Progress In Electromagnetics Research, Vol. 123, 279–298, 2012

ON LINEAR MAPPING OF FILTER CHARACTERISTIC TO POSITION OF TUNING ELEMENTS IN FILTER TUNING ALGORITHM

J. J. Michalski*

TeleMobile Electronics, Pomeranian Science and Technology Park, Gdynia, Poland

Abstract—This work presents a novel approach in building a multidimensional approximator which is used as a linear operator for mapping the vector of detuned filter characteristic to the vector of deviations of tuning elements. This has been done for the purpose of using it in postproduction filter tuning algorithm. With the use of collected sets of deviations of tuning elements and filter characteristics corresponding to them, the least squares method (LSM) is applied to determine the matrix which realizes the linear mapping between these vectors. The matrix found in this method approximates the vectors of both spaces (filter characteristics and corresponding deviations of tuning elements). In tuning process this matrix is used to determine the vector of tuning element deviations for a given detuned filter characteristic read from Vector Network Analyzer. To increase the "quality" of linear operator filter characteristics are transformed with the use of Karhunen-Loeve transform (Principal Component Analysis). In contrast to non-linear artificial intelligence approximators used in filter tuning and published to-date, this method does not require a time-consuming training process. Filter tuning experiments have been performed and proved the correctness of the presented approach.

1. INTRODUCTION

Microwave filters are devices which are still being designed, developed and improved. Although the filter theory is well established, the authors are still proposing new single-band [1, 2] and multi-band [3–5] filter designs. During the last couple of decades, together with the marked development of telecommunication, filter production has

Received 10 October 2011, Accepted 13 December 2011, Scheduled 24 December 2011
* Corresponding author: Jerzy Julian Michalski (jerzy.michalski@telemobile.net.pl).

reached a very high level and is still growing. In practice, the whole filter production process can be done automatically, but its last phase, filter tuning, is usually performed by human operators. These people are often not engineers and rely on their experience, adjusting a filter by looking directly at the characteristics, and changing the tuning elements until filter characteristics meet specified tuning goals. It takes a very long time for the operators to become experienced tuners and do this work without any support from computer algorithms. As a result, the production costs are significantly increased. During the last couple of decades the researchers reported many interesting methods for postproduction filter tuning, both in frequency [6] and time domain [7]. The method [6] was developed over the years into coupling matrix (CM) synthesis. Filter tuning algorithms based on (CM) are interesting due to the fact that the elements of CM directly correspond to physical filter tuning elements. Novel methods for CM synthesis are still very popular in researches, which is reflected in recently published papers [8–10].

The authors in [11] proposed the algorithm based on fuzzy logic system (FLS) that can be used in tuning of microwave filters. In their concept they used FLS to generate the entries of filter coupling matrix for given scattering parameters at the sampling frequencies. Lately we have proposed novel approaches in filter tuning, with the use of artificial intelligence algorithms (AIA). In [12] we presented a new approach to parallel filter tuning, where an algorithm for a given detuned filter characteristic generates the deviations of all tuning elements at the same time. In that concept we used Artificial Neural Network (ANN) for the purpose of mapping the filter characteristic to the tuning elements of a filter. The ANN was trained using input and output vector samples, collected randomly during the process of controlled filter detuning. The learning samples were collected with the use of a robot [13]. The input vectors were represented by the discretized detuned filter characteristics and the corresponding positions of the tuning elements defining the output vectors. The same algorithm, with the use of a new multidimensional approximator based on Neuro-Fuzzy System (NFS), was investigated and reported in [14]. The efficiency of the NFS tuning algorithm increased significantly which, in turn, made it possible to decrease the number of learning vectors. The consideration on the choice of the proper number of filter characteristic points according to its topology was reported in [15]. The concept of ANN mapping of filter characteristic to the position of tuning elements was further developed in the application to sequential filter tuning [16].

Although such methods can very efficiently model non-linear mapping of filter characteristic and tuning element positions, they require a learning process before they can be used. This process is always time consuming, especially for the learning sets of high cardinality.

In this work we propose a new concept of mapping filter characteristic to the position of tuning elements. Our investigation considers the search for such a matrix linear operator which, after projecting the vector with the elements of discretized filter characteristic, will produce a vector consisting of these tuning element deviations which are responsible for filter detuning. The matrix operator will be determined with the use of Least Squares Method (LSM) [17]. Following the above, the Principal Component Analysis (PCA) is introduced. Next, we present the results of "quality" of the approximation of introduced linear operator for modeling the relationship between vectors of detuned filter characteristics and the vectors of tuning elements deviations. The detailed results demonstrate the influence of PCA transform on the "quality" of the introduced linear approximator and the tuning process. Further, the results of tuning experiments for two filters are presented. Conclusions and discussion are included at the end of the paper.

2. CONCEPT OF LINEAR MAP BETWEEN FILTER CHARACTERISTIC AND DEVIATION OF TUNING ELEMENTS

Let us assume that we have M points representing the filter characteristic denoted by $\mathbf{s} \in \mathbb{R}^M$ and, by $\mathbf{s}_0 \in \mathbb{R}^M$, the characteristic of a properly tuned filter. We shall denote the values of N tuning element positions of a filter by $\mathbf{z} \in \mathbb{R}^N$. Next, we assume the general function dependence between tuning element position $\mathbf{z}$ and filter characteristic as $\mathbf{s} = f(\mathbf{z})$. In the approach published in the work [12] we introduced a non-linear $\mathbf{A}$ operator, which realizes the inverse dependence f^{-1}, i.e., $\mathbf{A} : \mathbf{s} \rightarrow \Delta\mathbf{z}$. In that concept, the sampled detuned reflection characteristics of a filter were used as the input vectors. The corresponding outputs were the normalized screw deviations $\Delta\mathbf{z}$. Operator $\mathbf{A}$ for filter characteristic $\mathbf{s}$ generates a tuning element interval $\Delta\mathbf{z}$, which, after applying current tuning elements $\mathbf{z} + \Delta\mathbf{z}$, makes the filter properly tuned $\mathbf{s}_0 = f(\mathbf{z} + \Delta\mathbf{z})$. For the characteristic of tuned filter $\mathbf{s}_0, \mathbf{A}$ operator generates $\Delta\mathbf{z} = 0$.

