Demand Forecasting for Control of the Use of Transmission System for Electric Distribution Utilities

Vitor Hugo Ferreira
Electrical Engineering Department
Fluminense Federal University (UFF)
Niterói, RJ, Brazil
vitor@vm.uff.br

Silvio Michel de Rocco
Companhia Paranaense de Energia (COPEL)
Curitiba, PR, Brazil
rocco.silvio@copel.com

Alexandre Rasi Aoki
Electric & Electronic Department
Institute of Technology for Development (LACTEC)
Curitiba, PR, Brazil
aoki@lactec.org.br

Abstract— **The Brazilian electric sector reform specifying that the remuneration of distribution utilities must be through the management of their systems increased the necessity of control and management of load flows through the connection points between their systems and the basic grid as a function of the contracted amounts. The objective of this control is to avoid that these flows exceed some thresholds along the contracted values, avoiding monetary penalties to the utility or unnecessary amounts of contracted flows that overrates the costumers. This question highlights the necessity of forecast the flows in these connection points in sufficient time to permit the operator to take decisions to avoid flows beyond the contracted ones. In this context, this work presents the development of a neural network based load flow forecaster, being tested two time-series neural models: support vector machines and Bayesian inference applied to multilayered perceptron.**

Keywords: Load forecasting, artificial neural networks, complexity control, input selection, Bayesian methods, support vector machines.

I. INTRODUCTION *(HEADING 1)*

The restructuring of the Brazilian electric sector established that the costs involved with the use of the basic transmission network must be divided equally with the agents. In this context, the distribution utilities had started to contract with the Brazilian Independent System Operator (ISO), named National System Operator (NSO), global hourly integrated demands for the peak load period. In fact, the peak load values were contracted with the transmission utilities which were represented by the legal intervening, in this case the NSO.

Since 2003, a resolution of the Brazilian National Electric Energy Agency (NEEA) established that previous contracted peak load have to be cancelled in a 25% per year basis, since these contracts paid a different tariff. The amounts necessary to complement the market of the distribution utilities, previously exempt of the transmission system use tariff (TSUT), became to be contracted with a new nodal tariff for any connection point with the Basic Transmission Network, i.e, transmission system with voltage level superior to 230 [kV]. From January 1st, 1997, the use of the transmission system become to be contracted with nodal taxes by connection points with the Basic Transmission Network established for the peak load period, i.e., 6:00 pm to 9:00 pm.

The amount of use of the transmission system to be contracted is established by the maximum integrated load (15 minutes integration period) for the peak load period for each connection of the system with the Basic Transmission Network. In the peak load period, if the flow in the connection point exceeds 5% of the contracted amount the utility is penalized with a tariff that equals three times the tariff for that period [1]. For other periods including weekends, the utilities must contract values for the flow in their connection points with the Basic Transmission Network, but occasional flows beyond the contracted ones are not penalized. These facts highlights the necessity of the distribution utilities to forecast their flows in the connection points with the Basic Transmission Network in time to permit the operators to actuate in the system in order to avoid flows beyond the contracted ones.

Several short-term load forecasting methods have been proposed during the last four decades. Such a long experience in dealing with the load forecasting problem has revealed some useful models such as the ones based on multilinear regression, Box-Jenkins method, Artificial Neural Networks (ANNs) [2], fuzzy systems, and hybrid models. However, autonomous load forecasters, i.e., automatic input selection and model complexity control, are still needed to avoid expert intervention and to extend the application to the bus load level [3].

This paper presents a methodology based on some of the most suitable techniques for controlling ANN complexity, with simultaneous selection of appropriate explanatory input variables for short-term load forecasting applied to the prediction of the amounts of use of the transmission network. In order to automatically minimize the out-of-sample prediction error, Bayesian Multi-Layered Perceptron (BMLP) is applied [4]. This training method includes complexity control terms in his objective function and automatic techniques to rank and select most relevant variables, which allow autonomous modeling and adaptation. The methodology is compared with popular Support Vector Machine (SVM) learning [5], [6] that recently wins a load forecasting competition [7].

This paper is organized as follows. In Section II, the Bayesian inference applied to MLP specification and training is present. In Section III, the Support Vector Regression is discussed, with the Sections IV and V dedicated to present and discuss the results in order to consolidate the conclusions.

