Technical University of Cluj-Napoca

Prof. Gabriel OLTEAN, Ph.D.

HABILITATION THESIS

Computational Intelligence Techniques in the Analysis and Design of Electronic Circuits and Systems

Faculty of Electronics, Telecommunications and Information Technology

Cluj-Napoca

2016

Contents

List of figures

List of tables

Abstract

This work presents my scientific, academic and professional activity since 2002, when the Technical University of Cluj-Napoca conferred me the academic title of Doctor (Ph.D.) in the field of Electronics and Telecommunications Engineering, with Magna Cum Laude distinction.

My research interests primarily consisted in the investigation and exploitation of the possibilities offered by computational intelligence (CI) techniques as fuzzy logic (FL), genetic algorithms (GA), and artificial neural networks (ANN) to address various problems/issues in the field of electronics. My scientific achievements can be organized under some areas of interests, exposed in the "*Scientific achievements*" section.

Automatic design optimization of electronic circuits using CI techniques

This area is related with my doctoral research, where I developed circuit design optimization algorithms, using fuzzy logic. In the post-doctoral research, my contribution was the development of a hybrid intelligent method for the design optimization of analog modules. The method combines some qualities of FL and GA: flexibility to formulate the objective functions and a known range of values, using fuzzy sets; low computational complexity and high accuracy in computing circuit performances by using fuzzy systems; efficiency to search complex solution spaces without being trapped in local minima, using GA.

The proposed method was implemented to act either as single objective optimization or real multiobjective optimization, which generates solutions on the Pareto frontier. The implementation was extended so that a standard circuit simulator was included in the optimization loop to evaluate the circuit performances.

The main results were published in 4 papers, 2 of them included in ISI database.

Functional modeling of analog modules using fuzzy systems

The objective of this research area was to develop a modeling procedure and functional models for some basic analog modules. The proposed modeling procedure is based on fuzzy logic systems (FLS) that are automatically build using supervised training and data sets previously obtained by circuit simulations.

The fuzzy functional models describe the input voltage - output voltage relation in terms of amplitude, frequency and phase-shift and include the temperature effect. All over the frequency and temperature ranges, the models are able to provide the output voltage for a sinusoidal input waveform that keeps the amplifier in its linear region. At low frequency, the models are able to provide the output for a sinusoidal / triangular input, even if the amplifier enters the nonlinear regions.

The results proved the high accuracy and the efficiency of the fuzzy models. These models can be very useful in the design / verification of complex systems at system level where the simulation is performed at behavioral level and tools like Matlab/Simulink are often used.

This research was partially supported by a national research grant, to which I was the grant manager. The main results were published in a book and in 5 papers, 2 of them included in ISI database.

Efficient waveform prediction in electronic systems using CI techniques

My contribution was to develop data-driven methods to build fast and simple to evaluate, but also accurate metamodels, capable of generating not-yet simulated waveforms as a function of different values of the parameters of the system. The novelty is that this represents the first systematic approach of generating signals/waveforms in different points of a complex system using cheap and fast substitutes for extensive simulation.

The method consists of describing the waveform family by a reduced set of features (relevant coefficients of Fourier / wavelet transform), training an ANN to generate this set of features, and finally generating the waveform, using the inverse transform, for any values of input parameters.

Two kind of metamodels were developed: Fourier transform based metamodel, that uses a simple selection of relevant coefficients and optimal wavelet transform based metamodel that uses a GA optimization to detect optimal wavelet transform and to select the most relevant coefficients.

These cost-effective, but reliable metamodels empower the design team to perform extensive analyses of system response, under a theoretically infinite number of different values of input parameters, in a timely manner.

The research was initiated in collaboration with Infineon Technologies, who provides us with a series of industrial early-stage simulation data. The main results were included in a dissertation thesis and published in 4 papers, 2 of them included in ISI database.

Other applications of CI techniques

Other issues that I addressed with computational intelligence technique were: pattern recognition using FLS and ANN, control systems using a FLS as controller, alarming events detection based on audio signal recognition using ANN, and decision making systems based on FLS.

The "*Academic and professional achievements*" section reviews my activities and contributions to the didactic and institutional development components.

Regarding the didactic component, my main achievements were: development and improvement of the subjects I am in charge of, introducing new subjects in the curricula, implication in students' graduation process, publishing 3 books, 7 application manuals, and 3 papers in the education domain (1 in the ISI database).

For the institutional development component the emphasis is on the administrative positions held, representation in national structures, and collaboration with the industry.

The next section "*Management of scientific and academic activities*", refers to my capacity to coordinate research teams and organize and manage teaching and research activities.

For the academic component there were considered: students' appreciation for my teaching activity, supervised diploma projects and master theses and their appreciation by the graduation committees, coordination of students for SSET competition and the received awards.

The scientific component addresses coordination of research activities and research teams in 2 national research grants and my involvement in doctoral programs, especially by co-advising 4 Ph.D. students (we co-authored 11 papers, 6 of them included in the ISI database).

The section "*Career development plan*" brings in my future development plan. My career development relies on all components: didactic, scientific, and institutional. I identified 7 goals to be fulfilled: Orientation towards a student-centred learning; Development of research-informed teaching; Modernization and internationalisation of subjects/study programs; Contribution to knowledge development through fundamental / applied research; Increasing the research visibility; Development of resources; Development of collaborations with industry.

The final section, "*References*", contains the list of the 141 references, out of which I have authored / co-authored 42.

Rezumat

Lucrarea de față prezintă activitatea științifică, academică și profesională, începând din anul 2002 când Universitatea Tehnică din Cluj-Napoca mi-a conferit titlul științific de Doctor în domeniul Inginerie electronică și telecomunicații cu distincția “Magna Cum Laudae”.

Principalele interese în cercetare au constat în investigarea și exploatarea posibilităților oferite de tehnicile de inteligență computațională (IC) precum logica fuzzy (LF), algoritmi genetici (AG) și rețele neuronale artificiale (RNA), pentru a rezolva diferite probleme/aspecte din domeniul electronicii. Realizările științifice au fost organizate pe câteva domenii de interes în secțiunea “*Realizări științifice”*.

Optimizarea proiectării automate a circuitelor electronice utilizând tehnici de IC

Acest domeniu este corelat cu cercetarea doctorală, unde am dezvoltat algoritmi de optimizare a proiectării circuitelor utilizând logica fuzzy. În cercetarea post-doctorală, contribuția a constat în dezvoltarea unei metode inteligente hibride pentru optimizarea proiectării modulelor analogice. Metoda combină avantaje ale LF și AG: flexibilitate în formularea funcțiilor obiectiv și cunoașterea domeniului de valori; complexitate redusă și precizie ridicată în determinarea performanțelor circuitului, utilizând sisteme fuzzy; explorare eficientă a spațiului soluțiilor, fără blocare în minime locale, utilizând AG.

Metoda a fost implementată pentru optimizare cu un singur obiectiv și pentru optimizare multiobiectiv reală, când generează soluții pe frontul Pareto. Implementarea a fost extinsă pentru evaluarea performanțelor utilizând un simulator inclus în bucla de optimizare.

Principalele rezultate au fost publicate în 4 articole, 2 fiind în baza de date ISI.

Modelarea funcțională a modulelor analogice utilizând sisteme fuzzy

Obiectivul a constat în dezvoltarea unei proceduri de modelare și a unor modele funcționale pentru module analogice de bază. Procedura se bazează pe utilizarea de sisteme cu logica fuzzy (SLF) construite automat prin instruirea supervizată și seturi de date numerice.

Modelele funcționale fuzzy descriu relația intrare - ieșire în termenii amplitudinii, frecvenței și defazajului, incluzând și efectul temperaturii. Modelele furnizează tensiunea de ieșire pentru o tensiune de intrare sinusoidală care menține amplificatorul în regiunea liniară, în întreg domeniul de frecvențe. La frecvențe joase, modelele pot furniza tensiunea de ieșire chiar dacă amplificatorul intra în regiunile neliniare.

Rezultatele au confirmat eficiența și gradul ridicat de precizie al modelelor fuzzy. Acestea pot fi foarte utile în proiectarea / verificarea sistemelor complexe la nivel de sistem, unde simularea este realizată la nivel comportamental și instrumente precum Matlab/Simulink sunt utilizate adeseori.

Cercetare a fost susținută printr-un proiect de cercetare național, la care am fost director. Principalele rezultate au fost publicate într-o carte și în 5 articole, 2 fiind în baza de date ISI.

Predicția eficientă a formelor de undă în sisteme electronice utilizând tehnici de IC

Contribuția mea a constat în dezvoltarea a două metode bazate pe date numerice pentru construirea de metamodele rapide și simplu de utilizat, dar precise, capabile să genereze forme de undă noi, în funcție de diferite valori a parametrilor sistemului. Noutatea constă în faptul ca aceasta este prima abordare sistematică de generare a semnalelor în diverse puncte ale unui sistem complex utilizând un substitut ieftin și rapid a unei simulări extensive.

Metodele utilizează caracterizarea familiei de forme de undă printr-un set redus de trăsături (coeficienții relevanți ai transformării Fourier / wavelet), instruirea unei RNA pentru a genera setul de trăsăturii și, în final generarea formei de undă, utilizând transformata inversă.

Au fost dezvoltate două tipuri de metamodele: metamodele bazate pe transformata Fourier, cu selectare simplă a coeficienților și metamodele bazate pe transformarea wavelet optimă, cu optimizare cu AG pentru găsirea transformatei wavelet optime și selecția coeficienților.

Aceste metamodele eficiente oferă proiectantului posibilitatea de a realiza în timp util analize extensive ale sistemului, teoretic pentru un număr infinit de valori a parametrilor de intrare.

Cercetarea a fost inițiată în colaborare cu Infineon Technologies, care a furnizat o serie de date industriale, din etape de simulare timpurie a unui sistem electronic. Principalele rezultate au fost incluse într-o teză de disertație și publicate în 4 articole, 2 fiind în baza de date ISI.

Alte aplicații ale tehnicilor de IC

Alte aplicații realizate prin utilizarea tehnicilor de IC au fost: recunoaștere de forme utilizând SLF și RNA, sisteme de control utilizând SLF, detecția evenimentelor alarmante bazată pe recunoașterea semnalelor audio cu RNA și sisteme de luare a deciziei bazate pe SLF.

Secțiunea „*Realizări academice și profesionale”* trece în revistă principalele activități și contribuții referitoare la componentele didactic și dezvoltare instituțională.

Principalele realizări au fost: dezvoltarea și perfecționarea disciplinelor la care sunt responsabil, introducerea unor noi discipline în planurile de învățământ, implicarea în procesul de absolvire al studenților, publicarea a 3 cărți, 7 manuale de aplicații și 3 articole în domeniul educațional (1 fiind în baza de date ISI).

Referitor la dezvoltarea instituțională sunt menționate pozițiile administrative deținute, reprezentarea în structuri naționale și colaborările cu mediul industrial.

Secțiunea „*Managementul activităților de cercetare și academice*” se referă la capacitatea de a coordona echipe de cercetare, de a organiza și gestiona activități didactice şi de cercetare.

Pentru componenta academică sunt menționate: aprecierea studenților asupra activității mele didactice, conducerea de lucrări de absolvire ale studenților și coordonarea unor studenți pentru concursul SSET, precum și premiile obținute.

Referitor la managementul cercetării se menționează coordonarea activităților și echipelor de cercetare în două proiecte de cercetare naționale la care am fost director, precum și implicarea în programe doctorale, în special prin contribuția la coordonarea a 4 doctoranzi, cu care am publicat în colaborare 11 articole (6 fiind în baza de date ISI).

Planul de dezvoltare a carierei, expus în secțiunea „*Plan de dezvoltare a carierei*”, se bazează pe realizarea următoarelor scopuri: Orientarea înspre instruirea centrată pe student; Integrarea rezultatelor cercetării în activitatea didactică; Modernizarea și internaționalizarea unor discipline/programe de studiu; Contribuții la dezvoltarea cunoașterii prin cercetare; Creșterea vizibilității rezultatelor cercetării; Dezvoltarea resurselor; Dezvoltarea colaborării cu mediul industrial.

Ultima secțiune „*Bibliografie*” conține lista celor 141 referințe, 42 fiind publicații proprii în calitate de autor / coautor.

Scientific achievements

The research activity presented in this work refers to the period starting in 2002, when the Technical University of Cluj-Napoca conferred me the academic title of Doctor (Ph.D.) in the field of Electronics and Telecommunications with Magna Cum Laude distinction. The title of the Ph.D. thesis was "Fuzzy Logic in Analysis and Design of some Electronic Circuits" and it presented the development of fuzzy logic based methods for automatic analysis and design of some electronic circuits.

After concluding the Ph.D. thesis, the research interests primarily consisted in the investigation and exploitation of the possibilities offered by computational intelligence (CI) techniques as fuzzy logic, genetic algorithms, and neural networks to address various problems/issues in the field of electronics. As a consequence, my scientific achievements can be organized under the following areas of interests:

- Automatic design optimization of electronic circuits using CI techniques;
- Functional modeling of analog modules using fuzzy systems;
- Efficient waveforms prediction in electronic systems using CI techniques;
- Other applications of CI techniques.

1. Automatic design optimization of electronic circuits using CI techniques

Bio-Inspired Computing Technologies led to the spectacular progress of the Computational Intelligence (CI) nowadays and to its implementations in form of Hybrid Intelligent Systems [Neg09]. Computational intelligence has been an outstanding success in the engineering domain, particularly in the electronic design. Over the last years improved techniques have raised the productivity of designers to a remarkable degree. Indeed in the area of digital, analog, radio-frequency, and mixt-signal engineering there is a focused effort on trying to automate all levels of design flow of electronic circuits, a field where it was long assumed that progress demanded a skilled designer's expertise [Fak15].

Most of the knowledge-intensive and challenging design effort spent in such systems design is due to the analog building blocks [Bal04]. Analog design has been traditionally a difficult discipline of integrated circuit design. In circuit design optimization, a circuit and its performance specifications are given and the goal is to automatically determine the device sizes in order to meet the given performance specifications while minimizing a cost of objective function, such as a weighted sum of the active area or power dissipation [Bag07]. This is a difficult and critical step for several reasons: 1) most analog circuits require a custom optimized design; 2) the design problem is typically underconstrained with many degrees of freedom; and 3) it is common that many (often conflicting) performance requirements must to be taken into account, and tradeoffs must be made that satisfy the designer [Rut07].

Consequently, the development of CAD tools at the cell-level, that automate and speed up the design process of analog portions of circuits and systems remains as an active research area in both industry and academia [Bal04], [Bao08], [Tak09], [Per10], [Jaf10].

1.1. Previous work

The research activity in the field of fuzzy logic applications in the automatic analyses and design of analog circuits was the subject of my Ph.D. thesis [Olt02a], and also the research theme of a national research grant: "Development of a computer aided design tool for design optimization of analog circuit design using fuzzy techniques" CNCSIS, 2001-2002, grant manager: Gabriel Oltean.

Works showing the possibility of application of fuzzy logic in computer aided design of electronic circuits started to appear in late 1980s and early 1990s. An argument for fuzzy logic application in CAD is derived from the nature of the algorithm used for solving design problems. The majority of algorithms for design synthesis use heuristics that are based on human knowledge acquired through experience and understanding of problems. The natural language, a fuzzy logic language is the most convenient way to express such knowledge. Linguistic descriptions are usually given in fuzzy terms not only because this is the most common form of representation of human knowledge, but also because our knowledge about many aspects of the design is fuzzy [Shr98]. Linguistic information, while not precise, represents an important source of knowledge. Another important source of knowledge is numerical data. Fuzzy logic systems are appropriate in such situations because they are able to deal simultaneously with both types of information: linguistic and numerical.

Also, fuzzy systems being universal approximators can model any nonlinear functions of arbitrary complexity [Men95], [Lag08]. This is very useful in modeling complex, nonlinear circuit functions of high accuracy at low cost, necessary to evaluate the circuit performances.

1.1.1. Design optimization algorithm based on fuzzy logic

Design optimization of an electronic circuit is a technique used to find the design parameter values (length and width of MOS transistors, bias current, capacitor values, etc.) in such a way that the final circuit performances (dc gain, gain-bandwidth, slew rate, phase margin etc.) meet the design requirements as close as possible.

There is no general design procedure independent of the circuit; also, there is no formal representation to connect the circuit functions with its structure in a consistent manner. The major obstacle consists in the peculiarity of the analog signals: the continuous domain of the signals' amplitude and their continuous time dependency. Hereby the analog circuit design is known like an iterative, multi-phase task that necessitates a large spectrum of knowledge and abilities of designers.

As stated in [Gie90] there are two basic modalities to deal with the analog design: knowledge based approaches and optimization based approaches. In the present research we are focused on the last one. The optimization algorithm begins with the formulation of the optimization objectives and optimization problem, followed by the initialization of the design parameters. During iterations an evaluation engine computes the actual circuit performances based on the actual design parameter values. If the objectives are fulfilled, the solution consists in the set (or sets – in case of a real multiobjective optimization) of the actual design parameter values and the algorithm stops. If the objectives are not fulfilled, new values of the design parameters have to be computed by the optimization engine, so the optimization loop is covered once again.

The novelty is the utilization of fuzzy logic, involving different fuzzy techniques in every phase of the optimization algorithm, as it is shown in Fig. 1. Fuzzy sets are involved in the formulation of the optimization objectives (to compute the unfulfillment degrees) and in the initial guess of the design parameters (to compute the matching degrees). Fuzzy systems are used to build efficient fuzzy models to address the performance evaluation (evaluation engine) and also they are involved in the new parameter values computation (optimization engine) [Olt02a], [Olt05a], [Olt08a].

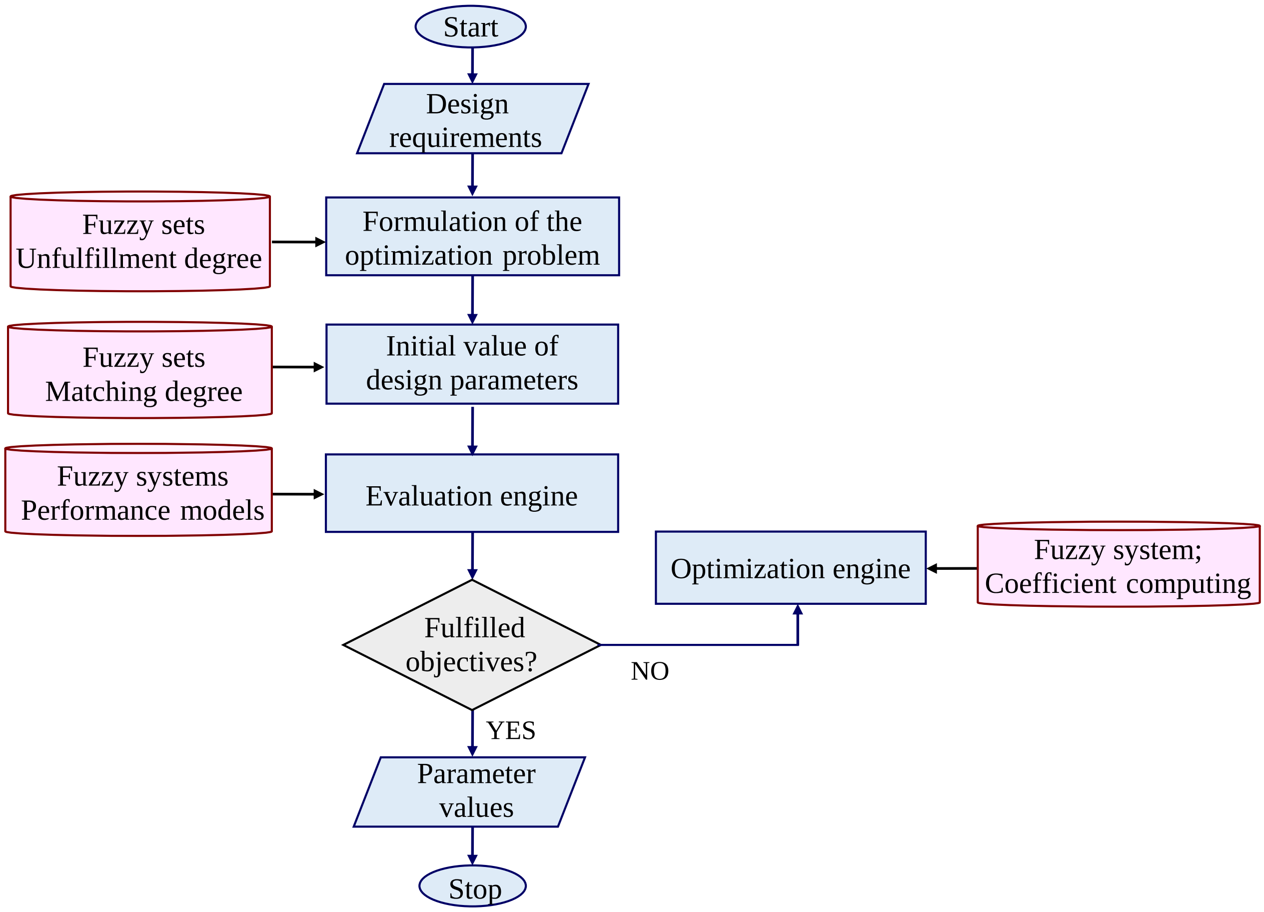


Fig. 1 Diagram of the design optimization algorithm, using fuzzy logic in all phases.

Optimization problem

The fuzzy technique used to define the optimization objectives supposes the fuzzification of the design requirements using fuzzy sets, getting this way the possibility to consider different degrees for requirement achievements and acceptability degrees for a particular solution.

As an example, the requirements "greater or equal" $f_k(x) \geq f_k^r$ and "equal" $f_k(x) = f_k^r$ have the corresponding fuzzy objective functions presented in Fig 2, with:

x – the vector of the design parameters;
f_k - the k^{th} performance function;
f_k^r - the k^{th} design requirement;
x^* - the current value of the design parameters vector.

The membership degree μ represents the error degree in the fulfillment of the fuzzy objective. A value μ =1 means the objective is not satisfied at all, while a value μ = 0 means the objective is fully satisfied.

The fuzzy objective functions are defined as:

$$\mu_k\left(f_k(x)\right):\ D_{f_k} \rightarrow [0,1] \tag{1}$$

where D_{f_k} is the range of possible values for $f_k(x)$.

The membership function $\mu_k\left(f_k(x)\right)$ indicates the error degree in accomplishing the k^{th} requirement, so we call it "unfulfillment degree" (*UD*). A value $\mu_k = 0$ means full achievement of fuzzy objective, while a value $\mu_k = 1$ means that the fuzzy objective is not achieved at all.

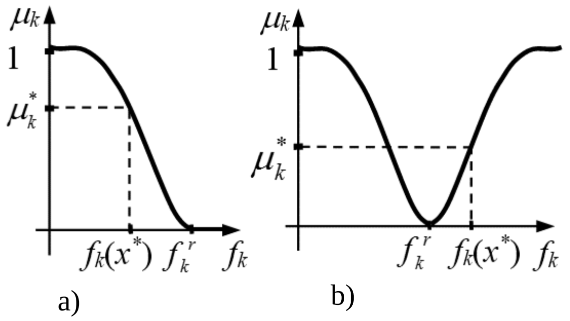


Fig. 2 Fuzzy objective functions:
a) $f_k \geq f_k^r$; b) $f_k = f_k^r$

The optimization problem can now be formulated as [Olt02a], [Olt05a], [Olt08a]:

$$
\begin{array}{ll}
\text{Find } x \text{ that} & \\
\text{minimizes} & \{\mu_1(f_1(x)),\ \mu_2(f_2(x)), \ldots, \mu_n(f_n(x))\} \\
\text{subject to} & \\
\quad \text{inequality constrains:} & g_i(x) \leq 0, \quad 1 = 1, \ldots, m \\
\quad \text{equality constrains:} & h_i(x) = 0, \quad 1 = 1, \ldots, p \\
\quad \text{bound constrains:} & lb_l \leq x_l \leq ub_l, \quad l = 1, \ldots, q
\end{array}
\tag{2}
$$

There are two ways to solve a multi-objective optimization problem. The first one is to convert the problem into a single objective optimization problem, by combining all the individual objectives into a single objective function (for example using a weighted sum). The second approach involves a real multiobjective optimization method, which separately considers each objective function for ranking a certain solution.

Initial values of design parameters

Different methods to obtain promising initial solutions are presented in literature. In [Far95], [Tor96], and [She88] an experienced user is the one that provides a good initial guess. Approximate analytic design equations are used in [Far95], [Has89], and [Mic96] while some randomly generated initial solution are preferred in [Fu99], [Koh90], [Agu94], and [Phe00].

I have proposed a fuzzy based method for the initial solutions. It is based on the selection of the initial solutions from a much larger set, previously generated using Latin Hypercube Sample technique. The selection criterion is the matching degree of the corresponding initial performances with the design requirements; the matching degree being defined by means of fuzzy sets [Olt02a], [Olt05a], [Olt08a]. The advantages of this method are: no necessity for an experienced designer, good quality of the initial solutions and no user intervention.

Evaluation engine

An accurate estimation of the circuit performances requires the use of complex and accurate models of circuit performance functions. This leads to an excessively large computation time, in the iterative process of design optimization as long as in each iteration each performance function should be evaluated. One way to reduce the computation time is to use simple models of circuits and/or devices. To satisfy both main requirements (accuracy and speed), many researches proposed several methods to evaluate circuit performances.

Simple analytical models are used in [Has89], [Tor96], [She88], [Bin07], and [Zeb98], while some version of polynomial and monomial models are involved in [Her04], [Her01] and [Bag07]. A lot of automatic design tools include a Spice-like simulator to accurately compute the performances [Bar06], [Bar05], [Zha05], [Pra04] and [Bag07]. Least-square support vector machine are also involved in [Kie04], [Ren07], and [Ber03].

Fuzzy systems are very useful to model the circuit performances because they imply just a few simple mathematical operations and can model any complex, multivariable and nonlinear function at any level of accuracy. Such fuzzy models are used in [Bal04], [Tor96], and [Ven99].

I synthesized a method to develop fuzzy models of circuit performance functions for some analog modules. These fuzzy models can be automatically built using input-output data sets. Each circuit performance is modeled by a first order Takagi-Sugeno fuzzy system, having the circuit parameters as inputs and the performance function as output. The proposed modeling procedure contains several steps. First, the necessary data sets are generated by using Spice simulation for combinations of parameters established by a DoE (Design of Experiments) phase based on the LHS (Latin Hipercube Sample) method. Using a fuzzy subtractive clustering method, an initial structure for the fuzzy system was generated. Then, using the ANFIS (Adaptive Neuro-Fuzzy Inference System) method, the training of the fuzzy system took place in order to improve its accuracy. In the training phase, two different data sets are involved: a training data set that is directly used to compute the approximation errors and adjust the parameters of the fuzzy system, and a checking data set that supervises the training process to detect a possible overtraining (overfitting) phenomenon. The resulting fuzzy system is finally tested from the accuracy point of view. If the accuracy is not satisfactory, the procedure must be resumed by generating a new initial fuzzy system or even by determining new data sets. If the accuracy is acceptable, the modeling procedure stops and it provides the desired fuzzy model of that circuit performance function.

The proposed method is illustrated in the development of fuzzy models of circuit performances for a simple transconductance amplifier (SOTA) in a 0.25μm CMOS technology, circuit presented in Fig. 3 [Olt02a], [Olt02b], [Olt08b], [Olt09a].

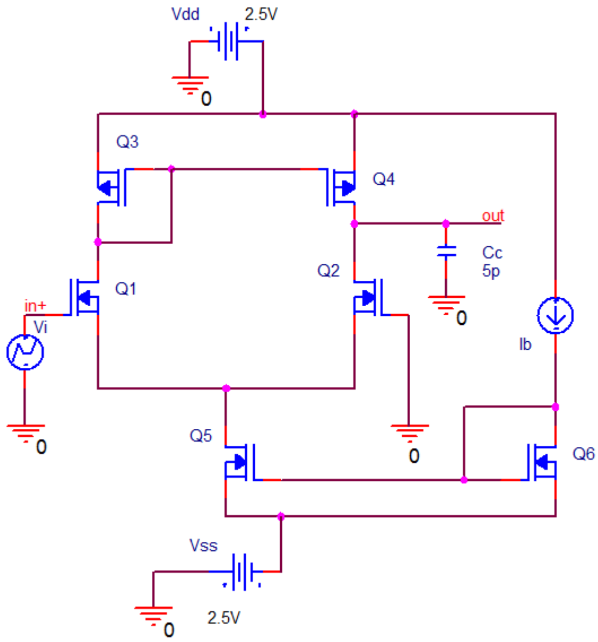


Fig. 3 The SOTA in 250nm CMOS technology.

The design parameters are the dimensions of the transistors *(W/L)* and the bias current I_b. The input transistors Q_1 and Q_2 must be identical, therefore $(W/L)_1=(W/L)_2$ resulting the first parameter $W12=W_1=W_2$, setting $L_1=L_2=0.5\,\mu$. The transistors Q_3 and Q_4 (active load) must be paired, resulting $(W/L)_3=(W/L)_4$, so the second parameter will be $W34=W_3=W_4$, setting $L_3=L_4=0.75\,\mu$. For the current mirror, Q_5 and Q_6, we consider the current (I_b) equal trough both transistors so $(W/L)_5=(W/L)_6$. To keep a minimal area, we have taken $W=L$ so our third parameter is $W56=W_5=W_6=L_5=L_6$. The fourth and final parameter is I_b. The range of variation of the design parameters was quite large: ±78%, ±82%, ±98%, respectively ±56% around a nominal value.

As circuit performances, the important ones were considered: the voltage gain A_{v0}, the unity gain bandwidth GBW, the phase margin *PM*, and the common mode rejection ratio $CMRR$. The voltage gain was determined considering that the whole differential input voltage is applied to the noninverting input (in+) with the inverting input connected to the ground; the drain of the Q2 transistor is considered the single ended output of the circuit (out); the output is loaded with a 5pF capacitance (Cc). The circuit benefits from a fully differential supply (±2.5 V), so no dc common-mode voltage is necessary to bias the input stage.

Table 1 contains some numerical results referring to the accuracy of fuzzy models. *RMSE* represents the root-mean square error encountered for the fuzzy model as a result of training. Let's mention that *RMSE* is dependent of the magnitude order of the performance function, and thus, the greater the values of circuit function the greater the *RMSE*. Also, the final error is directly correlated with the dynamic domain of the performance function. The mean and maximum values of the percent error, for a new, independent data set, are relatively small, confirming good modeling accuracy for these multi-variable, complex nonlinear functions.

It is worth to mention that the data sets were obtained semi-automatically obtained, the user intervention being necessary in the process of collecting the numerical values. This way, the data sets were limited to reasonable dimensions. If higher modeling accuracy proved to be necessary, larger data sets can be automatically obtained (without user intervention) by driving the simulator (Spice) and formatting data from the Matlab environment, similar with the approach used in [Olt05b], [Olt07a].