In the elaboration [12] the operator $\mathbf{A}$ was constructed with the use of artificial neural network (ANN). In the concept presented here we investigate linear matrix operator $\mathbf{A}$ for the purpose of mapping filter characteristic to deviations of tuning elements. The projection

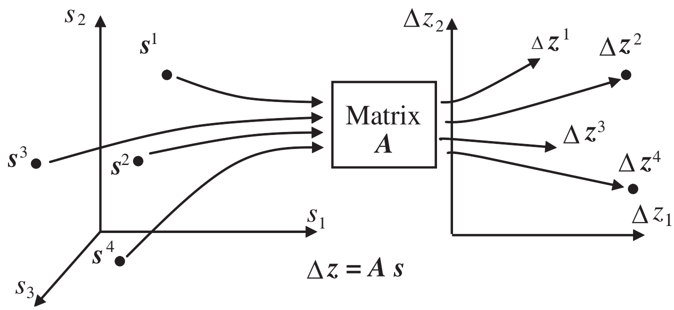


Figure 1. Mapping process between elements $\mathbf{s}$ of space $\mathbb{R}^3$ to the elements $\Delta\mathbf{z}$ of space $\mathbb{R}^2$ with the use of matrix approximator $\mathbf{A} \in \mathbb{R}^{2\times3}$.

of detuned filter characteristic $\mathbf{s}$ onto $\mathbf{A}$ operator yields the tuning position interval $\Delta\mathbf{z} = \mathbf{A}\mathbf{s}$, necessary to have the filter tuned $\mathbf{s}_0 = f(\mathbf{z} + \Delta\mathbf{z})$.

If we assume the $\mathbf{s}$ vector is of a length M and $\Delta\mathbf{z}$ of a length N, the matrix $\mathbf{A}$ will be the element of the space $\mathbb{R}^{M\times N}$. The matrix $\mathbf{A}$ performs the correspondence from the space $\mathbb{R}^M$ to $\mathbb{R}^N$. Above, Fig. 1 presents an exemplary mapping process of the elements from the space $\mathbf{s} \in \mathbb{R}^3$ to the space $\Delta\mathbf{z} \in \mathbb{R}^2$.

2.1. Preparation of Filter Tuning Algorithm

In the concept proposed in this paper the linear operator $\mathbf{A}$ will be described with the use of LSM presented below in Section 3. Additionally, the "outliers elimination" procedure proposed in Section 4 will be applied. The filter characteristics will be transformed with the use of PCA introduced in Section 5. The flow chart with the procedures of preparing the tuning algorithm proposed here is presented below in Fig. 2.

3. LINEAR OPERATOR AND LEAST SQUARES METHOD

In mathematics, linear maps are one of the most interesting correspondences between elements of two spaces. If we consider vector spaces, the map can be realized by matrix operator, which is a linear operator, i.e., the following conditions are fulfilled

$$\begin{cases} \mathbf{A}(\mathbf{x}+\mathbf{y}) = \mathbf{A}(\mathbf{x}) + \mathbf{A}(\mathbf{y}) \\ \mathbf{A}(c\mathbf{x}) = c\mathbf{A}(\mathbf{x}) \end{cases} \tag{1}$$

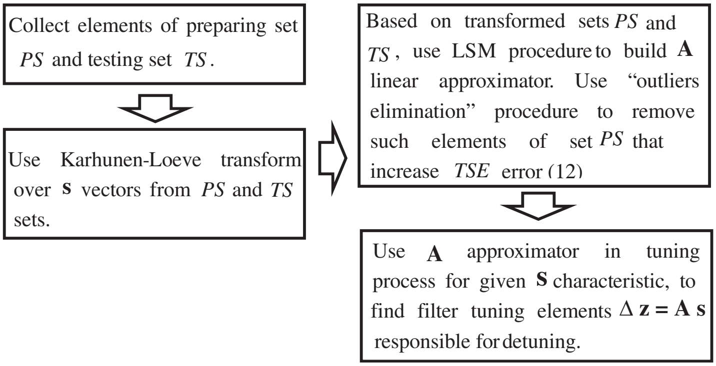


Figure 2. The flow chart with the procedures of preparing the tuning algorithm.

where $\mathbf{A}$ is a matrix linear operator, $\mathbf{x}, \mathbf{y} \in \mathbf{V}$ are any chosen elements of vector space $\mathbf{V}$ and c is any given scalar value. A real matrix of dimension $M \times N$ defines the linear map from the space $\mathbb{R}^N$ to $\mathbb{R}^M$. In the approach proposed here such matrix operator will be applied to map the filter characteristics (one vector space) to tuning element deviations (second vector space). This operator will be defined with the use of a set of sample elements from both vector spaces which are considered.

Let us assume that we have G pairs of vectors $(\mathbf{s}^g, \Delta\mathbf{z}^g)$, where $\mathbf{s}^g \in \mathbb{R}^M$, $\mathbf{S}^g = [s_1^g, s_2^g, \ldots, s_M^g]^T$, and $\Delta\mathbf{z}^g \in \mathbb{R}^N$, $\Delta\mathbf{z}^g = [\Delta z_1^g, z_2^g, \ldots, z_M^g]^T$, for $g = 1, 2, \ldots, G$. The detuned characteristics of a filter and corresponding tuning element deviations are represented by $\mathbf{s}^g$ and $\Delta\mathbf{z}^g$ vectors respectively. We are looking for such a matrix $\mathbf{A}$ which fulfils, for all pairs $(\mathbf{s}^g, \Delta\mathbf{z}^g)$ the following relation

$$\Delta\mathbf{z}^g = \mathbf{A}\mathbf{s}^g \tag{1}$$

where

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & \cdots & a_{1,N} \\ \vdots & \ddots & \vdots \\ a_{M,1} & \cdots & a_{M,N} \end{bmatrix} \tag{2}$$

As we must assume that the number of vector pairs G may be larger than both M and N, the matrix $\mathbf{A}$ will approximate the relation between the elements of space $\mathbf{S}$ and $\Delta\mathbf{Z}$, so for all vector elements of these spaces the following condition will be fulfilled

$$\mathbf{A}\mathbf{s}^g - \Delta\mathbf{z}^g = \mathbf{w}^g \tag{3}$$

In our consideration we are looking for such a matrix $\mathbf{A}$ which would best approximate all the elements of both spaces in the sense of $\min(\mathbf{w}^g)^2$ for all $g = 1, 2, \ldots, G$. In this approach we will use Least Squares Method [17] to find such a matrix.

