific application. In this way the periodicity of the sequence which is a nonrandom feature will not come into play.

Example: Consider the linear recurring sequence, s(0), s(1), ... in F_2, i.e all the elements are 0 and 1, with:

s(n+5) = s(n+2) + s(n) for n = 0, 1, …

and initial values s(0)=s(2)=s(4)=1, and s(1)=s(3)=0.

The characteristic polynomial, f(x), is given by:

$$f(x) = x^5 - x^2 - 1$$

which is a primitive polynomial over the field F_2. Thus, this sequence is a maximal period sequence with least period $r = q^k - 1 = 2^5 - 1 = 31$.

For many simulation purposes, and specially for applicatons in numerical analysis, one needs random sequences of real numbers. These numbers should all belong to a given interval on the real line which may be taken to be the interval [0, 1]. Again maximal period sequences in finite fields can be used to generate sequences of uniform pseudorandom numbers. Let F_p be a finite prime field i.e. p is prime, and let s(0), s(1), ... be a kth order maximal period sequences in F_p i.e. the sequence elements are < p. Then the integers have to be transformed into numbers in the interval [0, 1]. One poplular method of doing this is the normalization method, in which one chooses p to be a large prime and normalizes s(n) by setting

$$w(n) = \frac{s(n)}{p} \in [0,1] \qquad \text{for all } n \geq 0$$

The w(0), w(1), ... is taken as a sequence of uniform pseudorandom numbers. This sequence is periodic with least period r = $p^k - 1$.

III. 2 Linear Recurring Sequences

Sequences whose terms depend in a simple manner on their predecessors are of importance for a variety of applications such as time series analysis and digital filters. In this study, the focus is on integer fields with finite number of elements. Of particular interest is the case where the terms depend linearly on a fixed number of predecessors, resulting in linear recurring sequences. These sequences are widely employed in coding theory, cryptography, and other areas.

Let k be a positive integer, and let $a, a_0, a_1, ..., a_{k-1}$ be given elements of a finite field F_q. A sequence s(0), s(1), … of elements of F_q satisfying the relation:

$$s(n+k) = a_{k-1}s(n+k-1) + a_{k-2}s(n+k-2) + ... + a_0 s(n) + a$$

for n= 0, 1, … (III. 3)

is called a kth order linear recurring sequence in F_q. The terms s(0), s(1), …, s(k-1) which determine the rest of the sequence uniquely, are referred to as the initial values. We define a homogenous linear recurrence relation if a=0.

Ultimately periodic and periodic sequences: A linear recurring sequence is ultimately periodic with a period r if there exists integers r>0 and $n_0 \geq 0$ such that s(n+r)=s(n) for all $n \geq n_0$, and it is periodic if coefficient $a_0 \neq 0$.

We now have to establish the properties of the recurrent sequence and how to find the period.

Theorem III. 1: Let F_q be any finite field and k any positive integer. Then every kth order linear recurring sequence in F_q is ultimately periodic with least period r satisfying $r \leq q^k$, and $r \leq (q^k - 1)$ if the sequence is homogeneous.

The period of a homogeneous recurrent sequence, r, could be obtained by finding the value of r that satisfies the following equation:

$$f(x)s(x) = (1 - x^r)h(x) \qquad \text{(III. 4)}$$

where the characterisitc polynomial, f(x):

$$f(x) = x^k - a_{k-1}x^{k-1} - a_{k-2}x^{k-2} - ... - a_0 \in \mathrm{F_q}[x] \qquad \text{(III. 2)}$$

and

$$s(x) = s(0)x^{r-1} + s(1)x^{r-2} + ... + s(r-2)x + s(r-1) \in F_q[x] \qquad \text{(III. 5)}$$

and

$$h(x) = \sum_{j=0}^{j=k-1} \sum_{i=0}^{i=k-j-1} a_{i+j+1}s(i)x^j \in F_q[x] \qquad \text{(III. 6)}$$

with $a_k = -1$.

The calulation of the period r is much easier if the characteristic polynomial of the linear recurrent sequence is an irreducible polynomial, i.e. it can not be factored in the finite field $F_q[x]$.

