Modeling And Forecasting Exchange-Rate Shocks

Andreou, A. S. and Zombanakis, George A. and Likothanassis, S. D. and Georgakopoulos, E.

University of Patras, Bank of Greece, University of Patras, Computer Technology Institute of Patras

6 March 1998

Modeling And Forecasting Exchange-Rate Shocks

Andreou A.[1], Zombanakis G.[2], Likothanassis S.[1,3] and Georgopoulos E.[1]

1. University of Patras, Dept. of Computer Engineering & Informatics, Patras 26500, Greece, tel.: +61 997755, fax: +61 997706, e-mail: aandreou@ceid.upatras.gr
2. Bank of Greece, Research Dept. 21 Panepistimiou Str. 102 50, Athens, Greece, tel: +01 3235809, fax: +01 3233025
3. Computer Technology Institute, 3 Kolokotroni Str., 26221, Patras, Greece

Abstract

This paper considers the extent to which the application of neural networks methodology can be used in order to forecast exchange-rate shocks. Four major foreign currency exchange rates against the Greek Drachma as well as the overnight interest rate in the Greek market are employed in an attempt to predict the extent to which the local currency may be suffering an attack. The forecasting is extended to the estimation of future exchange rates and interest rates. The MLP proved to be highly successful in predicting the shocks, while exhange-rates and interest-rates forecasts with MLP and RBF optimized by a genetic algorithm resulted in good approximations.

1. Introduction*

The scope of the present paper is the study of the behaviour of exogenous disturbances which affect the historical course of the exchange rate of a certain currency and, what is most important, the examination of the way in which such disturbances or "shocks" are expected to influence the future course of the exchange rate of the currency involved.

The analysis will take place using the method of artificial neural networks, which, being a data-driven approach, has been considered preferable to traditional model-driven approaches used for forecasting purposes, for reasons which will be later analysed. Two different neural network architectures will be used and results are interpreted in a comparison manner too. The time series chosen represent the drachma exchange rate versus the US dollar as well as versus three major European currencies, namely the Deutche Mark, the French Franc and the British Pound. The policy instrument employed to rectify the course of such exchange rates when disturbed by external shocks is the overnight interbank interest rate, a typical authorities tool used in cases of intervention. It is most unfortunate that the foreign reserves spent by the Central Bank to stabilise the exchange rate in cases of crises have not been available, since this piece of information is considered to be restricted. This leaves us with only one instrument of intervention to express the policy makers corrective measures taken in cases of crises.

All series consist of about 2.5 thousand observations covering the period between January 1990 and December 1997, a period which incorporates a wide variety of disturbances or "shocks", each of different character, duration and intensity. The most important of these shocks have been selected to represent the source of the noise affecting the course of the exchange rate of the

* Thanks are due to Mr. John Farsarotas and Mrs. Helen Gazopoulou-Karahaliou of the Foreign-Exchange Department and the Economic Research Department of the Bank of Greece respectively for valuable assistance during the preparation of this paper.

drachma against the four currencies chosen. The data sets include an additional flag that is set to zero for the period corresponding to a normal state and to one for the period under a shock.

The model will be trained to identify the existence of such shocks and to predict the course of the echoing rate of the drachma in each case. A primary forecast on the interest rate as well will be attempted, in order to compare results to the ones of the exchange rates. As far as the extent to which a future shock, its duration and its initial time may be forecasted, the results derived seem to be promising. They require, however, additional information before they can be used to achieve reliable forecasts.

The rest of the paper is organized as follows: Section 2 provides an historical overview of the past and present history in the Greek economy and analyzes the kinds of shocks available in the data used in the present work. Section 3 incorporates with a brief presentation of the neural networks architectures used and their implementation techniques. Section 4 includes the numerical experiments and summarizes the forecasting performance of the networks. Finally we conclude in section 5.

2. The Greek Environment

2.1 Some History

The exchange rate has been used as a major policy instrument in Greece since March 1975, when the drachma ceased to be pegged to the US dollar in terms of a fixed exchange rate. The reasoning behind the extensive use of exchange-rate policy by the authorities was based on its effectiveness thanks to its speed, flexibility and politically costless action. This action aimed at improving or even restoring the degree of competitiveness of Greek goods and services in the international markets, restricting the balance of payments deficit or even curtailing the inflation rate, depending on the circumstances. Since 1975 and until the early 90's, the drachma had been following a depreciating trend which, in parallel to the implementation of an expansive monetary policy, has contributed to preserving high inflation rates (Bank of Greece, 1984, p.31).

The competitiveness gain thanks to the January 1983 15.5% drachma devaluation in effective exchange rate terms was to its largest part wasted since up and until August of the same year the drachma was linked to the US dollar which was considerably revalued in effective exchange rate terms, thus dragging the drachma rates high up again. The continuing depreciation path of the drachma vis-a-vis both the US dollar, as well as the major European currencies since 1984, reinforced the inflationary procedure and revived inflationary expectations. As it has been realised by the authorities later on (Bank of Greece, 1986, p.22), this had been a major problem, bringing to attention the constraints of the exchange-rate policy, indicating that such measures can not be regarded by any means as a substitute for a cost-reducing or inflation-fighting policy. Indeed this policy option proved to be rather unfortunate, since high production costs resulted to a considerable erosion of the degree of competitiveness of the Greek products in the international markets.

The 15% drachma devaluation in nominal exchange rate terms on the 11th of October 1985, seemed to be more promising in terms of effectiveness, compared to the 1983 devaluation, since the restoration of the degree of competitiveness in the international markets was pursued not just on the basis of the exchange rate policy alone, but, in addition, with the backing of a strict monetary policy and an effort to curtail the budget deficit. These measures aimed at delaying the

rate of increase of nominal wages and at restricting the effect of the devaluation on labour cost and prices so as to preserve, or even to improve the degree of competitiveness of the Greek goods and services (Bank of Greece, 1986, p. 29).
This stabilisation program, however, did not attribute much attention to the supply side, ignoring the need for structural reforms on the microeconomic level. It also underestimated the fact that since the import component of the Greek final products is high in terms of raw materials and semi-finished products, a devaluation or a depreciation is expected to lead to increases of the prices of these products, offsetting the benefits upon price competitiveness. Thus this program was abandoned long before it had attained its targets, this leading to a revival of the competitiveness deterioration and consequently to the burdening of the balance of payments. Thus, the rapid depreciation policy from 1988 onwards kept re-feeding inflation, since the business sector relied on the state support, regarding a cost subsidy in the form of a continuous drachma depreciation, as given. As a result the supply side neglected improving its productive efficiency and was reluctant to resist wage demands (Bank of Greece, 1991, p.33).

The increasing current account deficits accompanied by the accelerating inflation rates due to the "easy" exchange-rate policy followed from 1988 onwards, led the authorities to realise the ineffectiveness of the devaluation or even the rapid depreciation solutions. This ineffectiveness has been a fact extensively supported in the literature (Brissimis and Leventakis, 1989), (Karadeloglou, 1990) and quite recently, in connection to export-promoting strategies (Karadeloglou et al., 1998; Zombanakis, 1998). Thus, the authorities focused their attention on improving productivity, curtailing production costs and adjusting the supply side to changes in demand, since the early 90's. They have thus decided to start by following a non-accommodating depreciation policy in the sense that the rate of depreciation does not fully cover the inflationary gap between Greece and its trading partners. Additional intervention measures have also been imposed in order to face the impact of seasonal or irregular factors on the drachma exchange rate.

2.2 The Recent Past

During the past few years, the Bank of Greece, encouraged by the restrictive action of the non-accommodating exchange-rate policy upon inflation, has opted for an even tighter version, keeping the exchange rate of the drachma vis-a-vis the ECU, fixed. This has become known as the "hard- drachma policy", aiming at weakening both inflationary pressures and expectations concerning such pressures in the future. This policy has proven to be more than successful since, its drastic anti-inflationary impact was accompanied by a significant interest-rate reduction representing a relief for the budget deficit and a decrease of the capital cost of the business sector. Thanks, also, to the "hard-drachma", increases in servicing the foreign-currency denominated public debt have been avoided, the foreign-exchange risk has been restricted, while the cost of the imported raw materials for Greek export firms, the products of most of which bear a high import component, has been held constant.

Bearing in mind the benefits of the "hard-drachma policy", the position of the Central Bank on this matter has been adamant since deviating from such tactics would imply, among other things, that the authorities ignore international long-run policy targets (Bank of Greece, 1994). The Governor of the Central Bank, in fact, during the annual monetary program announcement for 1997 refers to the exchange-rate policy as restricted by our commitment to curb inflationary pressures announcing the intention of the authorities to keep the drachma/ECU exchange rate fixed for 1997, adding, however, that this policy will be exercised "allowing for considerably higher flexibility than what was the case in the past" (Bank of Greece, 1997). The translation of this enigmatic statement is that the exchange rate may be fixed in year-end terms, but fluctuations within the year will not be predetermined, their direction of change and magnitude depending on the inflationary pressures and the market conditions. This flexibility is expected to discourage speculative action which has been the source of a considerable degree of instability in the domestic market during the recent past.

2.3 Certain Undesired Complications

It seems, however, that the price that the authorities will have to pay for the success of the "hard-drachma policy" as an inflation-fighting device will be the continuous alert to face speculative attacks and their consequences. It seems that the market believes that the Greek currency is overvalued, and, therefore, an easy prey for speculative attacks, four of which have threatened the drachma during the last few months of 1997: End of May, Mid-July, end of October and beginning of December. The formal explanation is the international crisis that originated in the markets of Korea or Thailand, a direct impact of which has affected the Athens stock market and, consequently, the drachma. We feel that this explanation is rather superficial, since the bulk of the companies in the Athens stock market are not of a multinational caliber and as a result, have little or even nothing to do with these markets. Speculative movements, however, seem to be the explanation for the recent, repeated runs on the drachma, since investors attempted to make up for losses suffered in the markets of South-East Asia by selling foreign-currency denominated Greek bonds or stocks placed in the Athens stock exchange. This caused stock prices to drop, particularly during the December run, when the market has admittedly overreacted in the light of the painful experience of the previous attacks. Whatever the reason behind these runs, the fact remains that they created pressures on the drachma on all four occasions mentioned above which, however, have been successfully dealt with by the Central Bank with immediate market interventions by means of both raising the overnight rate and pumping foreign exchange reserves. As it will described later on in this paper, the intervention has been more than successful, since the drachma rates have only been marginally affected.

The drachma, however, is an ERM candidate, and it seems that beginning January 1999 it will join the ERM in order to comply with the two year ERM membership requirement before it catches the last train for the Euro, in January 2001. This means that during the first half of 1998, a period preceding the determination of the currencies' exchange rates in the Euro, speculative attacks like the ones described above are expected to continue. This forecast, no matter how alarming may sound, is based on the fact that the authorities have not yet proceeded to the necessary structural changes required concerning privatisation policy as well as capital and labour markets restructuring. Very little has also been attained with reference to government spending cuts. Thus the authorities seem to be relying too heavily on the "hard-drachma policy" as an anti-inflationary device, leading the markets to believe that the drachma is overvalued, as earlier indicated.

