STATE FORECASTING BASED ON ARTIFICIAL NEURAL NETWORKS

A.P. Alves da Silva¹ A.M. Leite da Silva¹ J.C.S. de Souza²,³ M.B. Do Coutto Filho²,³

¹Electrical Eng. Dept. - Catholic University of Rio de Janeiro - Rio de Janeiro - Brazil
²Electrical Eng. Dept. - Fluminense Federal University - Rio de Janeiro - Brazil
³Electrical & Computer Eng. Dept. - Northeastern University - Massachusetts - USA

ABSTRACT

State forecasting is a powerful tool to enhance the noise suppression ability of the state estimation process of a power system. Artificial neural networks can perceive complex nonlinear interactions among variables that improve the predictions accuracy and robustness. This paper investigates the applicability of artificial neural networks to state forecasting. Nonlinear autoregressive neural models are proposed and their performance are evaluated for an entire day using the IEEE-24 bus system data.

Keywords: Power System State Estimation, Forecasting-Aided State Estimation, Dynamic State Estimation, State Forecasting, Real-Time Monitoring, Artificial Neural Networks.

1. INTRODUCTION

Power System State Estimation (PSSE) is an intrinsic element of an energy management system (EMS) [1]. The role of a PSSE is to supply the EMS with a reliable real-time database to be used by the set of advanced functions for system security monitoring and control. Although PSSE has reached the category of mature research area, covered by a huge number of publications [2], challenges still persist [1, 3-5]:

- identification of bad critical measurements and multiple interacting bad data (including leverage points);
- partial "blindness" for topological error detection;
- difficulty in distinguishing among different types of error (analogical, topological and parametrical); and
- selection of pseudo-measurement values.

Solutions for these problems have been proposed based on pre-filtering schemes [4-11], instead of the conventional post-filtering procedures [12, 13]. These conventional procedures are used by the static state estimation (SSE) and, in essence, the measurement estimation residuals are analyzed through statistical methods of hypothesis testing. Unfortunately, these tests can not be considered fully reliable to overcome the problems mentioned above. Conversely, forecasting-aided state estimation (FASE, also known as dynamic state estimation) and pattern analysis (PA) can better deal with those problems since some extra information are used in a pre-filtering phase: e.g. state forecasting. Besides the benefits of state forecasting for security and economic purposes, pre-filtering schemes via FASE complement the post-filtering analysis.

The basic idea of FASE is to use the differences between the latest acquired measurement and the corresponding forecasted values (innovation vector) to detect, diagnose and eliminate inconsistent data. Therefore, the accuracy of the FASE prediction step is crucial for improving its data debugging capability. However, the availability of efficient prediction models has been the major shortcoming of the FASE approach so far. An approach for system modelling allowing adaptive model identification and verification in real-time is necessary. Artificial neural networks (ANN) have been successfully employed for system identification and forecasting [14]. ANN can perceive complex nonlinear interactions among variables that improve the accuracy and robustness of the prediction.

In this paper, a new dynamic model for the time behaviour of the system state using ANN is proposed. Real-time applicability, forecast accuracy, robustness and adaptability are among the topics considered in this work.

2. MATHEMATICAL MODELS

The time evolution of the operating state of a power system is basically determined by the continuous variation of the system loads. In daily operation, the loads vary according to cyclic patterns showing small, random fluctuations. Sudden changes in the system operating point seldom occur and are due to a predictable event, e.g. disconnection of a large industrial consumer, or as the result of some abnormal operating conditions, e.g. outages. System variables like power generation, line flows, voltages, etc. are adjusted to follow the variations, within the operating and network constraints.

Hence, under normal operating conditions, a power system exhibits slow dynamics, with transient oscillations of small amplitude compared with the overall change observed in a short time period (minutes). Therefore, it is reasonable to assume a hypothetical mode of operation (quasistatic) in which the load changes are met instantaneously by generation and other system variables. Considering that the system is operating under quasistatic conditions, its state of operation is perfectly characterized by the set of all complex nodal voltages (magnitudes and phase angles).

