Direct and Indirect Classification of High-Frequency LNA Performance Using Machine Learning Techniques

Peter C. Hung, Seán F. McLoone, Magdalena Sánchez,
Ronan Farrell, Guoyan Zhang

Institute of Microelectronics and Wireless Systems, Department of Electronic Engineering,
National University of Ireland Maynooth, Maynooth, Co. Kildare, Ireland
{phung, semcloone, msanchez, rfarrell, gzhang}@eeng.nuim.ie
http://imws.eeng.nuim.ie/

Abstract. The task of determining low noise amplifier (LNA) high-frequency performance in functional testing is as challenging as designing the circuit itself due to the difficulties associated with bringing high frequency signals off-chip. One possible strategy for circumventing these difficulties is to attempt to predict the high frequency performance measures using measurements taken at lower, more accessible, frequencies. This paper investigates the effectiveness of machine learning based classification techniques at predicting the gain of the amplifier, a key performance parameter, using such an approach. An indirect artificial neural network (ANN) and direct support vector machine (SVM) classification strategy are considered. Simulations show promising results with both methods, with SVMs outperforming ANNs for the more demanding classification scenarios.

1 Introduction

In recent years, functional testing of radio frequency integrated circuits (RFIC) has faced great challenges, especially for multi-gigahertz RF components. Two main problems exist: relaying the multi-gigahertz RF signal to the external tester without affecting the performance of tested RF circuits; building RF production testers operating in the gigahertz range that are not prohibitively expensive. While advances in technology and market requirements has seen rapid growth in high-frequency and high integration RFIC designs, testing practice has not followed suit. Indeed, reliable high-frequency testing has become the dominant factor in the cost and time-to-market of novel wireless products [1]. Consequently, developing cost-efficient testing solutions is becoming an increasing important research topic [2-4]. Some of the proposed schemes for RFIC testing are based on an end-to-end strategy in which the output of the transmitter and the input of the receiver are linked through a loop-back connection. In this configuration, the testing of the complete system is carried out without any external stimulus by employing the on-chip digital hardware available. Unfortunately, this solution is not always applicable to all kinds of RF components. Other recent proposals for RF system testing have focused on the development of methodologies and algorithms for automated test, and Design for Testability (DfT). In Built-

In-Test (BIT), for example, additional circuitry is included that allows high frequency tests to be performed on-chip and then evaluated using lower frequency or DC external testers [3-5]. However, when considering BIT testing, issues such as area overhead for embedding and BIT power consumption can add significantly to the cost of design.

In this paper a different approach is considered. Since many RFICs show strong correlation between their responses to circuit parameter variation at different frequencies, it is hypothesised that knowledge of responses at lower frequencies may provide sufficient information to allow classification of responses at higher frequencies.

To investigate this hypothesis, testing of a low noise amplifier (LNA), a key component in modern telecommunication systems, is used as a case study. A standard 2.4 GHz MOSFET design, simulated in ADS® using UMC's 0.18 µm silicon process technology, provided the data for our experiments [6]. The LNA circuit consisted of 2 bias transistors, 4 RF transistors (0.18 µm channel length and 0.5 µm channel width), 4 resistors, 3 capacitors and 4 inductors and was deemed to be functioning correctly if the value of S_{21} @ 2.4 GHz was in the range 14.7 dB to 17.2 dB and faulty otherwise. S_{21}, a critical RF circuit performance measure, is essentially the gain of the amplifier.

Random circuit parameter perturbations representative of typical manufacturing process variations were generated and the value of S_{21} recorded at different frequencies. Fig. 1 shows a plot of the correlation that exists between variations in gain (S_{21} @ 2.4 GHz) and variations in the same parameter computed at other frequencies. There is a strong, but decreasing, correlation evident as the circuit excitation frequency moves away from the operating frequency.