2.4 A "Shocking" Background

A thorough study of the recent history of the drachma exchange-rate, however, requires that one considers the drachma behaviour when exposed to various endogenous or exogenous disturbances as well as the reactions of the authorities to defend the drachma position in such cases. The most important incidents in the 1990s which have developed to "shocks" against the Greek currency, and which differ considerably between one another with respect to causes, duration and intensity have been the following:

The international crisis that started on the 16th of Septrember 1992 affected the Greek market rather moderately, since the Central Bank did not have to raise the overnight rate by more than about 75% on just one day, a score rather low compared to the cases we shall examine later on. The consequences of the turmoil in the international markets, however, kept affecting the domestic economy for about two months.

The 1993 parliamentary elections in October have influenced the domestic market, since the non-accommodating exchange-rate policy led the market to believe that a drachma devaluation might be imminent. This was strictly speaking a "family" business since the cause of this shock was purely domestic while the overnight rate increases did not exceed 45%. By the end of October, the market behaviour had returned back to normal again.

The 1994 shock, characterised as one of "institutional" character, has been more or less anticipated by the market and, certainly, by the authorities, since it was originally scheduled to take place at the beginning of July 1994 and the closer we moved to that day, the more nervous the market appeared as this was obvious judging from the pressures on the drachma during that period. Thus, in order to avoid speculative attacks, the authorities decided to surprise the market announcing the short-term capital movement liberalisation on Monday, May 16, 1994. The overnight interest rate rose by more than 11 percentage points on that day, reaching a maximum increase of about 670% by the end of that week. An attempt of the Bank to "loosen" the overnight rate on Monday, 23 proved to be premature since the pressure on the drachma seemed to persist. Thus, the overnight rate rose to an overall maximum of about 860% during the second week of that "crisis" to start a declining course since then which ended with gradually bringing the overnight rate back to pre-crisis levels by July 12. The successful treatment of this attack on the basis of the overnight rate was supplemented by the outflow of about $3 billion of the Bank's foreign exchange reserves to maintain the exchange rate of the drachma.

The repeated attacks on the drachma during 1997 are of purely speculative nature as it has already been explained, each one being much shorter in duration compared to the "institutional" shock previously analysed, each one not lasting more than four to five days . As regards the tactics used to confront the speculators, the overnight rate did not rise at all during the May attack, while it rose by just one percentage point during the July run. It appears, therefore, that the authorities relied exclusively on spending foreign exchange reserves to face these two attacks, since maintaining the drachma rates costed about $4 billion to the economy. The October run, however, seemed to be more persistent, leading the authorities to spend about $3 billion along with raising the overnight rate by about 950% on one occasion, in order to support the drachma exchange rate. On this particular occasion it took the overnight rate about a month to return to its pre-crisis levels. Finally, as regards the December attack its features have been very similar to the May and July ones.

3. Neural Network Methodology

In this section we investigate for data driven techniques, like the emerging technology of artificial neural networks. This technique belongs to a class of data driven approaches, as opposed to model driven approaches. The process of constructing such a machine, based on available data is addressed by certain general purpose algorithms. The problem is then reduced to the computation of the weights of a feedforward network to accomplish a desired input-output mapping and can be viewed as a high dimensional, non-linear system identification problem.

3.1 The Multi-Layer Perceptron Neural Network (MLP)

In a feedforward network, the units can be partitioned into layers, with links from each unit in the k^th^ layer being directed to each unit in the (k+1)^th^ layer. Inputs from the environment enter the first layer and outputs from the network are manifested at the last layer. A d-n-1 shown in Figure 1, refers to a network with m inputs, d units in the hidden layer and one unit in the output layer.

We use such m-d-1 networks to learn and then predict the behaviour of the time-series. The hidden and output layers realise non-linear functions of the form:

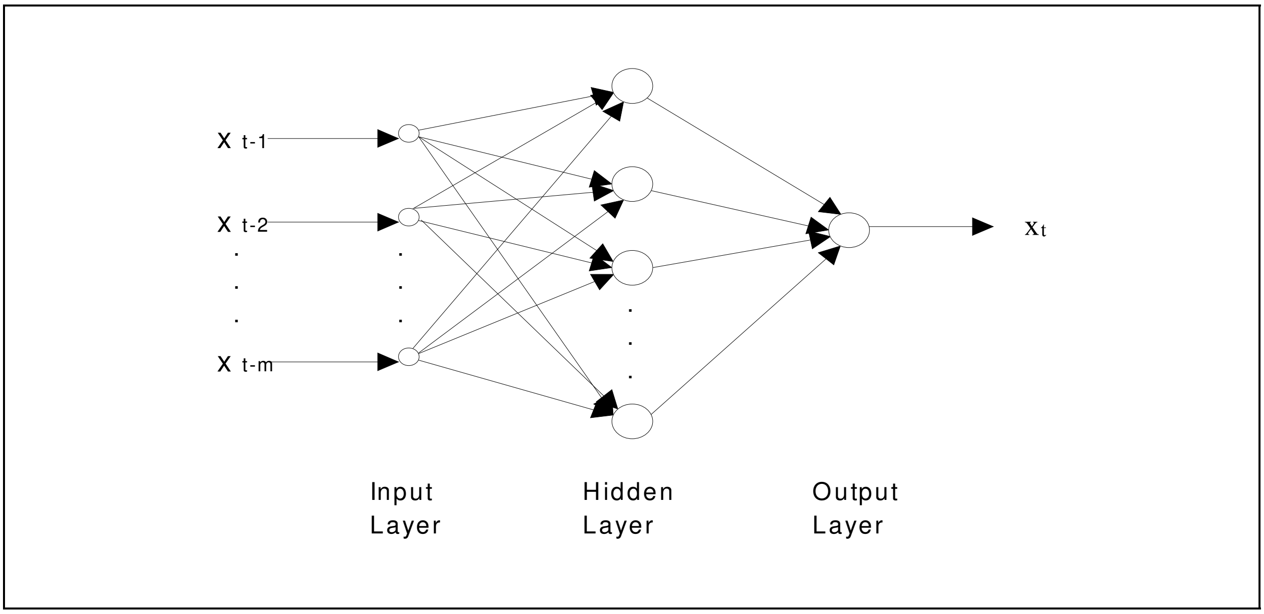

Figure 1 : ***The m-d-1 neural network architecture***

$$(1+\exp(-\sum_{i=1}^{m} w_i x_i + \Theta))^{-1} \qquad (1)$$

where w~i~'s denote real valued weights of edges incident on a node, Θ denotes the adjustable threshold for that node and m denotes the number of inputs to the node from the previous layer. As training algorithm we used the well known Back Propagation (Rumelhart and McLelland, 1986).

3.2 Radial Basis Function Networks

A basic Radial Basis Function (RBF) Neural Network may be depicted as shown in Figure 2 (Billings and Zheng, 1995), consists of an input, a hidden and an output layer. We can assume, without loss of generality, that the number of outputs of the network equal to one. The input vector to the network is propagated from the input nodes to the hidden nodes via unit connection weights. The hidden layer consists of a set of *radial basis functions.* Each hidden layer node is associated a parameter vector c_i, called centre. The hidden node calculates the Euclidean distance (norm) between the centre and the network input vector and then passes the result to a radial basis function. The radial basis functions in the hidden layer nodes are usually of the same type. Some common radial basis functions are :

a. The thin-plate-spline function: $\phi(v) = v*log(v)$ (2)
b. The Gaussian function: $\phi(v) = exp(-v^2/\beta^2)$ (3)
c. The multiquadric function: $\phi(v) = (v^2 + \beta^2)^{1/2}$ (4)
d. The inverse multiquadric function: $\phi(v) = 1/(v^2 + \beta^2)^{1/2}$ (5)
e. The cubic function: $\phi(v) = v^3$ (6)
f. The linear function: $\phi(v) = v$ (7)

where v is a non-negative number, that represents the distance between the input vector **x** and the centre c of the hidden node, while β is the width of the radial basis functions.

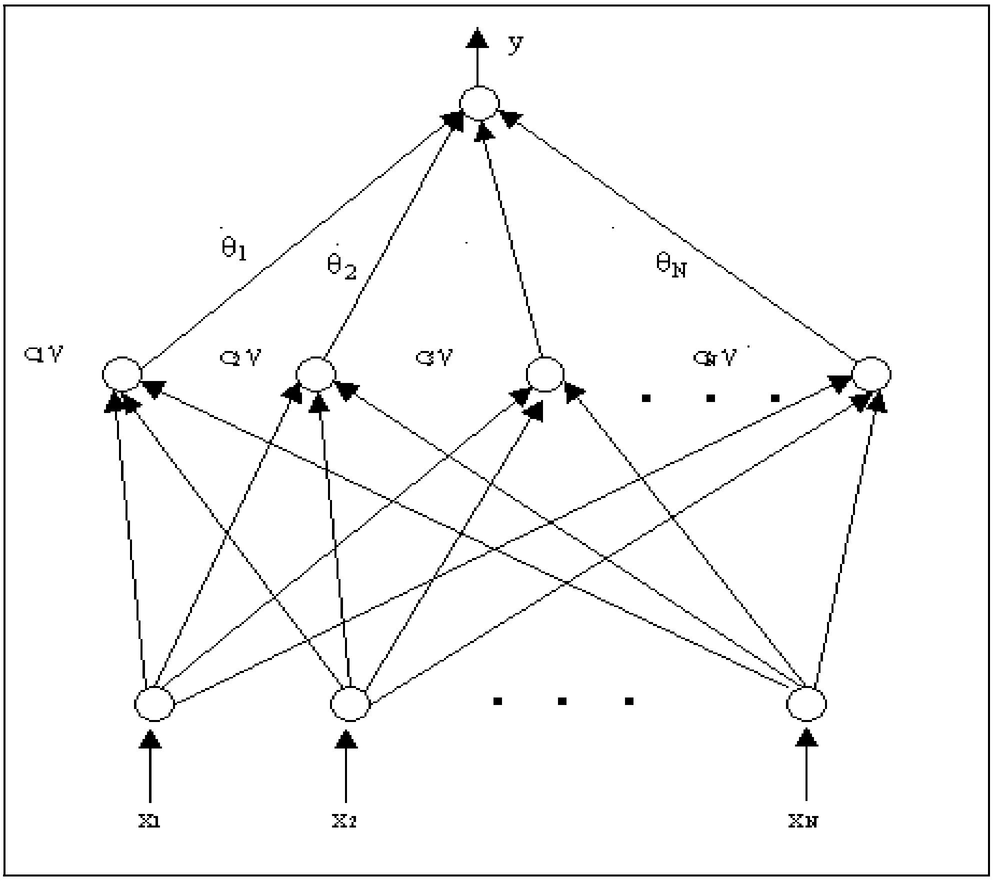

Figure 2: ***Architecture of a Radial Basis Function Neural Network***

In the present work the thin-plate-spline and the Gaussian function are used. However other functions, from these listed above, can be included by using a constant b parameter.
The response of the output layer is given by the following formula:

$$f(x) = \sum_{i=1}^{N} \theta_i \cdot \phi(\|x - c_i\|) \tag{8}$$

where N is the number of training patterns, and ||•|| denotes the Euclidean norm, $\mathbf{c}_i \in \mathbf{R}^n$ (i=1,...,N) is the ith centre (and in this particular network structure is the ith training pattern), $\mathbf{x} \in \mathbf{R}^n$ is the input vector, and θ_i (i=1,...,N) are the weights associated with the ith Radial Basis Function centre.