For a given network configuration, the unique way of assessing the system state is through the measurements gathered from around the system. For instance, at time sample $k$, the collected measurements are filtered using a state estimation process which then creates a historical series of filtered system states. This series is used to predict the system state of the next time sample $k+1$. Based on this sequence of prediction and filtering, the following dynamic system can be established:

$$x_{k+1} = \phi(x_k, x_{k-1}, \ldots) + w_k$$

(1)

$$z_k = h(x_k) + v_k$$

(2)

where $x$ and $z$ are the state and measurement vectors with dimensions $(nx1)$ and $(mx1)$, respectively; $\phi$ and $h$ are functions describing the state transition process and the measurement equations, respectively; $w$ and $v$ are white Gaussian noise vectors with zero mean and covariance matrices $Q$ and $R$, respectively. The components of the state vector are the phases and magnitudes of the nodal voltages, whereas active and reactive power flows and injections and voltage magnitudes form the measurement vector.

The FASE involves three basic steps: model parameter estimation, state forecasting and state filtering. They are discussed as follows.

2.1 Parameter Estimation and ANN Architectures

Parameter estimation and functional estimation are the two main groups (not mutually exclusive) in the system identification techniques. The parameter estimation approach assumes a known system model with unknown parameter. On the other hand, the functional approach estimates both the system model and its parameters.

Nowadays, there is a strong trend away from the restrictive assumptions of the parameter estimation approach. However, purely functional estimation [15,16] can be very time consuming, therefore, inadequate for real-time requirements. A good compromise between a pure parameter estimation and the functional approach is the Optimal Estimate Training 2 (OET2) algorithm [17,18], where only the order of the system model complexity is known (number of hidden neurons). The OET2 is a least-squares based algorithm suitable for on-line training.
Feed-forward ANN architectures, are employed to solve the state forecasting problem. Figure 1(a) represents a nonlinear model with the input layer formed by data conduits, a hidden layer with neurons in which \( f(z) = \) hyperbolic tangent of \( z \) in Fig. 1(b), and the output unit operating as an adder (\( f(z) = z \) in Fig. 1(b)). It has been shown that this type of architecture can provide an accurate approximation to any continuous function, provided that the number of hidden neurons is large enough [19]. The reason for adopting a straight line as the activation function, \( f(z) \), of the output neuron is that there is no a priori justification for not treating equally the output (forecast) range. More details on the estimation of the state transition function \( \phi \) will be discussed in the next section.

2.2 State Forecasting

The form of the function represented by a feed-forward ANN is determined by its architecture. It corresponds precisely to a specific family of regression curves. The input channels are specified as an autoregressive process. Therefore, consider the series of "r" values of the 4th component of the state vector, i.e.: \( x_k^0, x_k^1, \ldots, x_{k+r}^0 \). At time sample \( k \), the following state transition equation can be written for this particular component, where superscripts \( l \) are omitted to simplify the notation:

\[
x_{k+l} = \sum_{j=0}^{r} \beta_j f\left( \gamma_0 + \sum_{i=0}^{r} \gamma_i x_{k+l+i} \right) + w_k
\]  

(3)

where \( \gamma_0 \) denotes the interconnection weight from the input channel \( i \) to the hidden neuron \( j \) (\( i=0 \) is associated with the input always equal to 1), \( \beta_j \) denotes the interconnection weight from the hidden neuron \( j \) to the output neuron, and the function \( f(\cdot) \) is the hyperbolic tangent. Equation 3 represents the nonlinear prediction model associated with Fig. 1(a).

The ANN's employed in this work have only one output neuron since each ANN is responsible for predicting only one state variable at each time. The number of input channels of the ANN is "r+1".

The parameters (interconnections weights \( \gamma_0 \) and \( \beta_j \)) of the nonlinear autoregressive model described by eqn. 3 are determined by the OET2 algorithm, which is briefly described next. The following notation is used:

Output layer

\[
\begin{bmatrix}
X_{k+1} \\
X_k \\
X_{k-1}
\end{bmatrix}
\]

Hidden layer

\[
\begin{bmatrix}
f(z)
\end{bmatrix}
\]

Input layer

\[
\begin{bmatrix}
x_k \\
x_{k-1}
\end{bmatrix}
\]

Figure 1: (a) Fully connected feed-forward net with one hidden layer; (b) Representation of a neuron in the hidden and output layers.