II. BAYESIAN MLP SPECIFICATION AND TRAINING

Let $\underline{x} \in \mathbb{R}^n$ be the vector containing the input signals and $\underline{w} \in \mathbb{R}^M$ the vector including all the weights and biases of a single hidden layer MLP with m hidden neurons, then $M = mn + 2m + 1$. Representing the biases of the hidden neurons by b_k, with b representing the bias of the linear neuron of the output layer, w_k the weights connecting the hidden with the output layer and w_{ik} the ones connecting the input layer with the hidden, the final output of the MLP is given by:

$$y = f(\underline{x}, \underline{w}) = \sum_{k=1}^{m} \left[w_k \varphi \left(a_k \sum_{i=1}^{n} (w_{ik} x_i) + b_k \right) \right] + b \tag{1}$$

In a Bayesian inference point of view, the MLP training is seem as an estimation of, the conditional probability density function (PDF) of $\underline{w}$, given a dataset U, $p(\underline{w}|D, X)$ using Bayes' rule:

$$p(\underline{w}|D, X) = \frac{p(D|\underline{w}, X)\, p(\underline{w}|X)}{p(D|X)} \tag{2}$$

Since X is conditioning all probabilities in Eq. (2), it will be omitted from this point on. Therefore $p(D|\underline{w})$ is the likelihood of D given $\underline{w}$, $p(\underline{w})$ is $\underline{w}$'s *a priori* PDF, and $p(D) = \int p(D|\underline{w})\, p(\underline{w})\, d\underline{w}$ is enforcing $\int p(\underline{w}|D)\, d\underline{w} = 1$.

It is initially assumed that $\underline{w}$ presents a Gaussian distribution with zero mean and diagonal covariance matrix equal to $\alpha^{-1}\underline{\underline{I}}$, where $\underline{\underline{I}}$ is the $M\times M$ identity matrix, i.e.:

$$p(\underline{w}) = \frac{1}{Z_{\underline{w}}(\alpha)} e^{-\left(\frac{\alpha}{2}\|\underline{w}\|^2\right)}, \text{ where } Z_{\underline{w}}(\alpha) = \left(\frac{2\pi}{\alpha}\right)^{\frac{M}{2}} \tag{3}$$

The desired outputs can be represented by $d_j = f(\underline{x}_j, \underline{w}) + \zeta_j$, where ζ is Gaussian white noise with zero mean and variance equal to β^{-1}. The regularization factors α and β (learning parameters, also called hyperparameters), on the contrary of the other regularization techniques, are estimated along with the model parameters $\underline{w}$. Considering the previous hypotheses and assuming that the dataset patterns are independent, then:

$$p(D|\underline{w}) = \frac{e^{\left\{-\frac{\beta}{2}\sum_{j=1}^{N}\left[d_j - f(\underline{x}_j, \underline{w})\right]^2\right\}}}{Z_Y(\beta)}, \text{ where } Z_Y(\beta) = \left(\frac{2\pi}{\beta}\right)^{\frac{N}{2}} \tag{4}$$

Consequently, based on Eq. (2),

$$p(\underline{w}|D) = \frac{e^{[-S(\underline{w})]}}{\int e^{-S(\underline{w})} d\underline{w}} \tag{5}$$

where

$$S(\underline{w}) = \frac{\beta}{2}\sum_{j=1}^{N}\left[d_j - f(\underline{x}_j, \underline{w})\right]^2 + \frac{\alpha}{2}\sum_{l=1}^{M} w_l^2 \tag{6}$$

Therefore, the maximization of the *a posteriori* distribution of $\underline{w}$, $p(\underline{w}|D)$, is equivalent to the minimization of $S(\underline{w})$ [8]. The regularization term in Eq. (6), known as *weight decay*, favors neural models with small magnitudes for the connection weights. Small values for the connection weights tend to propagate the input signals through the almost linear segment of the sigmoidal activation functions. Notice that the requirement of prior information in Bayesian training is the primary instrument for controlling the ANN complexity.

One of the advantages of Bayesian training of an ANN is the embedded iterative mechanism for estimating λ, i.e., α and β, which avoids cross-validation. For multivariate problems such as load forecasting, the use of one single hyperparameter α for dealing with all connection weights is not recommended. Load and weather related input variables, such as temperature, require different priors. Even among the same type of variables, different levels of interdependency are involved (e.g., *P(k)* against *P(k+1)* and *P(K–23)* against *P(k+1)*, for an hourly basis load).