Table 1 Accuracy of fuzzy models for SOTA

Performance function		*RMSE* for the trained fuzzy model		Percent error for an independent data set [%]	
Notation	Range of values	Training data	Checking data	Mean	Max
A_{v0}	19 - 63	1.18	0.93	0.83	1.27
GBW [kHz]	1438 - 5435	142	58	3.07	8.53
PM [^{o}]	90.8 – 92.5	0.09	0.033	0.03	0.1
CMRR	2160 - 1880613	33400	43875	3.04	8.74

In the iterative process of optimization design, using fuzzy models in the evaluation engine presents outstanding advantages from the computing time point of view. In the case of SOTA circuit, the time to evaluate the four circuit performances using fuzzy models is approximately 30 times smaller than using Spice simulation.

Optimization engine based on fuzzy systems

The optimization engine (computing the new values of design parameters) is the heart of the optimization algorithm. It should be chosen so that the optimization will converge to an optimal (or at least acceptable) solution in a reduced number of iterations. This task is not an easy one due to complex relations between design parameters and circuit performances. The same design parameter can affect more than one circuit performance at a time, so when a parameter is modified to improve a performance it can worsen another.

The literature abounds in optimization methods. Some classic (local) optimization methods involved in analog design automation are: steepest descent [Koh90], [Mic96] and [Swi93], sequential quadratic programming [Tor96b], [Mau93], Lagrange multiplier [Mic96], conjugate gradient [Tor96], feasible direction [Far95], or simplex [Shr98]. Global optimization based on classical methods are also present in the field: simulated annealing [Tor96], stochastic pattern search [Phe00], geometric programming in convex form [Her04], [Her01], and [Bag07]. Computational based method as genetic algorithm and evolutionary strategy [Bal04], [Bar06], [Som07], and [Eec06] proved to be another class of efficient optimization methods.

The author proposed two optimization methods based on fuzzy inference systems, both being real multiobjective optimization method. The first one, GGFO (Global Gradients Fuzzy Optimization) uses global qualitative dependencies (qualitative gradients) of the performance functions on the design parameters. The second optimization engine, LGFO (Local Gradients Fuzzy Optimization) is based on local quantitative gradients.

The GGFO method proposes a zero order Takagi-Sugeno fuzzy system for every design parameter to compute a coefficient to modify it. The rule base of the fuzzy system incorporate qualitative dependencies between the design parameter and circuit performances. This method implies a reduced computation volume, because no gradient are determined, using only qualitative ones (increase, decrease). This method is a local optimization ones and it holds only for monotone performance function in respect with all parameters. The chances to search for the global optimum solution can be increased if the optimization procedure is re-run several times starting with different initial design parameters.

For the LGFO method, the sign and the value to modify a certain design parameter take into account the unfulfillment degrees of design requirements, the local quantitative gradients and the relative importance of the design parameters in relations with the optimization objectives [Olt05a], [Olt04a].

The method acts as a human expert when optimizes the parameter values:

- the parameter with greater importance is modified more, because it can really affect the performance, the modification magnitude being in direct relation with the unfulfillment degrees (*UD*) of the corresponding requirements;
- the parameter with lower importance is modified less or not at all, because its influence on circuit performance is insignificant;
- the final modification of a parameter is a weighted sum of the partial modification (imposed by every objective function), the weights being the relative importance of performances in the modification of the design parameter.

Such human expert knowledge is captured and incorporated in the optimization engine by means of a fuzzy logic system. The 3D surface generated by this fuzzy system is presented in Fig. 4. It shows the value of the partial coefficient (*coef-part*) as a function of the unfulfillment degree (*UD*) and the importance of a design parameter (importance) in the modification of circuit function. This optimization method acts in an adaptive manner: when the *UD*s are high (towards 1), large coefficients to modify the parameters result; for small *UD*s small coefficients result. Thus we can focus our search so that the solution converges to the optimal point.

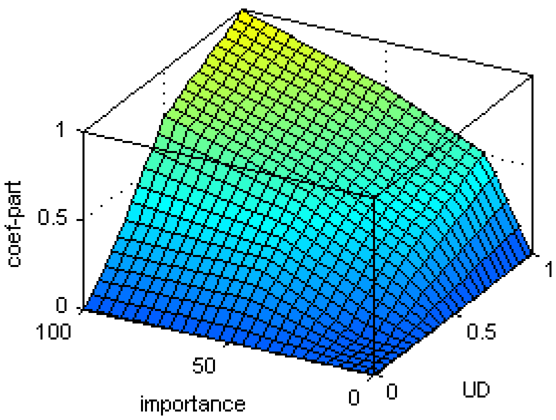


Fig. 4 Partial coefficient modification as a function of unfulfillment degree of a requirement and design parameter importance.

The LGFO method also falls in the category of local optimization method, but there is no restriction for the circuit performance functions.

In order to substantially increase the chance to find optimum solutions (not only one local optimum) placed on the Pareto front for our multiobjective optimization problem, the LGFO method was enhanced to deal with multiple parallel search paths by means of a population of solutions, consisting of candidate solutions (denoted MLGFO) [Olt02c]. In each iteration, for every candidate solution, the actual performances, the *UD*s and new parameter values are computed. If the *UD*s for one candidate solution cannot be decreased any further, a local Pareto optimal point is found and future iterations will not visit this candidate solution any more, shortening the total optimization time.

Note that, during the optimization, there is no information change between candidate solutions, each solution follows its own search path.

The optimization algorithm stops in one of the following situations:

i) All *UD*s become zero for one candidate solution. This candidate solution is considered a global Pareto optimal point and it is the final solution;
ii) None of the candidate solutions can be further improved, meaning that a subset of local Pareto optimal points was obtained. As final optimal solution we chose the one with the minimum value of the mean of unfulfillment degrees;
iii) Maximum number of iterations exceeded.

The fuzzy optimization with qualitative gradients (GGFO) is fast, because it implies low computation volume, but it is restricted to monotone functions and can only find a local solution. On the contrary, the fuzzy optimization with quantitative gradients and multiple search paths (MLGFO) has the advantage to avoid a local solution and find a set of Pareto optimal points. It can also cover a wide spectrum of the performance functions to be optimized.

The above presented optimization methods were implemented in the Matlab environment as CAD tools, with user-friendly graphical interfaces as communication bridges between the user and computer. The experimental results obtained for sizing few common analog modules (SOTA, MOTA – Miller OTA, CE amplifier, can be found in [Olt02a], [Olt02b], [Olt02c], [Olt02d],[Olt05a],[Olt08a]. Due to lack of space, only some results for a design optimization of SOTA circuit using the MLGFO method are presented here.

Table 2 Design optimization for SOTA using MLGFO

<table>
<tr><td rowspan="4">Circuit function</td><td rowspan="4">Require-ments</td><td rowspan="4">Time moment</td><td colspan="2">Candidate solution 3</td><td colspan="2">Candidate solution 9</td><td colspan="2">Candidate solution 11</td><td colspan="2">Candidate solution 18</td></tr>
<tr><td colspan="2">iterations: 152</td><td colspan="2">iterations: 9</td><td colspan="2">iterations: 9</td><td colspan="2">iterations: 250</td></tr>
<tr><td colspan="8">Settings: candidate solutions: 30; maximum iterations: 250</td></tr>
<tr><td>perf.</td><td>mean UD</td><td>perf.</td><td>mean UD</td><td>perf.</td><td>mean UD</td><td>perf.</td><td>mean UD</td></tr>
<tr><td rowspan="2">Avo</td><td rowspan="2">> 50</td><td>initial</td><td>30.18</td><td rowspan="8">iniţial 0.749
final 0.0221</td><td>44.73</td><td rowspan="8">iniţial 0.284
final 0.0113</td><td>46.97</td><td rowspan="8">iniţial 0.271
final 0.0</td><td>39.62</td><td rowspan="8">iniţial 0.634
final 0.0122</td></tr>
<tr><td>final</td><td>56.37</td><td>49.33</td><td>52.20</td><td>55.75</td></tr>
<tr><td rowspan="2">GBW [kHz]</td><td rowspan="2">> 4500</td><td>initial</td><td>1174.23</td><td>5188.7</td><td>4245.9</td><td>1693.5</td></tr>
<tr><td>final</td><td>4151.20</td><td>4919.8</td><td>4585.2</td><td>4219.2</td></tr>
<tr><td rowspan="2">PM [°]</td><td rowspan="2">> 60</td><td>initial</td><td>91.77</td><td>91.16</td><td>91.07</td><td>91.49</td></tr>
<tr><td>final</td><td>90.80</td><td>90.93</td><td>90,.89</td><td>90.84</td></tr>
<tr><td rowspan="2">CMRR</td><td rowspan="2">> 1000000</td><td>initial</td><td>420629</td><td>346566</td><td>612111</td><td>211336</td></tr>
<tr><td>final</td><td>967388</td><td>941386</td><td>1000040</td><td>1005601</td></tr>
</table>

The circuit was designed for a set of design requirements, all requirements being express as “greater than”, as they are presented in Table 2. The optimization was run for a number of 30 candidate solutions and a maximum number of 250 iterations. The data presented in Table 2 (requirements, performances, mean *UD*s, iterations number) refers to four candidate solutions (3, 9, 11 and 18). Due to the relatively large population the algorithm locates a final solution (candidate solution 11) in a very short time (9 iterations); this solution presents mean *UD* = 0, namely all design requirements fully satisfied. In the usual operation, the algorithm would be stopped, but, to better understand how it works, we further investigated all the other search paths up to the last iteration. Others individuals in the population also provide good (acceptable) solutions: candidate 9 stops in 9 iteration with a mean *UD*=0.0113; candidate 3 stops in 152 iteration with a mean *UD*=0.0221; candidate 18 goes up to the last iteration with mean *UD*=0.0122.

The optimization time, on a very modest computer (850MHz, 256MB RAM) for the entire optimization process was 252s. Anyway, in usual operation if the optimization was stopped after 9 iterations the time would have been around 10 s.

1.2. GA based optimization engine

The design optimization algorithm illustrated in Fig. 1 was entirely based on fuzzy logic. The proposed fuzzy logic based approaches for the optimization engines GGFO and LGFO could provide (very) good solutions, but they presented an appreciable possibility to discover only some local optima. Even the MLGFO method, that explores multiple search path to increase the chance to find a global optimum, is not a very efficient method because each candidate solution act independently.

To efficiently search the whole solution space, a powerful global optimization technique should be considered. Genetic algorithms (GA) are based on the Darwinian principle of natural selection and the concepts of natural genetics. GAs have many desirable characteristics and offer significant advantages over traditional methods in the optimization domain. They are inherently robust and have been shown to efficiently search large solution spaces containing discrete or discontinuous parameters and linear and non-linear constraints, without being trapped in local minima [Eri08]. GA does not require derivative information and have often found non-intuitive solutions to engineering problems. Genetic algorithms have already been employed in many CAD applications [Bal04], [Som07], [Eec06], [Tak04], and [Tah03].

The novelty of the approach discussed hereinafter is to develop a hybrid intelligent system (a combination of fuzzy logic and genetic algorithms) for the design optimization of some analog modules. It combines the best qualities of some CI techniques: flexibility to formulate the objective functions and a known range of their values using fuzzy sets; low computational complexity and accuracy in computing circuit performance by means of neuro-fuzzy models; and a powerful global optimization engine based on genetic algorithm.

The optimization engine (in the general design optimization process) consists now of a genetic algorithm (GA) that can perform single or multi-objective optimization. The general flowchart of the GA used in the design optimization is represented in Fig. 5.

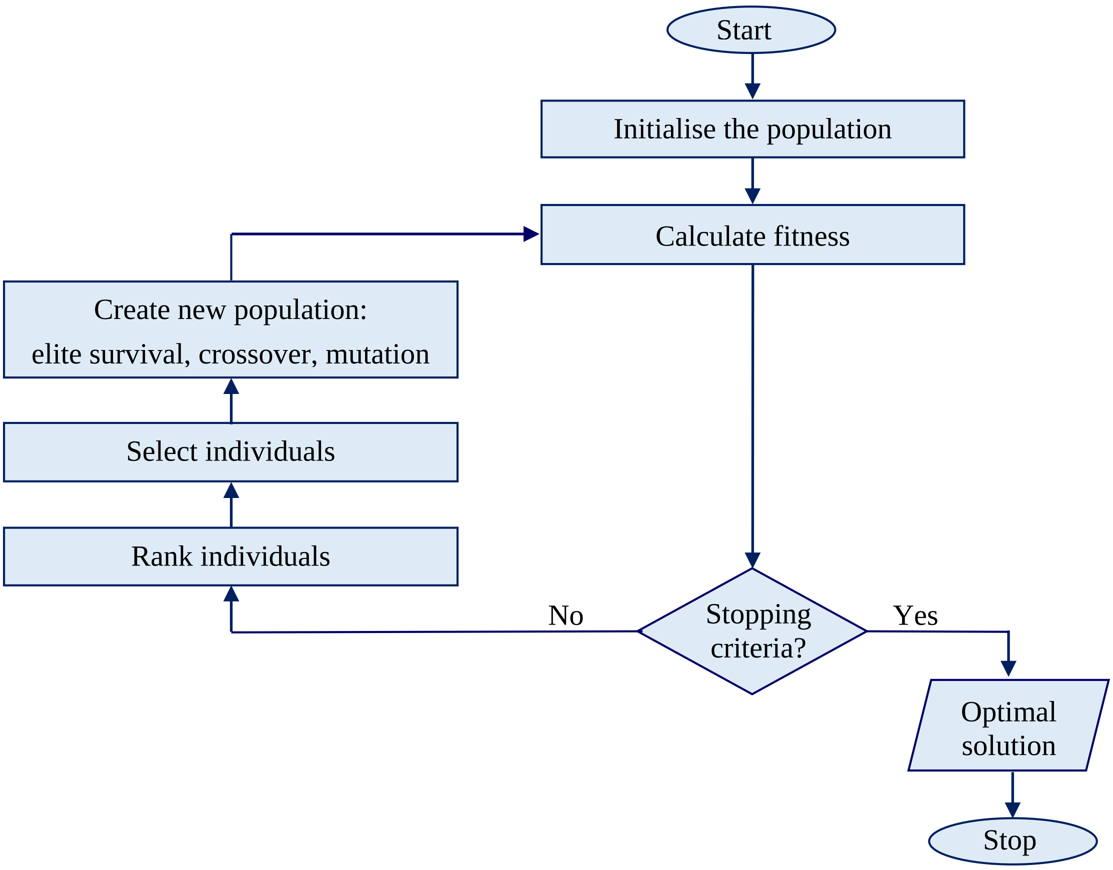


Fig. 5 Flowchart of GA used in the design optimization.

First, the population is initialized, usually randomly with a uniform distribution. It is sometimes feasible to seed the initial population with "promising" values that are known to be in the hyperspace region close to the desired one. These solutions can also be chosen by taking into account the previous experience of the designer, which sometimes can be a disadvantage in finding new and unconventional solutions [Neg09].

The next step is to calculate the fitness of each individual in the current population. This requires the evaluation of the objective function (or functions in the case of MOO), for each individual, by comparing the performances of the circuits with the design requirements. Since the optimization involves minimizing the objective function(s), the smaller the value(s), the better the fitness.

Then the stopping criteria (fitness limit, maximum generation, stall generations, etc.) are checked. If one of them is satisfied, the fittest individual(s) in the current population is(are) considered to be the optimal solution and the algorithm stops.

If none of the stopping criteria is satisfied, the population should evolve towards a new (improved) population. First, it is necessary to rank the individuals according with their fitness. A new population (children) is now created by applying genetic operators: selection, crossover and/or mutation, or even elite survival. The fittest individuals receive greater probabilities to be selected as parents, but also less fitted individuals have their chance to be selected as parents. Also, it is possible for the best individuals to automatically survive in the next generation as elite children, to preserve the quality of the population. Over successive generations, the population evolves towards an optimal solution. Detailed information about how genetic algorithms work can be found in [Kon06], [Rut08], [GA].

One important aspect using the GA as optimization engine is the fact that the evaluation engine (see Fig. 1) cannot any longer be considered as a separate component in the full design optimization algorithm. The evaluation of the performance functions are now fully integrated in the GA optimization, since in each generation, for every individual, the circuit performances are needed to be known in order to compute the fitness of individuals in the current population.

1.2.1. Single objective optimization

A hybrid optimization algorithm based on computational intelligence techniques (fuzzy sets to define the objective functions, neuro-fuzzy models for performance evaluation, and a genetic algorithm for global optimization) was implemented in Matlab and applied for the design optimization of SOTA circuit (Fig. 3) for three design requirements: A_{v0}, *GBW*, and *CMRR* [Olt08b], [Olt09a]. According with the type of requirements ("greater or equal" or "equal") for each design requirements we define fuzzy objective functions $\mu_k\left(f_k(x)\right), k=1, 2, 3$, using appropriate fuzzy sets (see Fig. 2). The vector of the design parameter is:

$$x = [W12;\ W34;\ W45;\ I_b] \tag{3}$$

The design problem is a multiobjective one, three objective functions are to be minimized in the same time, subject to bounds constrains for the design parameters:

$$\begin{aligned} &\text{find } x \quad \text{that minimizes} \quad \left\{\mu_1\big(f_1(x)\big),\ \mu_2\big(f_2(x)\big),\ \mu_3\big(f_3(x)\big)\right\} \\ &\qquad\qquad \text{subject to:} \qquad lb_l \le x_l \le ub_l, \quad l = 1, \dots, 4 \end{aligned} \tag{4}$$

For the single-objective optimization approach, we combine the individual objective functions into a single cost function $F(x)$ using a weighted sum. The formulation of the multiobjective optimization problem now becomes:

$$\begin{aligned} &\text{find } x \quad \text{that minimizes} \quad F(x) = \sum_{k=1}^{3} w_k \mu_k\left(f_k(x)\right) \\ &\qquad\qquad \text{subject to:} \qquad lb_l \le x_l \le ub_l, \quad l = 1, \dots, 4 \end{aligned} \tag{5}$$

where w_k is the weight, or relative preference, associated with the k^{th} objective function.

For GA development, the vector *x* of the design parameters is used directly as a chromosome, each design parameter being a gene. Because the design parameters are real variables we chose real numbers for genes. The main settings used for GA was: one population; population size:100; scaling: linear ranking; selection: roulette; no elite survival; crossover: intermediate; mutation: gaussian.

Due to the very large number of circuit performance evaluations (the number of individuals times the number of generations) we need efficient models of circuit performance functions, so fuzzy models are involved here.

The design optimization is illustrated here for the set of design requirements presented in Table 3, for equal weighted objective functions. The optimization was run several times, for a population of 100 individuals. The algorithm proved to be a robust one, since it has always found a solution that fulfils all the requirements. Different number of iterations is necessary to search for the optimum solution in different runs, depending on the initial population and on the evolution process.

Table 3 Requirements and performances for the SOTA design optimization (4 runs).

Circuit functions	Requirements	Performances			
		*run*1	*run*2	*run*3	*run*4
A_{v0}	≥ 50	52.17	52.01	51.83	51.97
GBW [kHz]	≥ 4 600	4 601	4 620	4612	4608
CMRR	≥ 1 000 000	1 011 413	1 008 935	1 004 844	1 003 761
Number of iterations		94	90	87	63

Table 4 Solutions for SOTA design optimization (4 runs).

Design parameter	*run*1	*run*2	*run*3	*run*4
*W*12	39.28	39.73	39.33	39.07
*W*34	4.00	3.99	3.98	3.99
*W*56	6.90	6.86	7.21	7.23
I_b [μA]	96.33	97.22	94.74	94.00

For the first optimization run (*run*1), the dynamic behaviour of the algorithm is presented in Fig. 6. In the first iterations (up to 10), due to a high diversity of individuals, the new population does not always contain a better individual than in previous population (minimum value of the cost function does not decrease – Fig. 6 top).

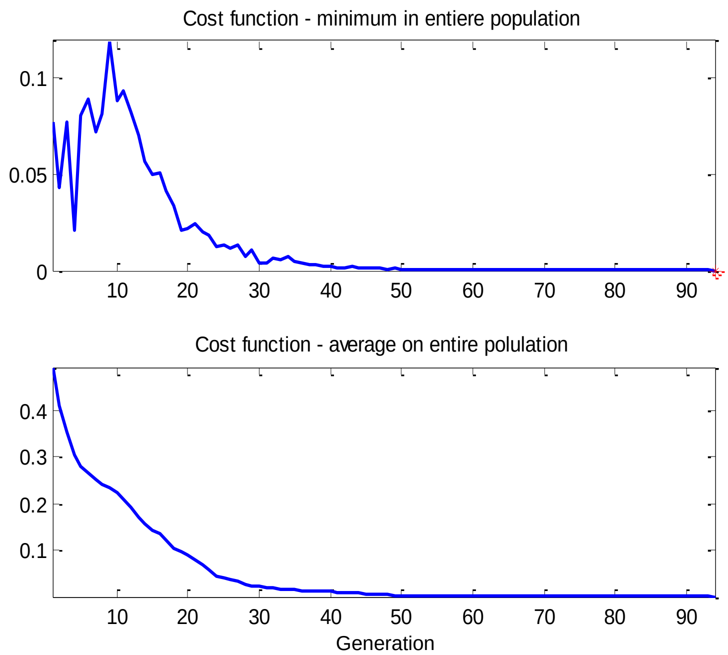


Fig. 6 Minimum and average evolution of cost function $F(x)$ in *run*1.

This happens because the population reproduction phase does not involve elitist survival. On the other hand, the population as a whole is improved continuously, the average value on the entire population of the cost function decreases in time, from one generation to the next (Fig. 6 bottom). As the population improves during evolution, all individuals move towards the optimal solution, decreasing both the minimum and average values of cost function. The improvement of population is obvious up to the generation 50, but not all the design requirements are fully satisfied. In the next time period, fine improvements happen up to generation 94, where all design requirements are fulfilled and the optimization stops.

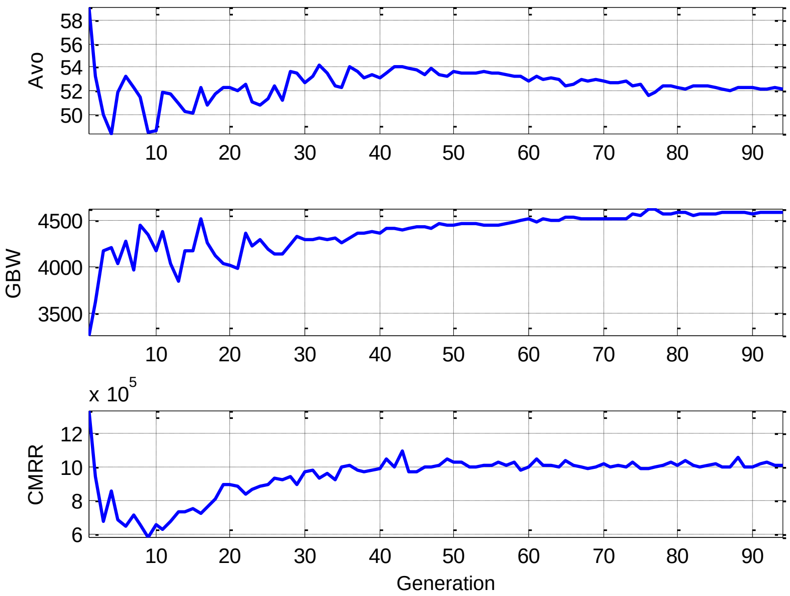


Fig. 7 Dynamic behavior of all performances in *run*1. In each generation the performances of the best individual was considered.

The evolution of all performance functions during the design optimization is presented in Fig. 7 where performances of the best individual in each generation were considered. For A_{v0}, the initial value was a satisfying one (59, that was greater than 50), but in the subsequent generation this value decreases, even bellow the requirement value (generation 4, generations 9-10). After some variations its value settles down around 52, stopping at 52.17 in the last generation. It seems that the most difficult requirement to meet was GBW, a value greater than the requirement appearing only in the final part of the optimization (from generation 76). One can see the nature of the optimal solution search, from generation to generation by the endeavour of the population to simultaneously satisfy all individual objective functions.

1.2.2. Multiobjective optimization

Single objective optimization does not allow multiple competing objectives to be accounted for explicitly; moreover they do not give the circuit designer the freedom to choose among different, equally feasible solutions. A big step forward in this direction can be achieved using a multiobjective approach. This technique allows different objectives to be treated separately but simultaneously during the optimization process [Nic08].

A multiobjective optimization algorithm should provide a set of nondominated individuals (Pareto front), or optimal solution set. A number of stochastic search strategies such as evolutionary algorithms, tabu search, simulated annealing, and ant colony optimization have been developed for multiobjective optimization [Zit04]. As evolutionary algorithms are assumed to yield good results on complex problems without explicit knowledge of the detailed interdependencies involved, they seem to be a tempting choice [Gre07].

Let's consider that the multiobjective optimization problem contains O objective functions to be minimized simultaneously:

$$\text{find } x \quad \text{that minimizes } F(x) = \{(f_1(x), f_2(x), \ldots, f_O(x))\} \tag{6}$$

Following the concept of Pareto dominance, an objective vector $F(x^1)$ is said to dominate another objective vector $F(x^2)$, if no component of $F(x^1)$ is greater that the corresponding component of $F(x^2)$, and at least one component is smaller. Accordingly, we can say that the solution x^1 is better than another solution x^2, i.e., x^1 dominates x^2 if $F(x^1)$ dominates $F(x^2)$ [3].

A solution x^* is said to be Pareto optimal, or a nondominated solution for a multiobjective optimization problem if, and only if there is no x such that

$$\begin{aligned} & f_i(x) \le f_i(x^*) \text{ for } \forall i = 1, \ldots, O \\ & \exists j \text{ for that } f_j(x) < f_j(x^*) \end{aligned} \tag{7}$$

A nondominated solution means a solution in which an improvement in one objective implies the degradation of another objective. Optimal solution, i.e., solution nondominated by any other solution, may be mapped to different objective vectors. In other words, several optimal objective vectors representing different trade-offs between the objectives may exist. The set of optimal solutions is usually denoted as Pareto set, and its image in the objective space is denoted as Pareto front. With many multiobjective optimization problems, knowledge about this set helps the decision maker (circuit designer) in choosing the best compromise for the final solution. In the following, we will assume that the goal of the optimization is to find (a subset of) the Pareto optimal solutions.

Illustratively, Fig. 8 presents a possible spread of solutions for a two-objective optimization problem (f1 and f_2 objective functions should be minimized).

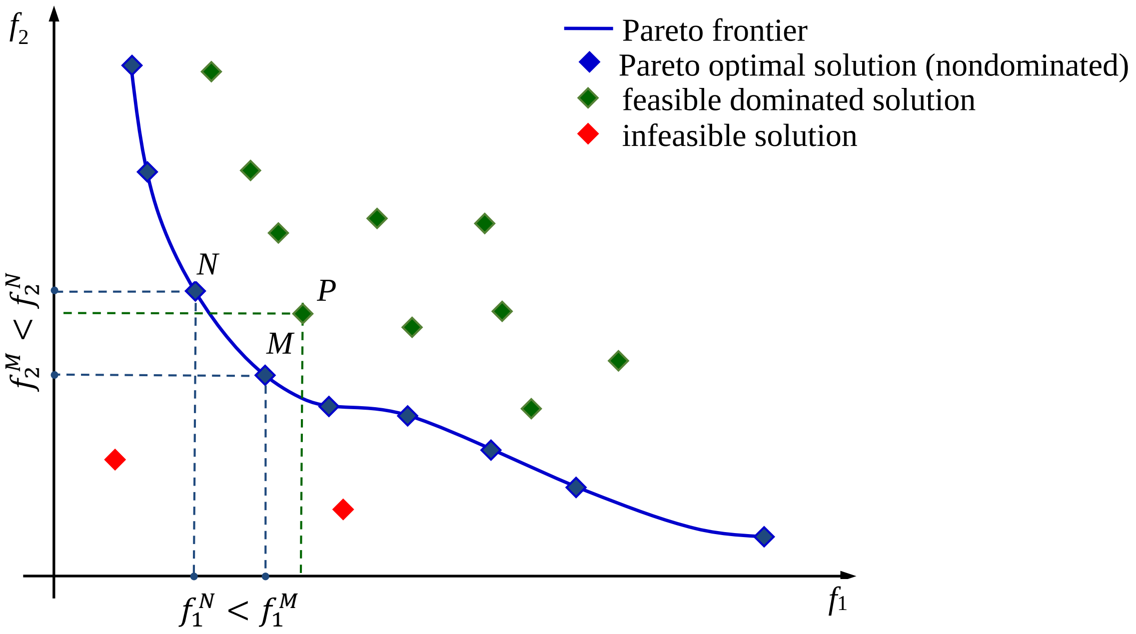


Fig. 8 Illustration of possible spread of solutions for a two-objective optimization: minimize $\{f_1, f_2\}$.

The blue curve represent the Pareto frontier, with a subset of Pareto optimal (nondominated) solutions (blue diamonds). If solutions M and N are compared, from the point of view of objective function f_1, solution N is better than solution M because $f_1^N < f_1^M$. From the point of view of f_2, solution M is better than solution N because $f_2^M < f_2^N$. As a whole, solutions M and N are equivalent and they are not dominated by any other solution. On the contrary, solution P for example, is not a Pareto optimal one because it is dominated by the solution M: $f_1^M < f_1^P$ and $f_2^M < f_2^P$.

All solutions represented by green diamonds in Fig. 8 are feasible solutions, but not optimal ones. The two solutions represented by red diamonds are infeasible solutions.

Genetic algorithms (GA) performing multiobjective optimization (MOO) have previously been used in analog circuit design to generate a set of Pareto optimal solutions. In [Nic08] the problem of analog IC design is formulated as a constrained MOO problem defined in a mixed integer/discrete/continuous domain. In [Yas07] analog circuit satisfying a specific frequency response, using free circuit structures and including some parasitic effects, are produced in a single design stage.

MOO is also successfully applied to design a Folded – Cascode op-amp using a non-dominated sorting genetic algorithm-II (NSGA-II) [Tak09] and an Operational Transconductance Amplifier, using genetic algorithm Pareto Envelope based Selection [Per10]. The solutions define the optimal Pareto front and reveal the trade-offs between the circuit's requirements.

A real multiobjective optimization based on GA, for design optimization of analog modules was implemented in Matlab. It provides a subset of optimal solutions (Pareto front) giving the possibility for the designer to select the final solution in accordance to further preferences or constrains. The objective functions are formulated using fuzzy sets (as the ones in Fig. 2), so we are using the *UD* (unfulfillment degree) to measure the error in satisfying the requirements. The evaluation of the current design, for each individual in the population is performed by means of neuro-fuzzy models of circuit performances.

As long as we are concerned with multiobjective optimization, the multiobjective ranking is used for the fitness assignment in our implementation. Each individual within a population receives a rank according to its quality. All solutions that are not dominated by any another solution constitute Pareto optimal solutions. All these nondominated solutions will receive a maximum value for their rank, $Rank_{max}$:

$$Rank_{max} = N_{Ind} - 1 \tag{8}$$

where N_{Ind} represents the number of individuals.

For the remaining solutions, the rank is computed using the relation:

$$Rank = Rank_{max} - N_{Dominating} \tag{9}$$

where $N_{Dominating}$ is the number of individuals that dominates the individual under consideration.