Theorem III. 2: Let s(0), s(1), … be a homogeneous linear recurring sequence in $F_q[x]$ with a nonzero initial state vector, and suppose the characteristic polynomial $f(x)$ is irreducible over $F_q[x]$ and satisfies $f(0) \neq 0$. Then the sequence is periodic with least period r equal to $ord(f(x))$.

Definition of Order of a Polynomial: Let $f \in F_q[x]$ be a polynomial of degree m>=1 with $f(0) \neq 0$, then the least positive integer e for which f(x) divides $(x^e - 1)$ is called the order of f and denoted by ord(f).

Example: Consider the linear recurrence relation:

s(n+3)=5 s(n) $\in F_7[x]$

which has the charateristic polynomial $f(x) = x^3 - 5 = x^3 + 2 \in F_7[x]$. The characteristic polynomial is irreducible and has $ord(f(x))$ =18 which is the same as the period r = 18.

It should be clear that the irreducible polynomial is not unique to the period, i.e. for the same period we can find more than one irreducible polynomial. Thus, several linear sequences with different characteristic polynomials, each might be irreducible, could have the same period.

III. 3 Estimation of the recurrent sequence parameters

In dealing with real data, one is confronted with a periodic signal that has a known period and a field that could be changed according to our scaling. For example, our exchange rate data could be scaled to be between 0 and 7 or between 0 and 23. Each will result in a different field while the period is still the same. Given both the field and the period, one could find the irreducible polynomials, and the corresponding linear sequences, that will generate the observed period. By comparing the data generated from the different tabulated polynomials and the actual observed data one could determine the right equation that describes the observed recurrent sequence. This method is called the look-up-table method.

III. 3a Look-up-table method

We know that the period r is a factor of $(q^k - 1)$, i.e. it is divisible by $(q^k - 1)$, where q is the number of elements of the filed, and k is the degree of the characteristic polynomial. If we assume that the characteristic polynomial of the linear recurrent sequence is irreducible, and since the period r is approximately known from the provided observations, then for each field order, q, there is a finite number of k values that satisfy the condition that r is divisible by $(q^k - 1)$, i.e.

$(q^k - 1)$ = 0 mod r

where mod is the modulus operation, for example 15= 0 mod 5 since 15 divides the modulus 5.
and for each polynomial degree k there is a finite number of irreducible polynomials.

Example: Let q=7 and let r=38, and assume that the linear recurrent sequence has an irreducible characteristic polynomial of unknown degree k<5. Thus, k must satisfy the equation

$7^k - 1 = 0 \bmod 38$

We have the following cases: k=1, yield 7-1=6 which does not satisfy the equation.
k=2, yields 49-1=48 which does not satisfy the equation. k=3, yields 343-1=342 which is equal to 9x38, i.e it satisfies the equation. k=4, yields 2401-1=2400 which does not satisfy the equation and so on.
For k=3, we have several irreducible polynomials; specifically:
(1) $f(x) = x^3 + 2x + 1$, which corresponds to the sequence s(n+3)=-2 s(n+1) - s(n) = 5 s(n+1) + 6 s(n),
(2) $f(x) = x^3 + x^2 + 3x + 1$, which corresponds to the sequence s(n+3) =- s(n+2) - 3 s(n+1) – 1 = 6 s(n+2) +4 s(n+1) + 6.
(3) $f(x) = x^3 + 3x^2 + x + 1$, which corresponds to the sequence s(n+3) = -3 s(n+2) -s(n+1) - s(n) = 4 s(n+2) + 6 s(n+1) + 6 s(n),
(4) $f(x) = x^3 + 3x^2 + 4x + 1$, which corresponds to the sequence s(n+3) = -3 s(n+2) - 4 s(n+1)- s(n) = 4 s(n+2) + 3 s(n+1) + 6 s(n),

(5) $f(x) = x^3 + 4x^2 + 3x + 1$, which corresponds to the sequence s(n+3) = -4 s(n+2) - 3 s(n+1) - s(n) = 3 s(n+2) + 4 s(n+1) + 6 s(n).
Thus, given that the sequence length is 34 and assuming that the data is generated through an irreducible polynomial, one searches in the possible five (5) solutions given above.

In many situations, however, the linear recurrent sequence does not necessarily have a characteristic polynomial that is irreducible. One might have the same period which is generated through different polynomials. However, there is a link between these different polynomials as is shown in the following theorems.