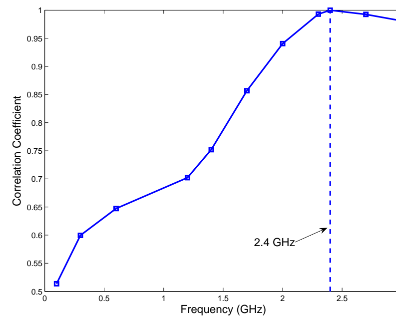
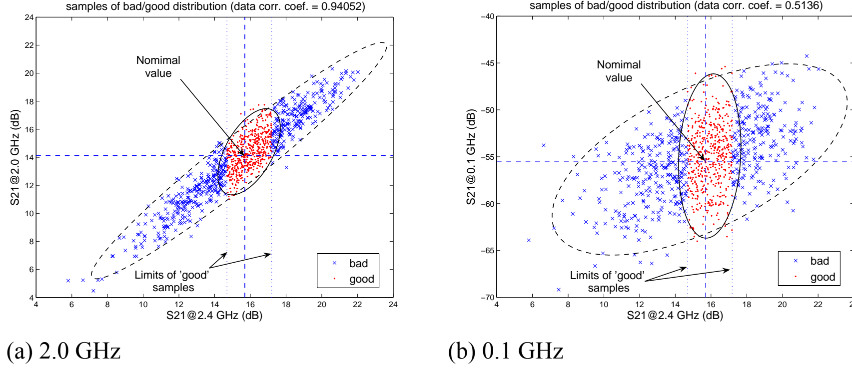


Fig. 1. Correlation coefficients between S_{21} value computed at different frequencies to the value computed at the target frequency (2.4 GHz)

Even when the correlation is high, circuit performance classification on the basis of these lower frequencies is not straightforward, as demonstrated in Fig. 2. This shows the relationship between S_{21} @ 2.4 GHz and the values computed at: (a) 2.0 GHz and (b) 0.1 GHz, respectively, and highlights the fact that even when the correlation is greater than 90% it is not possible to discriminate between the 'good' and 'bad' circuits effectively. In fact, simple thresholding on the basis of S_{21} @ 2.0 GHz leads to a misclassification rate of greater than 20%. The misclassification rate increases rapidly as the frequency is reduced and reaches 43.6% for of S_{21} @ 0.1 GHz.

The rapid deterioration in performance is a result of the localised influence of some parameter variations and the complex nonlinear interaction between circuit components.



(a) 2.0 GHz (b) 0.1 GHz

Fig. 2. S_{21} parameter relationships for the 2.4 GHz LNA model used in circuit simulations: (a) S_{21} @ 2.0 GHz and (b) S_{21} @ 0.1 GHz plotted against S_{21} at the operating frequency

Since the shape of the frequency response of an LNA is a deterministic nonlinear function of its component parameters, better classification performance can be expected if the information from several low frequency measurements can be combined. To that end, this paper considers the possibility of classifying circuit performance using machine learning techniques. Two strategies are investigated. In the first, an Artificial Neural Network (ANN) is trained to predict the value of S_{21} @ 2.4 GHz from the values measured at other frequencies and then a thresholding rule is applied to this prediction to perform the circuit classification, while in the second, a Support Vector Machine (SVM) is trained to directly classify circuit performance on the basis of the low frequency S_{21} measurements. These two machine learning techniques and the LNA classification methodology are introduced in Section 2. The simulation study is then described in Section 3 followed by the results in Section 4. Finally, the conclusions of the study are presented in Section 5.

2 Machine Learning LNA Performance Classification

Defining the set of N low-frequency S_{21} measurements of the ith LNA circuit as the feature vector $\mathbf{x}_i$ (row vector) and the corresponding class label y_i, with $y_i = +1$ indicating 'good' and $y_i = -1$ indicating 'bad', we can generate a set of L training data examples,

$$(\mathbf{X}, \mathbf{y}) = (\mathbf{x}_1, y_1), \ldots (\mathbf{x}_i, y_i), \ldots (\mathbf{x}_L, y_L) \quad \in \Re^N \quad , \tag{1}$$

with which to train a classifier to estimate a decision function

$$f(\mathbf{x}) : \Re^N \rightarrow \{\pm 1\} \quad (2)$$

that can then be used to classify new LNA circuits. Here, 'good' and 'bad' are determined by a threshold function, z_{th} applied to S_{21} @ 2.4 GHz, that is

$$z_{\text{th}}(x) = \begin{cases} +1 & \text{if} \quad 14.7 < x < 17.2 \\ -1 & \text{otherwise} \end{cases} \quad (3)$$

The decision function in (2) can be estimated directly from the training data by using, for example, a SVM classifier. Alternatively it can be estimated indirectly by first predicting the value of S_{21} @ 2.4 GHz from the feature vector,

$$g(\mathbf{x}) \rightarrow S_{21}@2.4\text{GHz} , \quad (4)$$

and then using a threshold function to perform the classification, that is

$$f(\mathbf{x}) : z_{\text{th}}(g(\mathbf{x})) \rightarrow \{\pm 1\} \ . \quad (5)$$

An ANN such as a Multilayer Perceptron can be used to learn the nonlinear mapping represented by (4).