Further on, all nondominated solutions benefit from a high selection probability, while the dominated solutions have a lower selection probability, decreasing with the number of dominating individuals.

For the selection, our approach uses the roulette-wheel method. Even if this method is the simplest selection scheme, it provides good results, without significant loss of population diversity, when it is used in conjunction with a rank-based fitness assignment, instead of proportional fitness assignment. For each individual *j*, a selection probability is computed as:

$$Selection\_probability_j = \frac{Rank_j}{\sum_{i=1}^{N} Rank(i)} \tag{10}$$

where *N* is the number of individuals.

Once the parents are selected, they go through the next genetic operation: intermediate crossover. Next, the Gaussian mutation is applied to generate the final offspring. Our GA uses a pure reinsertion scheme: it produces as many offspring as parents and replace all parents with offspring. Every individual lives one generation only.

Our design optimization algorithm accepts three types of requirements "greater than", "equal" and "smaller than" for each design requirement. It is used to design a CMOS simple transconductance amplifier SOTA (Fig. 3). The circuit is designed for a set of three requirements: voltage gain – *Av*, gain-bandwidth product – *GBW* and common mode rejection ratio – *CMRR*, using four design parameters: three transistor sizes $(W/L)_{12}$, $(W/L)_{34}$, $(W/L)_{56}$, and a biasing current *Ib*. All design parameters have lower and upper bounds, determined so that the transistors in the circuit will remain in their active region, regardless the combination of parameter values. These bounds are:

$$LB = [20, 0.5, 0.75, 1] \qquad HB = [70, 4, 7.5, 100] \tag{11}$$

The design optimization of SOTA is first illustrated for a set of “equal” type design requirements (Table 5) [Olt09b]. The optimization was run for a population of 400 individuals for 1000 generations.

Table 5 Performances and *UDs* for “equal” type requirements

Solution		Requirements		
		Av = 40	*GBW* = 5000 [kHz]	*CMRR* = 500000
*Sol*1	Performances	40.46	4891	487418
	UD	*0.0008*	*0.0012*	*0.0013*
*Sol*2	Performances	40.04	4613	472018
	UD	*7.39e-6*	*0.0156*	*0.0063*
*Sol*3	Performances	40.81	4980	466923
	UD	*0.0025*	*4.095e-5*	*0.0088*
*Sol*4	Performances	41.37	4716	500158
	UD	*0.0071*	*0.0084*	*2.96e-5*

At the end of the optimization, the algorithm found 34 individuals (out of 400 individuals) on the Pareto front. Due to lack of space, we present here the performances and the values of *UDs* for only four of them denoted *Sol*1, …, *Sol*4. *Sol*1 was selected here due to its minimum average *UD* (0.0011). *Sol*2 is the better one from the point of view of *Av* requirement, meaning that it has a minimum value of *UD* for *Av* in the entire Pareto set (7.39e-6). *Sol*3 is the one having the minimum *UD* for *GBW* requirement in the final Pareto set (4.095e-5). From the point of view of *CMRR* the best individual is *Sol*4 whose *UD* is 2.96e-5. Each individual constitutes a feasible design solution, the final decision being made by the circuit designer.

Table 6 Design parameters for “equal” type requirements

Solution	Design parameters			
	$(W/L)_{12}$	$(W/L)_{34}$	$(W/L)_{56}$	*Ib*[µA]
*Sol*1	62.3	2.6	7.3	100
*Sol*2	59	2.5	7.4	94.1
*Sol*3	63.1	2.7	7.3	94.1
*Sol*4	57.4	2.7	7.1	94.4

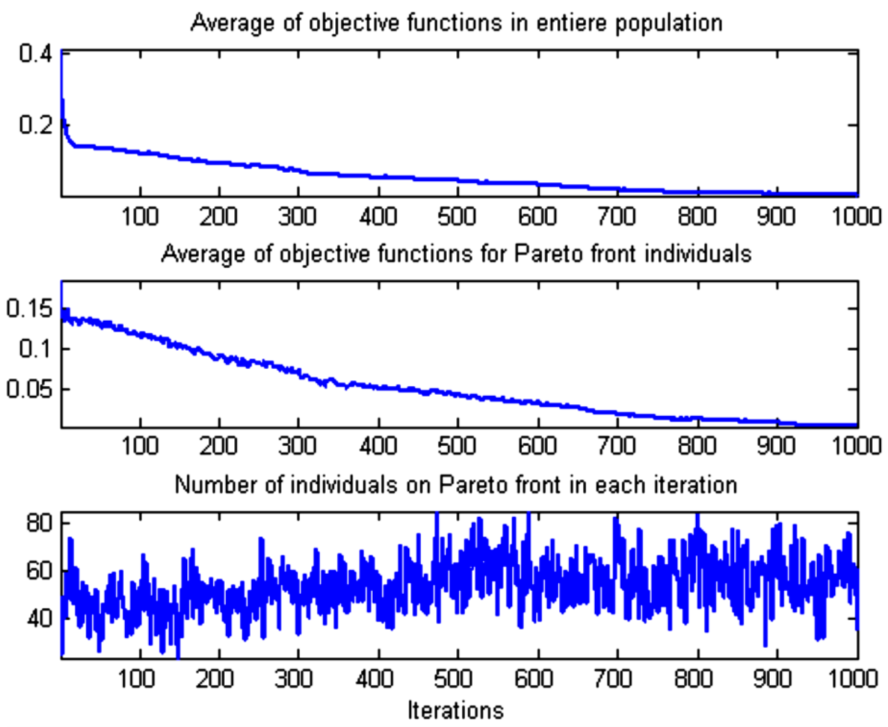


Fig. 9 Dynamic behavior of the GA MOO for SOTA design optimization.

The values of the design parameters are presented in Table 6. The dynamical behaviour of our optimization algorithm is presented in Fig. 9. The quality of the entire population is improved generation by generation especially at the beginning of the genetic evolution. The average of the *UD* in the entire population decreases continuously from an initial value of 0.4109 down to 0.0046 (Fig. 9 top). The quality of the Pareto front is improved in a slightly oscillating manner, meaning that not always a new Pareto front is better than the previous one. The average of the *UD* in the Pareto front decreases from an initial value of 0.18254 down to a final value of 0.0038 (Fig. 9 middle). The dimension of the Pareto front varies during the evolution (Fig. 9 bottom), with a minimum of 23 individuals in generation 149 and a maximum of 85 individuals in generations 474 and 588.

The design optimization of SOTA is also illustrated for a set of "greater than" requirements:

$$Av > 50,\ \ GBW > 3000[\text{KHz}],\ \ CMRR > 1200000 \tag{12}$$

The optimization was run for a population of 100 individuals for only 50 generations. In the population evolution there was only 1 individual on the Pareto front up to the iteration 41. From that point forward, the number of individuals on the Pareto front was increased continuously up to the final value of 97 individuals (out of a total of 100 individuals). The performances of five individuals and the corresponding design parameters from the final Pareto front are presented in Table 7; all design requirements being accomplished. The individuals are quite similar to each other, a possible interpretation being that during the evolution the population diversity was lost. The genetic algorithm can be improved if some condition to preserve the population diversity is introduced.

Table 7 Optimization results for "greater than" type requirements

Solution	Performances			Design parameters			
	Av	*GBW*	*CMRR*	$(W/L)_{12}$	$(W/L)_{34}$	$(W/L)_{56}$	*Ib*[μA]
*Sol*1	59.83	3369.16	1252521.98	20.20	3.60	5.70	84.20
*Sol*2	59.68	3389.05	1234074.67	20.30	3.60	5.80	82.50
*Sol*3	60.90	3468.18	1242884.89	20.20	3.80	5.70	82.10
*Sol*4	59.84	3369.10	1220738.45	20.20	3.60	5.60	82.30
*Sol*5	58.98	3363.40	1211318.90	20.40	3.50	5.90	81.20

1.2.3. Multiobjective optimization using a circuit simulator

The above presented MOO based on GA proves to be a reliable design optimization procedure. To keep the design optimization time at (much) reduced values (few minutes) the circuit performances were evaluated by means of fuzzy models, directly implemented in the Matlab environment. If a greater confidence in the numerical results is necessary, instead of using models of circuit functions, a circuit simulator should be involved in the optimization loop. Common circuit parameters, such as transistor dimensions, supply voltages or bias currents are provided by the GA and transferred to a standard circuit simulator (e.g. SPICE, ELDO, Spectre) that is used to simulate the circuit and to provide the circuit performances.

The proposed automated design optimization method using GA multiobjective optimization and the Spice simulator to evaluate the circuit performances is presented in Fig. 10. The Matlab environment is used both to control the entire process and to run the GA. At first, the initial population is generated. Next, for each individual, Matlab creates a netlist file, that is sent to SPICE to simulate the circuit, the simulator being launched from Matlab, for each individual in turn. The results of the simulation are written in a text output files, which are read from the Matlab. From these files, the circuit performances are extracted and used to compute the fuzzy objective functions (*UD*). The ranking of individuals uses multiobjective ranking, by ranking the individuals in one nondominated category and one or more dominated categories (ranks). Tournament selection and intermediate crossover are used together with mutation to generate the new population. The algorithm stops if the stop condition is reached (maximum number of generations or non-significant improvement in the fitness function value, over a certain number of generations), else, it goes back to the first step. The result consists in the noninferior (nondominated) solutions (Pareto optimal solutions). To the end, the user has the freedom (and responsibility) to make the final decision choosing one solution from the optimal Pareto front.

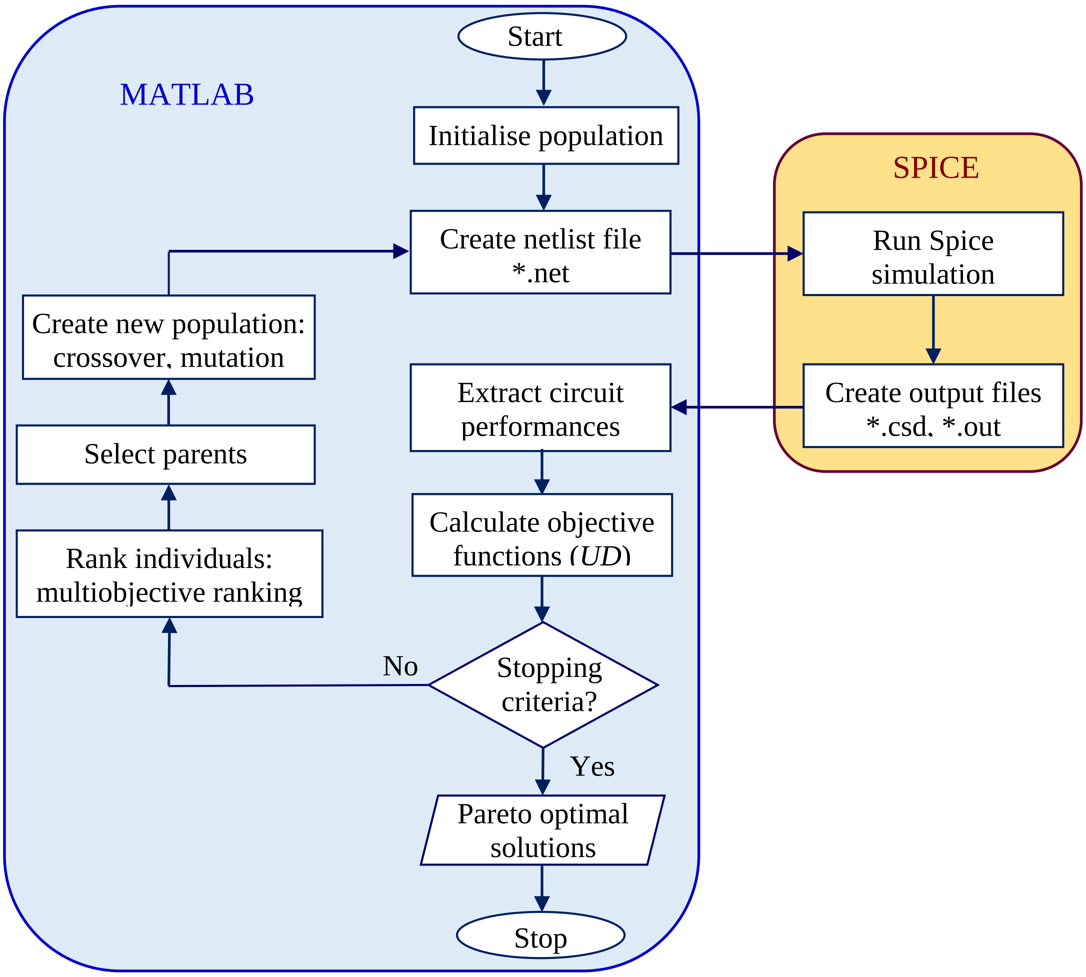


Fig. 10 Flowchart of GA MOO driven by Matlab and using Spice as circuit simulator.

The additional functions used to create the netlist file, launch the SPICE simulator and extract the necessary data for the algorithm to work, were defined in the Matlab environment and are external to the multiobjective GA optimization tool.

The proposed algorithm is used for the design optimization of a symmetrical CMOS operational transconductance amplifier in a 180nm technology (Fig. 11), operating on a capacitive load C_L (4 pF). Details about this circuit can be found elsewhere, for example in [San06], [Sed11], [Iva11].

Four design parameters are considered: width of the transistors M_1 and M_2 ($W12$), M_3 and M_4 ($W34$), M_7 and M_8 ($W78$) and the biasing current I, so the chromosome for GA optimization is:

$$x = [W12;\ \ W34;\ \ W78;\ \ I] \tag{13}$$

The ranges for the design parameter are were set to:

$$W12 \in [1, 100][\mu\text{m}];\ W34 \in [10, 1000][\mu\text{m}];\ W78 \in [1, 100][\mu\text{m}];\ \ I \in [1, 200][\mu\text{A}] \tag{14}$$

For the input stage we take into account the basic compromise in analog CMOS design: for high gain, the circuit has to be designed for small $V_{GS}-V_T$ and large channel length L. For high frequency, the requirements are exactly opposite. We set the channel length to $L1 = L2 = 1\mu\text{m}$ to obtain some gain. For the current mirrors (M_4-M_6, M_3-M_5), the mirrored current should be as close as possible to the value B times the reference current. This happens for quite large value of the transistor channel length. The larger the channel length, the smaller error in the mirrored current is. For our design we considered $L3 = L4 = L5 = L6 = 10\mu\text{m}$.

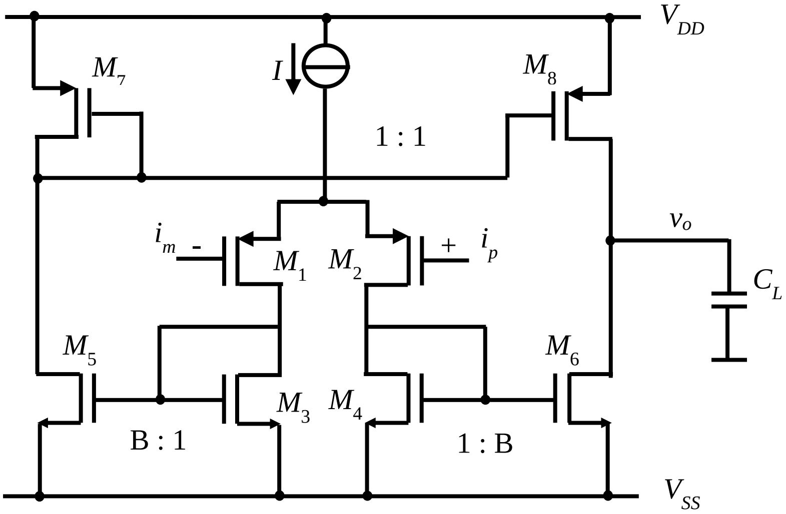


Fig. 11 Symmetric operational transconductance amplifier.

To keep our biasing current source operational (minimum positive voltage drop across it), there is a linear inequality constraint between two design parameters $W12$ and I:

$$\left(\frac{W}{L}\right)_{12} > 0.036I \tag{15}$$

The circuit should is designed considering four design requirements: gain at low frequency A_v, bandwidth *BW*, layout area *Area*, and power dissipation, *P*, as they are presented in Table 8. The optimization algorithm has to find the most appropriate values for the design parameters, in order to minimize the power consumption and layout area and to fulfil the “greater than” type requirements for the gain and bandwidth, subject to one nonlinear constraint and bound constrains for the four design parameters.

For the multiobjective optimization there are four individual objective functions. For the *Area* and *P* that needs minimisation, their values are directed considered as objective functions: *Area* determined in [μm^2] and *P* determined in [μW]. For the other two design requirements we are using the following objective function form:

$$f(x) = \begin{cases} 0; & perf(x) - \text{req} < 0 \\ |perf(x) - \text{req}|; & perf(x) - \text{req} < 0 \end{cases} \tag{16}$$

were *perf*(x) is the circuit performance computed for the design parameter vector x, and *req* is the required minimum value for that performance.

Table 8 Design requirements

No.	**Design requirements**	
	Name	**Value**
1	Gain	>200
2	Bandwidth	>150KHz
3	Layout area	minimized
4	Power consumption	minimized

The algorithm was run for a population of 60 individuals, over 100 generations. The plot in Fig. 12 shows the ranking of the individuals, in the final iteration. The Pareto front consists of the 21 individuals that have rank 1, which means that they are nondominated.

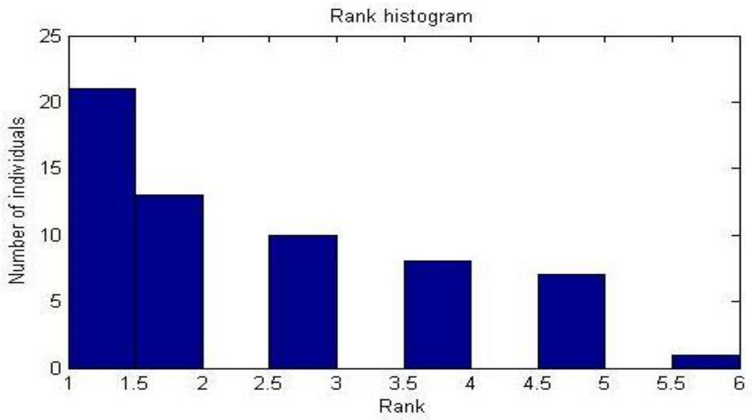


Fig. 12 Rank histogram.

Out of that 21 optimal solutions, Table 9 [Iva11] presents 10 solutions. Considering the gain and bandwidth specifications, there is one solution (Sol4: 204.04) that fully satisfies the former objective, and two that meet the latter (Sol5: 158.48 kHz and Sol10:158.48 kHz). Sol9 provides a gain (191.11) very closed to the requirement, while Sol 8 provides a bandwidth (138.03) very close to the requirement. From the point of view of layout area, best solutions, with the smallest values are Sol4 (1735.62 µm^2), Sol5 (5927.61µm^2), and Sol7 (7754.59µm^2). The power consumption has low values for Sol9 (284 µW), Sol4 (335µW), and Sol7 (455 µW). Sol4 is the best one considering three design requirements meaning the gain requirement is fulfilled and it provides the minimum layout area and minimum power consumption. AS for the fourth design requirement, bandwidth, that one is not satisfied.

Table 9 Optimal solutions on the Pareto front

Sol.	*W*12 [µm]	*W*34 [µm]	*W*78 [µm]	*I* [µA]	Gain	Bandwidth [kHz]	Layout area [µm^2]	Power consumption [µW]
1	3.24	601.17	13.38	56.39	23.36	104.71	36103.75	517
2	45.57	154.93	7.23	151.51	89.48	123.02	9401.83	1380
3	37.39	618.22	26.24	97.9	72.12	95.49	37220.98	897
4	44.36	27.41	1.01	37.05	**204.04**	18.19	**1735.62**	**335**
5	49.69	96.80	9.82	171.71	83.63	**158.48**	**5927.61**	1560
6	43.53	360.82	12.20	140.47	78.55	123.02	21760.86	1280
7	27.32	128.28	1.46	50.15	141.44	28.84	**7754.59**	**455**
8	46.14	282.22	22.88	124.93	75.21	**138.03**	17071.52	1140
9	43.32	856.66	1.18	31.22	**191.11**	10.96	51488.91	**284**
10	44.37	254.57	99.99	108.51	64.76	**158.48**	15563.19	994

It is worth to remember that the solutions provided by the multiobjective optimization reveals the trade-offs that occur between the conflicting design requirements. In our case to minimise the layout area, the first three design parameters (*W*12, *W*34, and *W*78) should be as small as possible, but this is contradictory with the gain to be high (greater than) were de first parameter (*W*12) should be high. Further on, to have high gain and low power consumption, the fourth design parameter (*I*), should be small, but this is contradictory with high bandwidth (greater than) that asks for large value of the current *I*.

By simply considering together the two different types of requirement (compulsory “greater than”) and simple “minimize” (without any imposed limit) the algorithm treats all of them similarly without any special effort to satisfy the former. To tackle this aspect two approaches can be used. One approach is to formulate the optimization problem so that the requirements for gain and bandwidth became prioritized with a higher probability to be fulfilled, only after that the optimization should focused on minimizing layout area and power. The second approach is to consider the requirement for the gain and bandwidth as (hard) nonlinear constraints for the optimization problem, keeping only the layout area and power to be minimised as objective functions. This means to solve a complex multiobjective optimization subject to one nonlinear constraint, bound constrains and two nonlinear constrains.

Our method proved to be a time-consuming (approximately 10 hours, depending on computer) due to the large number of calls to the external simulator and also to use the hard disk to write and read data files. Nevertheless, this time is not prohibitive for a real design task.

An alternative approach that can substantially reduce this time is to split the optimization into two phases: in the beginning of the optimization, only simplified models should be used to evaluate the circuit performances, drastically reducing the computation time. In the end of the optimization, when the algorithm is already close to optimal solutions, the simulator should be used for circuit evaluation. Also, by using an SSD hard disk when the circuit simulator is used, can substantially contribute to bring down the time.

Given its modular structure, the proposed method can easily be adjusted to fit other design specifications or to automate the design process of a different and more complex circuit.

2. Functional modeling of analog modules using fuzzy systems

High-level models (black box type) of input-output behaviour of analog modules play a key role in the modern approaches of electronic circuit design, as intellectual property (IP), virtual prototyping and simulation of mixed-signal systems [Kuo05].

Gielen and Rutenbar [Gie00] take into consideration three main reasons to develop high level models to describe the electronic circuits behaviour. In a top-down design methodology at higher levels of the design hierarchy, where the detailed lower-level circuit implementations are yet unknown, there is a need for higher-levels models describing the pin-to-pin behaviour of the circuits rather than the (yet unknown) internal structural implementation. Second, the verification of integrated mixed-signal systems also requires higher description levels for the analog sections, since such integrated systems are computationally too complex to allow a full simulation of the entire mixed-signal design in practical terms. Third, when providing or using analog IP macrocells in a SoC (System-on-Chip) context, the virtual component has to be accompanied by an executable model that efficiently models the pin-to-pin behaviour of the virtual component.

The current number of analog designers cannot keep up with the demand for analog components. Together with the increasing complexity of the analog blocks, this situation has created an analog-design bottleneck. As the chasm between requirements (analog blocks) and reality (lack of analog designers and increasing complexity) continues to widen, designers are asking analog IP to fill the gap [Naj04]. A virtual prototype is a software-simulation-based, architectural-level model of an electronic system and includes in its structure high level behavioural models of different constituent block [Lis04].

The resulted model has to possess two very important features. First, it should be computationally efficient to develop it and also to evaluate it, so that substantial computational savings can be achieved. Second, the model should be as accurate as possible [Che95].

Fuzzy systems appears to be perfect candidates for this job. According to [Men95], [God97] the fuzzy systems being universal approximators, they are appropriate to model any multivariable, non-linear function with any degree of accuracy, so the accuracy requirement can be accomplished. More than that, fuzzy systems can be automatically generated and trained using data sets, leading to a fully affordable time to develop them. The efficient fuzzy system evaluation, in terms of time and computation resources, are guaranteed by the simplicity of the involved mathematical operation: min, max, sum, product and division [Olt16a].

The idea to involve fuzzy systems in analog circuit modeling can be found in some previous works. The paper [Tor96b] shows how fuzzy systems are used to model the average dc transconductance and the nonlinearity error of a CMOS transconductance amplifier. The gain, bandwidth, common-mode rejection ratio and slew rate for a basic two stage and a Miller transconductance amplifiers are modeled in [Olt02a] using fuzzy systems. A three-layer neural-fuzzy network has been employed in

[Alp03] to model the gain-bandwidth and phase margin of a basic two stage operational amplifier and the open circuit bandwidth and short circuit bandwidth for a second-generation current conveyor. All these works deal with fuzzy modeling of some circuit performance functions of analog modules, not to the functional behaviour.

The scope of this research area was twofold: to develop a modeling procedure, and to develop some functional models for some basic analog modules. It is worth to mention that the research was partially supported by a national research grant: "Analog modules in nanometric technology - the development of functional models using fuzzy techniques" CNCSIS, 2005-2006, to which I was the grant manager. The main results were published in 5 scientific papers and a book.

2.1. Modeling procedure for fuzzy functional models

The modeling procedure for fuzzy functional models is presented in Fig. 13 [Olt05c], [Olt06a], [Olt06b], [Lup06], [Olt07a]. The analog module has to be created in a simulation environment – in our case, the Orcad environment.

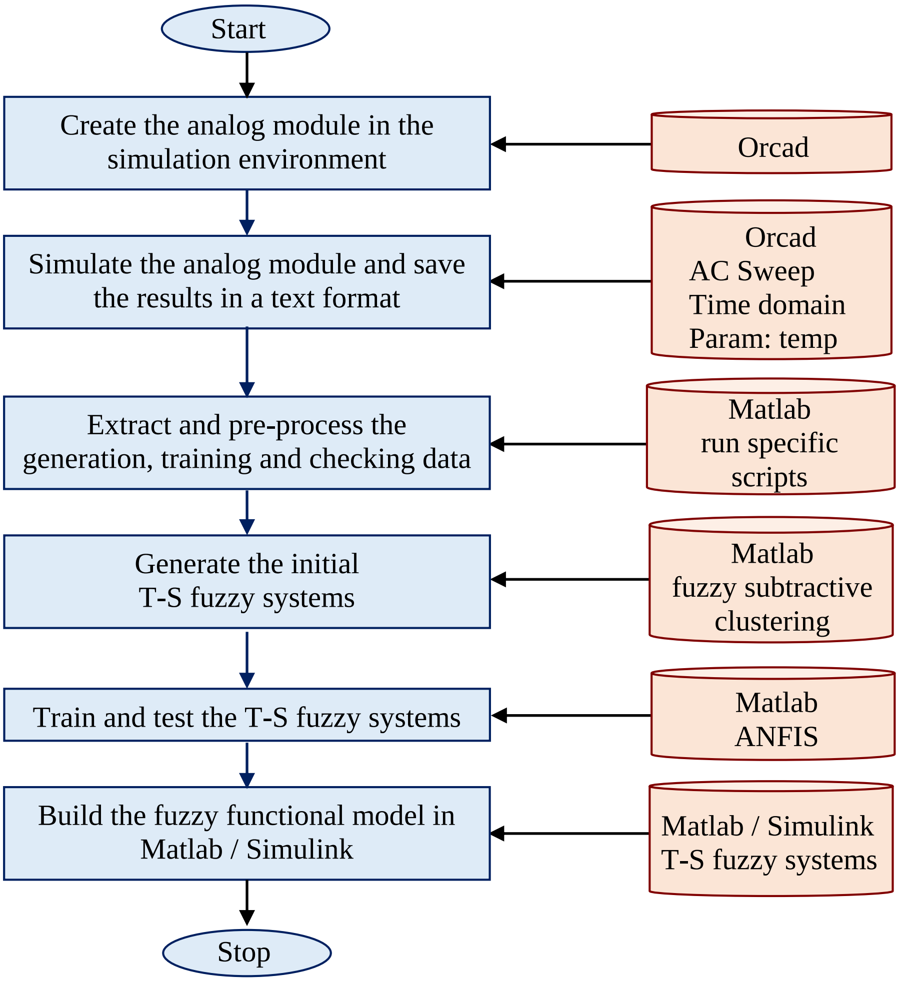


Fig. 13 The modeling procedure.

AC Sweep simulation is necessary to capture the frequency response of the circuit, keeping the circuit in its linear region. Each simulation run includes also a Temperature option to reiterate the simulation for different temperature values in the [-55; +125]^{0}C range. To intercept the amplifier behaviour in the nonlinear and saturation region it is necessary to use a time domain simulation with sufficiently high magnitude of the input signal. The simulation data files are saved in a text format.

To build the fuzzy model, we need a certain format of data. So it is necessary to extract that data from the text files (.csd) and then pre-process them to be further used for fuzzy models (fuzzy systems) development. This is done using some specific Matlab scripts, developed by the author.

The temperature will be translated up with 60°C to have only positive values. For the frequency a logarithmic transformation is used to dramatically reduce its range.

Between Mamdani type and Takagi-Sugeno type fuzzy inference systems, the latter one was selected in this application as it is more appropriate for automatic nonlinear function modeling (approximation) [Fen10], [Had11]. The initial T-S fuzzy systems are generated based on the numerical data, using a fuzzy subtractive clustering algorithm that identifies the data clusters and their centres. In the resulted fuzzy systems, each cluster is considered as a fuzzy rule. The initial fuzzy models are then improved for accuracy by training each of them with the Adaptive-Network-based Fuzzy Inference Systems (ANFIS), which is the major training routine for Sugeno-type fuzzy inference systems. ANFIS uses a hybrid-learning algorithm to identify parameters of Sugeno-type fuzzy inference systems. It applies a combination of least-squares and backpropagation gradient descent methods for training FIS membership function parameters to emulate a given training data set. ANFIS also performs a model validation using a checking data set, to detect the model overfitting. The fuzzy inference system is trained during a designated number of epochs until overfitting appears or the training error goal is achieved. More details about ANFIS can be found for example in [Jan93], [FLT].

Finally, once all necessary fuzzy systems are developed, the next step is to build the fuzzy functional models as Matlab functions and scripts and/or Simulink models.

2.2. Structure of the fuzzy functional models

The fuzzy functional models describe the input voltage - output voltage relation in terms of the amplitude, frequency and phase-shift and include the temperature effect in the [-55; +125]^{0}C industrial range. All along the wide frequency and temperature ranges the models are able to provide the output waveform for a sinusoidal input waveform that keeps the amplifier in its linear region. Supplementary, at low frequencies the models are able to provide the output waveform for a sinusoidal or triangular input waveform even if the amplitude of the input signal enters the nonlinear or saturation regions.