Let us denote

$$I = \sum_{g=1}^{G} \sum_{m=1}^{G} \left(\sum_{n=1}^{N} a_{m,n} s_n^g - \Delta z_m^g \right)^2 \tag{4}$$

where $a_{m,n}$ are the elements of matrix $\mathbf{A}$ (2).

Following the least squares concept, we are looking for such elements $a_{m,n}$ of $\mathbf{A}$ matrix which minimize the function I defined by (4). The minimum of the sum of squares (4) is found by setting all the gradients to zero $\dfrac{\partial I}{\partial a_{i,j}} = 0$. Since the model contains MN parameters, we have the same number of gradient equations

$$\frac{\partial I}{\partial a_{i,j}} = 2 \sum_{g=1}^{G} \sum_{n=1}^{N} \left(a_{i,n} s_n^g - \Delta z_i^g \right) s_j, \quad i = 1, 2, \ldots, N, \quad j = 1, 2, \ldots, N \tag{5}$$

These equations can be rewritten in the matrix form of the following system of inhomogeneous linear equations

$$\mathbf{C}_{MN\times MN} \mathbf{a}_{MN\times 1} = \mathbf{b}_{MN\times 1} \tag{6}$$

where the main square matrix $\mathbf{C}_{MN\times MN}$ is defined as follows

$$\mathbf{C}_{MN\times MN} = \begin{bmatrix} \mathbf{T}_{N\times N} & 0 & \ldots & 0 \\ 0 & \mathbf{T}_{N\times N} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \mathbf{T}_{N\times N} \end{bmatrix} \tag{7}$$

and consists of L matrixes $\mathbf{T}$ defined as

$$\mathbf{T}_{N\times N} = \begin{bmatrix} \sum\limits_{g=1}^{G} s_1^g s_1^g & \sum\limits_{g=1}^{G} s_2^g s_1^g & \ldots & \sum\limits_{g=1}^{G} s_N^g s_1^g \\ \sum\limits_{g=1}^{G} s_1^g s_2^g & \sum\limits_{g=1}^{G} s_2^g s_2^{qg} & \ldots & \sum\limits_{g=1}^{G} s_N^g s_1^g \\ \vdots & \vdots & \ddots & \vdots \\ \sum\limits_{g=1}^{G} s_1^g s_N^g & \sum\limits_{g=1}^{G} s_2^g s_N^g & \ldots & \sum\limits_{g=1}^{G} s_N^g s_N^g \end{bmatrix} \tag{8}$$

vector $\mathbf{b}$ is of the form

$$\mathbf{b}_{MN\times 1} = \left[\sum_{g=1}^{G} \Delta z_1^g s_1^g \; \sum_{g=1}^{G} \Delta z_1^g s_2^g \; \ldots \; \sum_{g=1}^{G} \Delta z_1^g s_N^g \; \ldots \sum_{g=1}^{G} \Delta z_M^g s_N^g \right]^T \tag{9}$$

and vector $\mathbf{a}_{MN\times 1}$ defines unknown variables; the elements of matrix $\mathbf{A}$ (2).

The system (6) has a solution if the determinant of matrix $\mathbf{C}$ does not equal 0, $\det(\mathbf{C}) \neq 0$. The necessary condition for $\det(\mathbf{C}) \neq 0$ is that

$$G \geq M \tag{10}$$

i.e., the number of vector pairs G is bigger than the length of the vector $\mathbf{s}$ containing discretized filter characteristic. For $G < M$ the rank of matrix $\mathbf{C}$ is lower than MN and the system (6) cannot be solved.

4. MINIMIZATION OF LSM APPROXIMATION ERROR BY ELIMINATION OF OUTLIERS

To build matrix $\mathbf{A}$ we will use G different vector pairs $(\mathbf{s}^g, \Delta\mathbf{z}^g)$, $g = 1, 2, \ldots, G$, collected from a properly tuned filter randomly, with the use of a robot. In general, the dependence between vectors $\mathbf{s}$ and $\Delta\mathbf{z}$ is non-linear, and introducing a linear operator $\mathbf{A}$ to model correspondence between them is a simplification, since matrix $\mathbf{A}$ works as a linear approximator. Because of this simplification some elements, randomly collected, are "unsuitable" candidates to build matrix $\mathbf{A}$. We propose the following method to eliminate such "unsuitable" candidates.

Let us define two independent sets containing pairs of detuned filter characteristics and corresponding deviations of tuning elements. The first set (Preparing Set), denoted by $PS = \{(\mathbf{s}_p^g, \Delta\mathbf{z}_p^g), g = 1, 2, \ldots, G\}$, contains the pairs used in the process of preparing matrix $\mathbf{A}$ according to the procedure presented above in Section 3. The next set (Testing Set), denoted by $TS = \{(\mathbf{s}_T^h, \Delta\mathbf{z}_T^h), h = 1, 2, \ldots, H\}$, contains the pairs which will be used to investigate the approximation quality of matrix $\mathbf{A}$. To measure this quality we introduce two error functions. The first one, PSE — Preparing Set Error (11), tells us how well matrix $\mathbf{A}$ approximates the PS set, i.e., the pairs $(\mathbf{s}_L^g, \Delta\mathbf{z}_L^g)$, which were used in the preparation of matrix $\mathbf{A}$. The next one, Testing Set Error — TSE (12) measures the approximation ability of $\mathbf{A}$, i.e., how "good" is the correspondence between the vectors which are the elements of the TS set, i.e., the pairs $(\mathbf{s}_T^h, \Delta\mathbf{z}_T^h)$, — the elements which were not used in preparing matrix $\mathbf{A}$.