In this work, each group of connection weights directly related to an input variable receives a different α_i. The same idea is applied to the groups of weights associated with the biases (one α_i for the connections with the hidden neurons and another for the output neuron connection). One last α_i is associated with all connection weights between the hidden and output layers. Therefore, for n dimensional input vectors $\underline{x}$, the total number of α_i s is n+3.

For a given model structure, the magnitudes of the α_i s can be compared to determine the relevance of the corresponding input variables (taken from a pre-defined set). As $p(\underline{w}_i)$ is supposed to be normally distributed with zero mean and $\alpha_i^{-1}\underline{\underline{I}}$

covariance, then, the largest α_i s lead to the smallest $\underline{w}_i$ s. For estimating the *a posteriori* PDF of $\underline{w}$, Bayesian training combines the *a priori* PDF with the information provided by the training set (Eq. 5). If an α_i is large, the prior information about $\underline{w}_i$ is almost certain, and the effect of the training data on the estimation of $\underline{w}_i$ is negligible. Another way to see the influence of α_i on $\underline{w}_i$ is through Eq. (9).

The impact on the output caused by input variables with very small $\underline{w}_i$ s, i.e., very large α_i s, is not significant. However, a reference level for defining a very large α_i has to be established. Probe random variables are applied to make the reference setting data drive. More details can be found in [9]. After training the model with the pre-defined set of input variables, continuous and dummy variables are separately ranked. For each rank, the variables with corresponding α_i s larger than α_{ref} (irrelevance level) are disregarded. After input selection, the ANN is retrained with the selected variables.

Bayesian inference can also be employed to determine the best structure among a pre-defined set of possibilities, e.g., $\mathrm{H} = \{H_1, H_2, ..., H_K\}$, for which the corresponding inputs have been previously selected, i.e.,

$$P(H_h|D) = \frac{p(D|H_h)P(H_h)}{p(D)} \tag{7}$$

In Eq. (7), $p(H_h)$ represents the a priori probability of model H_h and $p(D|H_h)$ is given by:

$$p(D|H_h) = \iint p(D|\alpha,\beta,H_h)\, p(\alpha,\beta|H_h)\, d\alpha d\beta \tag{8}$$

Using Gaussian approximation around the estimated hyperparameters (from training), analytic integration of Eq. (8) is possible, leading to:

$$\ln p(D|H_h) = -S(\underline{w}) - \frac{1}{2}\ln\left|\nabla\nabla S(\underline{w})\right| + \frac{1}{2}\sum_{i=1}^{n+3} M_i\alpha_i + \frac{N}{2}\ln\beta + \ln(m!) + 2\ln m + \frac{1}{2}\sum_{i=1}^{n+3}\ln\left(\frac{2}{\gamma_i}\right) + \frac{1}{2}\ln\left(\frac{2}{N-\gamma}\right) \tag{9}$$

where m denotes the number of hidden neurons in the ANN model H_h. Since all models, *a priori*, are assumed equally probable, H_h is selected by maximizing $P(D|H_h)$, which is equivalent to maximizing $\ln p(D|H_h)$. Consequently, Eq. (9) can be used for ranking and selecting among MLPs with different numbers of neurons in the hidden layer.

III. SUPPORT VECTOR MACHINES

In classification problems [5], maximum margin SVM classifiers are estimated to minimize the generalization error bounds. The training patterns that define the separation surface, based on which the maximum margin is obtained, are called support vectors. The other training patterns have no influence on the inference process.

In order to apply the same idea to regression problems, the concept of classification margin is adapted. A margin in regression means the amount by which the training and test accuracy can differ, i.e., different error functions are used for training and testing. During training, analogously to classification problems, an approximation error is not counted if it is inside a band of size ±ε (see Eq. (11)). Any training point lying outside this band (support vectors) has its corresponding error taken into account.

As a linear machine on feature space, i.e., the space defined by a set of nonlinear basis functions $\underline{\phi}(\underline{x})$ that allows the model to produce nonlinear mappings on the original input space of $\underline{x}$, the SVM output is given by:

$$y = \sum_{j=0}^{m} W_j \phi_j(\underline{x}) = \underline{W}^t \underline{\phi}(\underline{x}) \tag{10}$$

where $\underline{\phi}(\underline{x}) = \left[1, \phi_1(\underline{x}), \ldots, \phi_m(\underline{x})\right]^t$ and $\underline{W} = \left[b, W_1, \ldots, W_m\right]^t$.