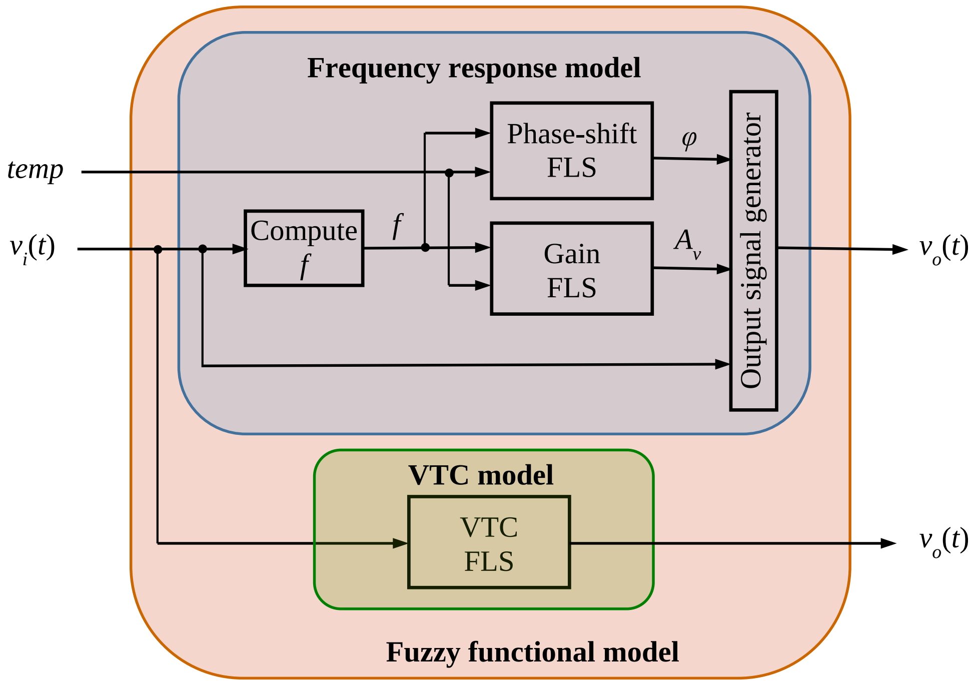


Fig. 14 Structure of fuzzy functional models.

The structure of the fuzzy functional model is presented in Fig. 14. It is composed of two main independent modules: "Frequency response model" and "VTC model".

The "Frequency response model" generates the output voltage in the form:

$$v_o(t) = \hat{V}_o \sin(2\pi f t + \varphi) \tag{17}$$

where $\widehat{V_o}$ is the amplitude, f is the frequency, and φ is the phase-shift of the resulted output voltage. This model is evaluated for an input voltage:

$$v_i(t) = \hat{V}_i \sin 2\pi f t \tag{18}$$

whose amplitude is small enough to keep the amplifier in the active region.

The frequency response model contain two fuzzy logic systems with two inputs and one output:

- Phase-shift FLS that models the nonlinear phase-shifting (φ) dependence on the frequency (f) and temperature (*temp*);
- Gain FLS that models the nonlinear gain (A_v) dependence on the frequency (f) and temperature (*temp*).

The block "Compute f" determines the value of the frequency f. The "Output signal generator" block merges all the information and builds the time variation of the output signal $v_o(t)$.

The "VTC model" contains one SISO FLS that models the nonlinear voltage transfer characteristic of the amplifier in the low frequency range (in the passband). This model generates the time variation (point by point) of the output voltage for a given input voltage, whose frequency spectrum is mainly located in the passband of the amplifier.

2.3. Implementation and some experimental results

Fuzzy functional models were developed for three common operational transconductance amplifiers: a simple operational transconductance amplifier (SOTA) in 250nm CMOS technology, a Miller operational transconductance amplifier (MOTA) in 130nm CMOS technology, and a folded cascode operational transconductance amplifier (FCOTA) in 130nm CMOS technology. Due to space limitation, this work includes some experimental results for FCOTA and SOTA. Details about fuzzy functional modeling of MOTA can be found in [Olt06b].

2.3.1. Fuzzy functional model for FCOTA

The fuzzy functional model for the folded cascode operational transconductance amplifier (FCOTA) contains only the module "Frequency response model" (see Fig. 14). The circuit, the application of the modeling procedure, the implementation and some experimental results for this analog module are discussed in the following.

FCOTA

Fig. 15 presents the schematic of our folded cascode operational transconductance amplifier in athe130nm CMOS technology. The M1 and M2 transistors composes the input differential stage, while the output is considered to be single-ended in the drain of M9 transistor. A detailed description of the amplifier can be found for example in [Sed11], [Olt07a], [Lup06]. We are interested in the gain A_{vo}:

$$A_{vo} = \frac{v_{out}}{v_{id}} = \frac{v_{out}}{v_{Ip} - v_{Im}} \tag{19}$$

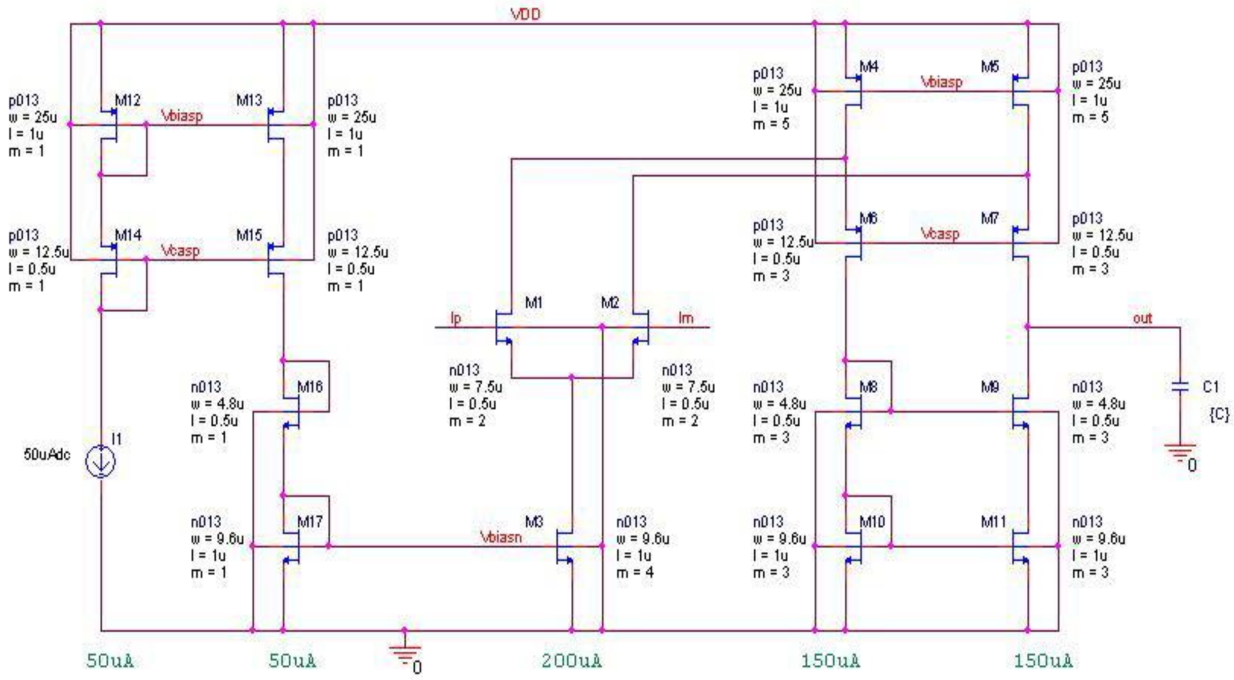


Fig. 15 FCOTA in 130nm CMOS technology.

Simulation and data sets

The circuit has a unipolar supply, only one dc voltage source VDD=3V being used. For the operation and simulation of the circuit, a dc biasing voltage (common mode voltage) should be provided, besides the small-signal to be amplified to the inputs. The total voltages applied to the positive and negative inputs are:

$$v_{Ip} = 1.5\ \mathrm{V} + \frac{v_{id}}{2} \qquad \text{and} \qquad v_{Im} = 1.5\ \mathrm{V} - \frac{v_{id}}{2} \tag{20}$$

The circuit is simulated using a parametric AC Sweep/Noise analysis, with temperature as parameter. The frequency range was [1 Hz; 1 GHz], while the temperature was considered in the industrial range of [-55^{o}C; +125^{o}C], with a step of 10^{o}C. The simulation results are presented in Fig. 16.

For the simulation, the amplitude of the differential input voltage was set to 1 V, this is way the output voltage appears to be so large (Fig. 16a)). In fact, the numerical value of output voltage at each frequency and temperature is numerically equal with the gain A_{vo} . The gain increases when temperature decreases. In the passband (low frequency), the maximum gain (7441.5 or 77.43dB) occurs at -55^{o}C. The phase-shift presents a much reduced temperature dependence, especially in the passband (Fig. 16b).

According with the modeling procedure, the simulation results are saved in a text file. The text file is read and processed in Matlab, where the data is extracted and prepared in a format required by the fuzzy systems generation and training steps. Three sets of data were produced: a data set for initial fuzzy system generation (8569 data points), a training data set (42769 data points), and a checking data set (9010 data points).

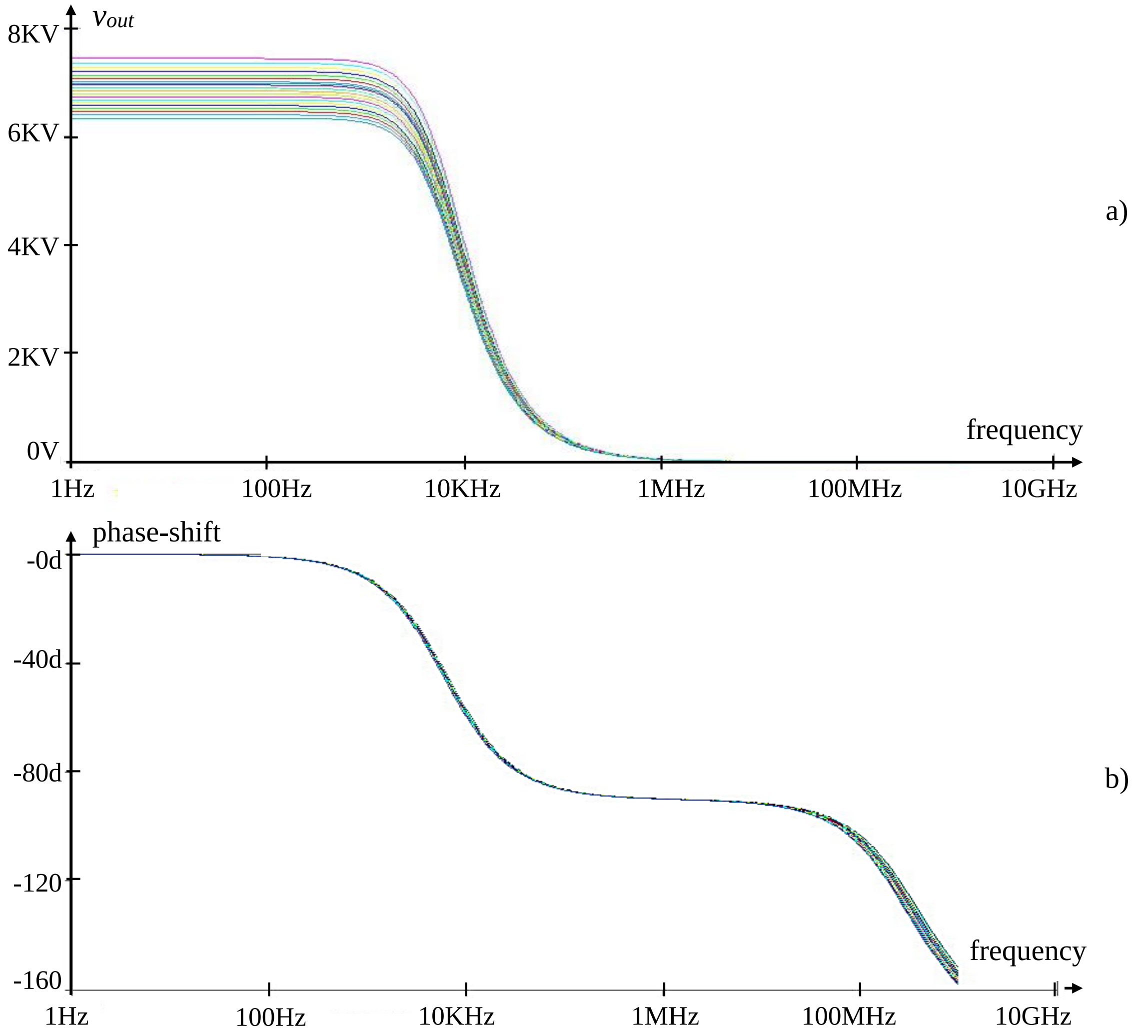


Fig. 16 Simulation results for FCOTA for a parametric AC Sweep analysis: a) output voltage as a function of frequency and temperature; b) phase-shift as a function of frequency and temperature.

Generate and train FLS

The initial fuzzy system is obtained by using a fuzzy subtractive clustering algorithm. The complexity of the resulted fuzzy system (number of fuzzy rules and fuzzy sets) can be controlled by the user by means of an algorithm parameter. The key point is to figure out a good compromise between the complexity of fuzzy systems, modeling accuracy and the size of training data set.

For the gain, a fuzzy system with 8 rules and 8 fuzzy sets on each input variable (frequency and temperature) was generated. The input fuzzy sets have gaussian membership functions, each fuzzy set being defined by 2 parameters (centre and standard deviation). The fuzzy sets for the output are first order polynomial of the input variable, each set being defined by three parameters.

The initial fuzzy system is then trained using two sets of data. The training data set is directly used to minimize the error between the reference values (values in each data point) and the values computed by the fuzzy system. The checking data set is used to supervise the training, to observe the generalization and interpolation capacities of the fuzzy model and identify if an overtraining (overfitting) phenomenon appears. Fig. 17 presents the *rmse* (root mean squared error) during the training of fuzzy system for 500 iterations. Even if the initial fuzzy system was build using a classification method, meaning the initial fuzzy system already incorporate the natural relations between input and output data, it is obvious that to increase the accuracy of the final fuzzy model, a training step is necessary. One can see a step improvement of approximation accuracy in the first 100 training iteration when both training and checking errors decrease from approximately 290 to approximately 40. After that, the training enters a fine tuning zone, still with accuracy improvement but at a much reduced rate. It is interesting to note both errors evolve similarly, meaning the data sets are consistent, all region of the function to be modeled being appropriately represented. No overfitting phenomenon is registered.

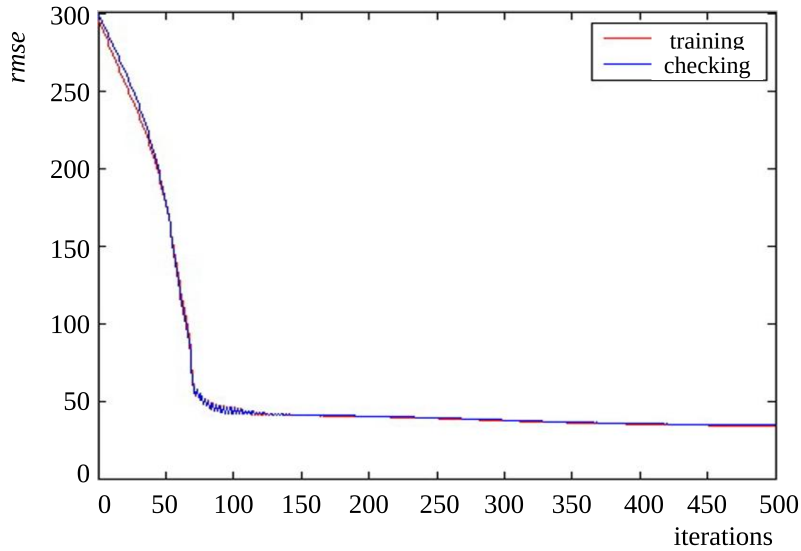


Fig. 17 Errors evolution training the gain(frequency, temperature) fuzzy logic system.

The 3D representation of the gain(frequency, temperature) fuzzy model can be seen in Fig. 18, for both fuzzy systems, the initial one (Fig. 18a)), and the final one, after the training (Fig. 18b)). One can observe the differences between the two surfaces and the effect of training the fuzzy systems, especially of smoothing the surface that is in full accordance with the circuit behaviour. Note that the frequency scale is a logarithmic one and for the temperature there is a positive offset, the initial temperature being shifted up with 60°C.

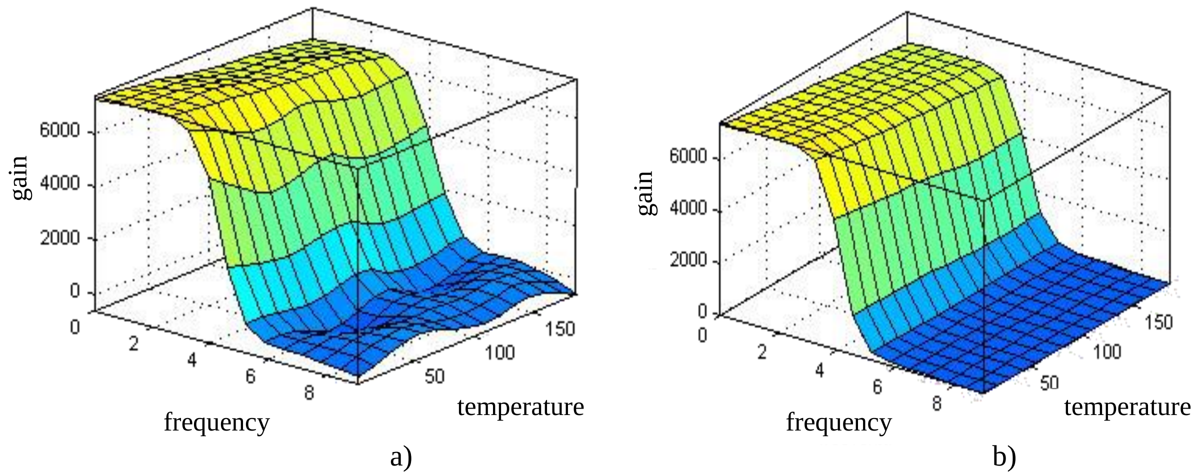
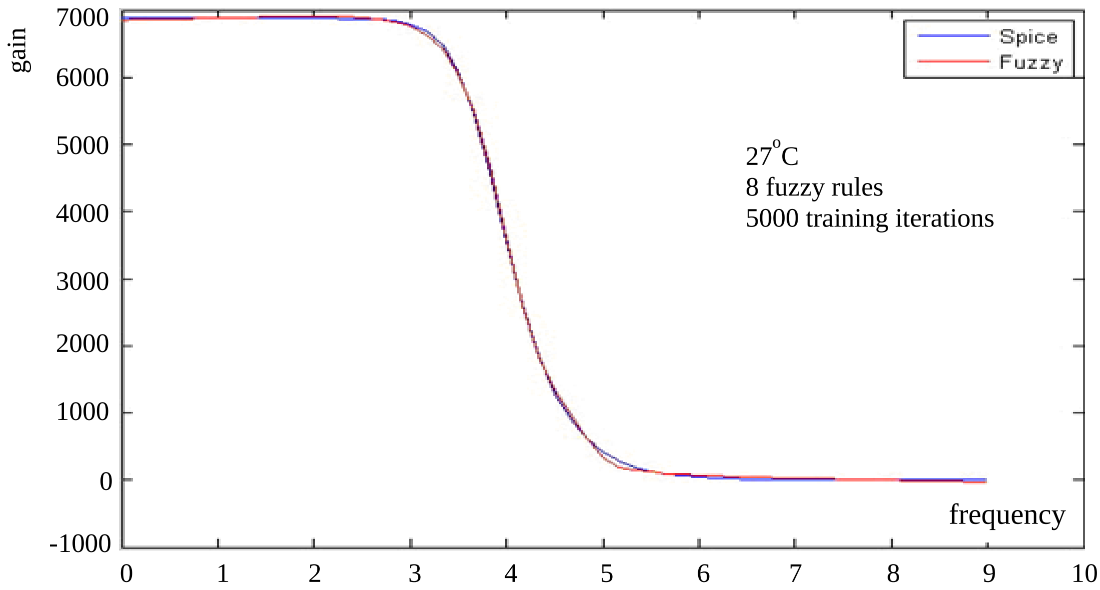
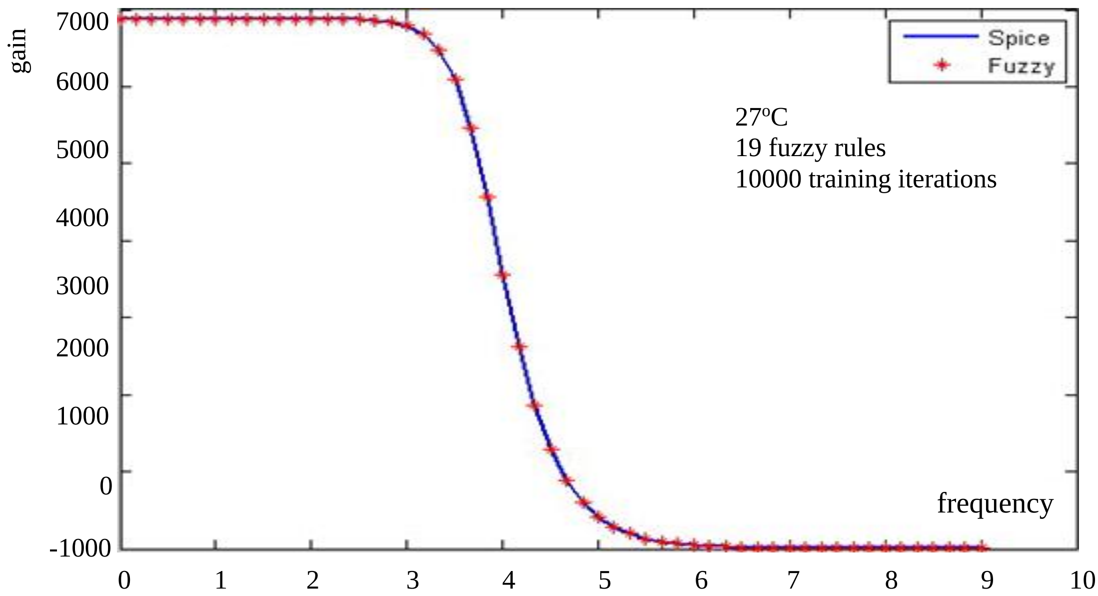


Fig. 18 Surface of the gain(frequency, temperature) fuzzy model: a) initial fuzzy system; b) final (trained) fuzzy system.

For a qualitative appreciation of the modeling accuracy, and also to give a hint about the structure and training of fuzzy systems, Fig.19 presents a direct comparison between the reference values (Spice simulated) and the values computed with the fuzzy model, at the nominal temperature (27°C), across the entire frequency range for the gain. Fig. 19a) corresponds to our previously presented fuzzy system (8 rules, trained for 500 iterations).



a)



b)

Fig. 19 Gain at 27^{o}C - comparison between the values generated by the fuzzy system and the reference values (Spice): a) fuzzy system with 8 fuzzy rules trained for 500 iterations; b) fuzzy system with 19 fuzzy rules trained for 10000 iterations.

This fuzzy model presents high accuracy, its output values being almost identical with the reference ones, in the linear regions of the curve. In the highly nonlinear regions there is an observable difference between the generated values and the reference ones. Even if the model accuracy appears to be a satisfactory one, the modeling accuracy can be further enhanced, if necessary, by trying out fuzzy systems with a different structure and increasing the training iterations. Fig.19b) presents the results obtained with a fuzzy system with 19 rules trained for 10000 iterations, using the same data sets as for the fuzzy system with 8 fuzzy rules. Now, one can observe that the fuzzy model presents excellent accuracy, the values computed by the fuzzy system being (almost) identical with the reference ones, even in the highly nonlinear regions.

A quantitative analysis of the fuzzy models accuracy is synthetized in Table 10 that includes some numerical results for both fuzzy systems FLS1 (8 rules, 500 iterations) and FLS2 (19 rules, 10000 iterations) together with the reference values. Table 10 also includes some results for the phase-shift fuzzy model (FLS), which will be shortly discussed in what follows.

Table 10 Fuzzy models values vs. Spice values for frequency response of FCOTA

Frequency	Temp.	Gain					Phase-shift[°]		
		Spice	8 rules 500 iterations		19 rules 10000 iterations		12 rules 1200 iterations		
			FLS1	Relative error [%]	FLS2	Relative error [%]	Spice	FLS	Absolute error
10Hz	-40°C	7316.4	7323.8	0.10	7317.9	0.02	-0.09	-0.73	-0.64
	27°C	6881.6	6887.2	0.08	6880.2	-0.02	-0.10	0.15	0.25
	100°C	6489.7	6495.7	0.09	6489.4	0.00	-0.10	-0.43	-0.33
200Hz	-40°C	7312.8	7353.7	0.56	7314.3	0.02	-1.79	-1.84	-0.05
	27°C	6877.8	6909.8	0.47	6880.4	0.04	-1.89	-2.49	-0.60
	100°C	6485.9	6508.5	0.35	6485.6	0.00	-1.94	-2.21	-0.27
6.038KHz	-40°C	5319.4	5369.4	0.94	5314.1	-0.10	-43.36	-43.53	-0.17
	27°C	4871.7	4975.7	2.14	4867.1	-0.09	-44.93	-44.25	0.68
	100°C	4535.7	4513.2	-0.50	4536.7	0.02	-45.66	-46.05	-0.39
75KHz	-40°C	621.7	583.4	-6.16	612.2	-1.52	-85.14	-86.17	-1.03
	27°C	553.6	532.3	-3.84	552.6	-0.18	-85.39	-84.68	0.71
	100°C	509.0	509.0	0.00	512.6	0.70	-85.52	-86.59	-1.07

Four representative frequencies were selected: 10 Hz and 200 Hz in the passband, 6.038 kHz around the cutoff frequency, and 75 kHz outside the passband. To appreciate the modeling precision of the gain, the relative error was used, while to appreciate the phase-shift modeling accuracy, the absolute error was used:

$$relative\ error = \frac{FLS_{value} - Spice_{value}}{Spice_{value}} \cdot 100[\%] \tag{21}$$

$$absolute\ error = FLS_{value} - Spice_{value} \tag{22}$$

All fuzzy models provides very good results, especially in the passband. Slightly higher error values are encountered in the increased nonlinearity regions of the curves to be modeled. For the gain, in the passband (frequency below 6.038 kHz), the magnitude of relative error is maximum 0.56% in the case of FLS1 and 0.04% for FLS2. For the phase-shift, the magnitude of the absolute error is in general less than 1°.

For the phase-shift, a fuzzy system with 12 fuzzy rules was generated and trained for 1200 iterations. The 3D representation of the phase-shift(frequency, temperature) fuzzy model can be seen in Fig. 20, with logarithmic scale for frequency and a positive offset of 60°C for the temperature. According to the simulations results (see Fig. 16b)) there is a strong nonlinear dependence with the frequency and a reduced temperature dependence.

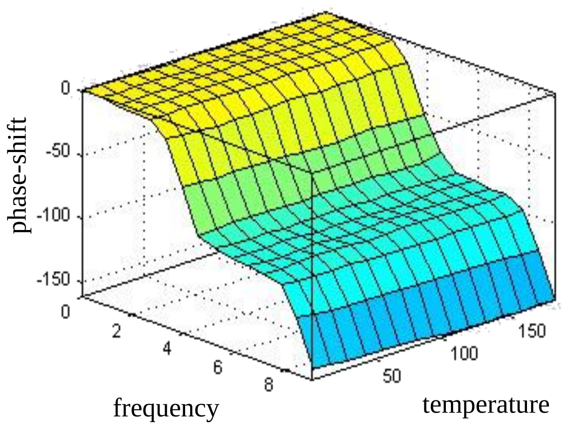


Fig. 20 Surface of the phase-shift(frequency, temperature) fuzzy model.

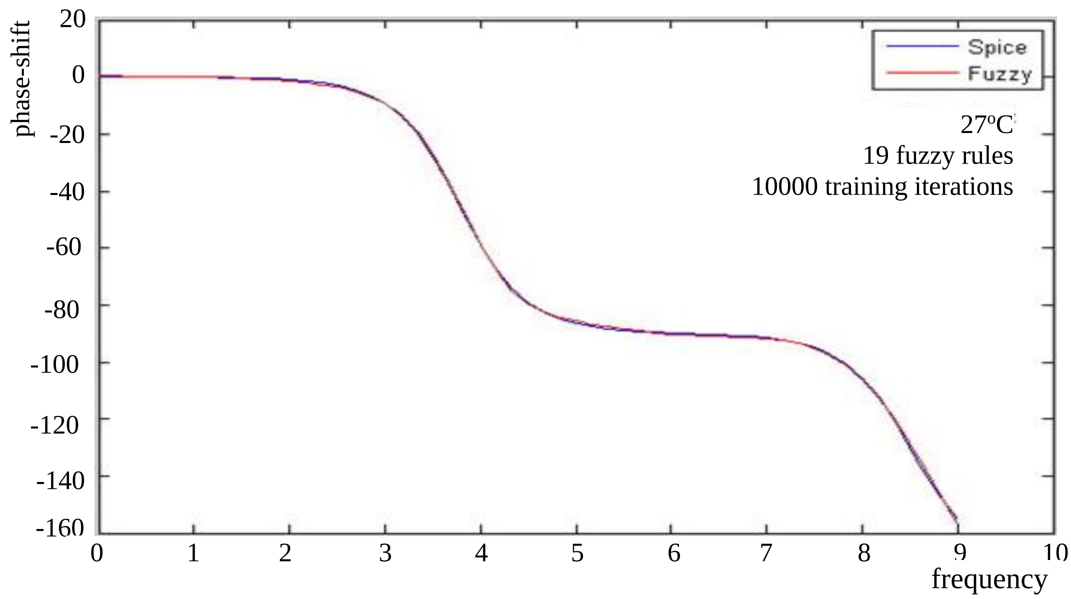


Fig. 21 Phase-shift at 27°C - comparison between the values generated by the fuzzy system and the reference values (Spice).

For a qualitative appreciation of the modeling accuracy, Fig. 21 presents a direct comparison between the reference values (Spice simulated) and the values computed with the fuzzy model, at the nominal temperature (27°C), across the entire frequency range. The fuzzy model present excellent accuracy, the values computed by the fuzzy system being (almost) identical with the reference ones, even in the highly nonlinear regions. The qualitative appreciation of the model precision is made for certain values of frequencies and temperature as they are presented in right-side of Table 10.

Simulink implementation of fuzzy functional model

The fuzzy functional model is implemented in both Matlab and Simulink. According with the problem to be solved, the user can chose the Matlab implementation (functions and scripts) or the Simulink model to be included in any complex system created in Simulink. The Simulink model is presented in Fig. 22.