Theorem III. 3: Respresentation of an arbitrary polynomial: Any arbitrary monic polynomial $f(x)$ that belongs to the field F_q and of positive degree with $f(0) \neq 0$ could be factored as:

$$f(x) = \prod_{i=1}^{i=h} g_i(x)^{b_i} \qquad \text{(III. 7)}$$

where the $g_i(x)$ are distinct monic irreducible polynomials over F_q and the b_i are positive integers. From this representation one could deduce how many sequences could be generated and the period of each sequence. Thus, given the data there is a finite number of linear sequences with different initial conditions and with different factorization that have the same period. The equations of these sequences are known in advance, and one could pick from them the proper equation that best fits the data.

Example: Let $f(x) = (x^2 + x + 1)^2 (x^4 + x^3 + 1) \in F_2[x]$, i.e. $f(x)$ is factored into two irreducible polynomials in the field $F_2[x]$, $(x^2 + x + 1)$ and $(x^4 + x^3 + 1)$. The first polynomial is raised to the power 2, i.e. $b_1 = 2$, and the second irreducible polynomial is raised to the power 1, i.e $b_2 = 1$. According to the theories of polynomial representation [Lidl and Niederreiter; 1997], the linear sequences that have the above charateristic polynomial but with different initial conditions could have one sequence of period 1, three sequences with period 3, twelve sequences with period 6, sixty sequences with period 15, and one hundred eighty sequences with period 30.

Another approach is to estimate the coefficients of the recurrent sequence directly using, for example, minimum square error criterion. This approach, however, suffers from the many local minima's that exist and one has to use the Genetic Algorithm or the Simulated Annealing method to find the desired integer coefficients. The simple Genetic Algorithm will be used in this report.

III. 3b The simple Genetic Algorithm

Given a sequence of N data points, all of which are integers and scaled to have a maximum value of (q-1) and a minimum value of 0, one is interested in modeling this sequence as a linear recurrent sequence that belongs to the field F_q. Let k be a positive integer, and let $a, a_0, a_1, ..., a_{k-1}$ be the unknown integer parameters that are elements of a finite field F_q. Then, the data sequence $\hat{s}(0), \hat{s}(1), ...$ of elements of F_q satisfy the relation:

$$\hat{s}(n+k) = a_{k-1}\, \hat{s}(n+k-1) + a_{k-2}\, \hat{s}(n+k-2) + ... + a_0\, \hat{s}(n) + a$$

for n= 0, 1, …, (N-1-k) (III. 8)

Note that the estimate of the sequence at time (n+k), $\hat{s}(n+k)$, appears in the left hand side of equation (III. 8) while on the right hand side, there appear also the estimated values of the sequence not the observed values. The initial conditions could be either estimated or set equal to the observed values.

Then, find the unknown integer parameters $a, a_0, a_1, ..., a_{k-1}$ that minimze
the cost function, J:

$$J = \sum_{n=0}^{n=N--1-k} [s(n+k) - \hat{s}(n+k)]^2 \qquad \text{(III. 9)}$$

Note that the cost function has more than one minima since equation (III. 8) is a nonlinear relation. The global minima is obtained through the simple Genetic Algorithm as briefly described next.

Genetic Algorithms have been suggested as a global optimization tool [IEEE; 1994]. The major advantage is that the search is guaranteed to converge to the global minima, and that no gradient is required. As in many optimization techniques, there are several tuning parameters that are problem dependent. In Genetic Algorithms, these tuning parameters are the probabilities of cross over and mutation. The proposed algorithm estimates, adaptively, these two probabilities, thus, increasing the convergence speed. The algorithm can be described as follows:

(1) Each unknown parameter represents a gene. The unknown parameters are the coefficients of the linear recurrent sequence model used. In the proposed algorithm, each of the probabilities of mutation, p_m, and cross over, p_c, is considered an unknown parameter, which is represented by a gene. The genes are not mixed together; i.e., each gene is also a chromosome. Each parameter is, independently, treated in terms of mating, cross over, and mutations. Each parameter is initialized at random with a population of size S. Usually the population size is around 20 times the number of unknown parameters. Thus, we have S vector solutions. Each parameter is coded through binary code, and each parameter has minimum and maximum values. For example, if a parameter is known to lie between 0 and 2, and if the code is 8 bits wide, then 00000000 corresponds to 0, and 11111111 corresponds to 2.