2.1 Support Vector Machines (SVM)

SVMs, first proposed by Vladimir Vapnik in 1963 [7] are a supervised linear learning technique widely used for classification problems. They are known to perform binary classification well in many practical applications. Consider a separating hyperplane that divides two classes of data:

$$\mathbf{w} \cdot \mathbf{x} - b = 0, \quad \mathbf{w} \in \Re^N, b \in \Re \ , \quad (6)$$

where $\mathbf{w}$ and b are unknown coefficients, and two additional hyperplanes that are parallel to the separating hyperplane:

$$\begin{aligned} \mathbf{w} \cdot \mathbf{x} - b &= 1 \\ \mathbf{w} \cdot \mathbf{x} - b &= -1 \end{aligned} \quad (7)$$

Defining the margin as the perpendicular distance between the parallel hyperplanes, the optimal hyperplane is the one which results in the maximum margin of separation between the two classes. Mathematically the problem can be expressed as

$$\max_{\mathbf{w}} \frac{2}{|\mathbf{w}|} \equiv \min_{\mathbf{w}} (\mathbf{w} \cdot \mathbf{w}), \quad \text{subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1, \quad i = 1,2,\ldots,L \ . \quad (8)$$

This is a constrained quadratic optimisation problem whose solution $\mathbf{w}$ has an expansion [8]

$$\mathbf{w} = \sum_i v_i \mathbf{x}_i \ , \quad (9)$$

where $\mathbf{x}_i$ are the subset of the training data, referred to as support vectors, located on the parallel hyperplanes, and v_i are the corresponding weighting factors. The linear SVM (LSVM) decision function is then given by

$$f_{\mathrm{LSVM}}(\mathbf{x}) = z_{\mathrm{SVM}}(\mathbf{w} \cdot \mathbf{x} - b) \tag{10}$$

where z_{SVM} is defined as

$$z_{\mathrm{SVM}}(\phi) = \begin{cases} +1 & \text{if } \phi \geq 0 \\ -1 & \text{if } \phi < 0 \end{cases} \tag{11}$$

The decision function in Eq. (10) can be rewritten as

$$f_{\mathrm{LSVM}}(\mathbf{x}) = z_{\mathrm{SVM}}\left(\sum_i v_i (\mathbf{x} \cdot \mathbf{x}_i) - b \right) \tag{12}$$

with the result that it is only dependent on dot products between the test data vector, **x**, and the support vectors. This important property allows SVMs to be extended to problems where nonlinear partitions of data sets are required. This is achieved by replacing the dot products by a kernel function $k(.)$ which meets the Mercer's condition [9]:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)\ , \tag{13}$$

thereby mapping the data into a higher dimension feature space where linear SVM classification can be performed. Note the resulting decision function, in the original data space, will be nonlinear and takes the form

$$f_{\mathrm{SVM}}(\mathbf{x}) = z_{\mathrm{SVM}}(r), \text{ where } r = \sum_i^L v_i . k(\mathbf{x}, \mathbf{x}_i) - b \tag{14}$$

In non-separable problems where different classes of data overlap, slack variables can be introduced so that a certain amount of training error or data residing within the margin is permitted. This gives rise to a 'soft margin' optimisation function [9, 10]. To give users the ability to adjust the amount of training error allowed in the optimisation, a smoothing parameter C is incorporated into the soft margin function, with a larger C corresponding to assigning a larger penalty to errors.

The Gaussian radial basis function (RBF) defined as

$$k_{\mathrm{RBF}}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{2\sigma^2} \right) \tag{15}$$

is a popular choice of SVM kernel and the one selected for this application. The parameter, σ, controls the width of the kernel and is determined as part of the classifier training process.

2.2 Artificial Neural Networks

Neural networks [11] are one of the best known and most commonly-used machine learning techniques. There are various configurations and structures of NNs, but all contain an array of neurons that are linked together, usually in multiple layers. In this application a single hidden layer Multilayer Perceptron (MLP) topology is chosen because of its universal function approximation capabilities, good generalisation properties and the availability of robust efficient training algorithms [12].