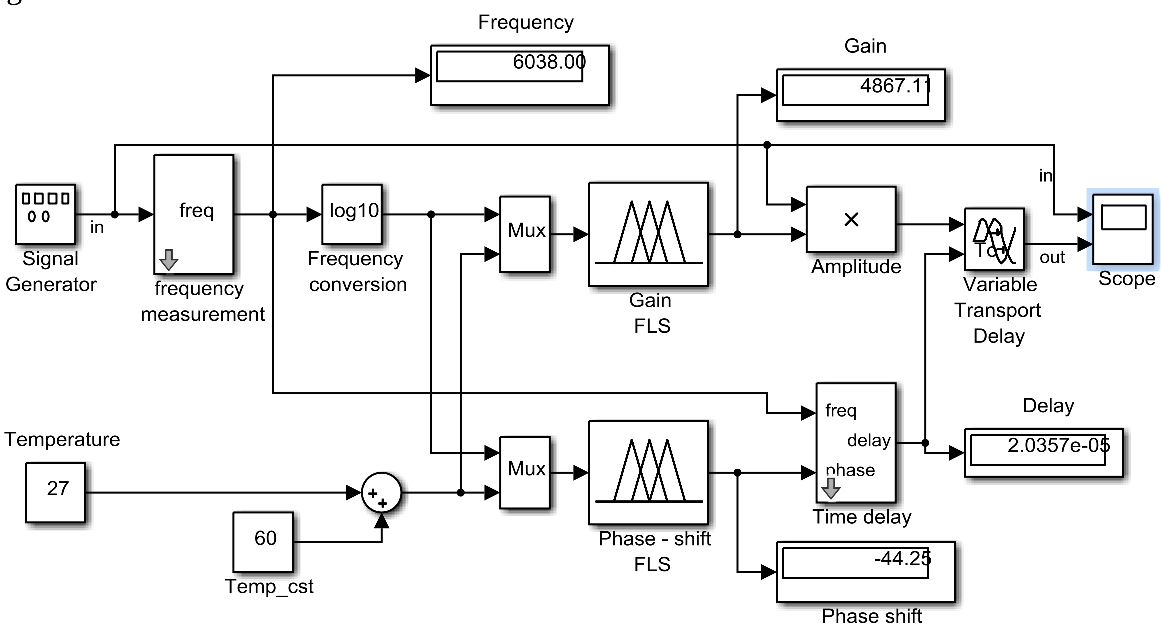


Fig. 22 The Simulink implementation of the fuzzy functional model.

The heart of the Simulink model consists of the two fuzzy logic systems (‘Gain FLS’ and ‘Phase-shift SLF’). Because they have two inputs, frequency and temperature, it is necessary to use a two-input multiplexer block (‘Mux’) in front of them. Let us remember that some data conversion was necessary to obtain accurate fuzzy systems. The operating temperature is set in the block called ‘Temperature’ (27°C in the exemplification presented in Fig. 22). The temperature is then shifted upward with 60°C (set in the ‘Temp_cst’) using a summing block.

The input voltage is generated by the ‘Signal generator’ block. To model the frequency behaviour of the amplifier we need the value of the frequency, so we have to compute the frequency for the time variation of the input signal. For this purpose the ‘frequency measurement’ block was developed and it computes the frequency based on the time measured between two successive zero crossing moments, meaning half of the signal period. It provides the frequency after the first half cycle of the input signal. This first half cycle will be treated as a transient regime, so the correct output signals will appear only after this transient regime. The decimal logarithmic conversion of the frequency is provided by the ‘Frequency conversion’ block. We can see the value of the frequency on the ‘Frequency’ display block (6038.00 Hz in Fig.22).

The ‘Amplitude’ multiplication block receives the input signal at one input and the value of the gain at the second input; its output signal being the amplified version of the input signal. To provide the correct output signal, the model has to include the phase-shift component, corresponding to the operating frequency. This task is carried out by the ‘Variable Transport Delay’ block, which delays the time-varying signal at its first input with the time value presented to its second input. The delay time is computed in the ‘Time delay’ block, using the phase-shift and the frequency.

The waveforms for the input and output voltages can be displayed on the “Scope” block. The values of the gain and phase-shift can be read on the ‘Gain’ and respectively ‘Phase shift’ display blocks (4867.11 for the gain and -44.25° for the phase-shift, in the exemplification in Fig. 22).

It is worth to mention that the Simulink implementation of the model is a general one; for a new analog module to be modeled, only the two fuzzy systems have to be replaced, without other modifications.

Some experimental results

The Simulink implementation of fuzzy functional model was tested for different scenarios: an input sine wave with 100 μV amplitude, but different frequencies and different temperatures. Fig. 23 presents the input and output voltages as they are seen on the oscilloscope (“Scope”) for 10 Hz frequency (passband) but different temperatures: 27°C and 100°C. There is no observable time delay (phase-shift) between the input and output voltages. For the nominal temperature (27°C) the gain computed by the Simulink model is 6880 (6882 in Spice), or 76.75 dB, so the amplitude of the output voltage was computed as 688.0 mV (688.2 mV in Spice). Increasing the temperature to 100°C, the computed gain decreases to 6489 (6490 in Spice), with the amplitude of the output voltage of 648.9 mV.

Another set of waveforms are plotted in Fig. 24 for a scenario with a frequency of 6.038 kHz at two different temperatures. This frequency was selected because it is the cutoff frequency at nominal temperature. For 27°C, the gain was calculated as 4867 (4872 in Spice) or 73.75 dB, the output voltage having 486.7 mV amplitude (Fig. 24a)). In the dB domains, the gain attenuation is -3 dB (from 76.75 dB in the passband down to 73.75 dB), confirming the cutoff frequency. The time delay computed by our Simulink model is 20.36 μs, corresponding to a phase shift of -44.25° (-44.93° in Spice). If the temperature decreases to -40°C, the gain is computed as 5314 (5319 in Spice) with an output voltage amplitude of 531.4 mV (Fig. 24b)). The computed time delay is 20.03 μs, corresponding to a phase-shift of -43.53° (-43.36° in Spice).

Fig. 25a) presents the waveforms at the nominal temperature at two frequencies in the transition band. For a frequency of 75 kHz, our model provides a gain of 553 (554 in Spice) and a time delay of 3.14 μs, meaning a phase-shift of -84.68 (-85.39 in Spice). For 1 MHz, the computed gain is 43 (46 in Spice) and the time delay is 0.25 μs that is a phase-shift of -90.24° (- 89.78 in Spice).

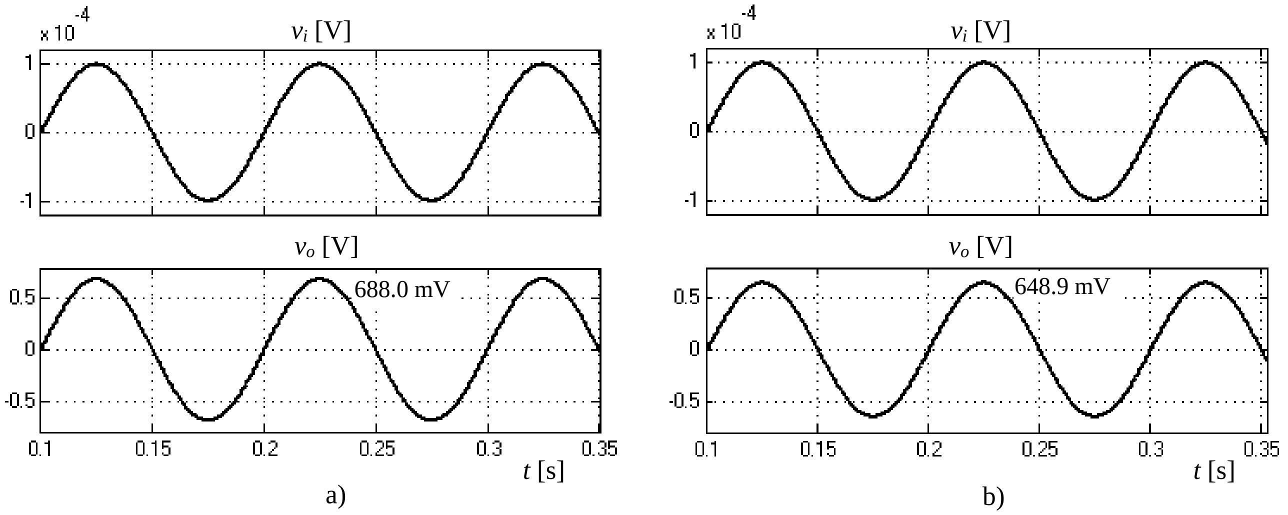


Fig. 23 Input and output waveforms of fuzzy functional model of FCOTA at 10 Hz: a) 27°C; b) 100°C.

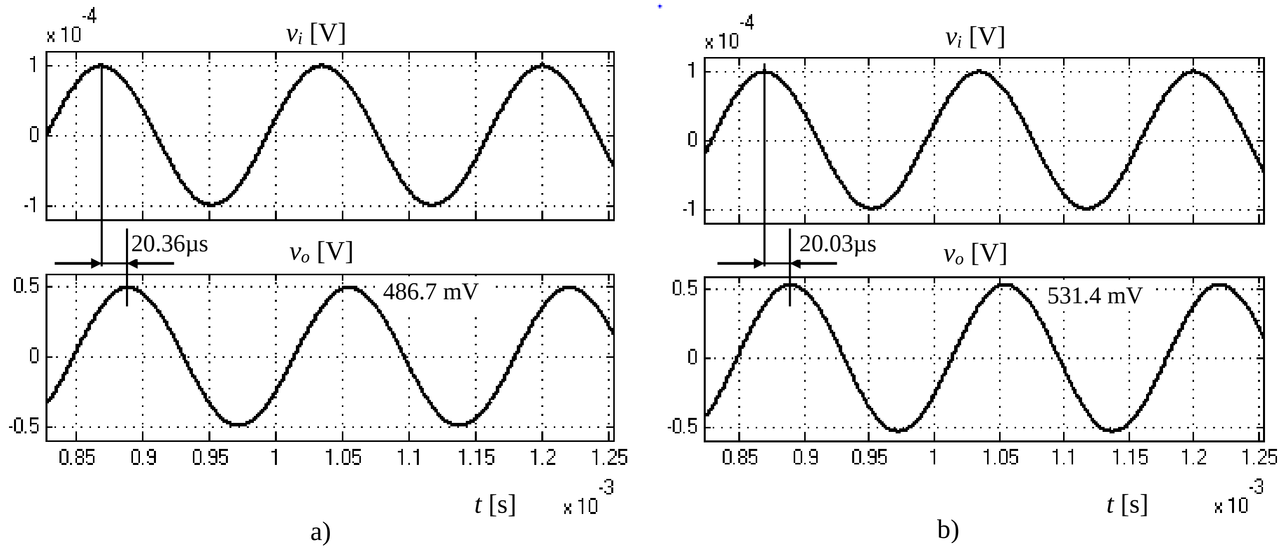


Fig. 24 Waveforms of fuzzy functional model of FCOTA at 6.038 kHz: a) 27°C; b) -40°C.

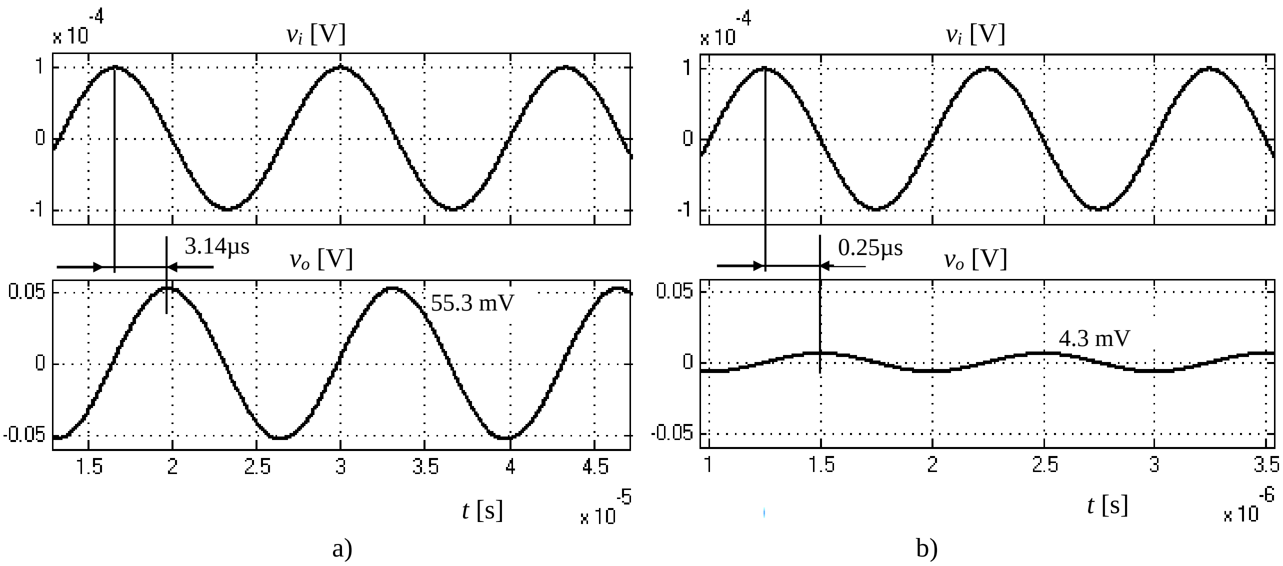


Fig. 25 Input and output waveforms of fuzzy functional model of FCOTA at 27°C a) 75 kHz; b) 1 MHz.

2.3.2. Fuzzy functional model for SOTA

The fuzzy functional model for the simple operational transconductance amplifier SOTA contains both modules (Fig.22). On one hand, it models the frequency response providing the output waveform for any sinusoidal input waveform small enough to maintain the amplifier in its linear region, across the full range of frequency and temperature. On the other hand, it models the voltage transfer characteristic in the passband, being capable to provide the output voltage, even if the amplitude of the input voltage (sinusoidal or triangular) sets the amplifier to operate outside of its linear region.

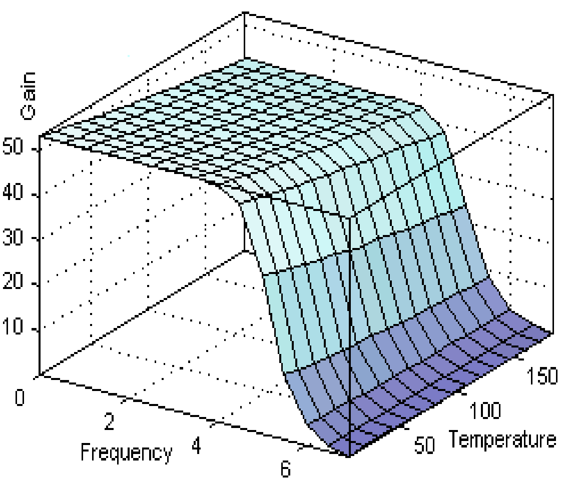
The development of the model, its implementation in Matlab / Simulink and experimental results were reported in a series of publications. The functional model that contains only the frequency response and temperature dependence is discussed in [Olt05c] for a Matlab implementation and in [Olt06a], [Olt06b] for the Simulink implementation. The functional model that include frequency response and voltage transfer characteristic are presented in [Olt05a], [Olt07] for a Matlab / Simulink implementation.

The schematic of the SOTA circuit is the one presented in Fig. 3. To obtain the data sets for the frequency response, the circuit was simulated using a parametric AC Sweep/Noise analysis, the parameter being the temperature. The frequency range was [1 Hz; 10 MHz], while the temperature was considered in the range of [-55^{o}C; +125^{o}C] with a step of 10^{o}C. Data sets for the VTC (Voltage Transfer Characteristic) were generated by simulating the circuit with a TRAN (transient) analysis and a PWL (piecewise linear) input voltage whose variation covers the entire input domain.

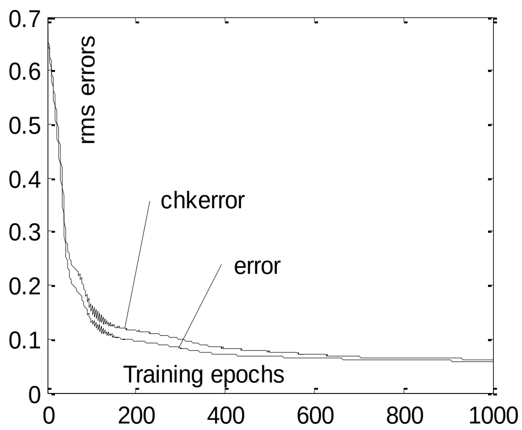
For the frequency response model, two fuzzy logic systems were generated and trained:

- Phase-shift FLS with17 rules trained 1000 training epochs;
- Gain FLS with 9 rules trained 1500 training epochs.

For each fuzzy system, 358 data pairs were used to generate the initial fuzzy systems, 35717 data pairs to train the fuzzy systems and 497 data pairs to check the fuzzy systems. As an illustration, Fig. 26a) presents the 3D representation of the final gain(frequency, temperature) fuzzy model and Fig. 26b) presents the error (*rmse*) evolution during training for both training and checking data sets. Let's remind the utilization of a logarithmic scale for the frequency and the existence of 60^{o}C offset for the temperature. The accuracy improvement of the model is very fast on the first 200 training epochs, and present, but less spectacular, for the next epochs (Fig. 26b). The errors in both data sets evolve similarly, meaning the data sets are consistent, all region of the function to be modeled being appropriately captured. No overtraining phenomenon appears, meaning that our system maintains good generalization capabilities.



a)



b)

Fig. 26 Gain FLS for SOTA: a) 3D final surface; b) errors evolution during training.

The fuzzy system for the phase-shift evolves in a similar manner, showing the dependence of the output signal phase on the frequency and on the temperature.

We tested our Simulink implementation of the functional model using a sine wave with various frequencies as the input signal, for different operating temperatures. A comparison between the results provided by our Simulink fuzzy model and the ones provided by Spice simulation (in Orcad environment) is presented in Table 11. One can notice the very good match of the values obtained with our fuzzy model and with Spice simulation, across the entire frequency range and across the entire temperature range. Three relevant frequency was selected for analysis: 1 kHz, in the passband; 75 kHz the cutoff frequency at 27°C; 1 MHz, in the transition band.

All fuzzy models provide very good results, especially in the passband. Slightly higher errors are encountered in the increased nonlinearity regions of the curves to be modeled. For the gain, the magnitude of relative error is maximum -0.50% at the cutoff frequency (75 kHz) at nominal temperature. Anyway, the difference between the model generated value (34.07) and the reference value (33.90) is in fact only 0.17, which confirms the high accuracy of the model. The maximum magnitude of the relative error is -1.36%, but this happens in the transition band (1 MHz) for a small value of the gain (2.95) the absolute error being only -0.04, which is negligible. For the phase-shift, the magnitude of absolute error is less than 0.4°, suggesting high modeling precision.

Table 11 Fuzzy models values vs. Spice values for frequency response of SOTA

Frequency	Temperature	Gain			Phase-shift [°]		
		Spice	Fuzzy model	Relative error [%]	Spice	Fuzzy model	Absolute error
1 kHz	-40°C	52.24	52.22	-0.04	-0.66	-0.81	-0.15
	27°C	48.10	48.10	0.00	-0.76	-0.66	0.10
	100°C	44.15	44.17	0.05	-0.86	-0.85	0.01
75 kHz	-40°C	39.47	39.30	-0.43	-40.90	-41.06	-0.16
	27°C	34.07	33.90	-0.50	-44.92	-45.09	-0.17
	100°C	29.40	29.47	0.24	-48.25	-48.65	-0.40
1 MHz	-40°C	4.51	4.49	-0.44	-85.07	-85.14	-0.07
	27°C	3.61	3.61	0.00	-85.70	-85.68	0.02
	100°C	2.95	2.91	-1.36	-86.20	-86.32	-0.12

The fuzzy functional model was first tested for the frequency response using at its input a sine wave with 10 mV amplitude, for certain combinations of frequency and temperature.

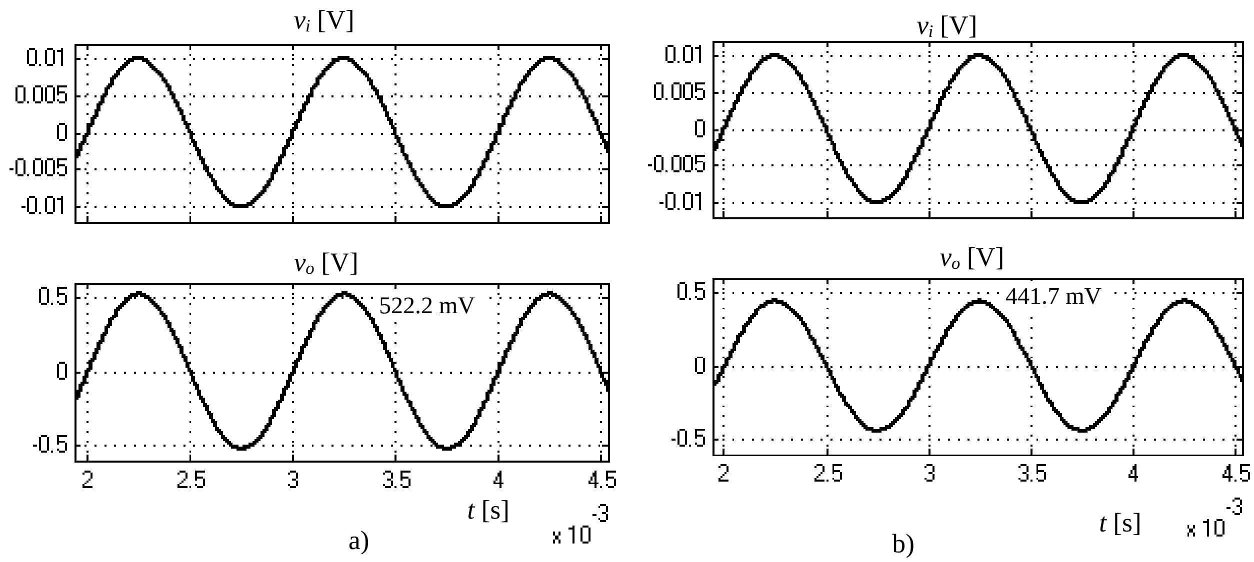


Fig. 27 Input and output waveforms of fuzzy functional model of SOTA at 1 kHz: a) -40°C; b) 100°C.

Fig. 27 presents the input and output voltages as they are seen on the oscilloscope in the Simulink model for 1 kHz frequency (passband) for two different temperatures: -40°C and 100°C. For -40°C temperature, the gain computed by the Simulink model is 52.22 (52.24 in Spice), so the amplitude of the output voltage was computed as 522.2 mV (522.4 mV in Spice). The phase-shift is -0.81° (-0.66° in Spice). Increasing the temperature to 100 °C, the computed gain decreases to 44.17 (44.15 in Spice), with the amplitude of the output voltage of 441.7 mV, while the phase-shift is -0.85° (-0.86° in Spice).

The waveforms at the nominal temperature (27°C) and cutoff frequency (75 kHz) are plotted in Fig. 28a). The gain was obtained as 33.9 (34.07 in Spice) or 30.60 dB, the output voltage having 339 mV amplitude. In the dB domain there is an attenuation of -3.04 dB (from 33.64 dB in the passband, down to 30.60 dB), confirming the cutoff frequency. The time delay computed by the Simulink model is 1.67 µs, corresponding to a phase shift of -45.09° (-44.92° in Spice).

Fig. 28b) presents the waveforms for the nominal temperature at a frequency in the transition band, 1 MHz. The model provides a gain of only 3.61 (3.61 in Spice) and a time delay of 0.27 µs, meaning a phase-shift of -85.68 (-85.6 in Spice).

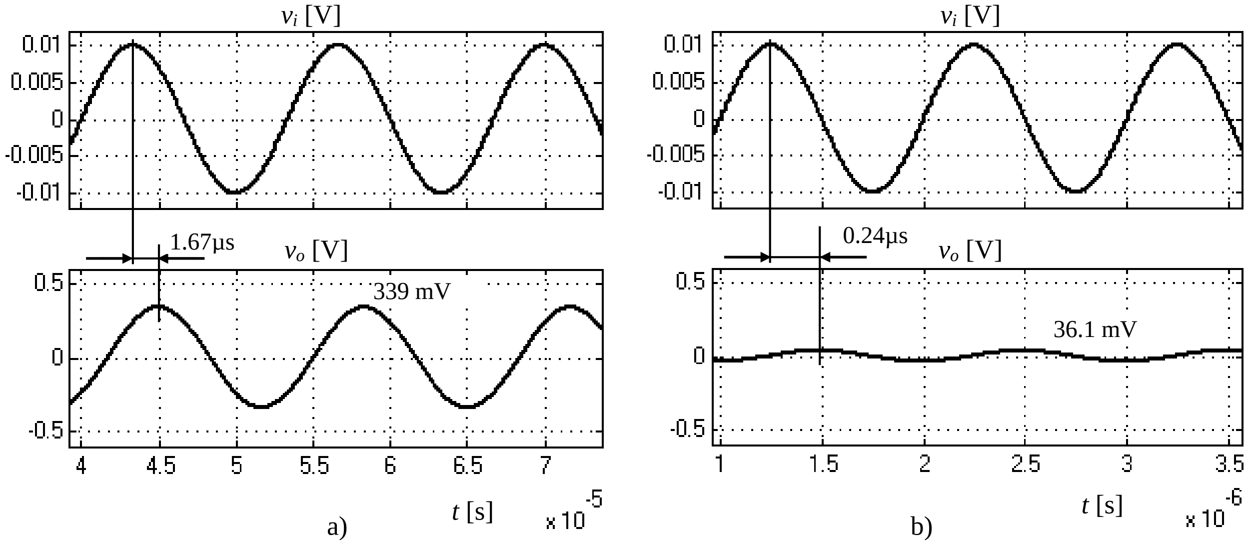


Fig. 28 Input and output waveforms of fuzzy functional model for SOTA at 27°C a) 75 kHz; b) 1 MHz.

For the VTC fuzzy model, a first set of 4008 data points as training data set, and a second set of 58 data points as checking data set, was generated. The range of the voltage applied to the input was set to $v_i \in$[-200 mV, 200 mV]. The final fuzzy system that models the v_i - v_o transfer characteristic contains 7 fuzzy rules.

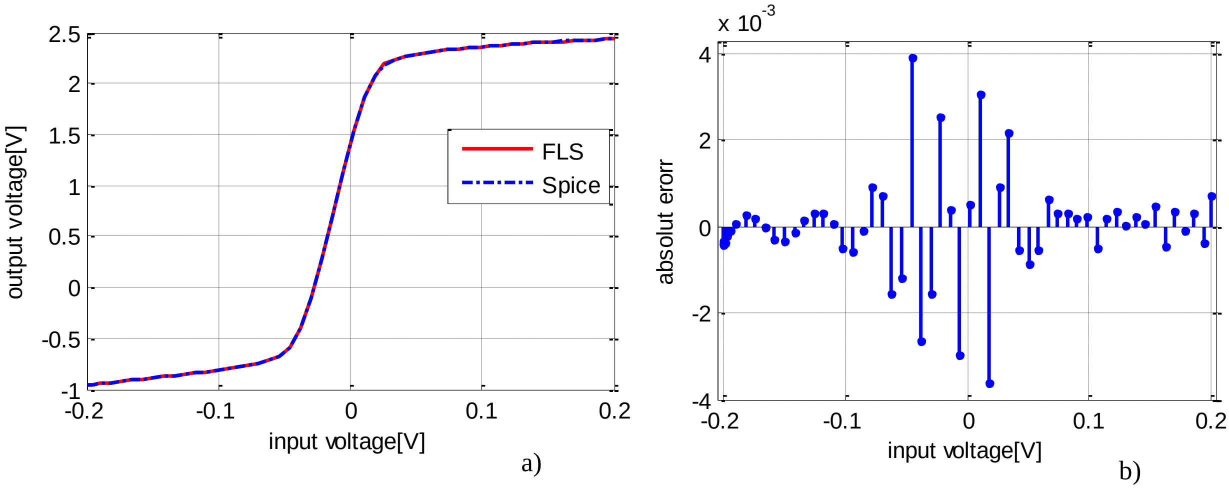
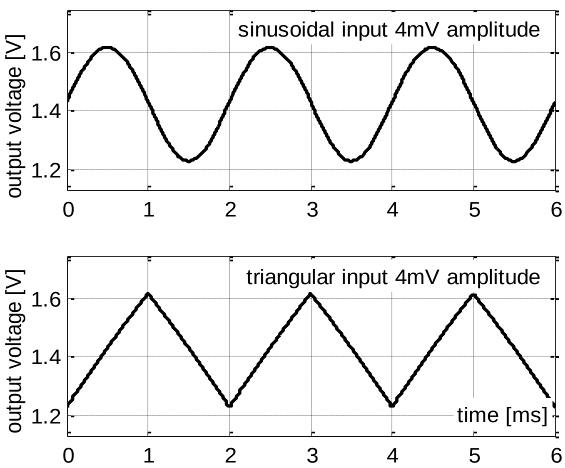


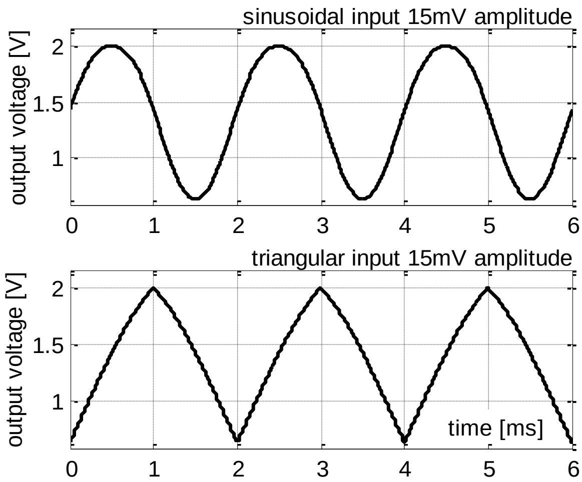
Fig. 29 VTC for SOTA, fuzzy model (FLS) vs. Spice: a) VTC curves; b) absolute errors.

The graphical representation of the v_i - v_o VTC generated by the fuzzy model versus the Spice is presented in Fig. 29a). As one can see the fuzzy model (FLS) presents high accuracy, the two curves being (almost) superimposed. The absolute error, as difference between the output voltage computed with the fuzzy model and the output voltage from Spice simulation, for the same input voltage is highlighted in Fig. 29b). The errors are very low, their order of magnitude being a few mV. The largest errors occur for the input voltage range [-0.07 V; +0.07 V], where the curve presents the maximum nonlinearities (the two knee regions) and the maximum variations of values (active region with largest slope). The top three maximum absolute errors are: 3.9 mV (-584.7 mV instead of -588.6 mV); -3.6 mV ((2.0774 V instead of 2.0810 V); 3 mV (1.862 V instead of 1.859 V).

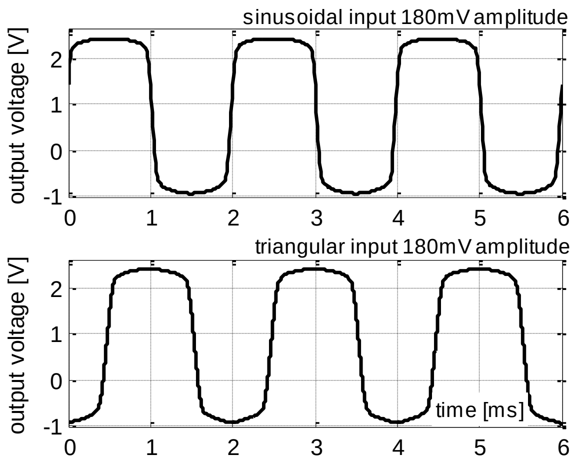
The VTC fuzzy model was validated using a 500 Hz sine wave and triangular wave with various amplitudes. Fig.30a) presents the output voltages for an input amplitude of 4 mV and 500 Hz frequency. The amplifier works in the linear region, the extreme values being 1.616 V and 1.228 V for both waveforms. The Spice simulation gives 1.614 V and 1.231 V respectively.



a)



b)



c)

Fig. 30 Output voltage for SOTA at 500 Hz input voltage (sine wave – top, triangular – bottom) for different amplitude of input voltage: a) 4 mV; b) 15 mV; c) 180 mV.