$$PSE = \frac{2K\sum\limits_{g=1}^{G}\sum\limits_{n=1}^{N}\left|\Delta\mathbf{z}_{0p}^g(n) - \Delta\mathbf{z}_{xp}^g(n)\right|}{GN}[u] \quad \text{where } \Delta\mathbf{z}_{xp}^g(n) = \mathbf{A}\mathbf{s}_p^g \tag{11}$$

$$TSE = \frac{2K \sum_{h=1}^{H} \sum_{n=1}^{N} \left| \Delta \mathbf{z}_{0T}^{h}(n) - \Delta \mathbf{z}_{xT}^{h}(n) \right|}{HN} [u] \quad \text{where } \Delta \mathbf{z}_{xT}^{h}(n) = \mathbf{A}\mathbf{s}_{T}^{h} \qquad (12)$$

where G — the number of vectors used in preparing matrix $\mathbf{A}$, $\Delta\mathbf{z}_{0p}^{g}$ is an element of PS — vector containing tuning element deviations — known value, $\Delta\mathbf{z}_{xp}^{g}$ — vector defining the product of projection of detuned filter characteristic onto matrix $\mathbf{A}$, i.e., approximate value of tuning element deviations, H — number of testing pairs in the set TS, $\Delta\mathbf{z}_{0T}^{h}$ is an element of TS (known proper value) — testing tuning element deviations, $\Delta\mathbf{z}_{xT}^{h}$ — tuning element deviation being the product of filter characteristic and matrix $\mathbf{A}$. The value of N is the number of tuning elements of a filter. The value of K is the multiple value of u and defines the maximum tuning element increment in both directions. The value of u defines the minimal angle change in (deg) each of the tuning elements. Both values depend on sensitivity of physical tuning elements and should be chosen experimentally.

An outlier is an element in the data set, which is numerically distant from the rest of the data. In the process of data set approximation, the outliers considerably decrease the quality of approximation. In the case considered in this work, an outlier denotes one pair ($\mathbf{s}_{L}^{i}$, $\Delta\mathbf{z}_{L}^{i}$) from the set which impairs the approximation ability of matrix $\mathbf{A}$. The method of outlier elimination, proposed here, is supposed to check whether the given pair, if removed, will improve the quality of matrix $\mathbf{A}$, i.e., the value of TSE defined by (12). If this criterion is satisfied, this pair is removed from the set PS. All the pairs ($\mathbf{s}_{L}^{i}$, $\Delta\mathbf{z}_{L}^{i}$), $i = 1, 2, \ldots, G$ have to be verified in this way.

5. KARHUNEN-LOEVE TRANSFORM OVER FILTER CHARACTERISTIC — PRINCIPAL COMPONENT ANALYSIS

The concept of filter characteristic transform for the purpose of filter tuning was introduced in [18, 19]. In general, the discrete filter characteristic, symbolised by numbers, can be represented in different forms exposing different features, with the use of mathematical tools called transforms. One of them is the transform called Principal Component Analysis (PCA), also known as the Karhunen-Loeve (KL) transform. It refers to a process whereby a data space is transformed into a feature space. The transformed data representation has exactly the same dimension as the original data. However, the transform is designed in such a way that the data set may be represented by a reduced number of "most effective" features [20, 21].

KL transform can be described as follows. Let $\mathbf{X}$ denote an $n \times m$ dimensional matrix representing the considered data set, n vectors of observations, each of length m. We assume that each column vector $\mathbf{x}_i$ of $\mathbf{X}$ (collecting the information at the same index for all n vectors of observations) has zero mean

$$E[\mathbf{X}_i] = 0, \quad i = 1, 2, \dots m \tag{13}$$

where E is the statistical expectation operator.

Let $\mathbf{q}$ denote a vector of dimension with Euclidean norm of unity and let us define the square matrix $\mathbf{R}$ of dimension $m \times m$, which is the correlation matrix of $\mathbf{X}$

$$\mathbf{R} = E\left[\mathbf{X}\mathbf{X}^T\right] \tag{14}$$

The next step in PCA algorithm is to find eigenvalues and eigenvectors of matrix $\mathbf{R}$, which will fulfil the following equation

$$\mathbf{R}\mathbf{q} = \lambda\mathbf{q} \tag{15}$$

This equation has non-trivial solutions $\mathbf{q} \neq 0$ for certain values of λ, called eigenvalues of matrix $\mathbf{R}$. The corresponding $\mathbf{q}$ vectors are called eigenvectors of matrix $\mathbf{R}$. Due to its symmetry, the correlation matrix $\mathbf{R}$ is characterized by real, nonnegative eigenvalues and the associated eigenvectors are unique. Let the eigenvalues of matrix $\mathbf{R}$ be denoted by $\lambda_1, \lambda_2, \dots, \lambda_m$ and the associated eigenvectors by $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_m$. We may then write the following equation

$$\mathbf{R}\mathbf{Q} = \mathbf{Q}\boldsymbol{\Lambda} \tag{16}$$

where $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_j, \dots \mathbf{q}_m]$ and $\Lambda = \mathrm{diag}[\lambda_1, \lambda_2, \dots, \lambda_j, \dots, \lambda_m]$.

The eigenvalues are arranged in a decreasing order and the corresponding eigenvectors follow this arrangement. The unique eigenvectors create unitary matrix $\mathbf{Q}$ which fulfils

$$\mathbf{Q}^T = \mathbf{Q}^{-1} \tag{17}$$

the Equation (16) can be rewritten in the form known as the orthogonal similarity transformation

$$\mathbf{Q}^T\mathbf{R}\mathbf{Q} = \boldsymbol{\Lambda} \tag{18}$$

which can be written as

$$\mathbf{q}_j^T\mathbf{R}\mathbf{q}_k = \begin{cases} \lambda_j, & k = j \\ 0 & k \neq j \end{cases} \tag{19}$$

The above equations transform the correlation matrix $\mathbf{R}$ into a diagonal matrix of eigenvalues and eigenvectors as follows

$$\mathbf{R} = \sum_{i=1}^{m} \lambda_i \mathbf{q}_i \mathbf{q}_i^T = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^T \tag{20}$$