The output of a single hidden layer MLP can be written as a linear combination of sigmoid functions (i.e. neurons),

$$g(\mathbf{x}, \mathbf{w}_{\mathrm{NN}}) = b^h + \sum_i w_i^h sig_i(\mathbf{x}), \tag{16}$$

where

$$sig_i(\mathbf{x}) = \frac{1}{1 + \exp(\mathbf{w}_i^u \cdot \mathbf{x} + b_i^u)}. \tag{17}$$

Here, w_i^h, $\mathbf{w}_i^u$, b_i^u,($i = 1,2,\cdots,M$) and b^h are weights and biases which collectively form the network weights vector, $\mathbf{w}_{\mathrm{NN}}$. Defining a Mean Squared Error (MSE) cost function over the training data

$$E(\mathbf{w}_{\mathrm{NN}}) = \frac{1}{L}\sum_{p=1}^{L}(g(\mathbf{x}_p, \mathbf{w}_{\mathrm{NN}}) - d_p)^2, \tag{18}$$

with d_p corresponding to the desired network output for the pth training pattern (i.e. S_{21} @ 2.4 GHz), the optimum weights can be determined using gradient based optimisation techniques.

3 Simulation Study

To evaluate the potential for employing multiple low frequency S_{21} measurements to classify LNA S_{21} performance at 2.4 GHz and to compare the performance of the proposed machine learning classifiers, a Monte Carlo simulation study was undertaken using a 2.4 GHz LNA model implemented in ADS®. Uniform random variations were introduced into 38 of the model parameters to represent typical LNA manufacturing process variations and 10,000 circuit simulations performed. While in practice circuit parameters might be expected to vary normally around their nominal values, uniform distributions were chosen to give an even coverage of the LNA parameter space. Catastrophic failures, such as short-circuits, were not considered as these can be identified relatively easily using existing IC testing techniques.

For each circuit the S_{21} performance parameter was recorded at 0.1, 0.3, 0.6, 1.2, 1.4, 1.7 and 2.0 GHz and also at the operating frequency (2.4 GHz). This data was

then normalised to have zero mean and unit variance and divided into training and test data sets, each containing 5,000 samples.

Two different feature vectors were considered in the study, containing S_{21} measurements up to 1.4 GHz and 2.0 GHz, respectively, that is:

$$\mathbf{x}_{1.4} = [S_{21}^{0.1}, S_{21}^{0.3}, S_{21}^{0.6}, S_{21}^{1.2}, S_{21}^{1.4}] \quad (19)$$

and

$$\mathbf{x}_{2.0} = [S_{21}^{0.1}, S_{21}^{0.3}, S_{21}^{0.6}, S_{21}^{1.2}, S_{21}^{1.4}, S_{21}^{1.7}, S_{21}^{2.0}]. \quad (20)$$

Here, S_{21}^{f} denotes the value of S_{21} at f GHz. In each case the target MLP model output is $S_{21}^{2.4}$ while the target labels for the SVM classifier are given by $z_{th}(S_{21}^{2.4})$.

3.1 MLP Training

MLP training was performed using the hybrid BFGS training algorithm [12] with stopped minimisation used to prevent over-fitting [13]. The optimum number of neurons (M) was determined for each model by systematically evaluating different network sizes and selecting the network with the minimum MSE on the test data set. Training was repeated ten times for each network size to allow for random weight initialisations and the best set of weights recorded in each case.

The optimum network sizes and resulting model fit, measured in terms of the correlation with the true value of $S_{21}^{2.4}$, are summarised in Table 1. For comparison purposes, the correlation between $S_{21}^{2.4}$ and the measurements at 1.4 and 2.0 GHz are also given.

Table 1. Optimum MLP classifier model dimensions and resulting model fit

Feature vector	Network dimensions	Model fit
$\mathbf{x}_{1.4}$	MLP(5,12,1)	0.9364
$\mathbf{x}_{2.0}$	MLP(7,15,1)	0.9979
$S_{21}^{1.4}$	-	0.7537
$S_{21}^{2.0}$	-	0.9408

As expected, the exploitation of multiple frequencies results in much better predictability of $S_{21}^{2.4}$ than using the measurement at a single frequency. Notably, the information provided by $S_{21}^{2.0}$ is still marginally greater than the combined information provided by all measurements up to 1.4 GHz. The classification performance of these networks, when employed in the indirect LNA classifier scheme, will be reported in Section 4.