If we increase the input amplitude up to 15 mV the amplifier enters the nonlinear regions and our fuzzy model gives the response presented in Fig. 30b). The extreme values are 2.004 V and 0.635 V for both waveforms. The Spice simulation gives 2.005 V and 0.634 V respectively. The reader can see the effect of the nonlinear gain, the output waveform being distorted, especially on the positive half wave, the effect being more conspicuous on the triangular waveform.

Finally, with 180 mV input amplitude we push the amplifier in the saturation region, on both sides, as one can see in Fig. 30c). The extreme values are 2.423 V and –0.929 for both waveforms, the Spice simulation giving the same values: 2.423 V and –0.929 V respectively.

As for the frequency behaviour, the fuzzy model of the voltage transfer characteristic is very accurate on the entire range of input voltages for both waveforms used.

3. Efficient waveform prediction in electronic systems

The electronics industry must reduce the time-to-market while increasing complexity, therefore the pressure on a thorough verification is high, especially for safety-relevant systems. To cope with this, pre-silicon verification, by modeling and simulation, starting from a system-level, has gained a lot of interest because of the opportunity to verify at early stages the application-fitness. While digital circuitry and software have established means of high-coverage verification, analog and mixed-signal systems need to be verified under all allowed operating conditions, given variations of components and loads. This introduces a lot of overhead since it is not possible to formally (analytically) verify such systems, and a reasonable coverage, when the number of variations is high, translates into a high number of long simulations.

The standard circuit-level simulation based design approach can only be used when the designer has sufficient time for running simulations to analyse and optimize the circuit. Unfortunately, this is usually not the case, since the timeline for a design process is very short. Simulation times for complex circuits are, most commonly, very long and it is not feasible to conduct an exhaustive search to find the optimal circuit [Moh14]. The use of a model that provides satisfactory simulation results, keeping the simulation resources at acceptable values, comes as a natural course of action. That particular model cannot be the circuit model, because it would lead to long and complex simulations, but something with a higher level of abstraction; moreover, that particular model should be something that doesn't actually simulate the circuits, but only produces their waveforms in different points of interest. Hence, the idea of a metamodel, or a model of the model, emerges.

The challenge is to generate the waveform by its numerous time samples as a function of different operating conditions described by a set of system parameters. Some important aspects are:

- the number of samples needed to build the model must be reduced i.e. the order of magnitude smaller than the Monte-Carlo circuit simulation methods;
- the evaluation time of the underlying results - model must be much smaller, in comparison to the simulation time;
- such model must be of reasonable complexity (or size), and must be sufficient to estimate new signals, with respect to the input variations;

The problem to be solved consists in the fact that the model has to generate the entire output waveform for every combination of the input parameters. In other words, we should simultaneously generate all the time samples of the waveform; so we need a model for a nonlinear multidimensional and multivariable function that implements a functional nonlinear mapping from a vector of input parameters to the vector of time samples of the waveform:

$$[s_1, s_2, ..., s_Q] = f(p_1, p_2, ..., p_M) \tag{23}$$

where $[s_1, s_2, ..., s_Q]$ is the output vector of the time samples of the waveform, Q is the number of time samples, $[p_1, p_2, ..., p_M]$ is the input parameter vector, and M is the number of parameters.

According with the literature, there are no systematic approaches of generating complete waveforms in different points of a complex system as a function of dependent parameter combinations, using a cheap and fast substitute (e.g. metamodeles) for extensive simulation. However, a literature review shows that efforts have been put into the approximation of waveforms, regardless of their nature (electrical or non-electrical), using various techniques. There are some methods [Sim01], [Raf09] which apply heuristics to the outputs of the simulation, in order to predict only not-yet simulated individual points, not the entire waveform. A series of approaches imply applying statistics on the output values, but do not build predictive models of the signal of interest, therefore can only assess the quality of the system under consideration, not to improve it [Nig11].

A method to predict post-layout waveform by System Identification, based on the fact that the waveforms of pre-layout and post-layout are always correlated, is described in [Hua15]. It uses linear models (linear ARX, impulse response, and transfer function) and non-linear models (non-linear ARX and Hammerstein-Wiener). The predicted waveforms cannot always achieve very high accuracy; however, they can essentially predict the trends of the waveforms, which can guide the designers to diagnose and optimize their designs.

A solution to the problem of quickly and accurately predicting gravitational waveforms within any given physical model is discussed in [Fie14]. The solution constructs a surrogate model for a fiducial set of waveforms in three offline steps. Results show that these surrogate models provide a reduction of the evaluation time with three orders of magnitude, compared to the standard methods, while maintaining a high accuracy.

The scope of this research area was twofold: to develop a metamodeling development procedure and to prove the efficiency and accuracy of metamodels in an industrial scenario. The research was initiated in collaboration with an important company (Infineon Technologies) who provided us with a series of industrial early-stage simulation data from the automotive domain. The system under study was a typical ECU (electronic control unit) for which the control signal is influenced both by the DUT (device under test) as well as the load variation, as described in [Raf10]. The exact switch-on time, given as value and pulse duration, is crucial when it comes to driving the squib of the airbag. The SystemC-AMS model is subject to simulations, to extract the output signals corresponding to applied variations on the DUT and Load parameters (Table II in [Raf10]).

3.1. The metamodeling procedure

Generally speaking, a metamodel, or a surrogate model, is a model of the model, i.e. a simplified model of an actual model of a circuit, system, or software like entity [Moh12], [Gar12]. A metamodel can be a mathematical relation or algorithm representing input - output relations. A model is an abstraction of phenomena in the real world; a metamodel is yet another abstraction, highlighting properties of the model itself. Various types of metamodels include polynomial equations, neural network, Kriging (interpolation for which the interpolated values are modeled by a Gaussian process governed by prior covariances), etc. [Moh12]. Metamodeling typically involves studying the input-output relationships and then fitting proper metamodels to represent that behaviour.

The metamodeling process implies a mathematical representation of the output, based on a prediction equation or algorithm, language and tool independent, reusable for different specifications, and which can be applied using non-EAD tools like MATLAB. The key points of metamodeling, according with [Moh12] are:

- accuracy - capability of generating the system response over the design space;
- efficiency - computational effort required for constructing the metamodel;
- transparency - capability of providing the information concerning contributions and variations of design variables and correlation among the variables;
- simplicity - simple methods should require less user input and be easily adapted to different problem.

To capture the characteristics of certain set of waveforms, common and differentiating, we have to use a quite high sample frequency that will generate thousands or even many more data points (time samples). To build an training-based model capable to fully generate all that values at once is not practical from the point of view of training resources (time, computer memory, size of the training data set) and also form the point of view of implementing and simulating such a huge model. To reduce the problem dimensionality the proposed solution is inspired from the pattern recognition domain, where to recognize (classify) a pattern, one have to describe it first the pattern by a reduced set of representative features [Gyo11],[Dud01].

The general diagram of the proposed metamodel development procedure is displayed in Fig. 31. The procedure needs all available primary data, consisting in the full waveform family together with the associated parameter combinations for each individual waveform. To describe each waveform by a set of features (not time sample values) a discrete transform (Fourier or wavelet) of the signal is applied, a transform that produces a set of decomposition coefficients. The transform itself does not make any dimensionality reduction. The dimensionality reduction consists in selecting a small subset of the most relevant features (coefficients). Once the coefficients are selected, the data sets across the entire waveform family should be generated and prepared for the next step. An optimal ANN is now created and trained so that it would be capable to generate the most important coefficients when it is fed with different parameter combinations, even for new parameter combinations that weren't part of the training data. The final ANN and other elements (especially some information about the used transform and the reverse transform) are integrated into the metamodel for waveform prediction.

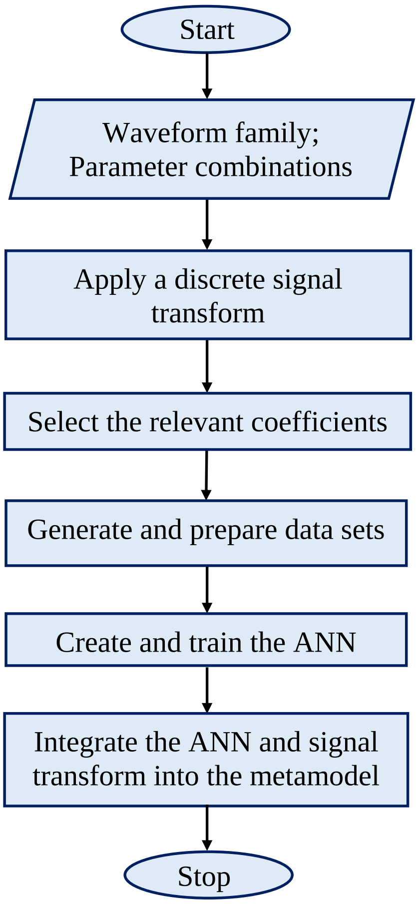


Fig. 31 The metamodeling procedure for waveform prediction.

Using the metamodeling procedure, two metamodels for waveform predictions were developed:

- SFTM – Simple selection Fourier Transform based Metamodel that uses a discrete Fourier transform and a simple selection of relevant coefficients using an imposed threshold for the coefficients magnitude. Different aspects and results involving this approach were published in a scientific paper [Olt13], in a dissertation thesis [Pro13], and presented in the SSET2013 student competition, where it was awarded with the 3rd prize in the Master/Doctor section.
- OWTM – Optimal Wavelet Transform based Metamodel that uses a discrete multilevel wavelet transform and a genetic algorithm optimization to detect the optimal wavelet transform and to identify the most relevant decomposition coefficients. Certain elements and results developing and using this metamodel were published in three scientific papers [Far14], [Olt15a], [Olt16].

As it results from the diagram in Fig. 31, the heart of our metamodel is an artificial neural network that plays the role of predicting right values for the most important coefficients only considering the input parameter combinations. Developing a fitted ANN requires a supervised learning procedure based on a set of numerical data. The quality of the trained ANN is directly connected with the quality of the data set involved in the training process. Some of the main characteristics that recommend the artificial neural networks are [Fer05], [Kha10],[Zai08]: ANNs are universal approximators that can learn data by example and can approximate any complex nonlinear multi-variable function with any desired accuracy. Supplementary information about ANN generation, training and performance evaluation can be found in [Kon06],[Rut08].

There are some approaches of using ANNs to tackle different issues in waveform processing. Back Propagation Neural Network and Radial Basis Function Neural Network are used to develop behavioural models of a RF power amplifier, the predicted output signal corresponding to sampling points of the amplifier output waveform value [Jiu10]. In [Mon12], an ANN is used for detection and classification of electrical disturbances in three-phase systems. Automatic detection of spikes in electroencephalograms (EEG) can be solved using neural networks, as described in [Sur13].

In [Sha10], a neural network provides a means of determining a degree of belief for each identified disturbance waveform in Power System. Three types of neural networks (multilayer perceptron, radial basis function and wavenet) are used in [Mog09] to estimate the feedback signal for a vector controlled induction motor drive.

In [Tha11] the configuration of an ANN (number of hidden units in the hidden layer, transfer function to use at the hidden layer, and transfer function to use at the output layer) is optimized such that the network's approximation error for signal approximation problems is minimized. Three different signals were considered there: Boolean XOR function, sinusoidal signal, and a signal representing the activity measurements on a server system.

[Tei15] proposes a method that combines ANNs with the wavelet decomposition to generate short-term global horizontal solar radiation forecasting, which is an essential information for evaluating the electrical power generated from the conversion of solar energy into electrical energy. The forecasts derived from the proposed method had a significantly higher correlation with the time series observations of global horizontal solar radiation when compared with the forecasts arising from using only the ANN (i.e., without considering the wavelet signals as input patterns).

Global or local thresholds can be used to separate the most important coefficients of the wavelet decomposition, which can further on be shank [Coi95] or normalized. Such a technique is applied in [Ye13] for noise reduction in cochlear implants. An energy-based wavelet denoising method is proposed in [San14], and applied to hydrological time series.

3.1.1. Simple Fourier Transform based Metamodel

SFTM structure

The block diagram (input - output) of the metamodel is presented in Fig. 32. It presents *M* inputs representing the input parameter $X = \{x_1, x_2, ..., x_M\}$ whose different combinations will generate specific output waveforms by its time samples $S = \{s_1, s_2, ..., s_K\}$ (*K* is the number of samples).

The development of the entire model, involves three phases:

1. Training data preparation;
2. Neural network training;
3. Output waveform prediction.

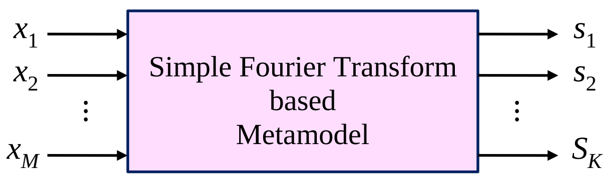


Fig. 32 Block diagram of the SFTM.

The detailed diagram of metamodel developing and utilization process is illustrated in Fig. 33 [Olt13], [Pro13]. The diagram also highlights the three phases, which are discussed further on.

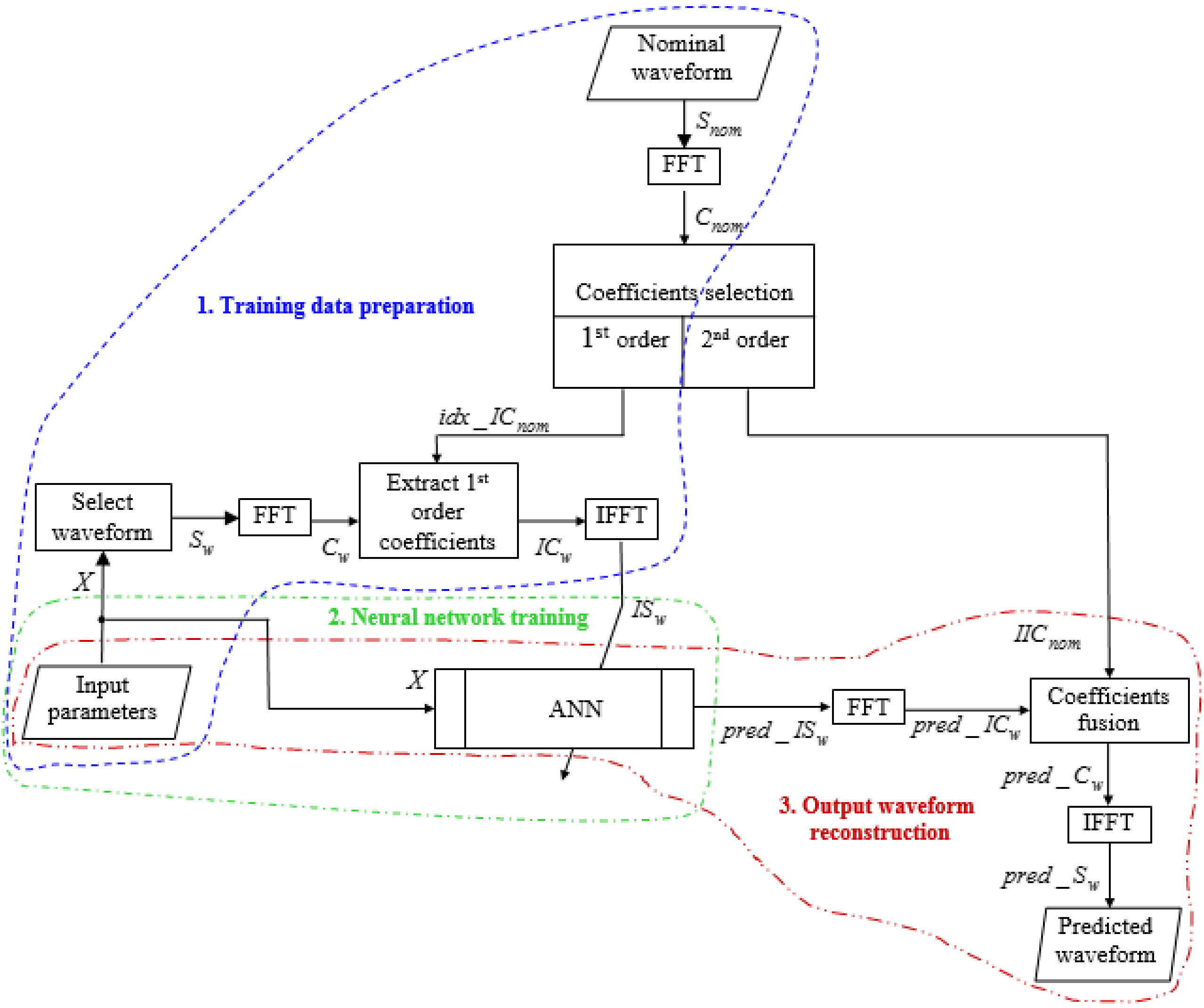


Fig. 33 The detailed diagram of SFTM developing and utilization process.

Training data preparation

In order to reduce the dimension of the data used to train the neural network, we resorted to a nominal waveform represented in terms of its time samples S_{nom} (see Fig. 33). Applying the Fast Fourier Transform (FFT) on this waveform results in domain change, from time samples to a new domain consisting of FFT coefficients C_{nom}. A coefficient (feature) selection algorithm was used to select from the new domain those coefficients considered the most important ones (according to their magnitude), of 1st order, denoted IC_{nom}. For these 1st order coefficients, we are not interested in their value, but in their indices (positions), stored as a vector idx_IC_{nom}. The remaining coefficients, considered as 2nd order, IIC_{nom} (values and indices) are stored for later use, in the waveform reconstruction phase.

The waveforms in the waveform family (data set) are determined by a combination of input parameters. For each individual set of parameters X, the corresponding waveform is selected. The coefficients vector C_w results from applying the FFT on the time samples of waveform S_w. From these coefficients only the ones corresponding to the 1st order indices extracted from the nominal waveform were taken into account (IC_w). In this way, the size of the resulted data set decreases substantially. Because the resulted data set is composed of complex numbers and the artificial neural network (ANN) cannot be trained with values of this format, an Inverse Fast Fourier Transform (IFFT) is applied to the IC_w data set resulting the associated 1st order time samples (IS_w), which are further used to train the neural network.

Neural network training

In our approach, we have used a feed-forward, back-propagation ANN, namely a multi-layer perceptron, with three layers: input, hidden and output. According with [Pri15], where it is stated that, from a function approximation perspective, the single hidden layer is quite adequate as the basic topology and due to the fact that a two-layer feed-forward network (one hidden layer and one output layer) with sigmoid hidden neurons and linear output neurons, can fit multi-dimensional mapping problems arbitrarily well [NNT], this is the solution for the neural network architecture adopted here. The number of neurons on the input layer is the number of input parameters, while the number of neurons on the output layer equals the number of selected coefficients. For the hidden layer, the number of neurons was determined via a series of trial runs, in order to obtain an optimal network structure, with minimum error.

The input data from the training data set are the parameter combinations, organized in a matrix *P*:

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1N} \\ p_{21} & p_{22} & \cdots & p_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ p_{M1} & p_{M2} & \cdots & p_{MN} \end{bmatrix} \tag{24}$$

The number of rows (*M*) represents the number of input parameters that are varied, while the number of columns is given by the number of combinations of input parameters *N*, which is, in fact, the number of available waveforms in the family.

The target of the neural network is the *T* matrix where each column contains the vector of 1st order time samples (IS_w) describing each waveform. *N* is the number of waveforms in the data set and *Q* is the number of 1st order time samples.

$$T = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1N} \\ t_{21} & t_{22} & \cdots & t_{2N} \\ \vdots & \vdots & & \vdots \\ t_{Q1} & t_{Q2} & & t_{QN} \end{bmatrix} \tag{25}$$

This way, after training the artificial neural network it can be further used to generate the necessary set of predicted time samples ($pred\_IS_w$) corresponding to the position of 1st order coefficients, for any parameters combination applied to its input.

Output Waveform Reconstruction

The reconstruction of the output waveform is realized using both the 2nd order components stored in the data preparation phase, and the 1st order components provided by the ANN (Fig. 33). A fusion operation (Coefficients fusion) between 1st order coefficients ($pred\_IC_w$) and 2nd order coefficients (IIC_{nom}) is involved. The 1st order coefficients were obtained by applying a FFT operation on the 1st order samples ($pred\_IS_w$) predicted by the neural network for the parameter combination applied to its input.

The output waveform is finally generated by its time samples ($pred\_S_w$) by applying the IFFT once again on the predicted coefficients $pred\_C_w$.

Experimental results

The data used for conducting the experiment consists of a set of 200 waveforms ($N = 200$), generated by 200 combinations of the 10 input parameters (*M* =10). The variation of these factors takes values between -1 and 1. These numbers do not represent absolute values of the parameters, but normalized values, -1 representing the minimum value, and +1 representing the maximum value. Supplementary, a nominal waveform has been provided in order to develop this system. Each waveform used to train the neural network is described by a number of ca.13600 samples, as they are presented in Fig 34.

According to the metamodel development procedure presented in Fig. 33, a number of 105 1st order coefficients was selected from the nominal waveform after FFT, the rest of the coefficients representing 2nd order components, used to reconstruct the waveform. As a consequence, the data set used to train the ANN contains only 105 samples, instead of 13600. This translates into a substantial data dimensionality reduction, by a factor of 130, which ensures the avoidance of risk concerning memory issues, and also assures a reduced time for training.

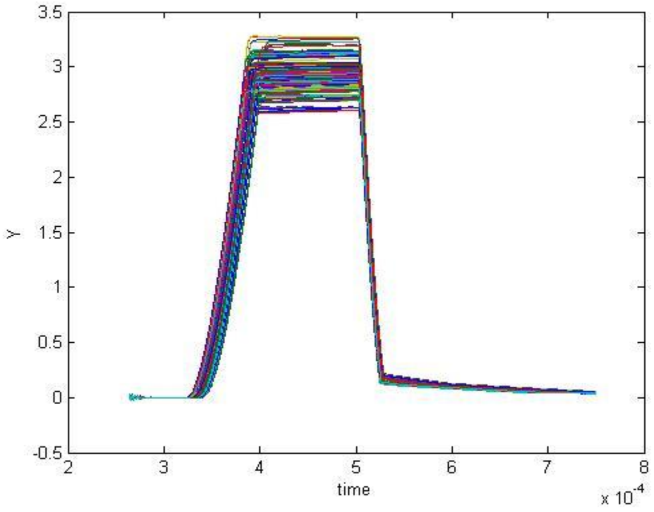


Fig. 34 Training data set described by 200 waveforms sampled in 13600 points.

The number of neurons in the hidden layer was selected from a series of trial runs in order to obtain an optimal network structure with minimum error. Finally, the network has 10 neurons in the hidden layer and 105 neurons in the output layer.

The full data set consists of 200 waveforms corresponding to the combinations of the input parameters, sampled in 105 points, describing the main characteristics of the waveforms. For the training procedure, the full data set was split into three data subsets: training subset (70% of the data set), validation subset (15% of the data set) and testing subset (15% of the data set). The training subset is used to train the neural network, adapting its parameters (weights and biases). The validation subset supervises the training, detecting the overfitting phenomenon. The testing subset, considered as an independent data subset, measures the performance of the neural network, inasmuch as that is not at all involved in the training process.

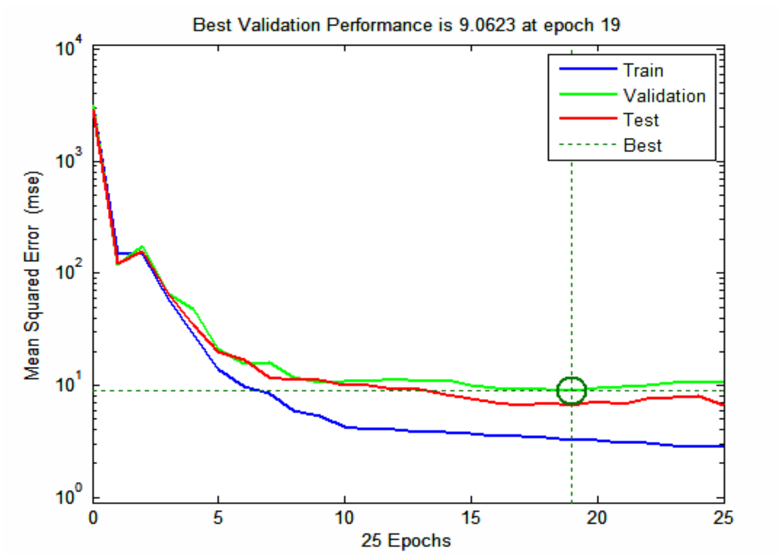


Fig. 35 Performance validation graph, training the ANN.

After successfully training the neural network, some analysis concerning the performances of this process was made. The performance validation graph is presented in Fig. 35. This figure illustrates the evolution of the mean squared error within the three subsets, during of the training. In the first 10 training epochs one can see a steep improvement (decrease) of the errors in all data subsets. Then, the training enters the phase of "fine tuning". When the overfitting phenomenon occurs (after 19 epochs), the training stops.

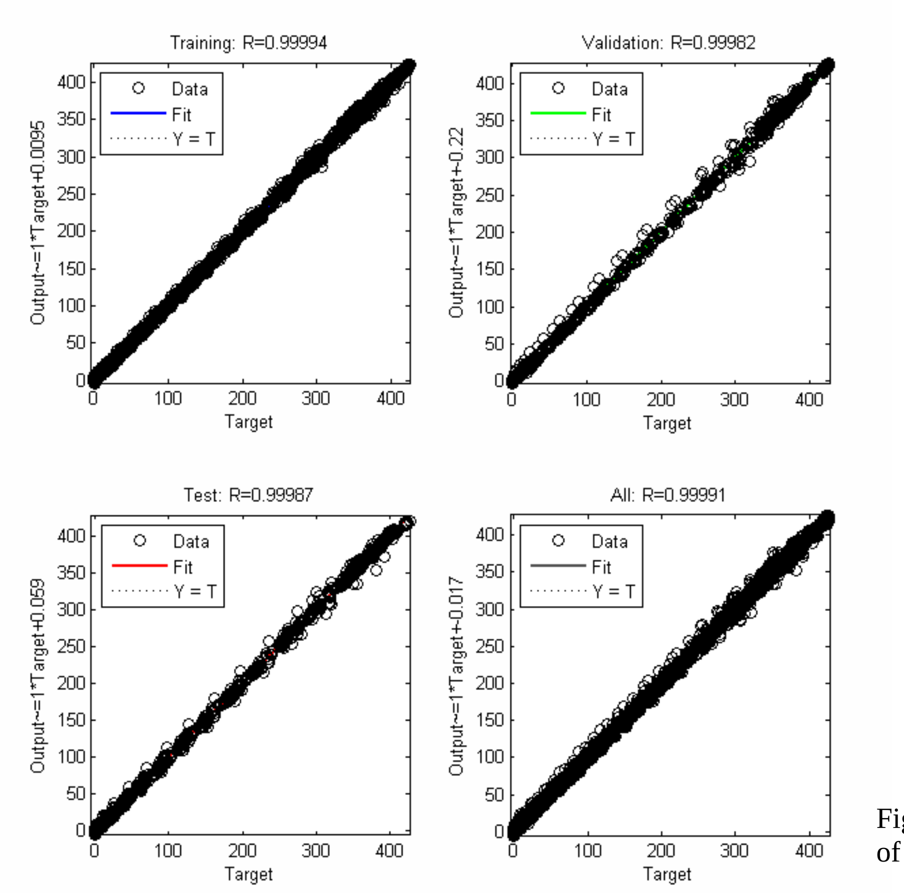


Fig. 36 Regression analysis of the neural network.

A regression analysis is also performed, the results being presented in Fig. 36. The graph illustrates the linear regression of targets (as reference values) relative to the outputs (predicted values) of the neural network. The regression equation is:

$$Output = a \cdot Target + b \tag{26}$$

where *a* is the slope of regression fit and *b* is the offset of regression fit.

An ideal fit (network outputs match the targets exactly) means *a* = 1 and *b* = 0, and a regression value *R* =1. It is easy to see that our neural network presents extremely good fitting performances in all data subsets. The slope of the regression fit is an ideal one (1 for all subsets), while the offsets have very low values for all subsets (0.0095 for the training subset, 0.22 for the validation subset and 0.059 for the testing subset). The regression value is almost 1 in all cases: $R = 0.99994$ for training, $R = 0.99982$ for validation, $R = 0.99987$ for testing, with a resulting $R = 0.99991$ for the entire data set.

This means the neural network provides very good generalization capabilities; in particular it can correctly predict a waveform corresponding to a new combination of input parameters, combination that was not included in the data set that actually trained the network. For all 30 waveforms from the testing subset, the results confirm the previous statement (see graph in bottom-left side of Fig. 36).

To appreciate the goodness of fit of the predicted waveform compared with the corresponding reference waveform, Fig. 37a) illustrates both an arbitrary chosen waveform from the data set (a reference waveform) and the corresponding waveform predicted by the metamodel. One can easily appreciate the quality of the prediction from the qualitative point of view, namely that the two waveforms are almost identical.

For a quantitative appreciation of the prediction accuracy, Fig. 37b) presents the sample-by-sample absolute error between the reference waveform and the predicted waveform. The error values are both positive and negative, around 0. The maximum error is 0.127, as a difference between the values of reference and predicted waveforms, for an expected value of 3.12. In this point we also have the maximum relative error of 4%. It is worth mentioning that the maximum error happens (as expected) in the most difficult region of the waveform with the maximum nonlinearity, where the waveform changes its shape from an almost vertical segment to a horizontal segment. The mean value of the absolute error of the whole waveform is 0.014. These results show that the metamodel can effectively predict a waveform, for any new combination of parameters that were not used in the metamodel development phase.