The following ε-insensitive cost function is adopted here:

$$L_\varepsilon(d,y) = \begin{cases} \left(|d-y|-\varepsilon\right)^2, & \text{for } |d-y|-\varepsilon \ge 0 \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

SVMs that use Eq. (11) as the error function are called L2-SVMs [10], in contrast with previously proposed SVM load forecasters (L1-SVMs) [7], which use an ε-insensitive linear loss function. L2-SVMs lead to differentiable analytical bounds for the generalization error. Such bounds cannot be derived for L1-SVMs. Then, the SVM hyperparameters can be directly estimated through mathematical programming techniques, avoiding cross-validation. This interesting theoretical feature of L2-SVMs is not applied in this work since the numeric optimization procedure still needs some improvements [9].

In the following development, ε and c_0 are assumed to be known, i.e., defined by the user. This assumption will be removed later. The training objective of an SVM model is the following constrained minimization of the empirical risk:

$$\min_{\underline{W}} \left\{ E_s(\underline{W}, D) = \frac{1}{N}\sum_{i=1}^{N} L_\varepsilon(d_i, y_i) \right\} \tag{12}$$

subject to

$$\|\underline{W}\|^2 \le c_0$$

where c_0 also affects the model complexity.

A. *Support Vector Regression*

The primal optimization problem formulated by Eq. (12) is transformed into its dual form, Eq. (13), to allow the incorporation of kernel functions, which avoid the requirement of knowing an appropriate $\underline{\phi}(\underline{x})$:

$$\max_{\underline{\alpha},\underline{\alpha}'}\left\{Q(\underline{\alpha},\underline{\alpha}')=\sum_{i=1}^{N}d_i(\alpha_i-\alpha_i')-\varepsilon\sum_{i=1}^{N}(\alpha_i+\alpha_i')\right.$$
$$\left.-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i-\alpha_i')(\alpha_j-\alpha_j')\left[K(\underline{x}_i,\underline{x}_j)+\frac{\delta_{ij}}{C}\right]\right\} \tag{13}$$

subject to

$$\sum_{i=1}^{N}(\alpha_i-\alpha_i')=0$$
$$\alpha_i\geq 0,\ \alpha_i'\geq 0,\ i=1,2,...,N$$

In Eq. (13), $K(\underline{x}_i,\underline{x}_j)=\underline{\phi}^t(\underline{x}_i)\underline{\phi}(\underline{x}_j)$ is the inner product kernel defined according to Mercer's theorem [6], δ_{ij} is the Kronecker delta function, and C is the regularization hyperparameter . Then, the output of an SVM is given by:

$$y=f(\underline{x},\underline{W})=\sum_{i=1}^{N}(\alpha_i-\alpha_i')K(\underline{x},\underline{x}_i) \tag{14}$$

As indicated in Eq. (14), the support vectors are the training patterns for which $\alpha_i\neq\alpha_i'$, i.e., the ones located outside the band defined by ε . In fact, an SVM model can be represented as a feedforward ANN model with hidden layer units activation functions defined by the kernel $K(\underline{x},\underline{x}_i)$. Notice that an SVM model, depending on the adopted kernel function, has the MLP and the RBF as special cases, when the kernels are specified as sigmoid and Gaussian functions, respectively. However, an important difference compared with traditional training algorithms for MLPs and RBFs is related to the convexity of the corresponding objective functions. While for error backpropagation and clustering algorithms local minima can be troublesome, in SVM training the solution is unique due to the corresponding quadratic optimization problem.

IV. RESULTS

The Bayesian inference based MLP specification (BMLP) and Support Vector Regression (SVR) are applied to the prediction of seven time-series related to connection points from COPEL Distribuição to Basic Transmission Network. These points are related with the amounts of use of the transmission system. The original data-base consists of hourly integrated demands [MW], from which are obtained the daily maximum hourly integrated demand for the peak load period. For each connection point, this daily time-series is denominated $S(k)$.