Having m unit vectors $\mathbf{q}$ we find m projections of the input data vector $\mathbf{x}$

$$\mathbf{a}_j = \mathbf{q}_j^T \mathbf{x} = \mathbf{x}^T \mathbf{q}_j, \quad j = 1, 2, \ldots, m \tag{21}$$

where $\mathbf{a}_j$ are the projections of $\mathbf{x}$ onto the principal directions represented by unit vectors $\mathbf{q}_j$. The $\mathbf{a}_j$ elements are called principal components and have the same dimension m as the input vector $\mathbf{x}$. The unit vectors $\mathbf{q}_j$ represent the basis of data space. The Equation (21) can be considered as a coordinate transformation, according to which a point $\mathbf{x}$ in the data space is transformed into a corresponding point $\mathbf{a}$ in the feature space. To reconstruct the original vector $\mathbf{x}$ from the projections $\mathbf{a}_j$ we have to apply the following equation

$$\mathbf{x} = \mathbf{Q}\mathbf{a} = \sum_{j=1}^{m} \mathbf{a}_j \mathbf{q}_j \tag{22}$$

5.1. Compression Mechanism

In order to reduce the number of dimensions of projection $\mathbf{a}$ we may reduce the number of features needed for effective data representation by discarding those linear combinations (22) which have small variances and retain only those elements which have large $\lambda_1, \lambda_2, \ldots, \lambda_l, l \leq m$ variances of correlation matrix $\mathbf{R}$. To reconstruct the approximate data vector $\check{\mathbf{x}}$ we use the truncated expansion of Equation (22) after l as follows

$$\check{\mathbf{x}} = \sum_{j=1}^{l} \mathbf{a}_j \mathbf{q}_j \tag{23}$$

The numerical experiments demonstrating compression concept, which are described here, can be found in the work [18].

6. NUMERICAL INVESTIGATIONS OF MATRIX A "QUALITY"

In order to verify our concept in practice, numerical investigations have been carried out. In our experiments we used RX and TX filter from 900 MHz diplexer, previously investigated in [15, 16]. The pictures of filters and their topologies are presented in Fig. 3.

For both filters the sets $PS = \{(\mathbf{s}_p^g, \Delta\mathbf{z}_p^g), g = 1, 2, \ldots, G\}$ and $TS = \{(\mathbf{s}_T^h, \Delta\mathbf{z}_T^h), h = 1, 2, \ldots, H\}$ were created with the use of properly tuned filters in the process of random detuning with the use of a robot [13]. In this process all the tuning elements can be

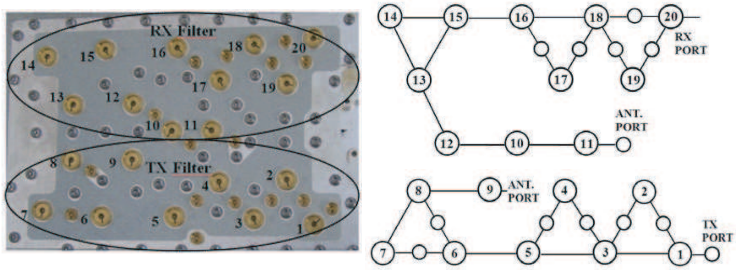


Figure 3. Picture and topology of the RX and TX filters used in the experiments. The small circles represent tunable couplings and cross-couplings. The bigger circles represent cavities. There are no tuning elements for couplings and cross-couplings between the cavities 10–11, 10–12, 12–13, 13–14, 13–15, 15–16, 16–18 in the RX filter and 1–3, 3–5, 5–6, 7–8, 8–9 in the TX filter. Two tuning elements common for both filters, i.e., No. 10 and No. 11, close to antenna port, were assigned to the RX filter.

changed simultaneously. In the experiment we considered only cavities; the couplings and cross-couplings had been pre-tuned and were not changed during the process of collecting the sets. The *PS* set had 1000 elements, 500 collected with $K = 5$, and the next 500 with $K = 10$. The *TS* set had 100 elements, 50 collected with $K = 5$, and the next 50 with $K = 10$. For the purpose of our experiments, the value $u = 18°$, which gives us the maximum screw change $\mp 180°$ during the process of collecting the elements of sets *PS* and *TS*.

Below, in Tables 1 and 2 we present Testing Set Error — *TSE* defined by (12) of **A** approximator for the RX filter, obtained with different configurations: for a different G number of the elements of *PS* set and a different number of filter characteristic points M, for filter characteristics without (first number in a cell) and with (second number in a cell) "outliers elimination" procedure. Since the results for TX filter are similar, they have been omitted.

Since for some combinations of G and M values, condition (10) is not fulfilled, the system (6) cannot be solved, and these cells in the tables are not filled in.

To check the influence of PCA filter characteristics transform on the "quality" of **A** approximator, all filter characteristics of a filter were then PCA transformed and cut after the 80th element (the original characteristics in frequency domain had 512 elements). Above, Table 2 presents the Testing Set Error for **A** approximator prepared with PCA transformed filter characteristics. Similarly to previous results, the first number is the error prior to "outlier elimination", the second number is the error after the "outlier elimination" procedure.

Table 1. Testing Set Error *TSE* for RX filter for different number *G* of *PS* set elements and different number *M* of the samples of filter characteristic. Results without and with “outlier elimination”.

$G\backslash M$	512	256	128	64	32	16	8
10							11.59 2.298
30						1.573 1.414	1.778 1.531
50					0.739 0.692	1.365 1.133	1.754 1.503
100				0.679 0.585	0.591 0.518	1.267 0.933	1.748 1.456
150			1.296 1.113	0.565 0.479	0.601 0.465	1.195 0.919	1.635 1.381
300		0.922 0.815	0.488 0.425	0.440 0.356	0.444 0.391	0.907 0.805	1.368 1.286
600	0.852 0.745	0.497 0.399	0.422 0.336	0.394 0.318	0.419 0.361	0.889 0.780	1.368 1.275
1000	0.489 0.352	0.404 0.293	0.378 0.286	0.373 0.299	0.409 0.349	0.872 0.762	1.387 1.273

Table 2. Testing Set Error *TSE* with PCA for RX filter for different number *G* of *PS* set elements and different number *M* of the samples of filter characteristic. Results without and with “outlier elimination”. Compression point $N = 80$.

$G\backslash M$	512		256		128	
150					0.751	0.588
300			0.471	0.394	0.444	0.383
600	0.419	0.343	0.418	0.328	0.398	0.332
1000	0.375	0.289	0.367	0.289	0.370	0.287

While analysing all the obtained results, we can observe that the application of the proposed “outlier elimination” procedure results in significant decrease of *TSE* error for both cases, with and without PCA transform. The application of PCA transform decreases both errors considerably, especially for preparing set *PS* of a small *G* number of vector pairs.