3.2 SVM Training

SVM training was performed using the Matlab® package simpleSVM [14]. The kernel width parameter σ and smoothing parameter C were fine tuned manually and optimised on the basis of classification performance on the test data set. The final values selected were $\sigma = 0.9$ and $C = 100{,}000$.

Initial SVM results were quite poor despite expectations of superior performance to indirect classification using MLPs (results presented in Section 4). It was determined that this was due to the bimodal distribution of the out-of-specification circuits forming the 'bad' class, i.e. it consists of two segments separated by the 'good' class, as shown in Fig. 3.

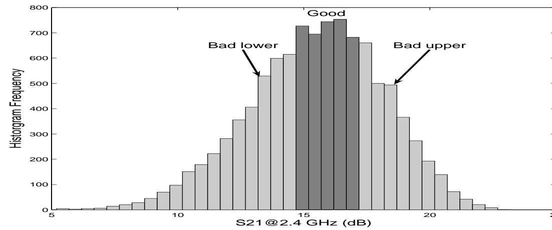


Fig. 3. Histogram of 'good' and 'bad' circuits as defined by S_{21} performance criteria

The *a priori* knowledge of the distribution of the 'bad' circuits can be taken into account by splitting the 'bad' samples into 'bad lower' and 'bad upper' samples, thereby introducing 3 classes – 'bad lower', 'good' and 'bad upper'. SVM classification is then performed in two-stages. Firstly, two binary SVMs are trained, one to classify LNAs as either 'bad lower' or 'not bad lower', and one to classify LNAs as either 'bad upper' or 'not bad upper'. Then the overall classification is obtained from a weighted linear combination of the individual decision functions, that is:

$$f_{\mathrm{SVM3}} = z_{\mathrm{SVM}}\left(Kr_{\mathrm{BL}} + r_{\mathrm{BU}}\right). \tag{21}$$

Here, r is as defined in Eq. (14), subscripts BL and BU represent the 'bad lower' and 'bad upper' classifiers and the constant K is a scalar which is chosen to maximise the correlation between f_{SVM3} and the true class labels over the training data.

This 3-class SVM approach is denoted SVM3 while the original two class SVM classifier will be referred to as SVM2.

4 Results

The performance of the MLP, SVM2 and SVM3 LNA classifiers was measured in terms of the following metrics computed on the test data set:

GPR: good pass rate - Percentage of good LNAs passed;
BFR: bad fail rate - Percentage of bad LNAs failed;
FR: failure rate - Percentage of passed LNAs incorrectly classified as 'good';
MCR: misclassification rate - Percentage of LNAs incorrectly classified.

Since the good pass rate (GPR) and bad fail rate (BFR) of a classifier vary as a function the classification threshold, with one increasing as the other deceases, the threshold can be adjusted to control one or other of these metrics. Here, the threshold of each classifier was adjusted to give a fixed BFR, reflecting the importance in the electronics industry of controlling the number of faulty components released to the market.

Table 2 shows the mean performance of each classifier when their thresholds were selected to give a BFR of 90% and 75%, respectively. The result for classification on the basis of single frequency measurements at 1.4 and 2.0 GHz are also included for comparison. To provide robust estimates, the metrics were computed by averaging over 100 batches of LNAs generated from the test data set using sampling with replacement. Each batch consisted of 500 'good' and 500 'bad' circuits randomly selected (with replacement) from a total of 1,795 'good' and 3,205 'bad' examples in the data set.