(2) Fitness-function calculations are performed for the S vector solutions. For each solution, we calculate the negative of the Distance-measure, or the least squares, which represents the fitness function ff_i.

(3) The probability of reproduction for the ith solution, p_{ri}, is calculated according to the formula

$$p_{ri} = \frac{ff_i}{\sum_{s=0}^{s=S-1} ff_s}$$

where S is the total number of solutions. If there is a big discrepancy between the maximum and minimum values of the fitness function, other forms of the probability of reproduction should be used.

(4) At each iteration, two solutions for each parameter are picked at random according to their probabilities of reproduction. These two solutions represent the two parents. This process is repeated S/2 times. Thus, at the end we get S parents.

(5) The two parents will mate through the procedure of cross over. The mating will occur with probability p_c. The location of the crossover is chosen according to the uniform distribution. If the size or code for each parent were 8 bits, then the cross over position would be between the second and the seventh position. The result of the mating will be two offsprings. Thus, if the two parents are xxxxxxxx and yyyyyyyy, and the cross over location is the third, then the resulting two offsprings will be xxxxxyyy, and yyyyyxxx. The two offsprings represent two new solutions and they replace the two parents.

(6) Mutation will be applied with probability p_m. In this operation, every single bit in every solution is subject to change with probability p_m. For example, if a solution parameter is 10001111, and the mutation will occur at the first bit, then the new solution parameter would be 10001110.

(7) The new S vector of solutions are used with the error criterion to find their corresponding fitness function and the whole process is repeated again.

(8) Stop the iterations when the number of iterations exceeds a limit or when the error is below a threshold. The obtained best solution, with maximum fitness, is the desired solution.

In summary, the main steps of the proposed algorithm are: (1) for a window of N data points, define the predetermined variables, including the lagged dependent variables, that are likely to affect the dependent variable; (2) Using the distance measure of equation (III. 9), this is the fitness function; (3) For the linear recurrent model, scale the data to have a minimum of zero and maximum of (q-1); (4) use the simple Genetic Algorithm to get the minimum square error estimate of the unknown coefficients, $a, a_0, a_1, ..., a_{k-1}$, in equation (III. 8). Stop the iterations at the time interval where the error reaches a minimum, or when the number of iterations exceeds a predetermined value.

IV. Discussion of Results

We study the behavior of the $US / EURO exchange rate based on data spanning the period April 28, 2000 (week 1) to March 28, 2001 (week 54). The period between April 28, 2000 (week 1) and February 2, 2001 (week 45) was used to find the models. The models were then used to forecast the $US/EURO between the period February 9, 2001 (week 46) to March 21, 2001 (week 54); a total of 9 weeks. Before modeling the data as a linear recurrent sequence, we could first find the trend in the data and study (1) the first difference in the data, and (2) the remainder between the actual data and the trend. Instead we will study the data itself with its trend.

The data, the $US/EURO exchange rate, was first scaled to be between 0 and 12, q=13, i.e, the field of interest is F_{13}. The exchange rate was then modeled as a nonhomogeneous linear recurrent sequence. The maximum lag was set to 3. Other degrees of lags could also be used. This, however, will increase the computation time. The lag of 3 yielded a characteristic polynomial of degree 3, where the general model is:

$$s(n+k) = a_{k-1}s(n+k-1) + a_{k-2}s(n+k-2) + ... + a_0 s(n) + a \in F_{13}$$

for n= 0, 1, … (IV. 1)

where s(n) is the $US/EURO exchange rate at time n.
The simple Genetic Algorithm was used to estimate the parameters of the linear recurrent sequence. The estimates are given below:

a_2	a_1	a_0	a
-1	5	1	-1

Then, since all the parameters and the data $\in F_{13}$, the linear recurrent equation that represents the $US/EURO data is estimated as:

$$\hat{s}(n+3) = 12\hat{s}(n+2) + 5\hat{s}(n+1) + \hat{s}(n) + 12$$

For n= 0, 1, … (IV. 2)

where $\hat{s}(n)$ is the estimated $US/EURO exchange rate at time n. Note that this equation is self sustained or self exited i.e. it does not need external input or external observations. We used the observations once to get an estimate of the unknown parameters. This is unlike conventional estimates where, on the right hand side of the estimation equation, we have the observations and on the left hand side we have the predicted or estimated value.