The $S(k)$ time-series are pre-processing for identification and estimation of missing values and outliers. For this, seasonal mean μ_{dm} and standard deviation σ_{dm} are calculated, considering data from a specific day *d* of the week from a specific month *m*. For outlier identification, the respective mean and standard deviation are estimated without the investigated point. Data lying outside the interval $[\mu_{dm}-3\sigma_{dm}\quad \mu_{dm}+3\sigma_{dm}]$ are substituted by the respective violated limit. After these pre-processing, the time-series is normalized $[-1\quad 1]$ as usual in development of neural network based regression models.

The forecasting problem consists of the prediction from one to seven days ahead of daily maximum flow in the seven connection points in the peak load period i.e., 6:00 pm to 9:00 pm. These predictions are used to support operators' decisions in order to avoid flows beyond the contracted ones, which implies in monetary penalties to the utility.

All tested configurations use as basic inputs seven consecutive lags, i.e., $S(k-1)$, $S(k-2)$,..., $S(k-7)$. In order to model the seasonal weather effects, twelve dummy variables (1 of n representation) are used to represent the month. The output is given by $S(k)$ and the predictions from two to seven days ahead are obtained by recursion of the respective forecasts.

Four configurations of the input space representation and database segmentation were tested. The first one includes in the input space representation seven dummy variables to represent the day of the week. This input-output database is divided in two subgroups, one containing prediction days related to holidays and another with patterns related to other days (normal days). For each subgroup is developed one independent model. This configuration is denominated Methodology 1.

The second modeling strategy treats the seasonal week dynamic by a segmentation of the training patterns in eight subgroups, seven related with each day of the week and one dedicated to holidays. In order to model the effects of the holidays in each day of the week, the holiday model uses seven dummy variables to codify the day of the week. For each subgroup is developed an independent neural model. This configuration is denominated Methodology 2.

The third configuration divides the training patterns in three subgroups, one for weekdays, other for weekends and the last one for holidays. The representation of each day of the week is made by dummy variables. So for the first subgroup, five dummy variables are used; for the second two dummy variables (Saturday and Sunday); for the last one seven dummy variables. For each subgroup is developed an independent neural model. This modeling strategy is denominated Methodology 3.

The last but not least strategy uses only the basic inputs and separates input-output patterns in three different subgroups. The first is related with workdays, the second is associated with weekends and the last one is related with holidays. For each subgroup is developed an independent neural model. This configuration is denominated Methodology 4.

The four modeling strategies listed above are uses as regression models the neural networks presented in Sections II and III. The BMLP searches models with number of hidden neurons varying from $m_{\min}=1$ to $m_{\max}=10$. The model with highest evidence (Eq. (9)) is selected to make predictions.

The SVR model needs a specification of hyperparameters ε, C e σ. In this works these values are estimated via cross-validation. This re-sampling technique is applied to any subgroup containing N training pattern, where $v = N - t$ are selected randomly from the original database to create the validation data and $t = \text{int}(2N/3)$ patterns are dedicated for training purposes, where $\text{int}(x): \mathbb{R} \to \mathbb{Z}$ is a function that returns the first integer greater or equal than x.

The neural regression models developed for the configurations listed above are evaluated using data from 01/01/1995 to 20/02/2007, with the predictions made from one to seven days ahead to the period of 01/03/2007 to 10/04/2007. The models are re-trained each day using the most recent data, i.e, the data from testing period is include in training data after the seven day ahead predictions were made. So, for each day ahead prediction is available 34 predictions for each of the seven connection points. Table I shows the mean absolute percentage error (MAPE) for the whole test period for each connection point. The two first lines of this table present some statistics estimated for all weekdays and only workdays, since only for workdays the utility is exposed to monetary penalties related to overflows. The last line of this Table presents the mean computation burden in minutes for estimation of the seven day ahead predictions with each neural model and modeling strategy.

TABLE I. MEAN ABSOLUTE PERCENTAGE ERROR - MAPE (%) – TEST PERIOD

	Methodology 1		Methodology 2		Methodology 3		Methodology 4	
	BMLP	SVM	BMLP	SVM	BMLP	SVM	BMLP	SVM
All weekdays	7.59	7.65	12.03	13.92	7.08	7.25	6.96	7.31
Only workdays	6.45	6.58	11.58	14.84	6.14	6.12	6.11	6.38
Time (min)	2.78	5.08	4.80	0.75	3.20	2.32	2.42	2.45

The results present in Table I show the superior performance of the BMLP with Methodology 4. In fact, this neural model presented greater results for 5 of 7 connection points investigated. Although the least computation burden showed by SVR using Methodology 2, the satisfactory result obtained by the BMLP with Methodology 4 recommends the use of this modeling strategy for the analyzed database.