\[X = \begin{bmatrix}
x_{k-1} & x_{k-2} & \ldots & x_{k-r} \\
x_{k-2} & x_{k-3} & \ldots & x_{k-r-1} \\
x_{k-3} & x_{k-4} & \ldots & x_{k-r-2} \\
\end{bmatrix}
\]

\[y = [x_k, x_{k-1}, x_{k-2}, \ldots, x_{k-r+1}]^T
\]

\[\beta = \text{vector } (r+1) \times 1 \text{ of the interconnection weights } \beta_i \text{ linking the hidden layer with the output layer}
\]

\[\Gamma = \text{matrix } (r+1) \times (r+1) \text{ of the interconnection weights } \gamma_{ij} \text{ linking the input layer with the hidden layer}
\]

\[D^{\text{hid}} = \text{matrix } s \times (r+1) \text{ of signals at the output of the neuron in the hidden layer during the backward phase of the OET2}
\]

\[C^{\text{hid}} = \text{matrix } s \times (r+1) \text{ of signals at the input of the neurons in the hidden layer during the forward phase of the OET2}
\]

\[D^{\text{hid}} = \text{matrix } s \times (r+1) \text{ meaning the propagation of } D^{\text{hid}} \text{ back through the hidden layer}
\]

\[C^{\text{hid}} = \text{matrix } s \times (r+1) \text{ meaning the propagation of } C^{\text{hid}} \text{ forward through the hidden layer}
\]

\[\text{col}_i(\cdot) = \text{ith column of a matrix}
\]

The OET2 algorithm has two phases: backward and forward. All minimizations in both phases are carried out in the least-squares sense. The OET2 steps are as follows:

**Backward Phase:**

1) Determine an input matrix \( X_{k+r+1} \), formed by \( (r' + 1) \) linearly independent input channels of the original matrix \( X_{k+r+1} \), using a stabilization method [20]. The linear dependent input channels represent redundant information that can be discarded.

2) Calculate an initial \( \beta_0 \), such that \( X\beta_0 = y \), assuming \( s \geq (r' + 1) \):

\[
\beta_0 = \min_{\beta} \| X \beta - y \|_2
\]

(4)

where \( \| \cdot \|_2 \) denotes the Euclidean norm. Note that \( \beta_0 \) is the optimum parameter for the associated autoregressive linear model.

3) Given \( \beta_0 \), obtain \( D^{\text{hid}} \) that produces \( y \), i.e. \( (\beta_0^T(D^{\text{hid}})^T)y = y'. \) For \( t = 1,2,\ldots,s: \)

\[
\min \| \text{col}(D^{\text{hid}})^T \|_2 \text{ s.t. } \beta_0^T \text{col}(D^{\text{hid}})^T = y_t
\]

(5)

4) Normalize the elements of \( D^{\text{hid}} \) to assure that they will be located within the output range of the hidden layer's activation functions (hyperbolic tangents),

\[
D^{\text{hid}} = \frac{0.99}{\lambda} D^{\text{hid}}
\]

(6)

where \( \lambda \) is the element of \( D^{\text{hid}} \) with the largest magnitude.

5) Calculate \( C^{\text{hid}} \) applying the inverse hyperbolic tangent, i.e.

\[
C^{\text{hid}} = \tanh^{-1}(D^{\text{hid}})
\]

(7)
6) Given $X$ and $C^{\text{hid}}$, calculate $\Gamma$, such that $X\Gamma = C^{\text{hid}}$. For $t = 1, \ldots, r+1$ and $i = 1, \ldots, r+1$,

$$\text{col}(\hat{\Gamma}) = \min_{\text{col}(\Gamma)} \| X \text{col}(\Gamma) - \text{col}(C^{\text{hid}}) \|_2,$$

... s.t. $|\gamma_i| \leq \text{Bound1}$ (8)

**Forward Phase:**

7) Obtain the actual signals at the output of the neurons in the hidden layer,

$$C^{\text{hid}} = X \hat{\Gamma}$$ (9)

$$D^{\text{hid}} = \tanh(C^{\text{hid}})$$ (10)

8) Recalculate $\beta$; this step is necessary in order to take into account the normalization performed in step 4. Therefore, $\beta$ is such that $D^{\text{hid}}\beta = y$. For $j = 1, \ldots, r+1$,

$$\beta = \min_{\beta} \| D^{\text{hid}} \beta - y \|_2, \text{ s.t. } |\beta_j| \leq \text{Bound2}$$ (11)

A heuristic procedure to find a good pair of bounds along with an algorithm to solve eqns. 8 and 11 is described in [18]. The ANN architecture is pre-defined by the number of input channels when using the OET2, since the number of hidden neurons has to be the same. Therefore, bounds on the interconnection weight absolute values are employed to control the extent of nonlinearity provided by the function that the ANN represents.