7. THE "QUALITY" OF LINEAR MAPPING WITH AND WITHOUT PCA

In general, the dependence between filter reflection characteristic and tuning element deviations, i.e., elements $\mathbf{s}^g$ and $\Delta\mathbf{z}^g$, is not linear. Our simplification $\Delta\mathbf{z}^g = \mathbf{A}\mathbf{s}^g$, where $\mathbf{A}$ is a linear matrix, is limited to filters which are detuned within a certain range, looking from the positions of these tuning elements which have been properly set.

In the following part of the paper, we will investigate the influence on the "quality" of the linear approximator $\mathbf{A}$ for the filters detuned with a different level, i.e., for the filters detuned with a different K value. In the experiment we will calculate the preparing set error PSE and the testing set error TSE for the vectors $\mathbf{s}^g$ and $\Delta\mathbf{z}^g$ taken from the RX filter, mapped using $\mathbf{A}$ matrix, with and without the outlier elimination procedure described above. Additionally, we will compare the error data with the results obtained with the use of ANN approximator introduced in [12]. In the Tables 3 and 4 below we have presented both PSE and TSE for used approximators. The numbers in each cell correspond to the following approximation method: "A" is the error for linear $\mathbf{A}$ approximator without "outlier elimination" procedure, "A OE" is the error for linear $\mathbf{A}$ approximator with "outlier elimination" procedure presented above, "PCA OE" is the error for $\mathbf{A}$ with PCA transformed filter characteristics cut after the 80th element with "outliers elimination" procedure, "ANN" is the error with the use of ANN non-linear approximator [12]. In the experiment the PS had 500 vector pairs, which were represented by discrete filter characteristics of 64 complex points, which gave, in total, the $M = 128$ length of vectors $\mathbf{s}^g$.

ANN used in the experiment was of the following topology: No. of input neurons $WI = 128$, No. of neurons in hidden layer $WU = 51$, No. of output neurons $WO = 11$ for RX filter and $WO = 9$ for TX filter (equal to the number of tuning elements). The learning process for this approximator was performed in 5000 learning epochs, which

Table 3. Preparing Set Error — PSE for RX filter for different levels of detuning K, $1u = 18°$.

Err\\K	5	10	20	30
A (u)	0.216	0.881	3.576	6.981
A OE (u)	0.095	0.401	1.660	3.213
PCA OE (u)	0.070	0.348	1.507	3.324
ANN (u)	0.213	0.410	1.115	2.463

Table 4. Testing Set Error —*TSE* for RX filter for different levels of detuning K, $1u = 18°$.

Err\\K	5	10	20	30
A (u)	0.298	1.203	4.402	9.861
A OE (u)	0.205	0.825	3.088	7.386
PCA OE (u)	0.191	0.788	3.035	7.367
ANN (u)	0.242	0.544	2.120	5.288

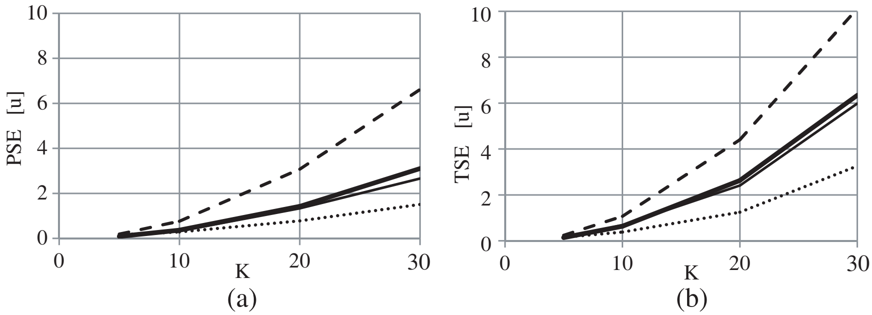


Figure 4. Preparing set error (a) and testing set error (b) for RX filter. Parameter K indicates filter detuning. LSM approximation without outlier elimination — dashed line, LSM approximation with outlier elimination — bold solid line, LSM PCA approximation with outlier elimination — thin solid line, ANN approximation [12] — dotted line.

took, in each experiment, about 26 minutes for each filter.

Above in Fig. 4, we have presented Preparing Set Error and Testing Set Error in the function of value K, which is the measure of the filter detuning level.

While analysing the error curves, we can conclude that, for the considered filter, the mapping between filter reflection characteristic vectors $\mathbf{s}^g$ and tuning element deviations $\Delta\mathbf{z}^g$ can be successfully approximated by the proposed linear matrix operator $\mathbf{A}$. We see that, in the cases where $K \leq 10(u)$, our liner approximator $\mathbf{A}$ gives us similar or even better levels *PSE* of and *TSE* errors, in comparison to ANN non-linear approximator.

7.1. Stability of s Vector Projection onto A approximator with and without PCA Transform

In this part of the paper we investigate the stability of results which are the tuning element vectors $\Delta\mathbf{z} = \mathbf{A}\mathbf{s}$ for the case of $\mathbf{A}$ approximator, created with and without PCA transform. We will examine the influence of a "slight" change in filter characteristic $\mathbf{s}$ on $\Delta\mathbf{z}$ for these two considered approximators.

Below in Fig. 5(a) we have presented two filter characteristics (real and imaginary part), which are very similar. The difference between them is presented in Fig. 5(b).

The solid line is the filter characteristic of a properly tuned filter, which was used in the process of preparing $\mathbf{A}$. The characteristic denoted by the dotted line is used as a probe to verify the stability of the $\Delta\mathbf{z} = \mathbf{A}\mathbf{s}$ values.

In this experiment we investigated two linear approximators: $\mathbf{A}$, which was prepared based on 512 elements of filter characteristics in frequency domain and $\tilde{\mathbf{A}}$ prepared with PCA transformed $\mathbf{s}$ vectors cut after 80th element. Table 5 presents two vectors of tuning element deviations $\Delta\mathbf{z}$, obtained with $\mathbf{A}$ and $\tilde{\mathbf{A}}$ for filter characteristic slightly detuned from Fig. 4(a).