Table II presents the maximum absolute percentage error for all connections points considered in the study. The results presented in Tables I and II highlights the performance of BMLP against SVR for the time-series considered, especially for BMLP with Methodology 4. Although the SVR obtained the least maximum absolute percentage error using this modeling strategy, BMLP still obtain a comparable performance in terms of the maximum absolute percentage error. The discrepancy with the results presented in Table II when compared the statistics for all weekdays against the ones obtained only for workdays may be explained by the absence of monetary penalties for flows beyond the contracted ones for weekends. These differences were presented in Table I too, but in Table II these discrepancies are clearer. This absence of penalty makes the operation of the system more flexible in weekends, when system maintenance including eventual load transfer and other maneuvers can be made without violation of the established contracts. This anomalous operation modifies considerably the dynamics of the maximum daily flow in the peak load period deteriorating the performance of the neural models developed.

TABLE II. MAXIMUM ABSOLUTE PERCENTAGE ERROR - MAE (%) – TEST PERIOD

	Methodology 1		Methodology 2		Methodology 3		Methodology 4	
	BMLP	SVM	BMLP	SVM	BMLP	SVM	BMLP	SVM
All weekdays	95.26	93.97	68.65	89.79	84.52	68.09	66.62	65.70
Only workdays	51.07	63.96	63.34	75.76	46.96	56.05	46.96	65.70

Table III shows the mean absolute percentage error for each one of the seven days ahead for the BMLP with Methodology 4, the modeling strategy that presented best results. Table IV presents the mean absolute percentage error for each one of the seven connection points considered in this work. Only for workdays, Table IV shows that the MAPE obtained for three of the seven connection points is below 5%. The inferior results obtained for the other connection points are related with the characteristics of the respective time-series, as illustrated in Figure 1. This Figure shows the daily time-series for connection point 3 and highlights the test period for this point. As illustrated in the Figure, at 01/01/2007 the dynamics of the time-series suffers an abrupt variation, possibly related to load transfers since the load variation persists some days ahead. This abrupt alteration occurs near the test period, compromising the development of models since the dynamics of the time series is seriously modified. Even in a presence of a filter to detect and smooth outliers and missing values, the identification and substitution of these load transfers is not a trivial task, demanding the expert intervention to filter the time-series.

TABLE III. MEAN ABSOLUTE PERCENTAGE ERROR - MAPE (%) - FOR EACH DAY AHEAD. TEST PERIOD – BMLP METHODOLOGY 4

	D+1	D+2	D+3	D+4	D+5	D+6	D+7
All weekdays	5.21	6.42	7.17	7.37	7.39	7.44	7.75
Only workdays	4.45	5.37	6.21	6.65	6.70	6.45	6.90

TABLE IV. MEAN ABSOLUTE PERCENTAGE ERROR - MAPE (%) – FOR EACH CONNECTION POINT. TEST PERIOD – BMLP METHODOLOGY 4

	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7
All weekdays	4.38	7.64	14.16	3.97	2.93	8.95	6.72
Only workdays	4.17	7.09	8.43	3.98	2.54	9.33	7.24

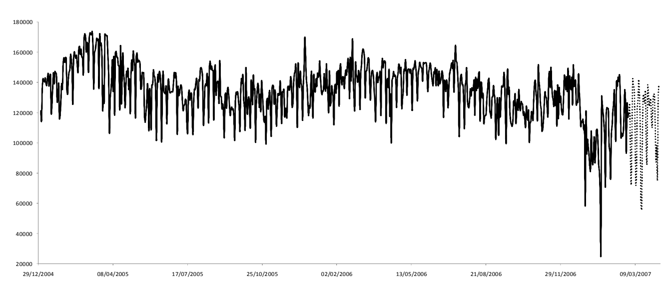

Fig. 1. Time-series and forecasts for connection point 3

V. CONCLUSION

This work presents the development of a neural network-based model for prediction of flows in connection points from COPEL Distribuição with Basic Transmission Network. These forecasts are needed to support the operators' decision making in order to avoid flows against the contracted ones, which implies in monetary penalties to the utility. Daily time-series from seven connection points were used to evaluate to neural models, Bayesian inference applied to MLP specification and training (BMLP) and Support Vector Regression (SVR). The first one proposes an autonomous modeling, i.e., automatic setting of relevant input variables and model structure in terms of number of hidden neurons, making possible the analytical treatment of several time-series automatically without cross-validation [9]. The SVR is a new neural regression paradigm, based on Structural Risk Minimization which main objective consists of development of models with good generalization capacity.