The complexity of the above architecture increases with the number of training data, thing that makes the implementation of the previous network unrealistic. In practical applications, it is often desirable to use a network with a finite number of RBF functions. Thus a natural approximated solution would be:

$$f^*(x) = \sum_{j=1}^{n_c} \theta_j \cdot \phi(\|x - c_j\|) \tag{9}$$

where n_c is the number of radial basis function centres, c_j is the jth centre that can be selected from the training patterns.
Assuming that we have a training set (x_i,y_i), where (i=1,2,...,N), $x_i \in \mathbf{R}^N$, $y_i \in \mathbf{R}$, and $x_i = (x_{i1},x_{i2},...,x_{iN})^T$ are the input patterns, the weights, centres and widths can be obtained by minimizing the function :

$$J_1(\theta, c) = \sum_{i=1}^{N} \left(y_i - f^*\right)^T \cdot \left(y_i - f^*\right) \tag{10}$$

where $\theta = (\theta_1,\theta_2, \dots , \theta_{nc})^T$, $c = (c_1,c_2, \dots ,c_{nc})^T$. This minimization problem can be solved using some non-linear optimization or gradient descent method. In the above network the number of hidden nodes (structure) is fixed (predetermined). That means that the structure can be determined only by trial and error. Therefore it is desirable to optimize the network structure, the centres and the connection weights simultaneously. For this case the objective function can take the following form (Billings and Zheng, 1995):

$$J_2(n_c, \theta, c) = \sum_{i=1}^{N} \left(y_i - \sum_{j=1}^{n_c} \theta_j \cdot \phi(\|x_i - c_i\|) \right)^T \cdot \left(y_i - \sum_{j=1}^{n_c} \theta_j \cdot \phi(\|x_i - c_i\|) \right) \tag{11}$$

The best structure that minimize the above function J_2 has N hidden layer nodes (equal to the number of training patterns) and the centres tend to the training patterns such that the network reverts to the one given in formula (8). This cause the network to be able to interpolate the training set and fail to capture the underlying functional relation of the samples, so the network will not be capable to generalize. In order to provide a compromize between network performance and complexity, Akaike's Information Criterion (AIC) can be used, and the function to be minimized can take the following form (Billings and Zheng, 1995):

$$J_3(n_c, \theta, c) = N \times log\left(\frac{1}{N} \cdot \sum_{i=1}^{N} \left(y_i - \sum_{j=1}^{n_c} \theta_j \cdot \phi(\|x_i - c_i\|) \right)^T \times \left(y_i - \sum_{j=1}^{n_c} \theta_j \cdot \phi(\|x_i - c_i\|) \right) \right) + 4 \times n_c \tag{12}$$

We used genetic algorithms in order to optimize the RBF network parameters. The number of inputs of the network (input nodes) is assumed to be known and the minimization of the objective function will determine both the network structure and the network parameters (weights and centres) simultaneously. The novelty in this paper is the training of the RBF networks using Genetic Algorithms, so in the subsequent paragraphs, we are going to focus primarily in the description of the modified genetic algorithm and the genetic operators that were implemented for the needs of this work.

3.3 The Genetically Trained RBF Network

Genetic Algorithms are a class of optimization algorithms which are good at exploring a large and complex space of a given problem in an intelligent way in order to find solutions close to the global optimum. In the present work we have implement a modified Genetic Algorithm (Michalewicz, 1994) that works as follows:

It maintains a population of individuals, $P(t)=\{i_1^t, \ldots ,i_n^t\}$ for iteration t. In our problem each individual is an RBF network with its own number of hidden nodes, centres and weights values. Each individual represents a potential solution to the problem at hand, and is implemented as some data structure S. Each solution (individual) i_j^t is evaluated to give a measure of its fitness. As such fitness function it is used the inverse Mean Square Error of the network. Then a new population (iteration t+1) is created by selecting the more fit individuals, according to their fitness (select step). Some members of the population undergo transformations (alter step) by means of “genetic” operators to form the new solutions (individuals). There are unary transformations m_i (mutation type), which create new individuals by a small change in a single individual (add or delete a random number of hidden nodes to or from the hidden layer, change the values of a random number of weights or centres of the network, e.t.c.) and higher order transformations c_j (crossover type), which create new individuals by combining parts from two or more individuals (take a random number n_1 of hidden nodes from a RBF network and a number n_2 from another RBF, and create a new RBF with n_1+n_2 hidden nodes). That way we can train the RBF networks and simultaneously change their structure. After some number of iterations the program converges. At this point is hopped that the best individual represents a near-optimum (reasonable) solution. These genetic operators will be described in full detail in the sequence.

Here, it would be interesting to state few things about the model of the RBF network that is processed by the modified genetic algorithm. The architecture of that model looks like the one of Figure 2; the novel thing is the implementation of that model. In the approach of (Billings and Zheng, 1995), the researchers code the RBF network as a string, use the genetic algorithm to manipulate that string, and then they decode the string back to a network in order to measure its performance.In our approach the basic philosophy is that we do not make use of some coding function in order to “go” from a neural network, to a floating point, or binary, string that represents that network, and then to have a genetic algorithm manipulating this binary string. On the contrary, it is used a modified genetic algorithm that handles directly the neural networks, and not some strings. The drawback of this approach is that we have to do with neural networks, structures much more complicated than strings, that costs in computer memory and computation time. This disadvantage is overcome by the great flexibility that give us; we can manipulate networks with different number neurons in the hidden region, add or delete hidden neurons and much more.

In the sequence, the genetic operators used to manipulate the RBF network, are presented in detail. (The name in the brackets correspond to the names of the functions used in the program.)

3.3.1 Initialization Process

Initialization is the first step of the algorithm. An initial population of potential solutions is generated, where each individual is a complete RBF neural network. The size of the population (pop_size) is a parameter of the algorithm. Each one of these individuals is created randomly. Only the number of inputs and outputs (i.e. the number of input and output neurons) are fixed (they depend on the function we want to teach to the network). The number of neurons in the hidden layer are chosen at random ,with uniform probability distribution, from an interval that has the general form [lower bound, upper bound], where lower and upper bounds are parameters of the algorithm. In the same manner, the centres are generated randomly in the range [minimum value of training pattern set, maximum value of training pattern set] and the connection weights in the interval [weight lower bound, weight upper bound], where these lower and upper weight bounds are also parameters of the algorithm. Thus, at the end of the initialization procedure we have a population of randomly generated individuals (neural networks).

3.3.2 Selection Process

The selection is an essential operation in genetic algorithms, since it constructs a new population with respect to the probability distribution based on fitness values of the individuals that belong to the previous population. We have implemented a number of different selection processes, but in our experiments we used the Elitist Selection Operator. In order to build that operator a roulette wheel with slots sized according to fitness is used. We constructed such a roulette wheel as follows (Michalewicz, 1994).

- Calculate the fitness value eval(I_i) for each individual I_i in the current population (i=1,...,pop_size). In our experiments as a fitness function we have used the following :

  $$\text{eval}(I_i) = 1/(1 + \text{MSE}) \qquad (13)$$

  Where MSE is the value of the Mean-Square-Error cost function.

- Find the total fitness of the current population:

  $$F = \sum_{i}^{pop\_size} eval(I_i) \qquad (14)$$

- Calculate the probability of a selection p_i for each individual I_i in the current population (i=1,...,pop_size) :

  $$p_i = eval(I_i)/F \qquad (15)$$

- Calculate a cumulative probability q_i for each individual I_i in the current population (i=1,...,pop_size) :

$$q_i = \sum_{j=1}^{i} p_j \qquad (16)$$

The selection process is based on spinning the roulette wheel (pop_size-1) times; each time we select a single individual for the new population in the following way:

- Generate a random float number r ,with uniform probability distribution, from the interval [0,1].
- If $r<q_1$ then select the first individual (I_1); otherwise select the ith individual I_i (where 2<= i <=pop_size) such that $q_{i-1} < r <= q_i$.

Obviously, some individuals would be selected more than once. This is in accordance with the Schema Theorem: the fittest individuals get more copies, the average stay even, and the worst die off.

Now in the empty pop_size-th place of the population we copy the best-ever individual. That way, the best-ever individual is conserved in every population.

3.3.3 Crossover Process

The crossover, or recombination, operator is a very important operator, that is applied to the new population. One of the parameters of the evolutionary algorithm is the probability of crossover p_c. This probability gives as an expected number (p_c*pop_size) of individuals which undergo crossover. We proceed in the following way:

For each individual in the new population:

- Generate a random float number r, with uniform probability distribution, in the interval [0,1].
- If $r < p_c$, select the specific individual for crossover.

In the sequence, we mate the selected individuals randomly : for each one in a pair of individuals (parents), we generate a random natural number in the range [1,...,number of hidden nodes-1]. This number indicates the position of the crossing point.Since the neural networks (individuals) in the population may not have the same number of hidden nodes, this crossing point is generated independently in each individual of the selected pair. Thus, supposing that we have two individuals : I N_1 N_2 N_3 O and I N_1' N_2' N_3' N_4' O, where I represents the input layer, O the output layer and N_i the i-th hidden node. If the crossing point for the first individual is 2 and for the second one is 3, then the “child” of these two individuals would look like this: I $N_1 N_2 N_1' N_2' N_3'$ O. Or schematically :

parent 1 : $\mathrm{I}\, N_1\, N_2 \mid N_3\, \mathrm{O} \rightarrow$

child : $\mathrm{I}\, N_1 N_2 N_1' N_2' N_3'\, \mathrm{O}$

parent 2 : $\mathrm{I}\, N_1'\, N_2'\, N_3' \mid N_4'\, \mathrm{O} \rightarrow$

For each pair of “parents” we produce two “children”, in the above way, which replace them in the population.

3.3.4 Mutation Processes

The mutation process was proved, by the experiments, to be a very significant operator that greatly influences the convergence of the algorithm. Thus we have implemented three different mutation operators, that are all used in the algorithm.