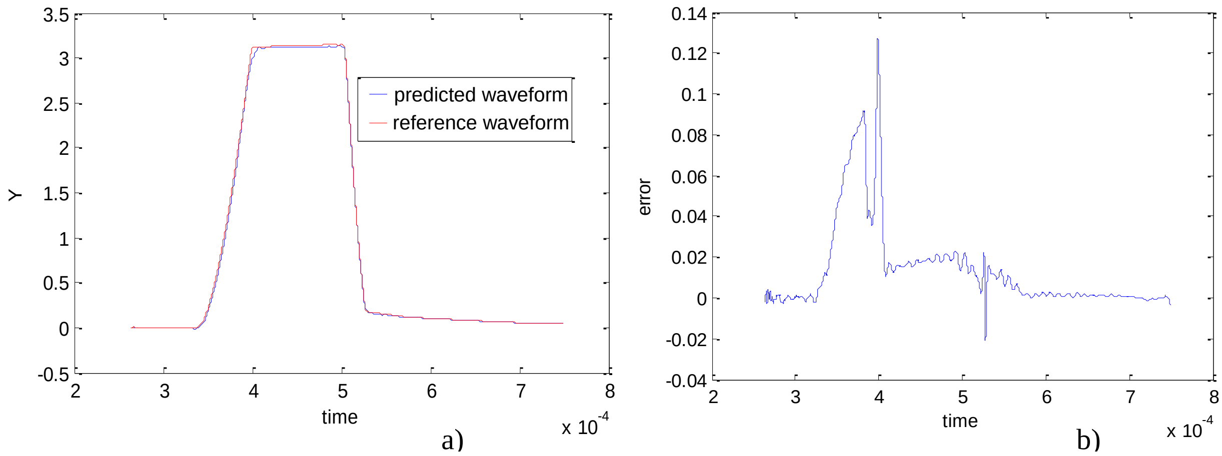


Fig. 37 Results for an arbitrary chosen waveform from the testing subset: a) predicted waveform vs. reference waveform; b) sample-by-sample absolute errors.

3.1.2. Optimal Wavelet Transform based Metamodel

Structure of OWTM

The block diagram of the Optimal Wavelet Transform based Metamodel (OWTM) is presented in Fig 38 [Olt16].

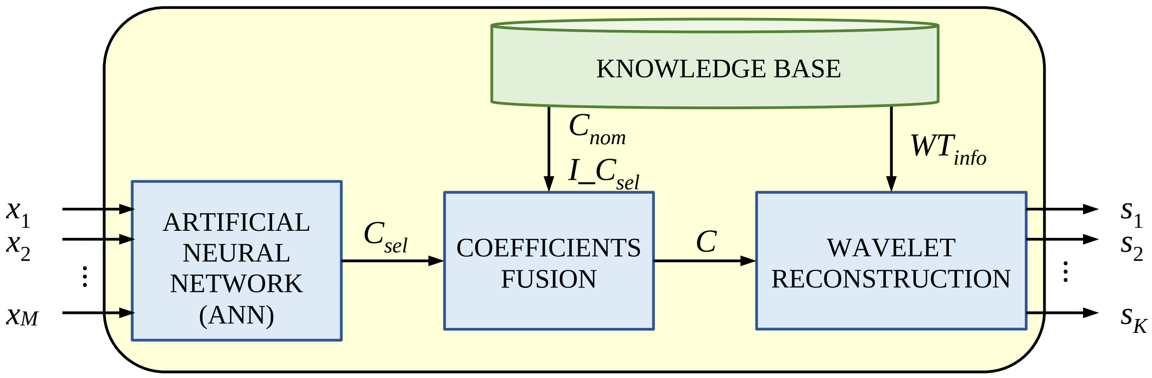


Fig. 38 Block diagram of OWTM.

The proposed metamodel for waveform generation has to rapidly produce an accurate approximation of the output waveform by its time samples $S = \{s_1, s_2, ..., s_K\}$ (K is the number of time samples) for any combination of the input parameters $X = \{x_1, x_2, ..., x_M\}$ (M is the total number of parameters).

The proposed metamodel is a knowledge-based one, containing one block that stores knowledge (KNOWLEDGE BASE) and three computation blocks (ARTIFICIAL NEURAL NETWORK, COEFFICIENTS FUSION, and WAVELET RECONSTRUCTION). The metamodel operation is based on the principle of generating all time samples of the output waveform using a wavelet reconstruction transform. To do this, it is necessary to use information about the wavelet transform, WT_{info} (mother wavelet and bookkeeping vector) and the vector of the coefficients, *C*. The vector of coefficients *C* is provided by the COEFFICIENTS FUSION block, through the fusion between the vector of nominal coefficients, C_{nom} and the vector of selected coefficients, C_{sel}. The fusion consists in replacing every nominal coefficient by its corresponding one, from the vector of selected coefficients C_{sel}, as indicated by the vector of indices of selected coefficients, I_C_{sel}. C_{nom} is the vector of coefficients resulted from the wavelet decomposition of the nominal waveform in the waveform family, according with WT_{info}. I_C_{sel} is the vector of the indices of the most relevant coefficients from an optimal wavelet decomposition, as it will be explained later, in the "GA optimization" section. The information about the wavelet transform, WT_{info}, the vector of nominal coefficients, C_{nom}, and the vector of indices of selected coefficients, I_C_{sel}, are previously determined and stored in the KNOWLEDGE BASE, in the phase of metamodel development, as presented hereinafter.

The vector of selected coefficients, C_{sel}, is generated by the ARTIFICIAL NEURAL NETWORK (ANN), for each combination of input parameters *X*.

OWTM development

In the frame of OWTM, besides the ANN a very important role is played by a discrete multilevel wavelet transform [Olt16]. The wavelet transform is a tool that cuts waveforms into different frequency components, and then studies each component with a resolution matched to its scale [Dau93].The wavelet transform preserves both time and frequency information, in its coefficients. On each level of decomposition, the signal is decomposed into low frequency coefficients (approximation) and high frequencies coefficients (details). The approximations are the high-scale, low-frequency components of the signal while the details are the low-scale, high-frequency components [Shi03].

In discrete multilevel wavelet decomposition (DMWD) the decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components [Dau93], [Shi03]. The decomposition can proceed until the individual details consist of a single sample. In practice, a suitable number of levels based on the nature of the signal will be selected. In general, giving a signal of length *K*, the DMWD consist of $\log_2 K$ levels at most. The first step produces two sets of coefficients: approximation coefficients A_1 and details coefficients D_1. These coefficients are obtained by convolving the signal with a low-pass filter for approximation and with a high-pass filter for details, followed by dyadic decimation (downsampling). The next step splits the approximation coefficients in two parts, using the same scheme, replacing the original signal by A_1, and producing the coefficients on the next level A_2 and D_2, and so on [Shi03], [WT].

The number of coefficients on the first decomposition level is given by the relation [WT]:

$$N_{A_1} = N_{D_1} = floor\left(\frac{K-1}{2}\right) + F \tag{27}$$

where 2*F* is the length of each filter (in the case of Daubechies family *F* is the order of mother wavelet).

Further, the number of coefficients for the j^{th} decomposition level can be determined with the relation:

$$N_{A_j} = N_{D_j} = floor\left(\frac{N_{A_{j-1}}-1}{2}\right) + F \tag{28}$$

The mathematical apparatus and further details regarding the wavelet transform can be found elsewhere [Dau93],[Nas08].

In the presented approach we have used DMWD, three wavelet families (Daubechies, Symlets and Coiflets), and maximum 10 decomposition levels.

GA optimization

The problem to be solved further involves two different aspects:

- find an optimum multilevel wavelet decomposition, in terms of mother wavelet and decomposition level;
- select a reduced number of relevant decomposition coefficients.

This two aspects cannot be separately treated, but unitedly, because the number and position of the selected coefficients are directly connected with the mother wavelet and decomposition level. As a consequence, an optimization method is required to quest for the best combination of mother wavelet, decomposition level and selected coefficients. The optimization problem has conflicting objectives: on one hand - high accuracy, meaning the most appropriate wavelet transform for the waveform family, keeping as many decomposition coefficients as possible; on the other hand - dimensionality reduction (small complexity), meaning as little selected coefficients as possible.

To find an at least good enough solution (not always necessarily the best one) to the above presented optimization problem, we are using a genetic algorithm (GA).

To maintain the population diversity and to avoid a fast polarization towards one multidimensional point in the solution space, the population is distributed into two sub-populations. The number of individuals will be determined after a series of trial-and-error runs, taking into account the computational resources and performance.

The individuals evolve with every iteration, by means of the genetic operators. First, a rank fitness scaling, followed by roulette selection is applied, in order to select the individuals that will contribute to the development of the new sub-populations. The best 2 individuals (elite count) will automatically be sent to the new population, without any changes. Intermediate crossover, with a crossover fraction of 0.7, and mutation are applied to the remaining selected individuals in each sub-population. Every 6 epochs (migration interval), individual migrate from one sub-population to the other. The algorithm stops when the average relative change in the fitness function value over 7 generations (stall generations) is insignificant.

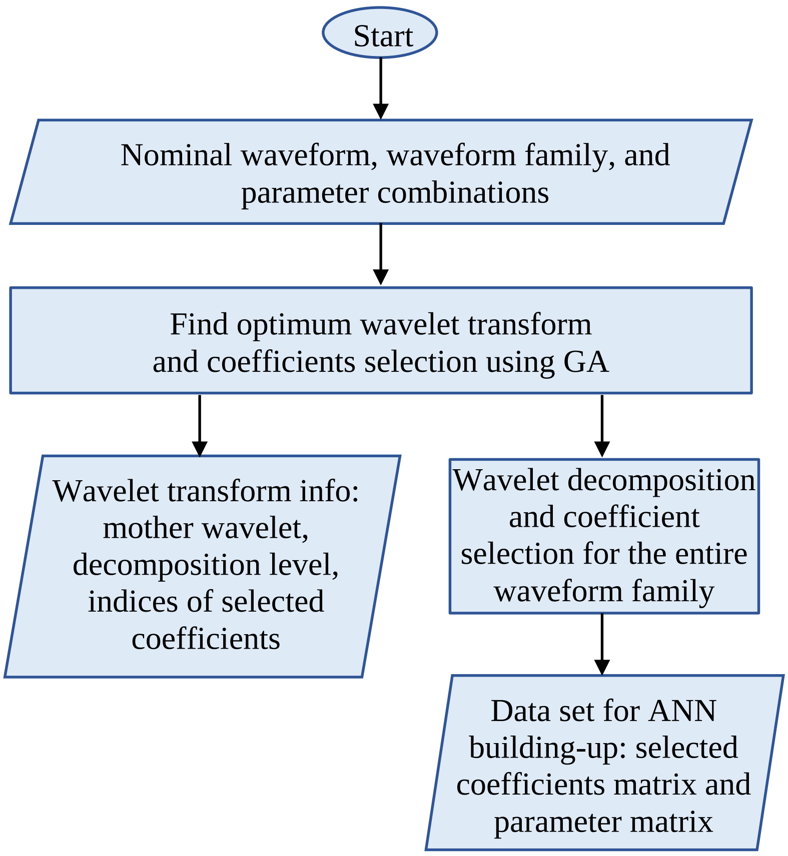


Fig. 39 The GA optimization process to obtain the optimum wavelet decomposition, selected coefficients, and training data set.

The GA optimization process is used to find the most appropriate wavelet transform, select the relevant coefficients, and eventually obtain the data set for ANN building-up. The optimization process, presented in Fig 39, starts from the available data: nominal waveform (corresponding to the nominal value of the parameters), waveform family (waveforms corresponding to different combinations of the parameter), and parameter combinations.

Next, we enter the optimization phase, to discover an optimum solution for the wavelet decomposition and selection of the coefficients. The chromosome consists of three genes:

[*wavelet_name*; *decomposition_level*; *selected_coefficients_ratio*]

The optimization problem is a multiobjective optimization one, composed by two objectives: high accuracy approximation of the wavelet decomposition and reduced number of selected coefficients. The accuracy of each approximation of the wavelet decomposition is evaluated by means of the mean squared error (*mse*) between the original and reconstructed waveforms:

$$mse = \frac{1}{K-1}\sum_{k=1}^{K}(s_k^r - s_k^o)^2 \tag{29}$$

where K is the number of time samples, s_k^r is the k^{th} reconstructed time sample, and s_k^o is the k^{th} original time sample. The reconstructed waveform is obtained using the same wavelet transform as for the decomposition, but taking only the selected coefficients from the original wavelet decomposition, the rest of the coefficients being the ones taken from the decomposition of the nominal waveform.

Across the entire waveform family we sum up all the *mse* resulted for each individual waveform, to obtain a global accuracy estimation, $gmse$, as defined in equation:

$$gmse = \sum_{j=1}^{N}\left(\frac{1}{K-1}\sum_{k=1}^{K}(s_{kj}^r - s_{kj}^o)^2\right) \tag{30}$$

where N is the total number of waveforms, K is the number of time samples, s_{kj}^r and s_{kj}^o are the k^{th} reconstructed, respectively original time samples for the j^{th} waveform in the family.

For the number of selected coefficients, we are using the selected coefficients ratio (*scr*), as the ratio between the number of selected coefficients and the number of time samples:

$$scr = \frac{number\ of\ selected\ coefficients}{number\ of\ time\ samples} \tag{31}$$

To select the coefficients, a global threshold across all decomposition levels and across the entire waveform family is used. For the j^{th} waveform in the family, the full vector of decomposition coefficients C_j is:

$$C_j = [c_{j1}, c_{j2}, ..., c_{jl}, ..., c_{jL}] \tag{32}$$

where c_{jl} is the l^{th} coefficient in the decomposition and L is the number of coefficients. We compute the vector of accumulated decomposition coefficients, AC by summing up the absolute values of the individual coefficients at every l location:

$$AC = [ac_1, ac_2, ..., ac_l, ..., ac_L], \qquad ac_l = \sum_{j=1}^{N}|c_{jl}| \tag{33}$$

The selected coefficients are the ones corresponding to the position where the value of the accumulated coefficient is greater than or equal to the global threshold. In order to select a number of coefficients in accordance with the *scr* value obtained from the GA, the global threshold is dynamically determined for each individual in the population.

This multiobjective optimization problem is then transformed into a single-objective optimization one, by combining the two metrics, $gmse$ and scr, into one objective function. The dynamic ranges for these metrics are very different, so, in order to make them comparable during the genetic evolution, we use first a domain transformation, by means of *s*-type fuzzy sets, followed by a weighted summation. Fig 40 presents the *s*-type fuzzy set used for the domain transformation of *scr*. It maps the transformation from the [0, scr_{max}] domain into the [0, 1] domain:

$$f_{scr} : [0,\ scr_{\max}] \rightarrow [0,\ 1] \tag{34}$$

where scr_{max} is the upper bound of *scr*, to be set by the user.

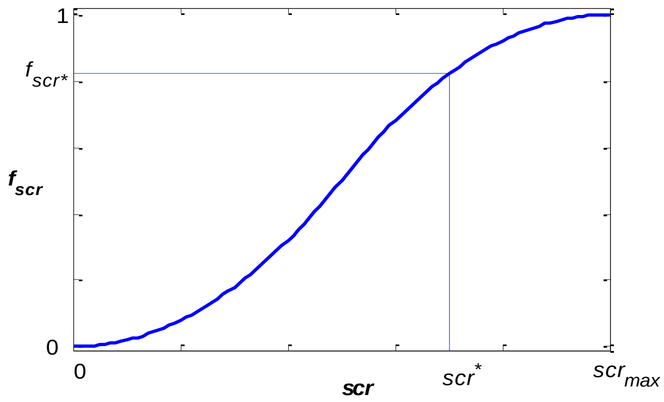


Fig. 40 Defining the objective function for *scr* using an *s*-type fuzzy set. If the current value for *scr* is scr^*, the value of the objective function results as f_{scr^*}.

A similar function, f_{gmse}, is used to define the objective function associated with *gmse*:

$$f_{gmse} : [0,\ gmse_{\max}] \rightarrow [0,\ 1] \tag{35}$$

where $gmse_{max}$ is the upper bound of *gmse*.

Now, both individual objective functions share the same dynamic range [0, 1]. Using a weighted sum, the final scalar objective function *f* results as:

$$f = w_1 f_{gmse} + w_2 f_{scr} \tag{36}$$

Where w_1 and w_2 are the weights (or relative preferences) associated with the individual objective functions.

The GA optimization process provides: the wavelet name, the decomposition level (bookkeeping vector) and the vector containing the indices of the selected coefficients. This information is stored into the knowledge base (WT_{info} and I_C_{sel} – see Fig 38), and also used for generation of the necessary data training set.

Building the ANN

Each waveform in the waveform family is decomposed using the optimized wavelet transform. From the resulted decomposition coefficients, the coefficients selection operation occurs, according with the indices of selected coefficients determined in the optimization phase. The selected coefficients for all waveforms in the family are then organized into the selected coefficient matrix, used as output data in the training data set. The input data in the training data set consists in the parameter combinations.

The structure of the ANN is the same as the one used in the frame of SFTM, with one hidden layer. The ANN is trained until its performance is considered acceptable. The resulted ANN is now ready to be used in the metamodel. Should the designer find the performance of the trained ANN not satisfactory, the entire process of building-up the neural network can be re-run with new initialization of the neural network adjustable parameters (weighs and biases) or even with a different number of neurons in the hidden layer.

3.2. Implementation and experimental results

The implementation was carried out in the MATLAB integrated development environment, making good use of the built-in functions available in the Genetic Algorithms, Neural Networks and Fuzzy Logic Toolboxes; a series of custom functions and scripts were also necessary, especially for the metamodel development part of the process, but also for graphical illustration purposes.

The data used for conducting the implementation and experimentation consists of three family of waveforms, each of them composed by a set of 200 waveforms ($N = 200$), simulated for 200 different combinations of 10 input parameters ($M = 10$) and a nominal waveform. Every waveform is described by a number of 8000 time samples.

It is worth to mention that the dynamic ranges for the waveform families are different, so, in order to have a consistent approach and to be able to make some direct comparisons between numerical results, we are using a unity-based normalization of the waveforms:

$$norm\_s_k = \frac{s_k - nom\_s_{\min}}{nom\_s_{\max} - nom\_s_{\min}} \tag{37}$$

where $norm\_s_k$ is the k^{th} sample of the normalized waveform, s_k is k^{th} sample of the original waveform, while $nom\_s_{\min}$ and $nom\_s_{\max}$ are the minimum and maximum values of the samples of the nominal waveform. As a consequence, all the samples of the normalized nominal waveform lays in the range $[0;\ 1]$.

Results for the 1st family

The original (not-normalized) waveforms in the 1st family are represented in Fig. 41 by the nominal waveform and other 4 representative curves: "max" – the waveform with maximum values in the intermediate region; "min" – the waveform with minimum values in the intermediate region; "right most" – the waveform with the furthest to the right values in the positive slope region, and "interm" – an intermediate waveform.

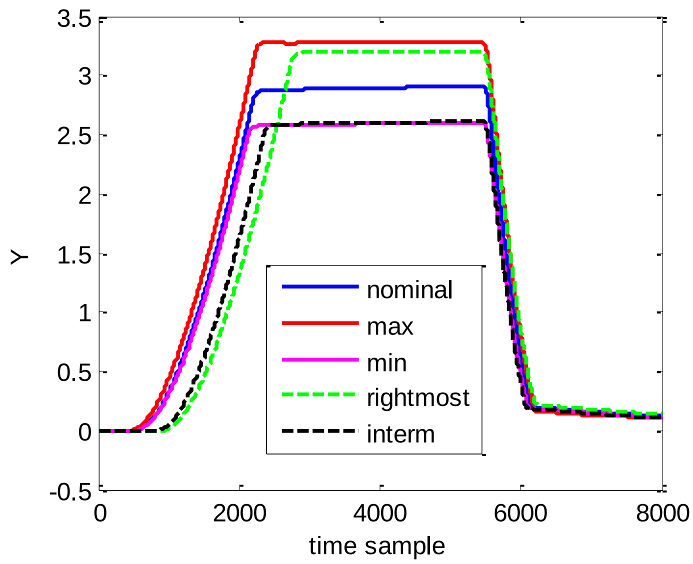


Fig. 41 The 1st waveform family, illustrated by the nominal waveform and others 4 representative curves.

The first stage in developing the metamodel is to find the optimum wavelet transform and the indices of selected coefficients using the above presented GA optimization, where each individual is described by three genes:

- *wavelet_name*, coded by an integer in the range [1; 24], each integer indicating a specific wavelet in the wavelet families Daubechies, Symlets, and Coiflets, of different order;
- *decomposition_level*, coded by an integer in the range [1; 10];
- *selected_coefficients_ratio*, coded by a positive real number in the range [0; 25];

Considering the stochastic nature of genetic algorithms, we ran the optimization multiple times. In Table12, the optimization results are presented for six solutions denoted *Sol*1, …, *Sol*6.

Table 12 Six solutions for optimum wavelet transform and coefficients selection (1st family).

Solution	Wavelet name	Decomp. level	Number of selected coefficients	Selected coefficients ratio	*mse* (normalized waveforms)	
					Nominal waveform	Entire family
*Sol*1	db13	8	142	1.77%	4.7825×10^{-9}	2.2972×10^{-5}
*Sol*2	db7	8	154	1.93%	8.0138×10^{-10}	7.2407×10^{-6}
*Sol*3	db13	7	105	1.31%	8.3319×10^{-9}	4.3397×10^{-5}
*Sol*4	sym4	8	102	1.28%	4.4554×10^{-9}	2.2763×10^{-5}
*Sol*5	db5	8	111	1.39%	2.1785×10^{-9}	1.9336×10^{-5}
*Sol*6	db5	6	370	4.63%	3.9386×10^{-12}	1.8145×10^{-7}

All presented solutions are quite similar from the point of view of accuracy and complexity, providing very good trade-offs between them.

The first 5 solutions (*Sol*1, …, *Sol5*) present an increased accuracy, evaluated by means of *mse* metrics for nominal waveform and also for the entire waveform family (the sum of the individual *mse* for all waveforms in the family). Across the entire family, the resulting *mse* is with 4 orders of magnitude higher than the one for the nominal waveform.

From the complexity point of view, the ratio of the selected coefficients is very small, between 1.28% (102 selected coefficients) for *Sol*4 and 1.93% (154 selected coefficients) for *Sol*2. This translates into a substantial data dimensionality reduction, as the ratio between the number of time samples and the number of selected coefficients is 78 for *Sol*4 and 52 for solution *Sol2*.

*Sol*6 in Table 3 provides even higher accuracy, presenting better (smaller) values for *mse*, with two orders of magnitude smaller than for the first 5 solutions. The price paid is an increased complexity: 370 selected coefficients (4.63%), that is roughly 3 times greater than for the other solutions.

The optimum found decomposition level is quite the same (8, 7 or 6). It is worth to remind that the most important (in respect with their magnitude) decomposition coefficients are selected. The coefficients responsible for the low-frequency content of the waveform (the approximation coefficients) are entirely selected. For the details coefficients, the ratio of the selected coefficients on each decomposition level decreases with the details order [Olt16].

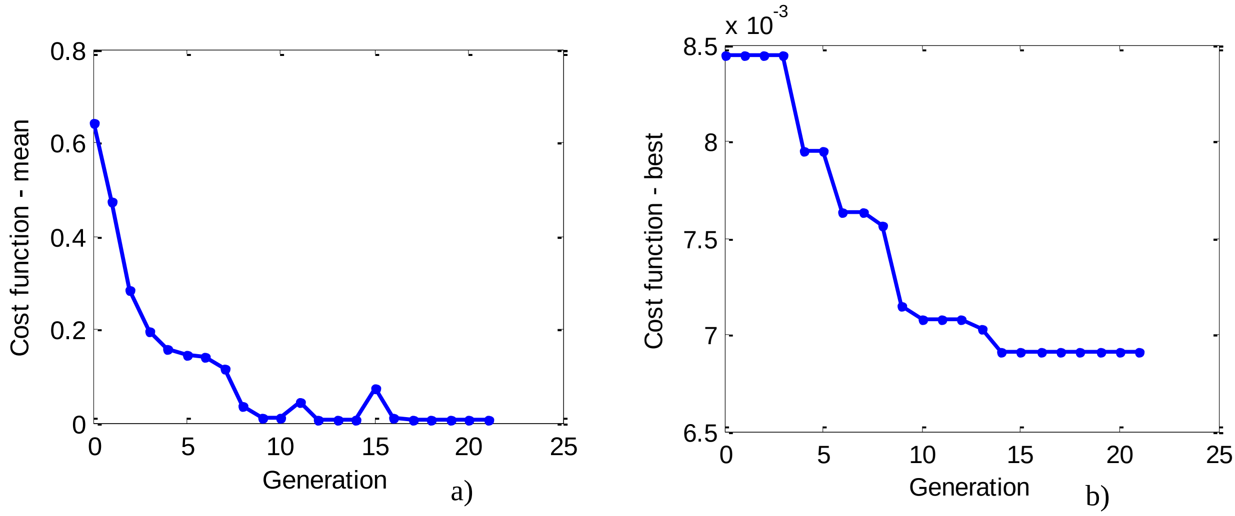


Fig. 42 Cost function evolution during the optimization for *Sol5*, 1st waveform family:
a) Mean value across the entire population; b) Best individual.

Some relevant information about the dynamics of the GA optimization is presented for solution *Sol*5. The evolution of the cost function (mean and best) is presented in Fig 42. One can notice a global improvement of the whole population (30 individuals split into two subpopulations) by a global decrease of the mean value of the cost function, from 0.642 for the initial population, down to 7.0x10^{-03}, in the last generation (21). Also, the cost function for the best individual in each generation improves constantly, from an initial value of 8.445x10^{-3}, down to the final value of 6.908x10^{-3} in generations 14 to 21. On the last 7 generations, no further improvement of the cost function (best individual) appears, so the optimization stops (stall generations was set to 7).

To evaluate the computational effort required for developing the metamodel, we mention here that the experiments were conducted on a general purpose computer (i5-4460 CPU @ 3.2GHz, 8GbRAM, 64-bit operating system). The time necessary to run the GA optimization has a medium value (across the above presented solutions) of 11 min, with a maximum of 17 min for *Sol*1 (27 generations). For the solution we use further (*Sol5*), the optimization time was 10 min.

For the ANN generation and training we will use the *Sol5* to prepare the data sets. The output data (targets) is organized in a matrix with $N = 200$ columns and $Q = 111$ rows (number of selected coefficients), by applying the wavelet decomposition and coefficients selection for all 200 waveforms in the family. The input data is organized in a matrix with $N = 200$ columns and $M = 10$ rows.

The neural network architecture (see [Plos16] for details) presents:

- 10 inputs - the number of input parameters;
- 111 outputs - one output for each selected coefficient;
- one hidden layer, composed by a number of 15 neurons; the number of neurons is chosen as a result of a series of trial runs.

For the training procedure, the full data set (200 data pairs) is split into three data subsets: training subset (90% of the data set), validation subset (5%), and testing subset (5%).

Training multiple times will generate different results due to different initial conditions (initiation of weights and biases with randomly generated values) and data sampling [NN]. In our implementation, the procedure involves multiple training trials, and the neural network that presents the highest regression value *R* across the entire data set is chosen as a final solution. Table 13 presents the results for 8 training trials.

Table 13 Results for 8 ANN training trials, *Sol*5, 1st waveform family.

Performance metrics		ANN training trial							
		01	02	03	04	05	06	07	08
R		.999949	.999945	.999943	.999949	.999954	.999959	.999944	.999949
mse	Training	2.7x10^{-3}	2.9x10^{-3}	3.2x10^{-3}	2.7x10^{-3}	2.5x10^{-3}	2.3x10^{-3}	3.1x10^{-3}	2.8x10^{-3}
	Validation	3.0x10^{-3}	5.6x10^{-3}	3.2x10^{-3}	3.2x10^{-3}	2.0x10^{-3}	1.9x10^{-3}	3.3x10^{-3}	3.6x10^{-3}
	Testing	6.7x10^{-3}	4.2x10^{-3}	4.4x10^{-3}	4.9x10^{-3}	4.7x10^{-3}	2.3x10^{-3}	4.3x10^{-3}	3.7x10^{-3}
Epochs		404	375	380	659	729	809	402	419

The regression value *R* indicates a very high accuracy of the ANN, laying between 0.999959 (trial 06) and 0.999943 (trial 03). The *mse* between the original coefficients (targets) and the coefficients computed by the ANN (outputs) is quite similar for all trials in the training data set but also in the validation and testing data sets. Conclusively, irrespective of initial condition and data sampling, the resulted ANN presents a high level of accuracy.

Hereinafter we illustrate the ANN training process and detailed performances for ANN training trial 06, *Sol*5. The performance validation graph is presented in Fig 43. In the first 200 training epochs (Fig. 43a)), one can see a steep improvement (lowering-in) of the *mse* in all data subsets. Then, the training enters the phase of “fine tuning”, continuously improving the performance, as it can be observed on Fig. 43b) that uses a linear scale to plot details on *mse* evolution.

The training ends after 909 epochs, but the best trained ANN is considered to be the one at iteration 809, where it presents the best validation performance ($1.928 x 10^{-3}$). It is observed that starting with epoch 809 the overfitting phenomenon appears, the *mse* in the validation subset (green curve) having a tendency to slightly increase on the next 100 consecutive training epochs.

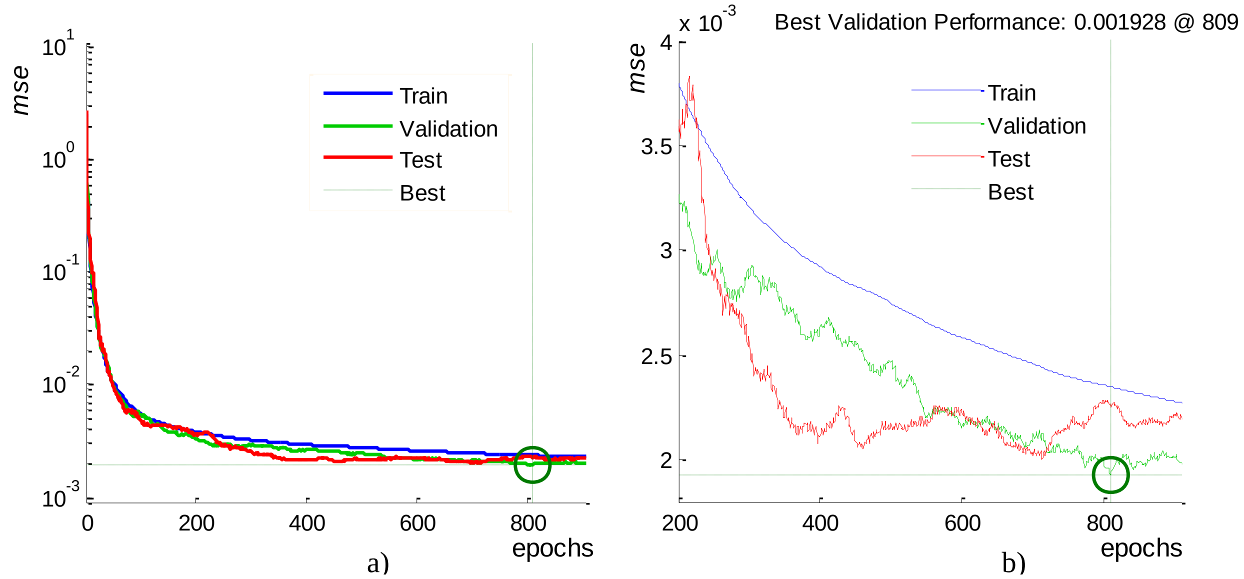


Fig. 43 Performance validation graph for ANN training trial 06, *Sol*5, 1st waveform family:
a) All training epochs (log scale for *mse*); b) Details for fine tuning epochs: 200 to 909 (linear scale for *mse*).