Numerically stable methods for solving a system of linear equations in the least-squares sense are described in [20]. This reference also describes efficient algorithms for triangularizing $X$ at some tolerable error. The triangularization process is performed in step 4. Therefore, bounds on the interconnection weight absolute values are employed to control the extent of nonlinearity provided by the function that the ANN represents.

**2.3 Automatic Model Selection**

Model adaptation shall be performed on-line to obtain efficient prediction models. A poorly selected model can generate wide confidence regions with no debugging capability. As mentioned before, the self-organization of the model can be very time consuming. To reduce computational costs without sacrificing generalization (interpolation or extrapolation ability), the ANN architecture has to be pre-specified using as much knowledge about the problem as possible. Unnecessary complex models must be avoided to not overfit the training patterns.

The selection of a prediction model is dependent on the following aspects:

- state estimator execution cycle;
- load curve shapes; and
- number of training instances (chosen).

The first aspect determines the time horizon of the predictions. A model for very short term forecast should be employed when the state estimator is executed every five to ten minutes. This case favors the selection of simple models since the system loads, and consequently the state variables, usually do not fluctuate much during this interval. On the other hand, more complex models are expected to generalize better if the state estimator cycle is greater than ten minutes. Regarding the second aspect, nonstationarity in level and in slope favors the selection of more complex models, such as the nonlinear ones. Finally, the selected number of training samples define the part of a state variable series that is being modeled. Therefore, more complex models are supposed to generalize better as the number of training instances increase.

The probability of performing interpolation instead of extrapolation raises as the number of training instances increase. At least in principle this is desirable since, in general, interpolation is more reliable than extrapolation. However, more training patterns also means more computational effort and less adaptability. This trade-off has to be balanced keeping in mind the practicality of the adaptive on-line training scheme described in Section 2.2. It is clear that the three aspects affecting the selection of a prediction model are interrelated. This makes the development of an automatic model selection procedure system dependent.

**2.4 State Filtering**

The preceding steps have shown how the state vector can be predicted at time $k+1$ based on past information obtained until time $k$. Now suppose that a new set of measurements $x_{k+1}$ is available. The predicted state vector $\hat{x}_{k+1}$ can be filtered and a new estimate $\hat{x}_{k+1}$ is then obtained. Therefore, consider the usual weighted least squares objective function for the filtering process at time instant $k+1$:

$$J(x) = [h(x)-z]^T R^{-1} [h(x)-z]$$ (15)

To simplify the notation, the time index $k+1$ has been omitted from all variables in eqn. 15. Using Taylor's series expansion to linearize $J(x)$ about a nominal set of state variables $x^*$ yields:

$$J(x) = [H(x^*)\Delta x - \Delta z]^T R^{-1} [H(x^*)\Delta x - \Delta z]$$ (16)

Now, the set of equations 13 is easily rewritten as eqn. 1, with $x_{k+1} = [x_{k+1}^{(1)}, x_{k+1}^{(2)}, \ldots, x_{k+1}^{(n)}]^T$, $x_k = [x_k^{(1)}, x_k^{(2)}, \ldots, x_k^{(n)}]^T$, and so on. The forecasted state vector $\hat{x}_{k+1}$ can be estimated as:

$$\hat{x}_{k+1} = \hat{\Phi} (x_{k+1}, \hat{x}_{k}, \ldots, \hat{x}_{k-r})$$ (14)
Equation 17 is in the form $\|Ax-b\|^2_2$. To be minimized, it can be iteratively solved using Gauss’ normal equation or applying an orthogonal transformation method; assuming that the Jacobian matrix $H$ is of full rank, i.e. an observable system. Since the forecasted state vector, $\hat{x}_{k+1}$, is available, it can be used as a suitable linearization or starting point, $\tilde{x}_{k+1}$, for the iterative filtering process, reducing the number of iterations and computer time required for convergence. It is expected that a single iteration will lead to sufficiently accurate results in practice.