The first one that we simply call *neuron_mutation* operates as follows. Lets suppose that an individual of the population is selected to undergo this mutation operator. First a random number of neurons n_d from its hidden layer, is removed, and then an other random number of neurons n_a is inserted. These two numbers take values in a specified interval whose lower and upper bounds are passed as parameters to the algorithm. The centres of the new nodes and the weights of the new connections, are generated randomly in the same way as in the *initilization process*.

The second mutation operator, is called *uniform_weight_mutation* and works as follows. Let us suppose again that an individual is selected to undergo this mutation operator. For each centre and weight of this individual we do the following:

I. Generate a random float number r in the range [0,1]
II. If $r < p_{u\_alter}$ (the probability to select a weight for uniform mutation) then give this weight a totally new random value in the same way as in the *initilization process*

Thus, at the end of the *uniform_weight_mutation* process a portion of the centres weights of the selected individual, will have been changed.

The third mutation operator, called *non_ uniform_weight_mutation*, is responsible for the fine tuning capabilities of the system. Let suppose, again, that an individual is selected to undergo this mutation operator. For each centre and weight of this individual we do the following:

I. Generate a random float number r in the range [0,1]
II. If $r < p_{u\_alter}$ (the probability to select a weight for non-uniform mutation) give this weight a new value w′ such that:

$$\begin{aligned} w' &= w + \Delta(t, ub-w), \quad \text{if a random digit is 0,} \\ w' &= w - \Delta(t, w-lb), \quad \text{if a random digit is 1,} \end{aligned} \tag{17}$$

where w is the current centre or weight value, lb and ub are the lower and upper bounds of the centre or weight values respectively, t is the generation number, and Δ(t,y) is a function that returns a value in the range [0,y], such that the probability of Δ(t,y) being close to 0 increases as t increases. This property causes this operator to search the solution space uniformly initially (while t is small) and very locally at later stages. We have used the following function (Michalewicz, 1994):

$$\Delta(t,y) = y \cdot \left(1 - r^{\left(1-\frac{t}{T}\right)^{b}} \right), \tag{18}$$

where r is a random number from [0,1], T is the maximal generation number, and b is a system parameter determining the degree of non-uniformity.

These three mutation operators working altogether like this: For each individual in the population do as follows:

- Generate a random float number r in the interval [0,1],
- If $r<p_{nm}$ then select this individual to undergo neuron_mutation, else if $r<p_{nm}+p_{uwm}$ then select this individual to undergo uniform_weight_mutation, else if $r<p_{nm}+p_{uwm}+ p_{nuwm}$, then

select the individual for non_uniform_weight_mutation.

- If $r>p_{nm}+p_{uwm}+p_{nuwm}$, then no mutation is performed.

3.4 System Design

From the given time series $x=\{x(t): 1\leq t\leq N\}$ we obtain two sets: a training set $x_{train}=\{x(t): 1\leq t\leq T\}$, and a test set $x_{test}=\{x(t):T\leq t\leq N\}$, where N is the length of the data record. The next step is the pattern selection, for the training phase. The network is asked to predict the next value in the time sequence, thus we have one output. The number of inputs m is one of the investigation aspects and is detailed in the next section. The problem of the pattern selection strategy for neural network training which is of the types, random and deterministic, has been comparatively presented in (Cachin, 1994). Simulation results show that convergence time and learning accuracy can be improved, but only by strategies of the deterministic type. So only this strategy was used in our experiments.

In the analysis performed, aiming at obtaining returns in continuous time, the data sets, both exchange rates and interest rates, have been transformed to the first differences of the natural logarithms of each series. All networks have a tripple input, consisting of the exchange rate, the interest rate and a "shock" flag, whose value is set to 0 for a normal period nad 1 for a period under shock, as mentioned earlier. Several input schemes, as regards the number of inputs of the exchange and/or interest rates, were adopted, while the shock flag was fed into the networks constantly as a one and unique input. Each training input pattern consists of "past" data items, according to a window value: i.e. the window of length 5 is of the form:
$(y_1, y_2, y_3, y_4, y_5 \rightarrow y_6)$,
$(y_6, y_7, y_8, y_9, y_{10} \rightarrow y_{11})$,
...etc.

In the following sections only the networks performed best are produced.

4. Numerical Experiments

As it has been mentioned earlier in this paper, the data series used are the drachma rates against four major currencies, i.e. the U.S. Dollar (USD), the British Pound (BP), the Deutsche Mark (DM) and the French Frank (FF), as well as the Greek overnight interest rate. These consist of daily observations covering a period of 8 years, starting on the 1st of January 1990 and ending on the 31st of December 1997. The shock flag is constructed based on the overnight interest-rates reactions. Thus, when the interest rate rises to face an incoming shock the flag is set to 1, whereas during non-shock periods, when the interest-rate fluctuations do not exibit pronounced variations, the flag is set to 0, representing a regular, non-shock period.

4.1 Forecasting an Incoming Shock

Our first step involves forecasting the shock flag, thus trying to train the neural network to behave, to the best possible extent, as a shock recognizer. The architecture selected has been the Multi-Layer Perceptron (MLP) with a Backprobagation training algorithm, as described earlier. Our choice is justified by the significant capabilities of the MLP to recognize patterns of information better than other widely used architectures. The data used as inputs to the networks are the first differences of the logarithms of the exchange rates and the interest rate, as well as the binary shock flag. Each input is unique, in the sense that it is fed as a single sample. This task has been performed for each and every currency at a time, requiring the network to predict the next value of the shock flag.

The results are summarized in Table 1, which includes only the best (internal) archtictures from the point of view of both results and time convergence.

Table 1: ***Forecasting results of shock flag using an MLP neural network****

* input-hidden-output stands for number of input, hidden and output nodes, while epochs are the iterations during the training phase

Currency	Network (input-hidden-output)	Epochs	Training patterns	Testing patterns	Hits of training	Hits of testing
USD	3-5-1	100	1906	50	1898 (99,5%)	48 (96%)
BP	3-5-1	100	1904	50	1898 (99,5%)	48 (96%)
FF	3-8-1	100	1904	50	1898 (99,5%)	48 (96%)
DM	3-8-1	100	1905	50	1898 (99,5%)	48 (96%)

The results are indeed very promising. The networks converge to a forecasting level of 96% during the testing phase . USD and BP need no more than 5 hidden neurons, while DM and FF increase this number to 8. This difference is explained by the fact that exchange-rate fluctuations are more pronounced in the USD and BP series, requiring a simpler architecture to interpret their behaviour.

The accuracy of the results in Table 1 suggests that we have a very good approximation of an incoming shock and the networks can easily be further used to simulate the market reactions to a future run against the drachma.

4.2 Forecasting Exchange Rates and the Interest Rate

The next step of our analysis is to evaluate the forecasting performance of the MLP and RBF (with genetic algorithms) networks. The three-input scheme (exchange rates, interest rate, shock flag) is retained, while the output is the next value of either the exchange rate or the interest rate, depending on the series, the behaviour of which we need to study. Multiple number of inputs have been examined and the results presented are only those exhibiting the best performance. Similar work during the recent past (ie. Andreou et al, 1997; Adamopoulos et. al 1997) has shown a rather good prediction performance mostly for the DM and FF, while poorer predictions yielded for USD and BP. The main difference in our case compared to earlier attempts on this topic is that we have introduced in the analysis the main policy instrument, the overnight interest rate, in addition to the various exchange rates, our main objective being to see to what extent the presence of this interest rate improves our forecasting ability.

In order to evaluate the performance of each network we have calculated the Mean Square Error (MSE) and the Mean Absolute Error (MAE). These prediction evaluators are given as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(t_i - o_i\right)^2 \tag{19}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|t_i - o_i\right| \tag{20}$$

where t_i is the actual value, o_i is the output of the network and n is the number of forecasts.

I. MLP

Applying the MLP networks aims at predicting the future exchange rate and the interest rate one step ahead in different networks, focusing on a set of five-input past values for each input, a scheme that performed better than all schemes employed. The window length is kept constant at 1. The data used are the logarithmic sets for the exchange rates and the interest rate, scaled to the range of 0 to 1 for best performance of the network, and the binary shock flag.

Table 2 summarizes the results for the exchange rates forecasting, while figures 3 to 6 offer a graphical presentation of the performance of the networks.

Table 2. ***Forecasting exchange rates errors using an MLP neural network***[*]

[*] input-hidden-output stands for number of input, hidden and output nodes, while epochs are the iterations during the training phase

Currency	Network (input-hidden-output)	Epochs	Number of Training Patterns	Number of Testing Patterns	MSE	MAE
USD	15-10-1	100	1000	300	0.00130	0.0230
BP	15-10-1	100	1000	300	0.00005	0.0052
DM	15-10-1	100	1000	300	0.00110	0.0043
FF	15-10-1	100	1000	300	0.00004	0.0026

The results in Table 2 indicate a relatively mediocre performance of the networks, with the forecasting performance of the FF being the best. The low error figures must be carefully considered, since the data values themselves range within a very small band. Turning to the graphs, one can easily see that there is a substantial delay in the learning process. There is a time lag involved after which the networks start to learn the patterns and to a certain extent they seem to have some tracking of the series, even if the degree of precision this is followed is not high. Comparing, however, these results to those obtained in earlier work on the subject (see Andreou et. al., 1997), one may observe that the contribution of the overnight interest-rate series improves the prediction accuracy to a significant extent.

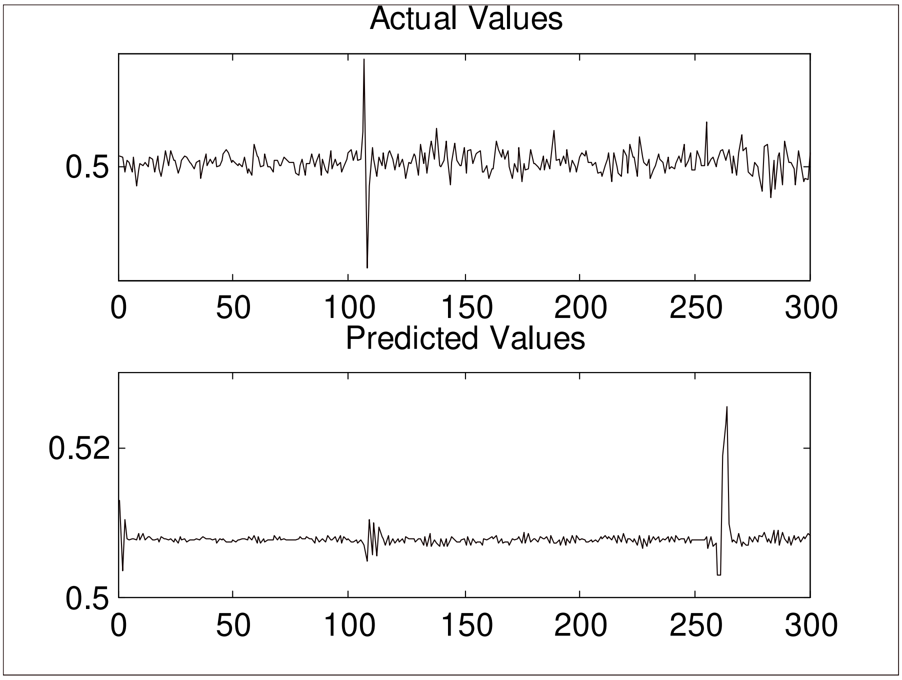

Figure 3: ***USD/GRD actual and predicted exchange rates series using an MLP neural network.***