Besides the theoretical appeal and promising results previously obtained by SVR in short-term load forecasting [7], for the data-base used in this work the BMLP presented the best results. The autonomous modeling makes possible the automatic selection of input representation and model structure most probable for each time-series, avoiding manually tuned solutions that do not necessary fit well for all time-series considered. The ARD prior [8] automatically controls the impact of each variable in the estimation of the output of the network, reducing the effect of least relevant variables included in the model. Beyond the input ranking, this prior makes the complexity control intrinsic to the model estimation avoiding analytically the overfitting problem.

The computational burden required by BMLP is compatible with practical requirements. The one to seven days ahead forecasts to all of 40 connection points of COPEL Distribuição with Basic Transmission Network Extension estimated by the BMLP with Methodology 4 will require approximately one hour and forty minutes. The R&D project where this work was developed specifies that this task has to be made in four hours in order to give information to system operators in sufficient time. Actually the BMLP with Methodology 4 is implemented in COPEL Distribuição estimating in a daily basis the mentioned forecasts.

Although the satisfactory results, the Figure 1 illustrates the necessity of filtering the database in order to clean missing values, outliers and mainly load transfers that is not easily identified and re-allocated by automatic models. Autonomous models treating several load series needs automatic methods for identify and substitute these anomalous data. Automatic clustering algorithms [12] can be used for this task making possible the extension of neural network based short-term load forecasting to the bus load level.

REFERENCES

[1] ANEEL, Agência Nacional de Energia Elétrica, Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional, Mó-dulo 3 – Acesso aos Sistemas de Distribuição, Access in 29/02/2008, available at http://www.aneel.gov.br/arquivos/PDF/Modulo3-TextoBasico-2005-06-03.pdf.

[2] H.S. Hippert, R.C. Souza, and C.E. Pedreira, "Neural Networks for Load Forecasting: A Review and Evaluation", IEEE Transactions on Power Systems, v.16, n.1, pp. 44-55, Feb. 2001.

[3] N. Amjady, "Short-Term Bus Load Forecasting of Power Systems by a New Hybrid Method", IEEE Trans. on Power Systems, v.22, n.1, pp. 333-341, Feb. 2007.

[4] V.H. Ferreira, A.P. Alves da Silva. "Toward Estimating Autonomous Neural Network-based Electric Load Forecasters". IEEE Transactions on Power Systems, v.22, n.4, p. 1554-1562, Nov. 2007.

[5] V.N. Vapnik, Statistical Learning Theory, John Wiley & Sons, 1998.

[6] B.Schölkopf and A.J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond, MIT Press, 2002.

[7] B.-J. Chen, M.-W. Chang, and C.-J. Lin, "Load Forecasting Using Support Vector Machines: A Study on EUNITE Competition 2001", IEEE Transactions on Power Systems, v.19, n.4, pp. 1821-1830, Nov. 2004.

[8] D.J.C. Mackay, Bayesian Methods for Adaptive Models, Ph.D. Dissertation, California Institute of Technology, Pasadena, USA, 1992.

[9] Ferreira, V.H., Alves da Silva, A.P.: Toward Estimating Autonomous Neural Network-based Electric Load Forecasters, IEEE Transactions on Power Systems, v.22, n.4, pp. 1554-1562, Nov. 2007.

[10] M.-W. Chang and C.-J. Lin, "Leave-One-Out Bounds for Support Vector Regression Model Selection", Neural Computation, v.17, n.5, pp. 1188-1222, May 2005.

[11] C.M. Bishop, Neural Networks for Pattern Recognition, New York, Oxford University Press, 1995.

[12] Y.-M. Cheung, Maximum Weighted Likelihood via Rival Penalized EM for Density Mixture Clustering with Automatic Model Selection, IEEE Transactions on Knowledge and Data Engineering, v.17, n.6, pp. 750-761, Jun. 2005.