3. DATA VALIDATION

PSSE can be considered a process to clean up the erroneous data. The result of the estimation process provides the real-time database to be used by the set of advanced functions for system security monitoring and control.

Conventional schemes for data validation are usually built through statistical analysis of measurement estimation residuals, $r_{k+1}$, defined as being the differences between telemetered and estimated quantities, i.e.

$$ r_{k+1} = z_{k+1} - \hat{z}_{k+1} \quad \text{with} \quad \hat{z}_{k+1} = h_{k+1}(\hat{x}_{k+1}) \quad (18) $$

These schemes have worked well under favourable estimation conditions, i.e. when there are only single or multiple-non-interacting bad data; when the network configuration is correct; when the parameter values are reasonable and when the acquired measurements are sufficient to provide system observability. However, if it does not happen, difficulties can be faced: carrying out block elimination of correlated bad data; providing adequate values for missed measurements or for those eliminated as bad data.

If the estimator has forecasting capability, data consistency can be verified before the filtering step by performing simple statistical tests with innovations, $\tilde{r}_{k+1}$, defined as being the differences between telemetered and predicted quantities, i.e.

$$ \tilde{r}_{k+1} = z_{k+1} - \tilde{z}_{k+1} \quad \text{with} \quad \tilde{z}_{k+1} = h_{k+1}(\tilde{x}_{k+1}) \quad (19) $$

The conjunction of the a priori (innovations) and a posteriori (residuals) data validation schemes has allowed the establishment of a framework for the diagnosis of different anomalies which had not been treated in an integrated way by the conventional data validation schemes [8,9]. In addition, the incorporation of decision rules to help the error diagnosis should be considered [5]. The practical consequences of the previous considerations are:

- elimination of bad data smearing;
- block identification of bad data;
- adequate bad data replacement;
- detection of bad critical measurements;
- improved discrimination between bad data and network configuration errors;
- identification of system sudden changes;
- rapid detection of difficult conditions to PSSE.

More recently [10,11], a new framework for solving data acquisition and processing problems in power systems has been proposed. Two different pattern analysis techniques have been developed to deal with very noisy environments: an ANN based topology classifier and a probabilistic associative memory for data estimation and debugging. These tools extend the concept of local redundancy, allowing efficient data processing under difficult conditions.

- sustained unobservability situations where the prediction time horizons will not be in accordance to the assumed modelling;
- estimation of erroneously measured and/or non-telemetered voltage (or phase shifter) transformer tap position;
- simultaneous occurrence of different types of error (analogical, topological and parametrical); and
- system sudden changes (when forecasting models become inoperative for a while).

4. SIMULATION RESULTS

The nonlinear models presented in the preceding sections were tested using simulated data from the IEEE 24-bus Reliability Test System (RTS-24) described in Reference 21 and shown in Figure 2.

4.1 Description of Simulation

The simulation study is typical made over a period of 288 time-sampled intervals, representing a winter day (Tuesday) in the RTS-24; a 5 minute state estimation cycle time is assumed. The dynamic behaviour of the state vector is determined from the system loads. The load curve at each bus is composed from a constant proportional relation to the system load plus a random fluctuation (jitter). A constant power factor is assumed (98%), so that the reactive power follows the real power. The jitter is represented by a normally distributed random number with zero mean and standard deviation of 0.5% of the load participation factor component value in each bus.

An active generation dispatch policy based on merit order rule is adopted; active generation is allocated to the units with the smallest operating costs according to [21]. The controlled voltages are held constant. Test sets of measurements are provided by a measurement simulation program in the following way. Consistent sets of voltages are taken from load flow outputs. Using the product of bound defined by the metering characteristics (eqn. 20) and a randomly generated number (normally distributed), a measurement is simulated by adding this product to the true value. A total number of 84 measurements are considered.

$$ 3\sigma = ACC \times |z| + FS \quad (20) $$

where $|z|$ denotes the absolute value of the true measurement at any given time instant $k$. Parameters ACC (accuracy) and FS (full scale) have the following values depending on the meter type: flows $\rightarrow$ ACC = 0.02 and FS = 0.035; injections $\rightarrow$ ACC = 0.01 and FS = 0.035; and voltage magnitudes $\rightarrow$ ACC = 0.01 and FS = 0.