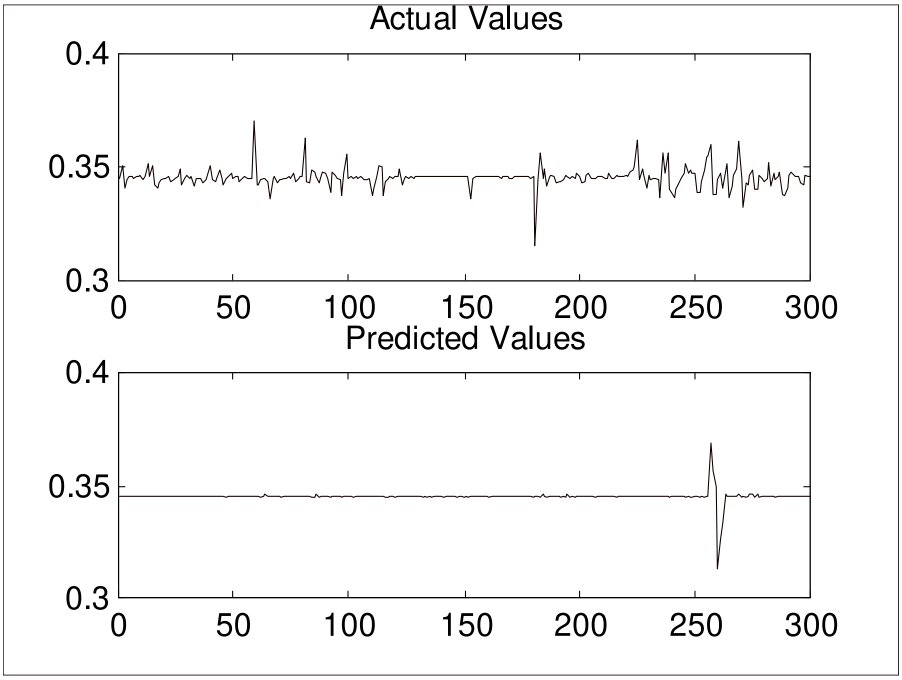

Figure 4: ***BP/GRD actual and predicted exchange rates series using an MLP neural network.***

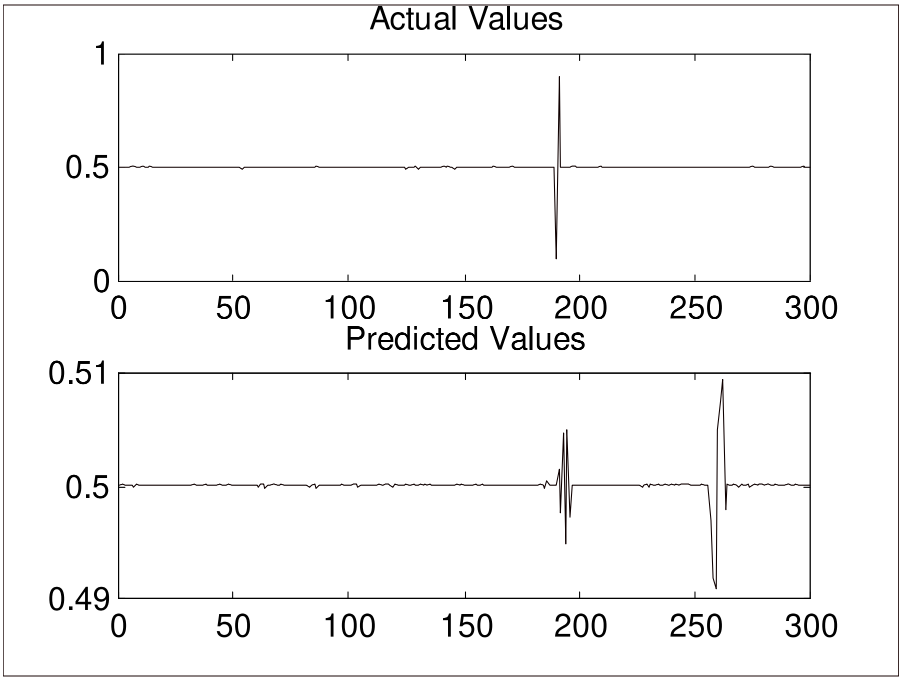

Figure 5: ***DM/GRD actual and predicted exchange rates series using an MLP neural network.***

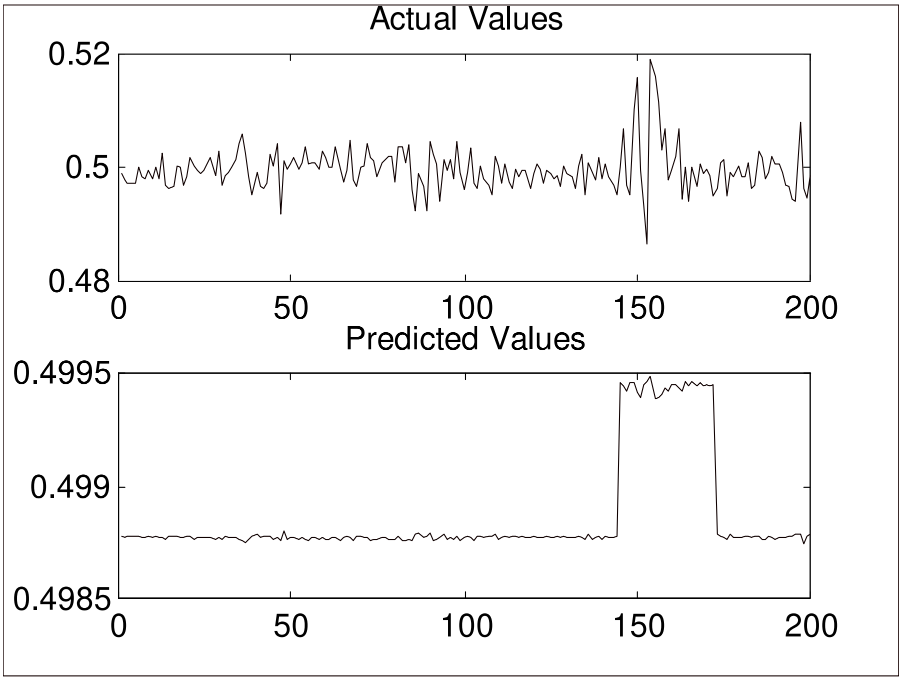

Figure 6: ***FF/GRD actual and predicted exchange rates series using an MLP neural network.***

Table 3 summarizes the results obtained in the case in which an overnight interest-rate forecasting is attempted on the basis of the fluctuations of the four currency exchange rates versus the Greek drachma, while Figures 7 to 10 depict the graphs of the networks performance.

Table 3. ***Forecasting interest rates errors using an MLP neural network***[*]

[*] input-hidden-output stands for number of input, hidden and output nodes, while epochs are the iterations during the training phase

Currency	**Network (input-hidden-output)**	**Epochs**	**Number of Training Patterns**	**Number of Testing Patterns**	**MSE**	**MAE**
USD	15-10-1	100	1000	300	0.000025	0.0023
BP	15-10-1	100	1000	300	0.000023	0.0024
DM	15-10-1	100	1000	300	0.000024	0.0025
FF	15-10-1	100	1000	300	0.000439	0.0134

As it was the case with Table 2, the results in Table 3 indicate a relatively mediocre performance of the networks, with the low error figures being the result of very weak fluctuations of the data involved. The FF performance, however, appears slightly worse compared to the rest of the drachma rates, among which there does not seem to be significant forecasting performance differences. Inspection of figures 7 through 10, however, shows that the forecasts are clearly worse than those indicated in figures 3 to 6 where the exchange-rate performance is designated. This result should, however, be regarded by no means as discouraging, since interest rates, being an extensively used policy instrument, are expected to be characterised by a great deal more noise in their time series compared to the exchange rates. The question of the time lag involved between the application of the interest rate policy and its impact, if any, on the exchange rates of the drachma is an open issue requiring extensive treatment before final conclusions may be drawn. There is still, therefore, a lot of room for improvement in the area of network architectures as well as input strategies before interest rate forecasting is improved.

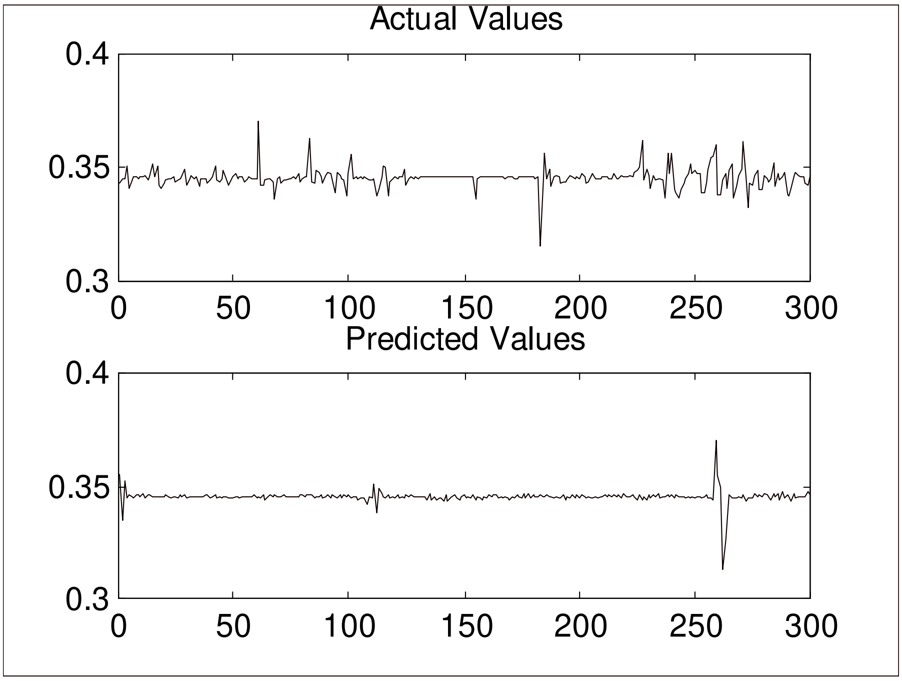

Figure 7: ***USD/GRD actual and predicted interest rates series using an MLP neural network.***