Although the *PSE* and *TSE* for $\mathbf{A}$ approximators with and without PCA transform are at a similar level, the results of $\Delta\mathbf{z} = \mathbf{A}\mathbf{s}$ can differ significantly for the $\mathbf{s}$ filter characteristics, which was not used in the preparing process of $\mathbf{A}$. Moreover, if we calculate $\Delta\mathbf{z} = \mathbf{A}\mathbf{s}$ "on line" for $\mathbf{s}$ taken directly from Vector Network Analyser, the values of $\Delta\mathbf{z}$ are very unstable and change within the range $\mp 1(u)$ from the mean values. This situation looks completely different for $\tilde{\mathbf{A}}$ prepared with PCA, where the $\Delta\mathbf{z} = \tilde{\mathbf{A}}\mathbf{s}$ are very stable and show proper values.

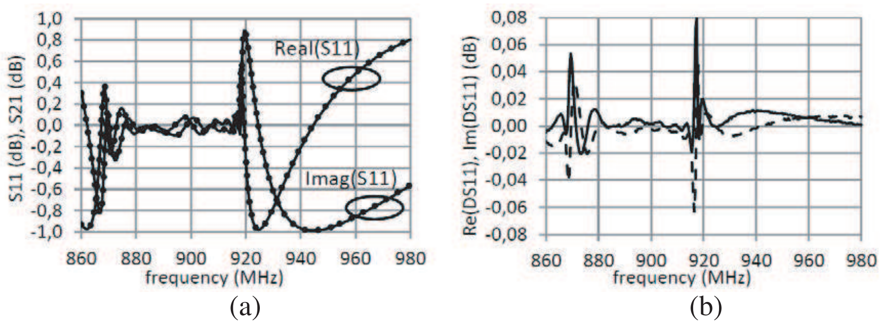


Figure 5. (a) Imaginary and real part of tuned and slightly detuned reflection characteristics of RX filter. The characteristic (tuned) used in the preparing process of $\mathbf{A}$ approximator — solid line, the characteristic (slightly detuned) used to verify the "quality" of $\mathbf{A}$ approximator — dotted line. Fig. 5(b) shows the difference between curves DS11 from Fig. 5(a). Solid line presents the real part and dashed line represents the imaginary part difference.

Table 5. Tuning element deviation $\Delta\mathbf{z}$ being the product of $\mathbf{A}$ and $\tilde{\mathbf{A}}$ approximator and filter characteristics $\mathbf{s}$ slightly detuned from Fig. 4(a). Values are given in $u = 18°$ units.

Tuning Element No.	1	2	3	4	5	6	7	8	9	10	11
$\Delta\mathbf{z}(\mathbf{m})$ $(\mathbf{u})$ **for A without PCA**	−3	−1	2.1	1.1	−1	1.8	2.2	0.8	0.3	−2	2
$\Delta\tilde{\mathbf{z}}(\mathbf{m})$ $(\mathbf{u})$ **for** $\tilde{\mathbf{A}}$ **with PCA**	−0.08	0.01	−0.01	0.03	0.04	−0.04	−0.01	0.07	−0.01	0.02	0.03

The possible reason for this can be the high dimension of matrix $\mathbf{A}$ (512 columns) without PCA, which can cause numerical errors in calculating $\Delta\mathbf{z}$.

8. FILTER TUNING EXPERIMENT

Filter tuning experiments were performed with the use of $\tilde{\mathbf{A}}$ approximator, which was prepared based on the *PS* and *TS* sets investigated in Section 6. The initial vectors $\mathbf{s}_p^g$ of the length 512 were PCA transformed and shortened to 20 first elements. This gave us the reduction of $\mathbf{A}$ from 512 to 20 columns. Then $\mathbf{A}$ approximator was constructed with the use of LSM.

Using the presented tuning concept, for $\mathbf{s}$ characteristic read from VNA, the tuning element deviation vector is calculated as $\Delta\mathbf{z} = \tilde{\mathbf{A}}\mathbf{s}$. By setting the tuning elements one by one in a proper position we set the zero values for elements in $\Delta\mathbf{z}$ vector. During the tuning, the order of the screws can be chosen randomly. The tuning ends if all the elements in $\Delta\mathbf{z}$ vector equal zero, $\Delta\mathbf{z} = 0$.

Tables 6 and 7 present $\Delta\mathbf{z}$ vectors for RX and TX filters before and after tuning; the corresponding reflection and transmission characteristics are presented in Figs. 6(a), (b).

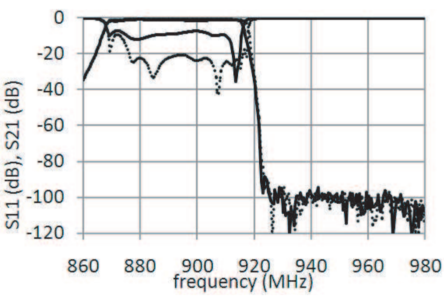
While analyzing the obtained tuning results we can observe that both filters were very well tuned. Normally the tuning is finished after one iteration (each tuning element is positioned only once). In some cases, after one tuning iteration it is necessary to run another iteration if, for some tuning elements, the value of $\Delta\mathbf{z}\,(i)$ does not equal zero.

Table 6. Tuning element deviation $\Delta\mathbf{z}$ which is the product of $\mathbf{A}$ approximator and filter characteristic $\mathbf{s}$, before and after tuning for RX filter.

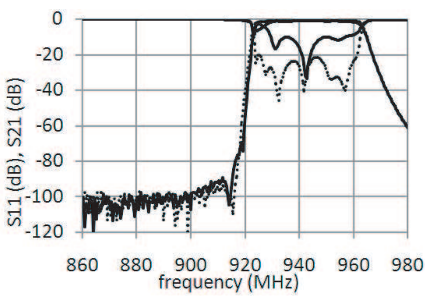
Tuning Element No.	1	2	3	4	5	6	7	8	9	10	11
$\Delta\mathbf{z}$ (m) before tuning (u)	−1	12	3	−6	8	−6	0	−7	6	6	−5
$\Delta\mathbf{z}$ (m) after tuning (u)	0	0	0	0	0	0	0	0	0	0	0

Table 7. Tuning elements deviation $\Delta\mathbf{z}$ which is the product of $\mathbf{A}$ approximator and filter characteristic $\mathbf{s}$, before and after tuning for TX filter.