Figure 2: IEEE RTS-24 (measurement configuration)
4.2 Results

Figure 3 shows the total system load in percent (%) where the daily peak corresponds to 2850 MW. It shows different trends and transitions that can be separated into segments or sections. One possibility is to use pre-defined forecasting models (i.e., number of inputs channels and training patterns) based on the specific characteristics of each section of the load curve. Another possibility is to find a robust model which can match the data complexity for any segment of the load curve. The latter avoids the extra effort of selecting different models. The following tests shown the robustness of an ANN model.

Figure 4 represents the filtered and forecast values of the angle at bus 21 in degrees (degs). The interval 25-100 corresponds to the early morning hours up to 9 a.m. The forecasting model uses three inputs and 15 training samples. The mean absolute error (MAE; the difference between the forecast and filtered values) is 0.23221 degs for the interval 25-100. This performance is very good besides the two different trends in Figure 4. It can be observed that the ANN model adapts very fast to the abrupt transition.

Figure 5 shows the time evolution for the voltage magnitude at the same bus. Again, a good performance (MAE = 0.00186 pu) is achieved by using the same ANN architecture and number of training samples.

The time behaviour of the angle at the bus 21 during the interval 155-240, comprising the peak hours, is shown in Figure 6. The bus load patterns, which includes a random fluctuation, together with the measurement noises make this period especially difficult for any forecasting model. Even though, the ANN model with 3 inputs and 15 training samples has shown an acceptable performance (MAE = 0.24902 degs.).

A comparative study has shown that an ANN with 3 inputs trained with 15 samples has been consistently more accurate than other alternatives. The global performance is summarized in Table 1.

![Figure 3: Total system load in percent](image_url)

![Figure 4: Filtered and forecast values for \( \theta_{21} \) (interval 25-100)](image_url)

![Figure 5: Filtered and forecast values for \( V_{21} \) (interval 25-100)](image_url)

<table>
<thead>
<tr>
<th></th>
<th>P - F</th>
<th>P - T</th>
<th>F - T</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angles (degs)</td>
<td>0.24583</td>
<td>0.19136</td>
<td>0.15335</td>
</tr>
<tr>
<td>Voltages (pu)</td>
<td>0.00159</td>
<td>0.00096</td>
<td>0.00124</td>
</tr>
</tbody>
</table>
The absolute errors represent the average among all variables (angles and voltages separately), considering all 288 time intervals. The errors "P - F", "P - T" and "F - T" denote the differences between the predicted and filtered values, predicted and true values, and filtered and true values, respectively. As can be seen from Table 1, the performance of the ANN forecasting models for bus 21 is consistent with the global average performance.

In general, one can expect "F - T" < "P - T", i.e., filtered errors smaller than predicted errors. However, due to the interpolative properties of the ANN forecasting models, it is possible that the predictive values be a better approximation to the true ones for curve sections with well-defined trends. This is illustrated in Table 1 for the voltage magnitudes.

Finally, considering the same time interval (155-240) as in Fig. 6, Figure 7 represents a sudden change in the operating point, at time instant 200, caused by a truly reported line outage; branch 10 - 21. Once more, an ANN with 3 inputs trained with 15 samples is applied. To avoid scale problems, the forecasting for instants 201 and 202 are omitted. These two values are limited by a ramp activation function (lower bound = 0 and upper bound = 40 degrees) for the output unit, in order to avoid unfeasible angle (or voltage magnitudes) forecasts. As can be seen, the ANN has a remarkable adaptability, considering that the forecast error is back to the normal range, without re-initializing of the model, three instants after the line outage.

Equations 8 and 11 are left unbounded in all tests.

5. CONCLUSIONS

In this work artificial neural networks have been applied to state forecasting in power systems. It has been shown empirically that the effective complexity of an ANN model adapts to the time series' complexity. Artificial neural networks have demonstrated remarkable robustness and adaptability. This seems to be the great advantage of nonlinear ANN models over linear models in forecasting. A comparative study with linear models along with data debugging ability evaluation will be the subject of a future work.

6. ACKNOWLEDGEMENTS

The authors acknowledge the financial assistance granted by CNPq, SCT and FAPERJ of Brazil in support of this research project. Dr. Do Coutto Filho is also indebted to the Catholic University of Rio de Janeiro for granting him a sabbatical leave.

7. REFERENCES