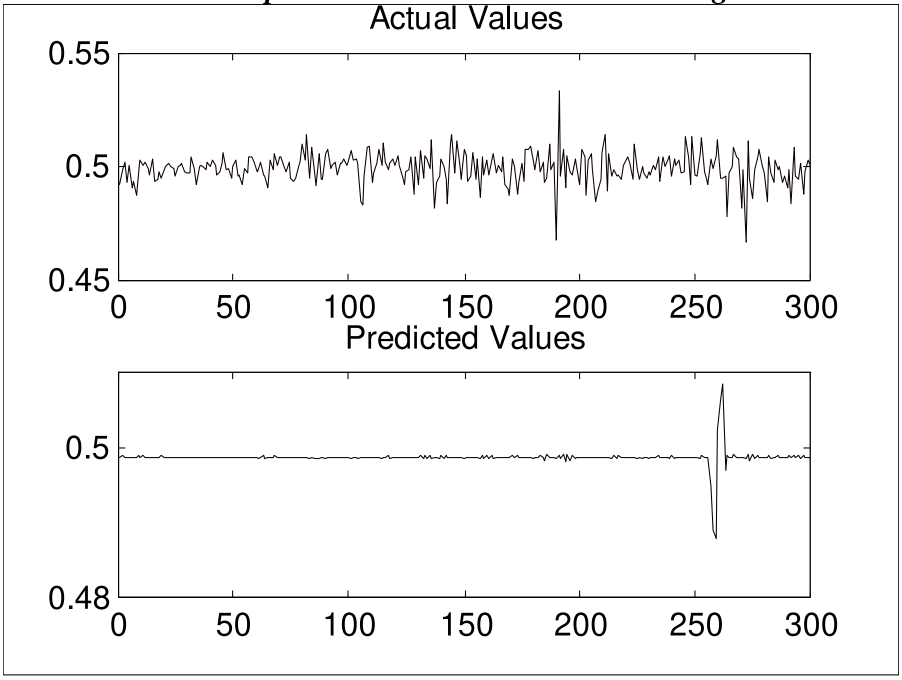

Figure 8: ***BP/GRD actual and predicted interest rates series using an MLP neural network.***

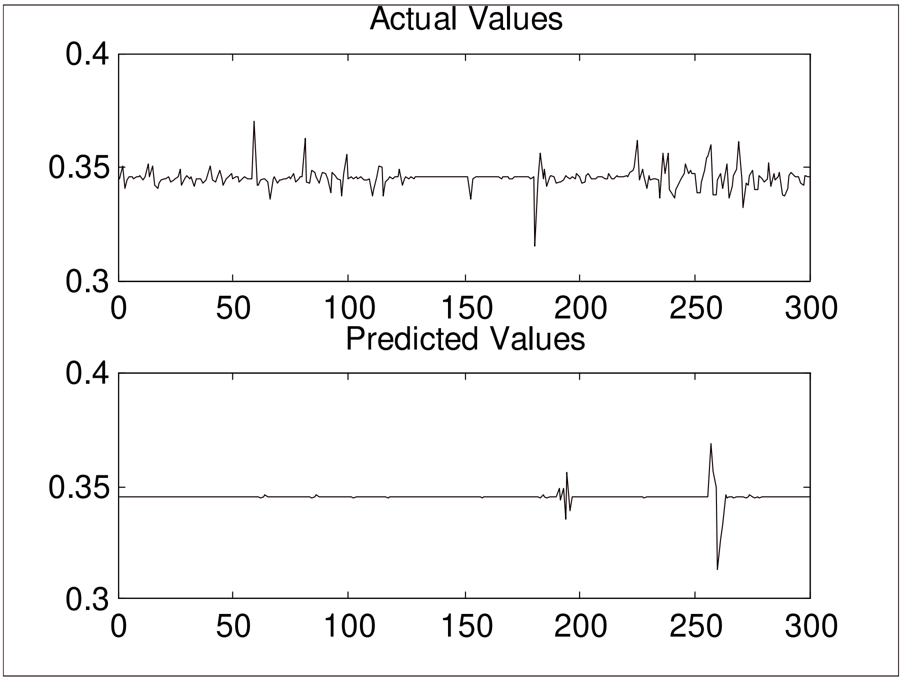

Figure 9: ***DM/GRD actual and predicted interest rates series using an MLP neural network.***

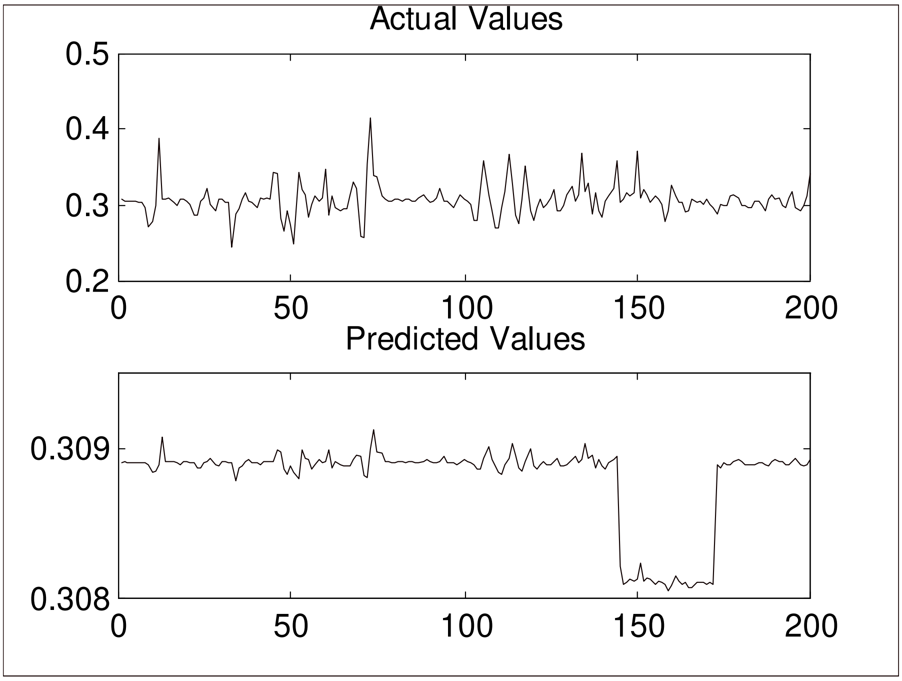

Figure 10: ***FF/GRD actual and predicted interest rates series using an MLP neural network.***

II. RBF with Genetic Algorithms

Based on the theoretical framework analysed earlier on in this paper, we have applied the RBF architecture, optimized by the genetic algorithm. Tables 4 and 5 present the most successful networks among those which have been used for predicting the exchange rates and the interest rate respectively. As in the MLP case, 15 inputs indicate 5 past values for each major input, exchange rate, interest rate and shock flag, while a window of length 1 was kept contant. The data used were the logarithmic sets for the exchange rates and the overnight interest rate, scaled to the range of -1 to 1 for best performance of the network, and the binary shock flag.

Table 4. ***Forecasting exchange rates errors using a RBF neural network optimized by genetic algorithms*** *

* input-hidden-output stands for number of input, hidden and output nodes

Currency	Network (input-hidden-output)	Population Size / Number of Generations	Number of Training Patterns	Number of Testing Patterns	MSE	MAE
USD	15-8-1	20 / 500	600	300	0.01310	0.0537
BP	15-8-1	20 / 500	600	300	0.00068	0.0192
DM	15-8-1	20 / 500	600	300	0.00002	0.0040
FF	15-8-1	20 / 500	600	300	0.00080	0.0220

Table 4 clearly shows a good behavior on forecasting ability. Notice that the data range along a broader band compared to the MLP case, in order to avoid direct comparison between the two methods. Figures 11 to 14 show that the networks exhibit a faster learning ability compared to that of the MLP with a higher forecasting precision. The forecasts follow the actual series easily, especially in the cases of the USD, FF and BP series, catching up with all peaks, with a lower intensity, however, with the exception of the FF series. The results obtained in our case are similar to those reported in Adamopoulos et. al. (1997), using, nevertheless, a smaller number of generations. Therefore we can conclude at this point that employing the interest rate as an additional input to this type of network, does not improve to a significant extent the performance of forecasts, compared to the one obtained using only the exchange rate values (see Adamopoulos et. al., 1997).

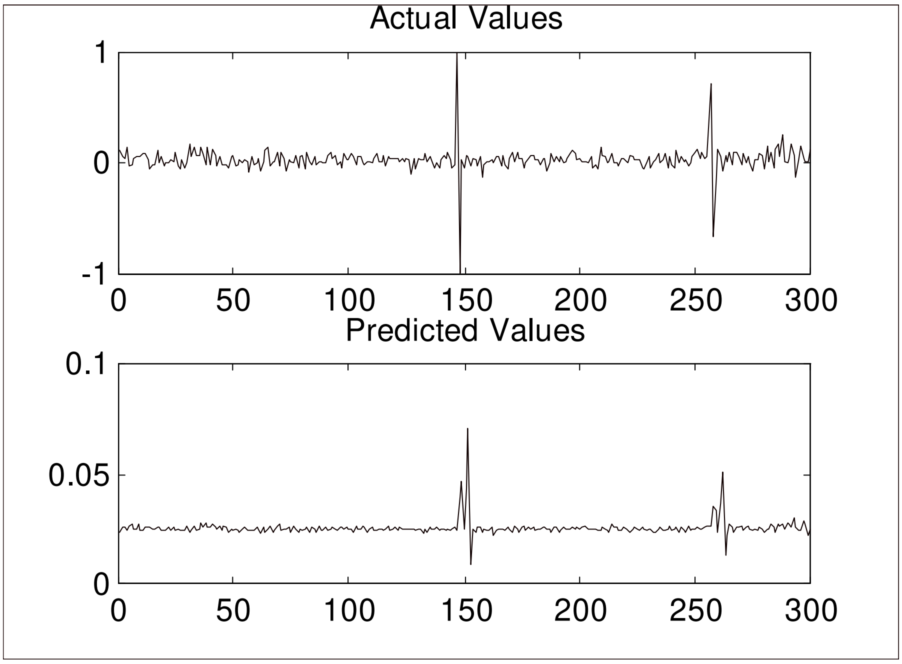

Figure 11: ***USD/GRD actual and predicted exchange rates series using a RBF neural network with genetic algorithms.***