Through trial and error, it was found that taking the average of the three previous values improves the accuracy of the estimates. Thus, the new $US/EURO estimate at time n, $\hat{y}(n)$, is given as:

$$\hat{y}(n) = [\hat{s}(n) + \hat{s}(n-1) + \hat{s}(n-2)]/3 \qquad \text{(IV. 3)}$$

After taking the average, we scaled the estimates back to the actual levels of the data. Equations (IV. 2) and (IV. 3), after scaling, were used to predict the exchange rate 9 weeks ahead. It could be used to predict as many weeks in the future as we like. But since the $US/EURO might be described by a time-varying linear recurrent equation, one should minimize the prediction steps as much as possible.

The conventional AR(3) model was used for comparison purposes. The least square error for the AR(3) model (0.011) was less than that of the smoothed linear recurrent sequence (0.021). Nevertheless the prediction of the AR(3) was less accurate than that of the linear recurrent sequence.

In Fig. 1, we presented the actual $US/EURO exchange rate and the smoothed linear recurrent model estimates. We also present the predictions in 9 weeks ahead using equations (IV. 2) and (IV. 3) and using the AR(3) model. As one notices, the AR(3) model predicts an upward trend while the smoothed linear recurrent model was able to capture the ups and downs of the $US/EURO exchange rate (weeks 46-54).

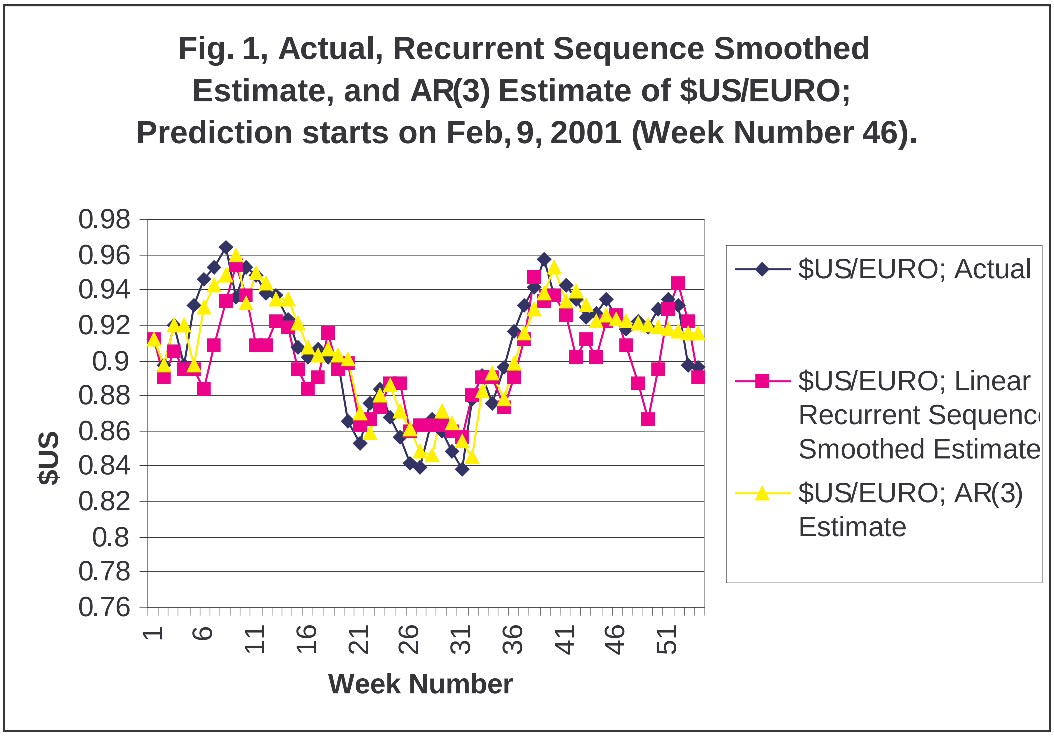

Fig. 1, Actual and Estimated $US/EURO; Linear Recurrent Sequence Smoothed Estimate, and AR(3) Estimate. Predictions start on Feb, 9, 2001 (Week Number 46) till March 21, 2001 (Week Number 54).