Tuning Element No.	1	2	3	4	5	6	7	8	9
$\Delta\mathbf{z}$ (m) before tuning (u)	−7	−7	7	0	−5	−5	7	2	8
$\Delta\mathbf{z}$ (m) after tuning (u)	0	0	0	0	0	0	0	0	0



(a)



(b)

Figure 6. Transmission and reflection characteristics for (a) RX filter and (b) TX filter. solid lines show scattering curves before tuning and dotted curves present the filter condition after tuning.

9. CONCLUSIONS

In this work th,e least squares method has been employed to realize linear mapping of detuned filter characteristic to tuning element deviations, for the purpose of applying it in filter tuning algorithm. Such mapping from one vector space to another, which is, in fact, non-linear, can be successfully approximated by a linear matrix operator found with the use of least squares method. This operator is built based on vector pairs which are PCA transformed filter characteristics and tuning element deviations corresponding to them. This approach, in contrast to non-linear artificial intelligence approximators presented earlier in literature [12, 14–16, 18, 19], does not require a training process, and thus significantly reduces the time of filter tuning algorithm customization. For the filters presented in the paper, the time needed for preparing **A** linear PCA approximator was only 10 seconds which, in comparison to 26 minutes for ANN [12] approximator, is a significant advantage. Moreover, the application of PCA transform over filter characteristics ensures stable projection results for $\Delta\mathbf{z} = \tilde{\mathbf{A}}\mathbf{s}$, while $\mathbf{s}$ is read "on line" from Vector Network Analyser. The filter tuning experiments, performed on two filters of high order, show the reliability of the presented approach.

ACKNOWLEDGMENT

This work was supported by Polish National Ministry for Science & Higher Education (Decision nr. 736/N-COST/2010/0), under the project name "New optimization methods and their investigation for the application in physical microwave devices which require tuning", performed within the COST Action RFCSET IC0803.

REFERENCES

1. Cui, D., Y. Liu, Y. Wu, S. Li, and C. Yu, "A compact bandstop filter based on two meandered parallel-coupled lines", *Progress In Electromagnetics Research*, Vol. 121, 271–279, 2011.
2. Kung, C. Y., Y. C. Chen, S. M. Wu, C. F. Yang, and J.-S. Sun, "A novel compact 2.4/5.2 GHz dual wideband bandpass filter with deep transmission zero," *Journal of Electromagnetic Waves and Applications*, Vol. 25, Nos. 5–6, 617–628, 2011.
3. Safia, O. H. A., A. A. Omar, and M. C. Scardelletti, "Design of dual-band bandstop coplanar waveguide filter using uniplanar series-connected resonators," *Progress In Electromagnetics Research Letters*, Vol. 27, 93–99, 2011.
4. Huang, C. Y., M. H. Weng, C. Y. Hung, and S. W. Lan, "Design of a dual-band bandpass filter for GSM and direct sequence ultra-wideband communication systems," *Journal of Electromagnetic Waves and Applications*, Vol. 25, Nos. 11–12, 1605–1615, 2011.
5. Hu, J. P., G. H. Li, H. P. Hu, and H. Zang, "A new wideband triple-band filter using SIR," *Journal of Electromagnetic Waves and Applications*, Vol. 25, No. 16, 2287–2295, 2011.

6. Thal, H. L., "Computer-aided filter alignment and diagnosis," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 26, No. 12, 958–963, Dec. 1978.
7. Dunsmore, J., "Tuning band pass filters in the time domain," *IEEE MTT-S Int. Microwave Symp. Digest*, 1351–1354, 1999.
8. Corrales, E., P. de Paco, and O. Menendez, "Direct coupling matrix synthesis of bandstop filters," *Progress In Electromagnetics Research Letters*, Vol. 27, 85–91, 2011.
9. Wang, R., J. Xu, C. L. Wei, M.-Y. Wang, and X.-C. Zhang, "Improved extraction of coupling matrix and unloaded Q from S-parameters of lossy resonator filters," *Progress In Electromagnetics Research*, Vol. 120, 67–81, 2011.
10. Wang, R. and J. Xu, "Extracting coupling matrix and unloaded Q from scattering parameters of lossy filters," *Progress In Electromagnetics Research*, Vol. 115, 303–315, 2011.
11. Miraftab, V. and R. R. Mansour, "A robust fuzzy-logic technique for computer-aided diagnosis of microwave filters," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 52, No. 1, 450–456, Jan. 2004.
12. Michalski, J. J., "Artificial neural networks approach in microwave filter tuning," *Progress In Electromagnetics Research M*, Vol. 13, 173–188, 2010.
13. Intelligent automatic filter tuning tool (IAFTT) is registered with "Priority certificate # 2516" (Hannover Cebit 2007), and patent pending "European Patent Application No. P382895 assigned by Polish National Patent Office".
14. Kacmajor, T. and J. J. Michalski, "Neuro-fuzzy approach in microwave filter tuning," *IEEE MTT-S Int. Microwave Symp. Digest*, Baltimore, 2011.
15. Michalski, J. J., T. Kacmajor, J. Gulgowski, M. Mazur, "Consideration on artificial neural network architecture in application for microwave filter tuning," *PIERS Proceedings*, 1313–1317, Marrakesh, Morocco, Mar. 20–23, 2011.
16. Michalski, J. J., "Inverse Modeling in Application for Sequential Filter Tuning," *Progress In Electromagnetics Research*, Vol. 115, 113–129, 2011.
17. Wolberg, J., *Data Analysis Using the Method of Least Squares: Extracting the Most Information from Experiments*, Springer, 2005, ISBN 3540256741.
18. Kacmajor, T. and J. J. Michalski, "Principal component analysis in application for filter tuning algorithms," *IMWS Conference,*

Proceedings, Sitges, Barcelona, Spain, September 15–16, 2011.

19. Michalski, J. J. and T. Kacmajor, "Filter tuning algorithm with compressed reflection characteristic by Daubechies D4 wavelet transform," *Proceedings of the 41st European Microwave Conference*, 778–781, Manchester, UK, Oct. 9–14, 2011.
20. Pearson, K., "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine*, Vol. 2, No. 6, 559–572, 1901.
21. Haykin, S., *Neural Networks and Learning Machines*, 3rd edition, Prentice Hall, Upper Saddle River, NJ, 2009.