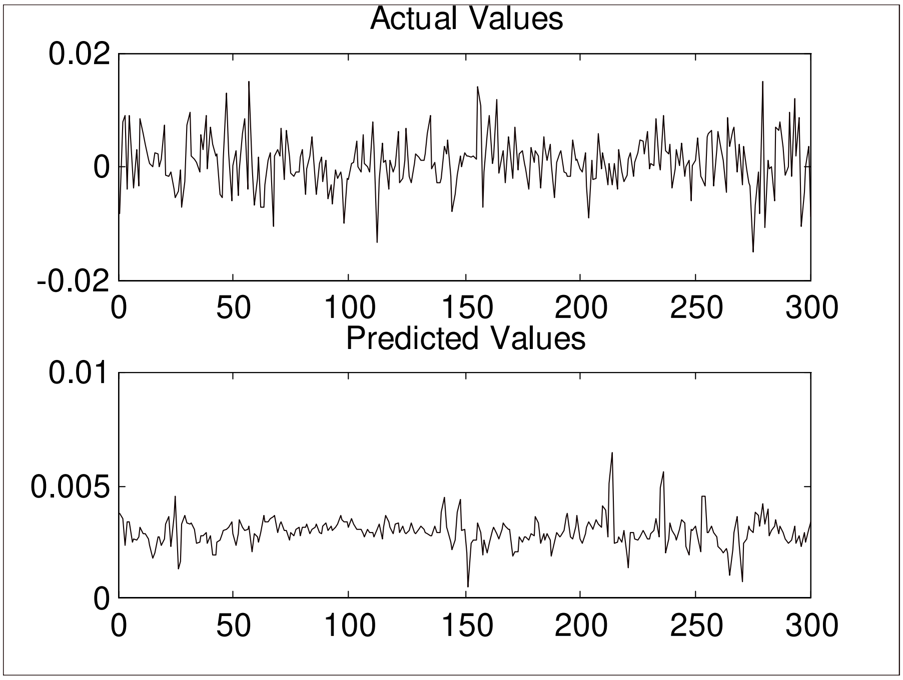

Figure 12: ***DM/GRD actual and predicted exchange rates series using a RBF neural network with genetic algorithms.***

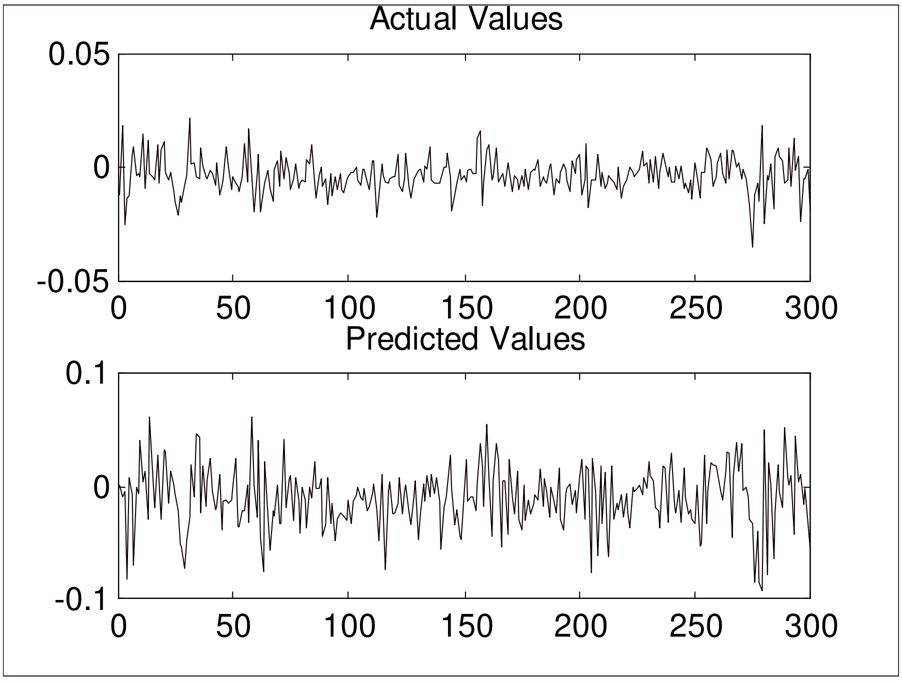

Figure 13: ***FF/GRD actual and predicted exchange rates series using a RBF neural network with genetic algorithms.***

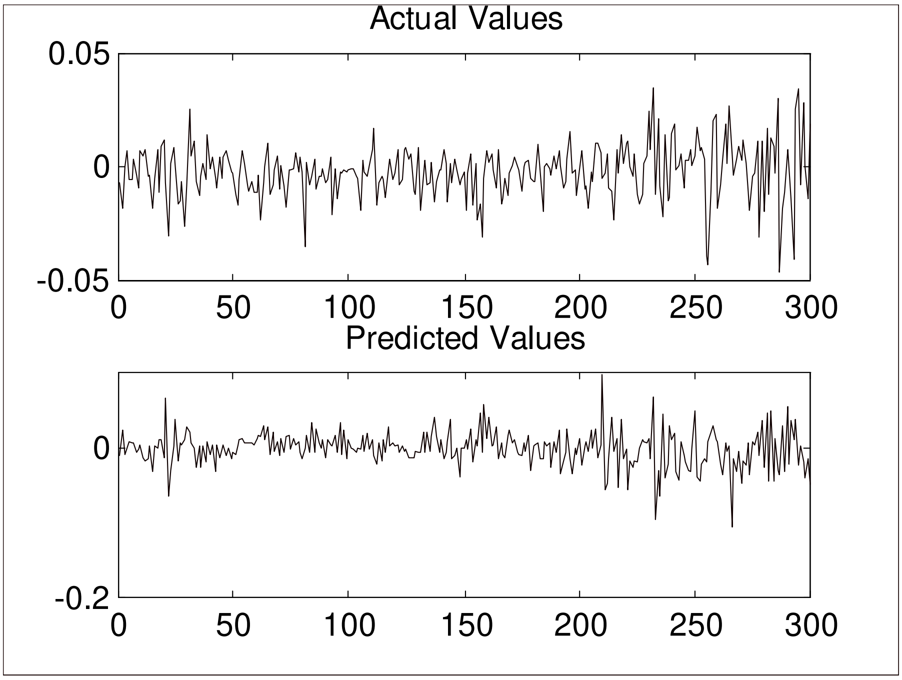

Figure 14: ***BP/GRD actual and predicted exchange rates series using a RBF neural network with genetic algorithms.***

Table 5 indicates a relatively good behavior of errors in forecasting ability for the interest rates. We should repeat however that the data used range through a broader band compared to the MLP case, in order to avoid direct comparison of errors between the two methods, as we did in table 4. Turning to the graphical presentations denoted in figures 15 to 18, these show that the networks exhibit a higher forecasting precision compared to that of the MLP, but forecasts follow the actual series in some cases only, catching up with various but not all the peaks. This result, despite its medoiocre success, constitutes an improvement in the network performance compared tthe MLP performance.

Table 5. ***Forecasting interest rates errors using a RBF neural network optimized by genetic algorithms***[*]

[*] input-hidden-output stands for number of input, hidden and output nodes

-

Currency	Network (input-hidden-output)	Population Size / Number of Generations	Number of Training Patterns	Number of Testing Patterns	MSE	MAE
USD	15-8-1	20 / 500	600	300	0.01350	0.0403
BP	15-8-1	20 / 500	600	300	0.01380	0.0906
DM	15-8-1	20 / 500	600	300	0.00066	0.0170
FF	15-8-1	20 / 500	600	300	0.00490	0.0559

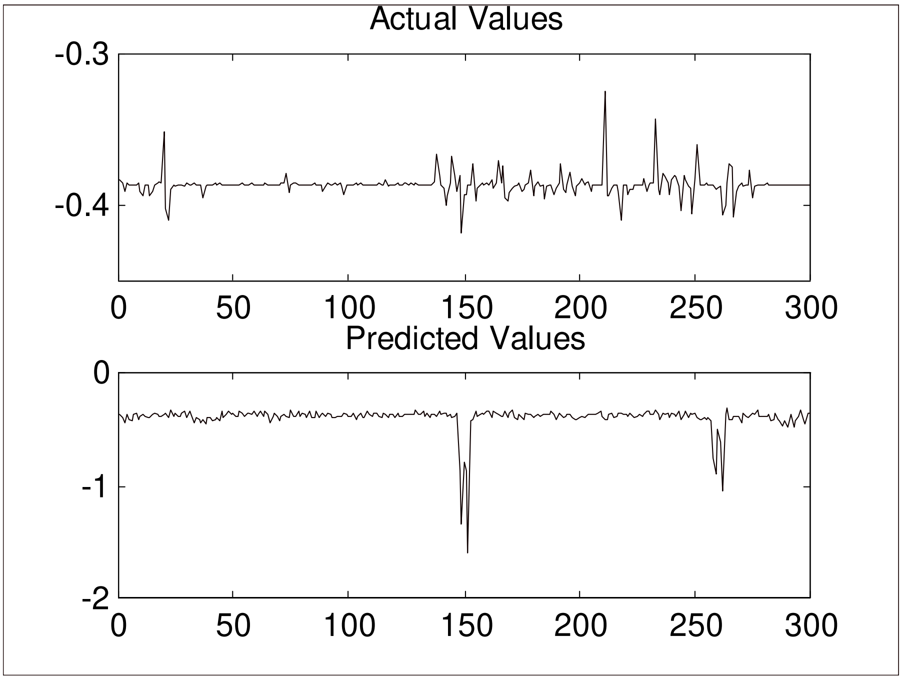

Figure 15: USD/*GRD actual and predicted interest rates series using a RBF neural network with genetic algorithms.*

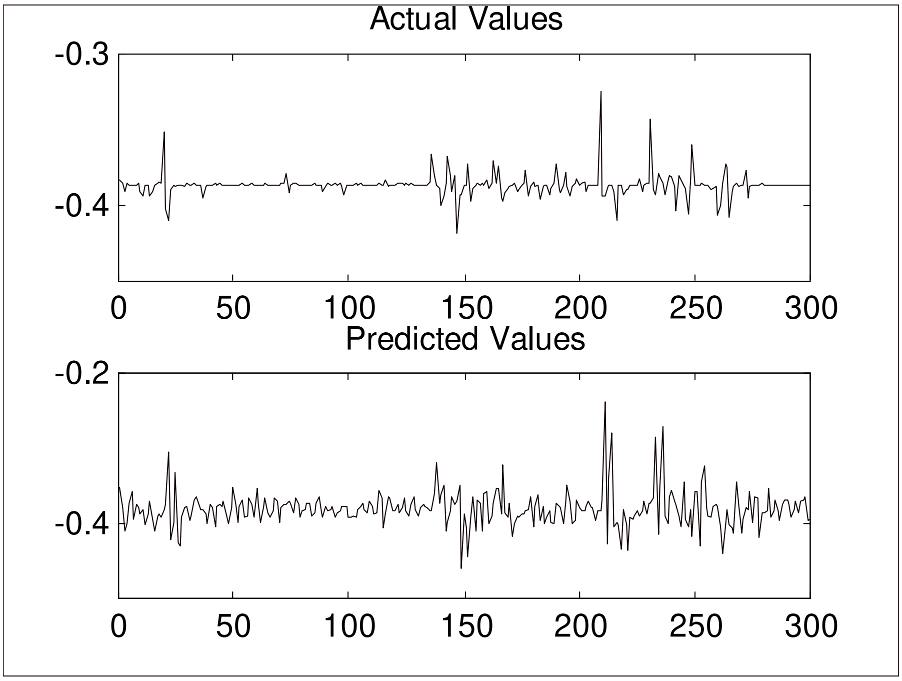

Figure 16: ***DM/GRD actual and predicted interest rates series using a RBF neural network with genetic algorithms.***