Table 2. Mean performance of the MLP, SVM2 and SVM3 LNA classifiers

Inputs	**BFR (%)**	90			75		
	Method	**GPR (%)**	**FR (%)**	**MCR (%)**	**GPR (%)**	**FR (%)**	**MCR (%)**
$\mathbf{x}_{1.4}$	MLP	57.11	14.88	26.45	81.03	23.55	21.99
	SVM2	44.18	18.43	27.96	77.32	24.41	11.39
	SVM3	**82.30**	10.82	13.85	**90.85**	21.56	17.08
$\mathbf{x}_{2.0}$	MLP	**99.70**	9.09	5.15	**100.00**	19.97	12.50
	SVM2	84.40	10.57	12.80	97.33	20.41	13.83
	SVM3	97.60	9.28	6.20	99.34	20.09	12.83
$S_{21}^{1.4}$	-	19.98	33.29	45.01	47.13	34.63	38.94
$S_{21}^{2.0}$	-	55.48	15.24	27.27	84.65	22.78	20.17

Comparing the different classifiers, it can be seen that SVM3 provides the most consistent performance. Although the MLP classifier produces the best results when using frequencies up to 2.0 GHz, SVM3 is only around 2% and 0.7% behind. More importantly, when LNA classification is performed using S_{21} measurements up to 1.4 GHz only, SVM3 outperforms the MLP solution by 25% and 9% respectively. It is noted that high GPRs are always accompanied by correspondingly low values of FR and MCR. As expected, the performance of all classifiers deteriorates when only the lower frequency S_{21} measurements ($\mathbf{x}_{1.4}$) are considered, though SVM3 and MLP still outperform the classifications obtained using a single S_{21} measurement at 2.0 GHz.

Interestingly, the SVM classifier was only able to outperform the MLP classifier when the *a priori* knowledge of the bimodal distribution of the out-of-specification LNAs was taken into account. In all cases SVM2 is substantially inferior to both the MLP and SVM3 classifiers. This suggests that while SVMs are the natural setting for classification they do not always yield the optimum results.

Although this study does not consider catastrophic IC failures, they can be identified relatively easily using other simple tests such as supply current tests. Overall, the results successfully demonstrate the usefulness of machine learning techniques for

LNA functional testing. For example, it can lower the cost of testing by extending the frequency range of existing ATE testers by as much as 70%.

5 Conclusions

Functional testing of high-frequency LNAs is becoming a prohibitively expensive and time-consuming exercise, due to the difficulties with bringing such signals off-chip. In this paper, a novel testing strategy is proposed in which machine learning classifiers are used to predict high-frequency performance by combining information from several lower frequency measurements. Promising results are obtained using both direct SVM and indirect MLP classifiers.

Acknowledgements

The authors gratefully acknowledge the financial support of Enterprise Ireland.

References

1. Ferrario, J., Wolf, R., Ding, H.: Moving from mixed signal to RF test hardware development. IEEE Int. Test Conference (2001) 948–956
2. Lau, W. Y.: Measurement challenges for on-wafer RF-SOC test. 27th Annual IEEE/SEMI Int. Elect. Manufact. Tech. Symp. (2002) 353–359
3. Negreiros, M., Carro, L., Susin, A.: Low cost on-line testing of RF circuits. 10th IEEE Int. On-Line Testing Symp. (2004) 73–78
4. Doskocil, D. C.: Advanced RF built in test. AUTOTESTCON '92 IEEE Sys. Readiness Tech. Conf. (1992) 213–217
5. Goff, M. E., Barratt, C. A.: DC to 40 GHz MMIC power sensor. Gallium Arsenide IC Symp. (1990) 105–108.
6. Allen, P. E., Holberg, D. R.: CMOS analog circuit design. 2nd edn. Oxford University Press, Oxford (2002)
7. Vapnil V., Lerner A.: Pattern recognition using generalised portrait method. Automation and Remote Control 24 (1963)
8. Hearst, M.A.: SVMs – a practical consequence of learning theory. IEEE Intelligent Sys. (1998) 18–21
9. Burges, C. J.: A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2 (1998) 121–167
10. Cortes C., Vapnik V.: Support vector networks. Machine Learning 20 (1995) 273–297
11. Haykin, S.: Neural Networks: A comprehensive foundation. 2nd edn. Prentice Hall, New Jersey (1998)
12. McLoone S., Brown M., Irwin G., Lightbody G.: A hybrid linear/nonlinear training algorithm for feedforward neural networks. IEEE Trans. on Neural Networks 9 (1998) 669-684
13. Sjöberg, J., Ljung, L.: Overtraining, regularization, and searching for minimum with application to neural networks. Int. J. Control 62 (1995) 1391-1407
14. Vishwanathan S. V. N., Smola A. J., Murty M. N.: SimpleSVM. Proc. 20th Int. Conf. Machine Learning (2003)