V. Conclusions:

In this paper, the linear recurrent sequence model is introduced and applied to the problem of modeling the $US/EURO exchange rate. Based on this model, one is able to forecast the $US/EURO exchange rate several weeks ahead. The new linear recurrent sequence model assumes that all the data and model parameters belong to a finite integer field. Properties of the model could be evaluated based on the lags and the value of the parameters. An important feature is the period of the recurrent sequence, which tells us about the possible repetition pattern in the data. This model takes into consideration the fact that the actual (observed) data suffer from the reversal effect phenomenon [Jegadeesh; 1990]. By using the simple Genetic Algorithm, we are able to find the coefficients that give the global minima of the error criterion. The linear recurrent sequence model could also have time-varying parameters, which, if estimated, might improve its prediction power. These issues are currently under investigation.

REFERENCES:

A. Abutaleb, 1997, "A Genetic Algorithm for the Maximum Likelihood Estimation of the Parameters of Sinusoids in Noisy Environment", Circuits, Systems, and Signal Processing, Vol. 16, #1, pp. 69-81.

A. Abutaleb, 2001, "Number Theory and Bootstrapping for Phase Unwrapping", IEEE Trans. Circuits and Systems, Part I, Accepted for Publication.

A. Abutaleb, and M. Papaioannou, 1990, "An Adaptive Filter for Time-Varying parameter Models ", Int. J. Adaptive Control and Signal Processing, Vol.4, pp. 335-343.

A. Abutaleb and M. Papaioannou, 2000, "Maximum Likelihood Estimation of Time-Varying Parameters: an Application to the Athens Stock Exchange Index",Applied Economics, Vol. 32, # 10, Aug. 2000, pp. 1323-1329.

1. Z. Bodie, A. Kane, and A. Marcus, 1995, Essentials of Investments, Irwin, Chicago, USA.
2. G. Chow, 1987, Econometrics, McGraw Hill, New Jersey, USA
3. D. Denning, 1982, Cryptography and Data Security, Addison-Wesley Publishing Company, Reading, Massachusetts, USA.
4. F. Diebold, 1998, Elements of Forecasting in Business, Economics, Government and Finance, South Western College Publishing, USA.
5. Harvey, 1992, Forecasting, Structural Time Series Models, and the Kalman Filter, Cambridge University Press, U. K.
6. J. Holland, 1969, “Adaptive plans optimal for payoff-only environments”, Proc. 2nd Hawaii Int. Conf. Systems Sciences, pp. 917-920.
7. J. Holland, 1975, Adaptation in Natural and Artificial Systems, Ann Arbor: The University of Michigan Press.
8. IEEE Transactions on Neural Network, Jan 1994, Special issue on Evolutionary Computation.
9. N. Jegadeesh, 1990, “Evidence of Predictable Behavior of Security Returns”, J. Finance, Vol. 45, pp. 881-898.

B. Lehman, 1990, “Fads, Martingales and Market Efficiency”, Quarterly Journal of Economics, Vol. 105, pp. 1-28.

10. W. Leveque, 1977, Fundamentals of Number Theory, Dover Publications Inc., New York, USA.
11. R. Lidl, and H. Niederreiter, 1997, Introduction to Finite Fields and their Applications, Cambridge University Press, United Kingdom.
12. Z. Michaelewiz, 1994, Genetic Algorithms + Data Structures = Evolution Programs, Springer Verlag, NY.
13. W. Press, B. Plannery, S. Teukolsky, and W. Vettering, 1988, Numerical Recipes in C, Cambridge University Press, UK.
14. M. Priestley, 1988, Non-linear and Non-stationary Time Series Analysis, Academic Press, New York, USA.
15. M. Rahman, M. Mustafa, and D. Burckel, 1997, "Dynamics of the Yen-Dollar Real Exchange Rate and the US-Japan Real Trade Balance", Applied Econometrics, Vol. 29, pp. 661-664.
16. P. Young, 1984, Recursive Estimation and Time-Series Analysis: An Introduction, Springer-Verlag, New York, USA.

Article received: 2006-05-29