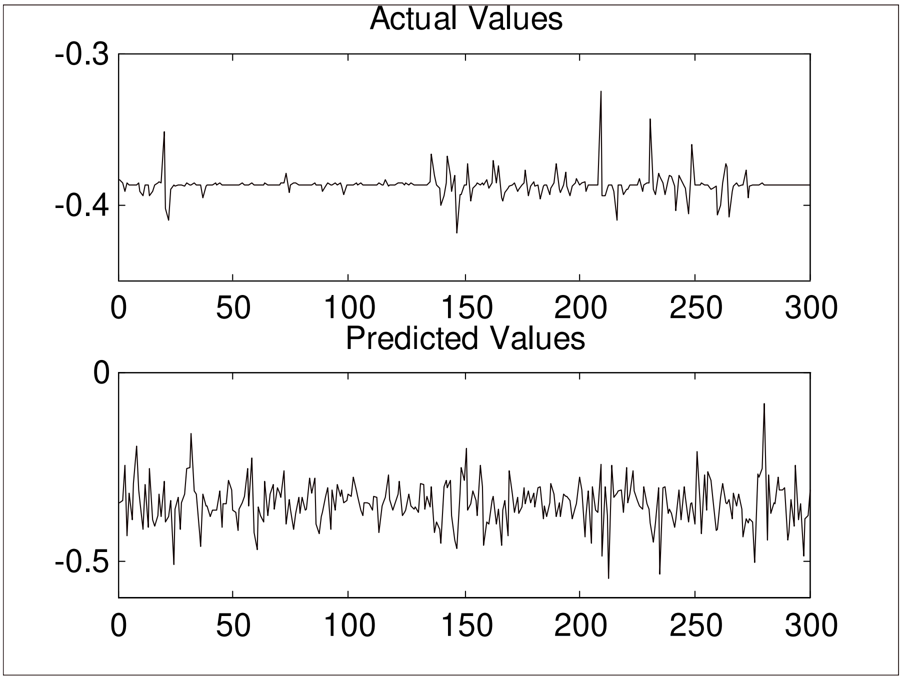

Figure 17: ***FF/GRD actual and predicted interest rates series using a RBF neural network with genetic algorithms.***

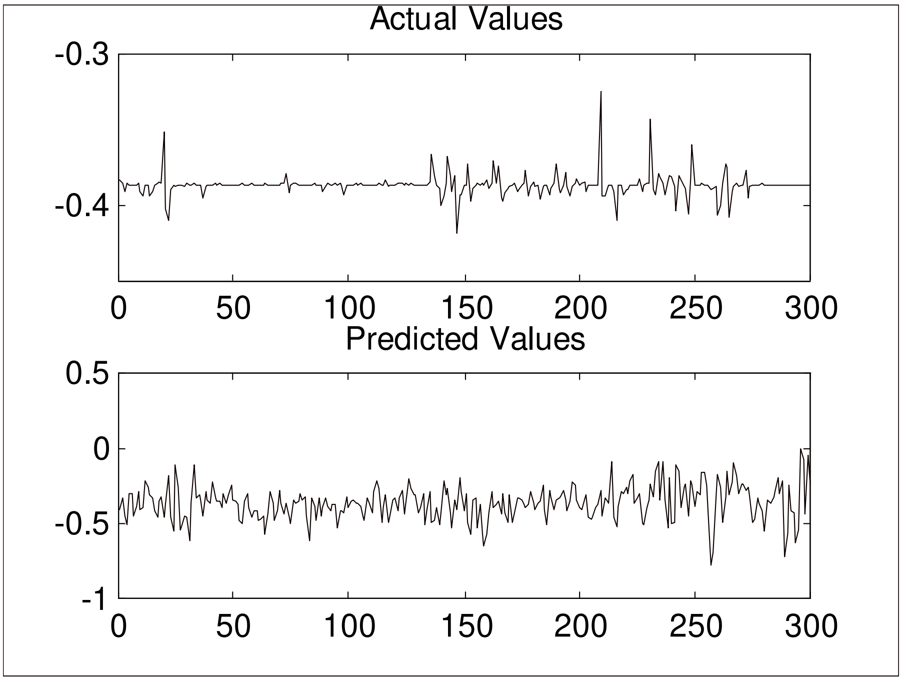

Figure 18: BP/*GRD actual and predicted interest rates series using a RBF neural network with genetic algorithms.*

5. Conclusions

In the present work we have studied the impact of exogenous disturbances on the behaviour of the drachma-exchange rate of certain major currencies. We have examined, furthermore, the impact of such "shocks" on the future course of the exchange-rate of the currencies involved. Finally, we have tested the predictability of the exchange rates and the overnight interest rate series, using both the well-known MLP algorithm and the genetically trained RBF network. recently introduced (Adamopoulos et. al., 1997).

The results obtained are rather encouraging concerning the predictive ability of both algorithms used and seem to be in line with previous research undertaken in this area (Andreou et al., 1997). With respect to the performance of the algorithms when the series involved are characterised by shocks, the MLP matches the pattern almost perfectly (96% success), since the network is particularly suitable for cases of pattern recognition. When it comes to predicting the exchange-rate and the interest-rate behaviour, the MLP requires a certain time lag after which tracking seems to be realised to a certain extent. This performance, however, can not be considered very successful as regards the degree of accuracy of "learning". In the case of the RBF with Genetics, the performance is improved since the network learns faster while its predictive ability is higher with respect to the MLP. The contribution of the interest rate, however, leaves a lot to be desired in both algorithms, as regards its degree of prediction and its introduction in the analysis does not seem to improve the performance of the network in comparison to earlier work on the topic (Andreou, et. al., 1997; Adamopoulos et al., 1997).

In terms of an overall assessment, the degree of prediction attained is satisfactory, following the peaks and troughs in the pattern of the series involved, displaying, however, tighter fluctuation bounds compared to the corresponding actual series. With reference to specific exchange rates, the dollar/drachma and the BP/drachma trends are less predictable than the DM/drachma and the FF/drachma ones. These results are to a large extent influenced by the ERM membership of the German and the French currencies, a fact which introduces a considerable degree of discipline to the fluctuations of these currencies in the international markets. As regards the failure of the overnight interest rates to add to the predictive ability of the network, it must be borne in mind that the Greek money market is not altogether free of distortions and the interest rates, are to a large extent administratively determined, despite the steps taken during the past few years to liberalise the market. The overnight interest rate, in particular, is used along with the foreign-exchange reserves, as a policy instrument to face speculative attacks against the drachma, and as such, its time series is bound to be noise-contaminated to a considerable extent. There may, however, be additional complications asociated with the use of the overnight interest rate as input in the analysis: This rate is used as a "shock-absorber" by the authorities which raise it once the pressure on the drachma exchange rates starts rising and, sometimes, even before such pressure is created, simply for precautionary reasons, as it happened in May 1994 with the abolition of all restrictions regarding capital movement. The question arising, therefore, is to what extent there is a lag or a lead involved between a crisis and the reactions of the authorities and what the duration of such a lag or lead may be. In cases in which lags are involved, the next issue to discuss refers to the sort of lag characterising the behaviour: One may concentrate, e.g. on a "recognition" lag, i.e. the lag involved between the origin of the shock and the time the authorities become aware of its existence. Or, maybe, the "administrative" lag will prove to be more important, i.e. the time period between the realisation of the shock and the effective reaction of the authorities. An "operational" lag, finally representing the time between the application of a policy measure and its impact on the exchange rate may be the appropriate one to concentrate on. It is obvious, therefore, that the issue of the time lag or, even, the time lead involved requires particular attention, since the faster the rise of the overnight interest rate the more efficient is it in the sence that it does not allow the shock to affect the drachma exchange rates. In that sence, one may be tempted to argue that the Greek authorities and more specifically the Central Bank have been able to interfere in the market both rapidly and efficiently, since the drachma exchange rates we have used exibit only weak fluctuations, even at the peak of each shock periods. It is only natural, therefore, that using these weak drachma exchange-rate variations in order to forecast the overnight interest rate should not yield impressive results. Our contribution might have been more substantial had we been allowed to use the foreign-exchange reserves pumped into the market in cases of crises, information which, unfortunatelly, is not available for the time being. An effort to gain access to this piece of information, along with the choice of the most appropriate time lag characterising the reactions of the authorities are issues on which we plan to place more emphasis during future reseach on the topic.

We feel, however, that the results presented in this paper are rather encouraging and the prospects of improving the predictive ability of the algorithms used will be enhanced if we expand our research along the lines just described. Further research on the topic would also require that we choose the pattern of certain types of shocks out of those described earlier in this paper in order to train our series to the shock which the domestic market is expected to suffer once the drachma becomes an ERM member. We have strong reasons to believe that training our series on the basis of our experience on the types of runs against the drachma suffered during the past few years is expected to contribute a great deal to predicting the intensity and duration of the attacks anticipated.

REFERENCES

- Bank of Greece, "Report of the Governor", Athens, 1984.
- Bank of Greece, "Report of the Governor", Athens, 1986.
- Bank of Greece , "Report of the Governor", Athens, 1991.
- Bank of Greece , "Report of the Governor", Athens, 1994.
- Bank of Greece , "Annual Monetary Program Announcement", Athens, 1997.
- Billings, S. A. and Zheng, G. L., "Radial basis function network configuration using genetic algorithms", *Neural Networks, vol. 8, no 6*, pp. 877-890, 1995.
- Brissimis S. N. and J. A.Leventakis, "The Effectiveness of Devaluation: A General Equilibrium Assessment with Reference to Greece", *Journal of Policy Modeling*, 11(2), pp.247-271, 1989.
- Cachin, C., "Pedagogical Pattern Selection Strategies", *Neural Network, Vol. 7*, No. 1, pp. 175-181, 1994.
- Karadeloglou P., "On the Existence of an Inverse J-curve", *Greek Economic Review*, *V.12,2, p.p.* 285-305, 1990.
- Karadeloglou P., C. Papazoglou, G. A. Zombanakis, "Is the Exchange Rate An Effective Anti-Inflationary Policy Instrument?", *Economia*, 1998 (Forthcoming).
- Likothanassis, S. D., Georgopoulos, E. and Fotakis, D., Optimizing the Structure of Neural Networks using Evolution Techniques, 5th Int. Conf. on HPC in Engineering, Spain, 4-7 July, 1997.
- Michalewicz, Z., Genetic Algorithms + Data Structures = Evolution Programs, Second Ext. Edition , Springer Verlag, 1994.
- Rumelhart, D.E. and McLelland, J.,"Parallel Distributed Processing", *Cambridge, MA, MIT Press*, 1986.
- Zombanakis G. A., "Is The Greek Exporters' Price Policy Asymmetric?", *Greek Economic Review* (Forthcoming), 1998.