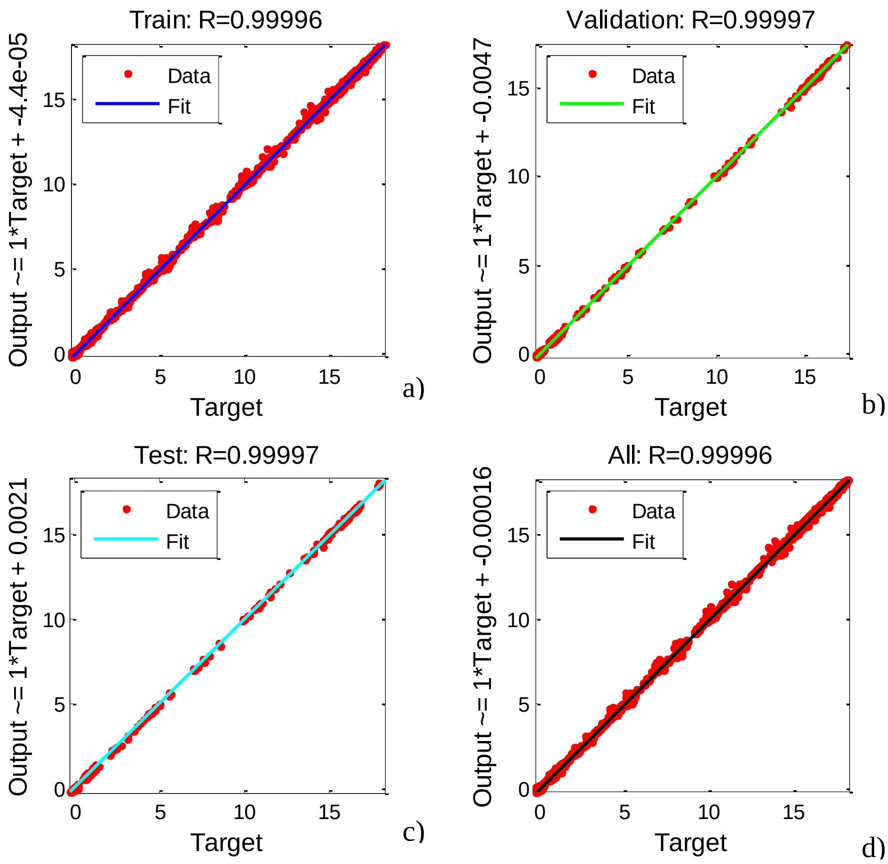


Fig. 44 Linear regression in all data subsets for ANN training trial 06, *Sol*5, 1st waveform family:
a) Training subset; b) Validation subset; c) Testing subset d) All data.

The trained neural network presents excellent fitting performances in all data subsets. This can be seen by inspecting the results for linear regression of outputs of the neural network (predicted values) relative to the targets (original values) presented in Fig. 44.

The slope of the regression equation is 1 in all data subsets and the offset is very close to zero, with a maximum value of 4.7×10^{-3} in the validation subset and 1.6×10^{-4} across all data. The regression value, *R*, is a very good one also: 0.99996 in the training subset and across all data and 0.99997 in the validation and testing subsets.

Fig. 44 also provides a qualitative appreciation of the ANN accuracy. The ideal result should place all data points on the first bisector. In our case, almost all data points are correctly placed on the first bisector, with only few points slightly shifted from their ideal position.

The linear regression offers information on two extremes. On the one hand it provides a global appreciation of the accuracy (through the regression value *R* and through the slope and offset of the regression equation), but on the other hand it compares the position of each generated data point with its target counterpart. The global appreciation can “hide” some errors, while the detailed plot of each point can lead to very difficult to analyze images. For example, in the testing subset (Fig. 44c) 1110 data points are represented (111 outputs of the ANN, with 10 value each).

To fully describe the results, beside the linear regression, I introduced new intermediate error representations. Fig. 45 presents the normalized root mean squared error (*nrmse*) computed for the 10 sample data points (waveforms) in the testing subset:

$$nrmse = \frac{\sqrt{\frac{1}{Q}\sum_{i=1}^{Q}(o_i - t_i)^2}}{t_{\max} - t_{\min}} \tag{38}$$

where Q is the number of ANN outputs, o_i is the i^{th} generated output, t_i is the i^{th} target, $t_{\max}$ is the maximum value of target values, and $t_{\min}$ is the minimum value of target values.

For all 10 testing waveforms the *nrmse* is very small, laying in the range of 1.67×10^{-3} for waveform 3 and 4.39×10^{-3} for waveform 9. It results that the ANN presents high accuracy and very good generalization (prediction) capability for all waveforms in the testing subsets, as these waveforms was not at all involved in the training process.

To characterize the ANN from the point of view of its individual outputs, Fig. 46 presents the mean squared error encountered for the testing subset for each of the 111 outputs. The largest *mse* values (Fig. 46a) appears for the target presenting the largest values and widest dynamic range (Fig. 46b). These correlation is normal and it is mainly justified by the fact that *mse* is scale dependent. The maximum *mse* (0.021985) appears on output 17 where the target presents maximum value and maximum dynamic range (14.09 to 17.87).

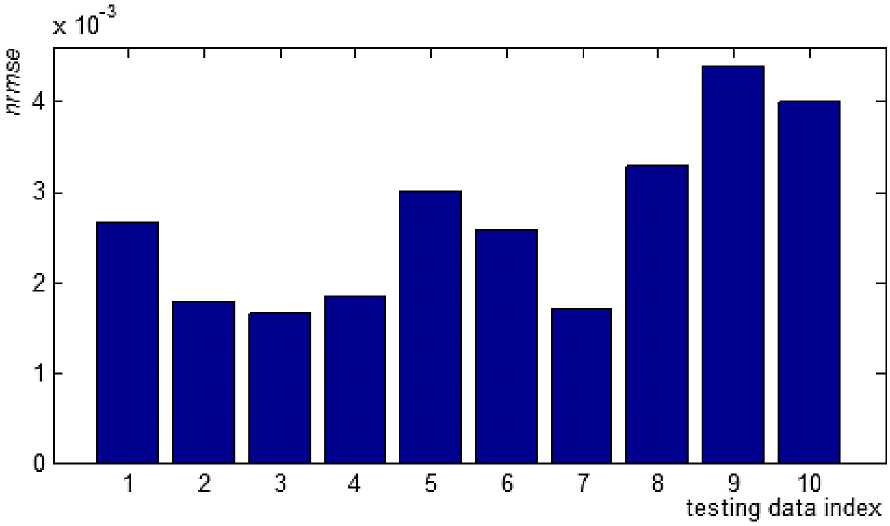
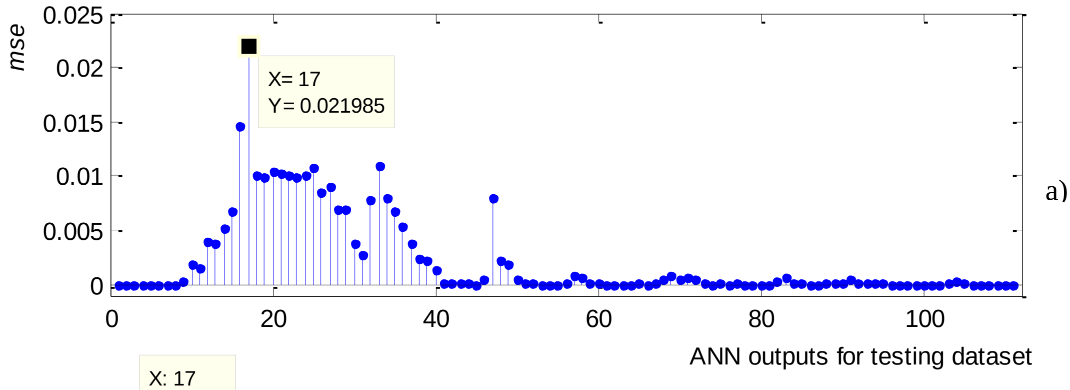


Fig. 45 Values of *nrmse* for the 10 sample data points (waveform) in the testing subset.



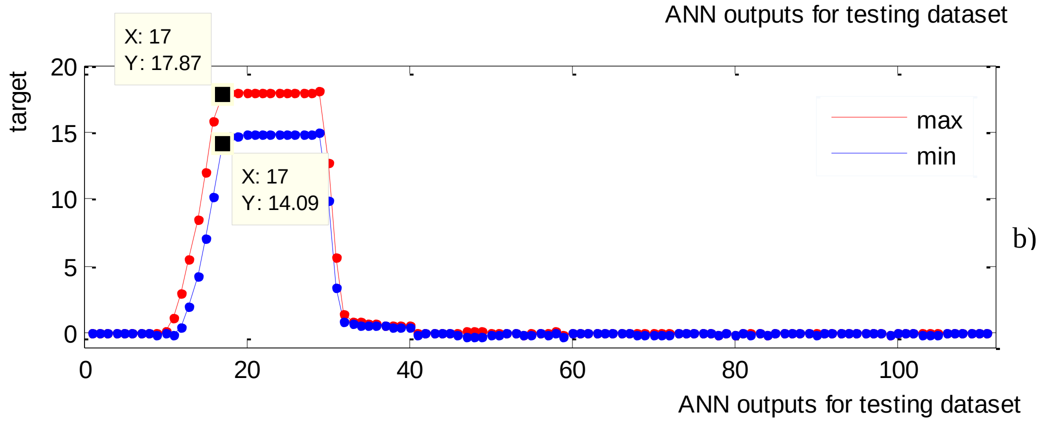


Fig. 46 Prediction accuracy of ANN for testing data subset: a) *mse* error for ANN outputs (111 outputs); b) Dynamic interval of targets: max values (red) and min values (blue).

From the point of view of the training time, the total time for training (8 trials) and selecting the final solution for the ANN is less than 1min, small enough to arise no issues regarding the metamodel development time. Referring to the total time necessary for the metamodel development, once the waveform family is available, the most time consuming tasks are the GA optimization and ANN training. Considering the previously mentioned consumed time, a maximum of 17 min for GA optimization and 1 min for ANN training, results in a total time of maximum 18 min on a general purpose computer. It is obvious that the computational effort required for constructing the metamodel is absolutely affordable.

The final step in our metamodel implementation refers to the metamodel integration, meaning (according to Fig. 38) to fusion the coefficients generated by the ANN with the remaining necessary coefficients (taken form the nominal waveform decomposition), in full accordance with the indices of the selected coefficients. To generate the output waveform by its time samples, the wavelet reconstruction is applied on the full coefficients vector, using the wavelet transform (wavelet and bookkeeping vector) stored in the knowledge database.

To appreciate the accuracy of the whole metamodel, we used it for all our 200 available parameter combinations. Fig. 47 presents the value of the “goodness of fit” (*mse*) for all generated waveforms, in both normalized and original waveform versions.

The mean value of *mse* across the entire family is 3.1837×10^{-5} in the normalized version and 2.693×10^{-4} for the original version. All individual and mean values of *mse* indicate a high accuracy of the metamodel. The distribution of the *mse* values across the entire waveform family (few cases with relatively high values, few cases with medium values, a lot of cases with low values, and no cases with zero values) indicates that our metamodel present very good generalization capabilities.

The greatest *mse* is encountered for waveform 184 (2.8418×10^{-4} for normalized version, respectively 2.4038×10^{-3} in the original version). Even for this extreme case, the accuracy is quite high: the correlations between the original waveform and the predicted one are very good, as the regression factor is very close to 1, R=0.9993.

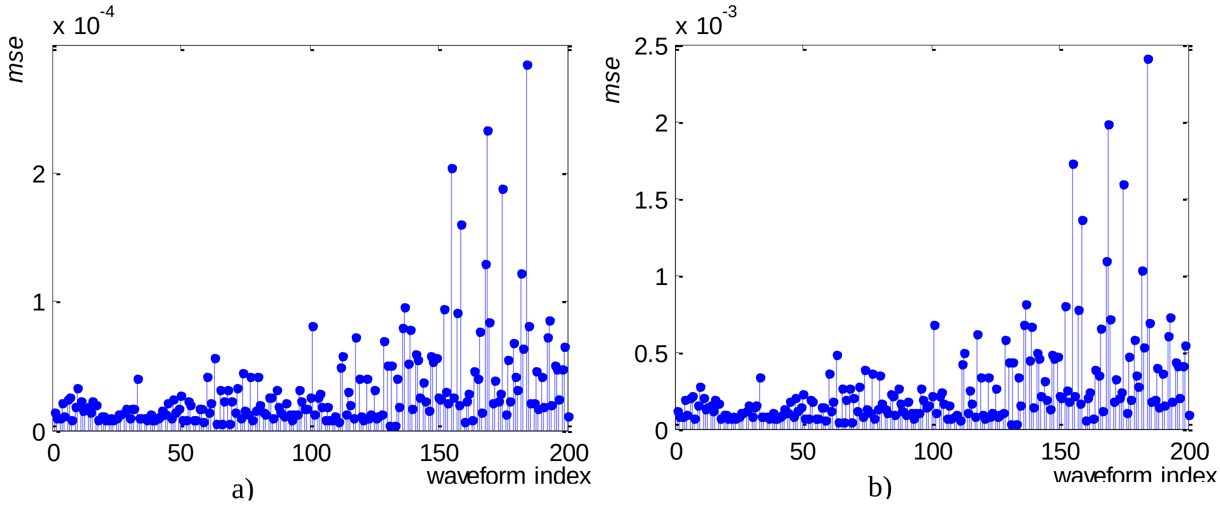


Fig. 47 Mean squared errors (*mse*) obtained using the metamodel for the generation of all 200 waveforms in the 1st family: a) Normalized waveforms; b) Original waveforms (no normalization).

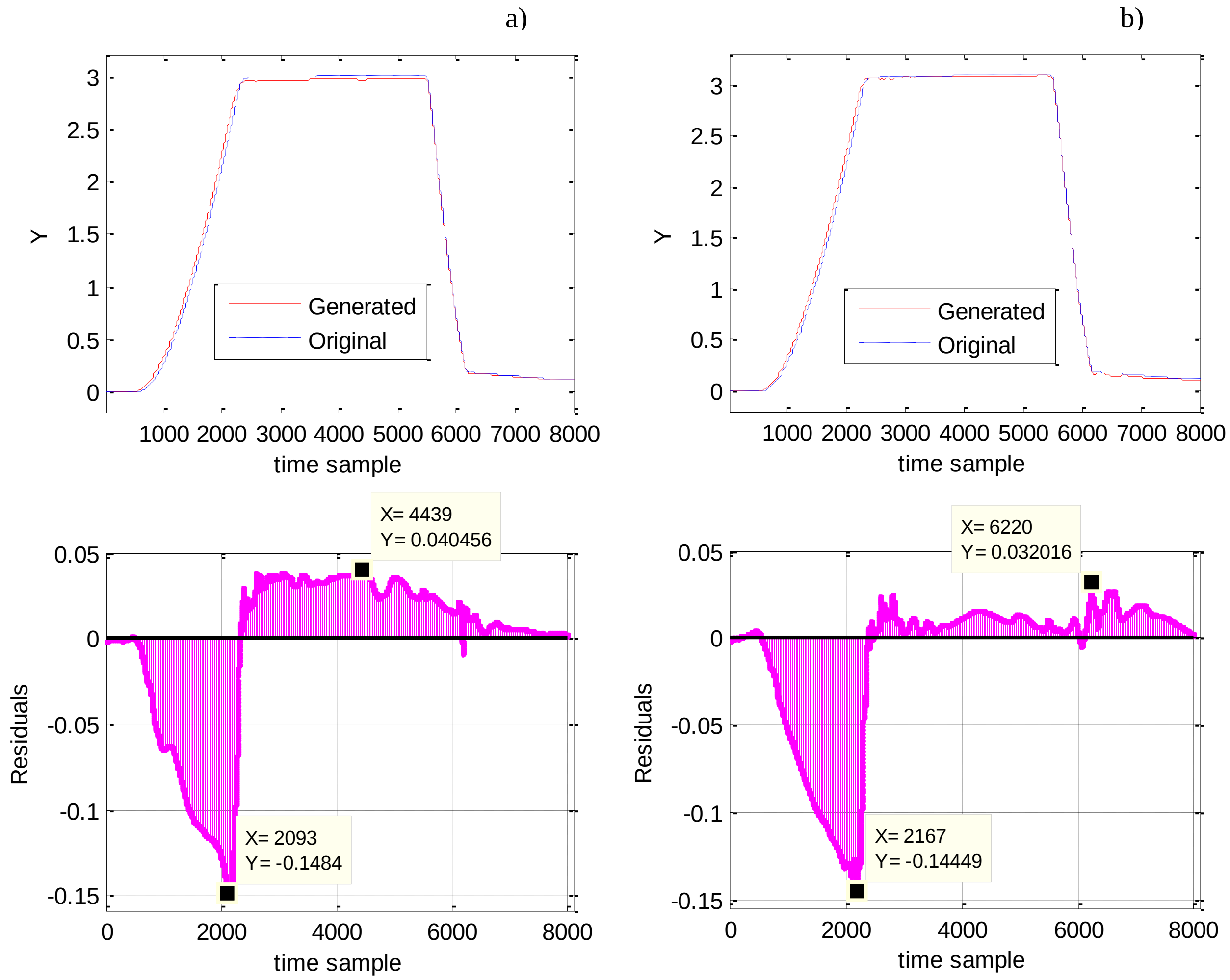


Fig. 48 Comparison between the predicted and original waveforms in two cases:
a) Highest *mse* – waveform 184 (generated vs. original waveform – top; residuals - bottom);
b) Second highest *mse* – waveform 169 (generated vs. original waveform – top; residuals - bottom).

Another perspective for the metamodel accuracy is offered by the plots in Fig. 48, which depict a direct comparison between the predicted and original waveforms, in two cases: waveform 184 which presents the highest *mse* value and waveform 169 which presents the second highest *mse* value. In both cases, the predicted waveforms are (almost) identical with the original ones. For waveform 184, the differences appear in the region with positive slope, between samples 600 – 2300 and in the region with almost constant high values (samples 2300 - 5500), as one can see in Fig. 48a) top. The associated residuals are presented in Fig 48a) bottom. The residuals are mainly negative in the first region (positive slope) with a maximum magnitude of 0.1484 for sample 2093, and positive in the rest with a maximum magnitude of 0.040456 at sample 4439. For waveform 169 the differences between predicted and original waveforms are mainly presented in the region of positive slope (samples 600 - 2350), see Fig. 48b) top. The associated residuals (Fig. 48b) bottom) are negative in that region, with a maximum magnitude of 0.14449 at time sample 2167. In the remaining part of the waveform, the residuals are positive, with a maximum magnitude of 0.032016 at time sample 6220.

By computing the sample-by-sample error (residuals), in fact one computes the differences on the vertical axis, thus obtaining a larger error that it really is, considering the entire region in discussion. It is obvious that comparing two waveforms using the sample-by-sample difference on the vertical axis is not the best approach, at least not in the regions presenting steep slopes. Hence, another measure to compare two waveforms should be developed and used.

It is worth to mention that the *mse* in the worst case (2.4038×10^{-3} for waveform 184) is with two magnitude orders greater than the *mse* for the best case (2.5607×10^{-5} for waveform 131). This is due to the fact that in the available data set (200 waveforms) there are a reduced number of examples similar with waveform 184. The solution for better metamodel accuracy as a whole is to obtain and use a larger data set for developing (training) the metamodel.

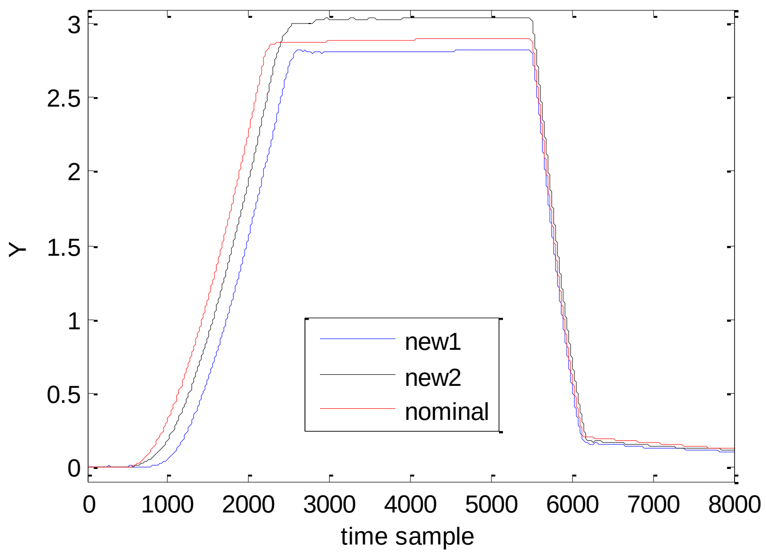


Fig. 49 Two new generated waveforms and the nominal waveform (1st waveform family).

To better understand and appreciate the quality of the metamodel we are using it to generate brand new waveforms for completely different parameter combinations. The new values of the parameters are randomly generated and checked to be different from the 200 combinations that were already used for building the metamodel. As an illustration, Fig. 49 presents two such waveforms, denoted new1 and new2. For the sake of comparison, the nominal waveform is also plotted on the same figure. According with the shape and values of new waveforms, compared with the nominal one and with the other waveforms in the family (see Fig. 41) it is clear that they belong to the 1st waveform family, so the metamodel can be credited as a very reliable one.

Results for the 2nd and 3rd families

The 2nd family is presented in Fig. 50a) through the nominal waveform and other 4 representative curves: “max” – maximum values in the intermediate region; “min” – minimum values in the intermediate region; “rightmost” – rightmost values in positive slope region, and “interm” – intermediate values. Compared with the 1st family (Fig. 41) the 2nd family is more complex, each waveform presenting an overshot at the end of the rise time and also at the end of the fall time.

Fig. 50b) presents the 3rd waveform family through its nominal waveform and other 4 representative curves: “max1” – maximum values in the intermediate region (samples 2250 - 5500); “max2” – maximum values in the right side region (samples 5500 - 8000); “min” – minimum values in the intermediate and left side regions (samples 2500 - 8000), and “rightmost” – rightmost values in the positive slope region (samples 0 - 2500). The waveforms in this family present large variations inside of each individual waveform but also from one waveform to another, especially in its second half.

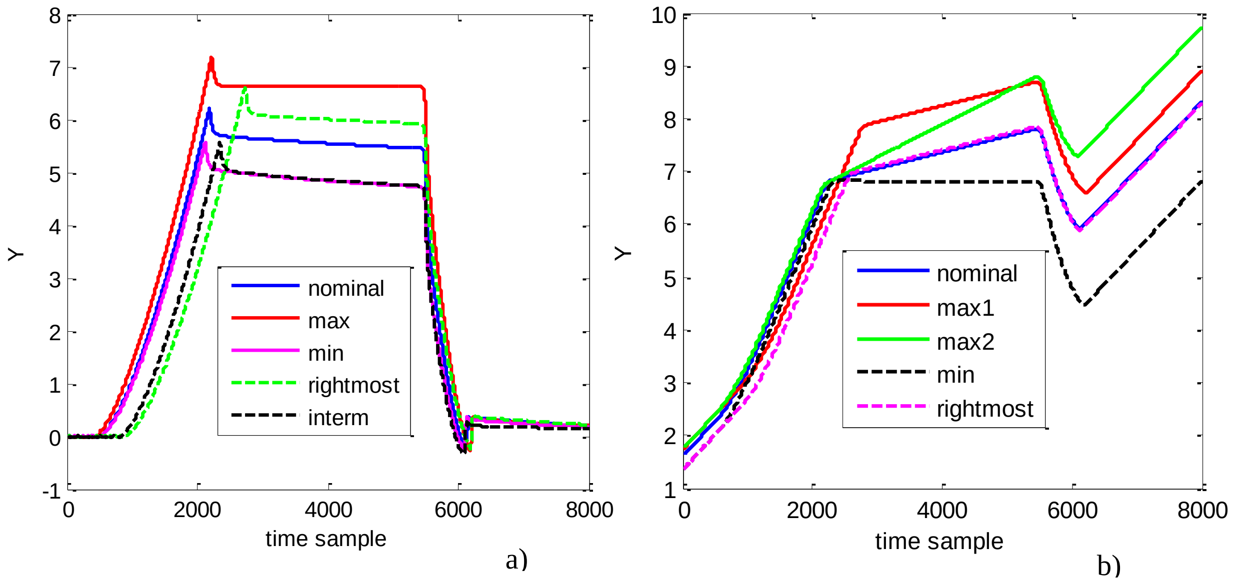


Fig. 50 Waveform families illustrated by the nominal waveform and 4 more representative curves: a) 2nd waveform family; b) 3rd waveform family.

Using the same settings for the GA optimization as the ones used for the 1st family, we run the optimization six times for each waveform family. Table 14 contains the final optimal solutions for wavelet transform and coefficients selections for both families.

Table 14 Optimal solutions for wavelet transform and coefficients selection using GA optimization for the 2nd and 3rd waveform families.

Waveform family	Wavelet name	Decomposition level	Number of selected coefficients	Selected coefficients ratio	*mse*	
					Nominal waveform	Entire family
2nd	sym7	8	239	2.83%	5.8499×10^{-9}	4.2533×10^{-5}
3rd	sym3	7	181	2.26%	7.5886×10^{-10}	8.1433×10^{-6}

The optimal solutions in both families are quite alike: same type of mother wavelet (sym7, respectively sym3); 8, respectively 7 decomposition levels; 239, respectively 181 selected coefficients. From the complexity point of view, the ratio of the selected coefficients is small, 2.83% (239 selected coefficients) for the 2nd family and 2.26% (181 selected coefficients) for the 3rd family. This translates into an important data dimensionality reduction, the ratio between the total number of waveform samples and the number of selected coefficients being 33.5 for the 2nd family and 44.4 for the 3rd family. The accuracy of approximating 8000 time samples by a drastically reduced number of wavelet coefficients is a very high for nominal waveforms (5.8499×10^{-9}, respectively 7.5886×10^{-10}), but also across the entire family (4.2533×10^{-5}, respectively 8.1433×10^{-6}).

For each family, 8 training trials took place in the quest of the final ANN. Table 15 presents the final results for each family, both neural network topology and training performances. For the 2nd family there are 10 inputs, 13 neurons in the hidden layer, and 239 neurons in the output layer. The ANN for the 3rd waveform family presents 10 inputs, 16 neurons in the hidden layer, and 181 neurons in the output layer.

Table 15 ANN topologies and training performances for 2nd and 3rd waveform families.

Waveform family	ANN topology			Training performances				
	Inputs	Hidden layer neurons	Outputs	Regression	Training *mse*	Validation *mse*	Testing *mse*	Epochs
2nd	10	13	239	.999898	$2.5x10^{-3}$	$2.7x10^{-3}$	$2.0x10^{-3}$	331
3rd	10	16	181	.999970	$1.1x10^{-3}$	$5.5x10^{-4}$	$5.8x10^{-4}$	896

The results confirm that the training process is a robust one, always convergent, producing neural networks with almost the same final performances. The regression value *R* indicate a very high accuracy of the trained ANNs being 0.999898 (2nd family) and 0.999970 (3rd family). In the case of the 2nd waveform family the *mse* between original coefficients (targets) and the coefficients computed by the ANN (outputs) is quite similar in the training data subset $2.5x10^{-3}$ and in the validation and testing data subsets ($2.7x10^{-3}$, respectively $2.0x10^{-3}$). For the 3rd family there is a small difference, the *mse* in the validation and testing subsets being approximately half of the one in the training subset.

The total time necessary to develop the metamodel (in the worst case) is less than 12 min (maximum 11 min for GA optimization and 30s for ANN training) for the 2nd family and less than 14min (maximum 13min for GA optimization and 1min for ANN training) for the 3rd family.

To globally appreciate the accuracy of our metamodels we apply them for all our 200 available waveforms in each family. Fig. 51 presents the value of the "goodness of fit" (*mse*) for all waveforms, for normalized versions.

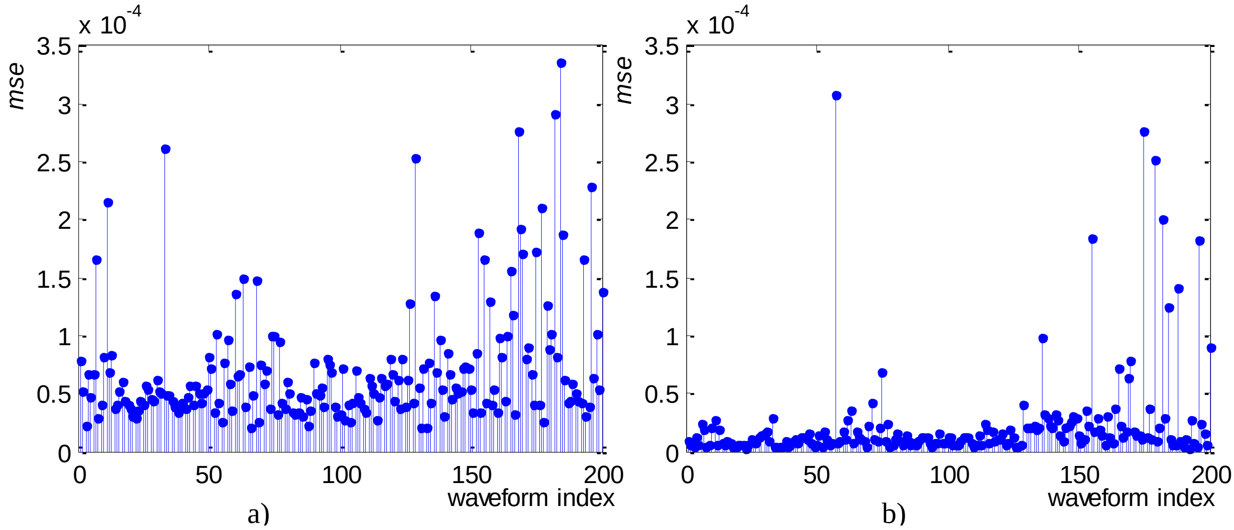


Fig. 51 Mean squared errors (*mse*) obtained using the metamodel for the generation of the normalized waveforms: a) 2nd family; b) 3rd family.

The mean value of *mse* is $7.0648x10^{-5}$ across the 2nd family and $2.2924x10^{-5}$ across the 3rd family. All individual and mean values of *mse* indicate high prediction accuracy of our metamodels. The greatest *mse* occurs for waveform 184 ($3.3616x10^{-4}$, in the normalized version and $1.4057x10^{-2}$ in the original version) in the 2nd family and for waveform 57 ($3.0750x10^{-4}$ in the normalized version and $1.3853x10^{-2}$ in the original waveform) in the 3rd family.

The images in Fig. 52 present a direct comparison between the original and predicted waveforms, for the predicted waveform with lowest accuracy in each family. For both families, even in the case of lowest accuracy, the generated waveforms are close enough to their original counterparts.

For the 2nd family some differences can be observed on the positive slope, overshooting, and almost constant high value regions (Fig 52a) top). The residuals (Fig 52a) bottom) are negative in the positive slope region with a maximum magnitude of 0.37656 for time sample 2090, and positive in the rest, with a maximum magnitude of 0.35919 at time sample 2284. The metamodel for this 2nd waveform family is an accurate one, possessing a good generalization capability.

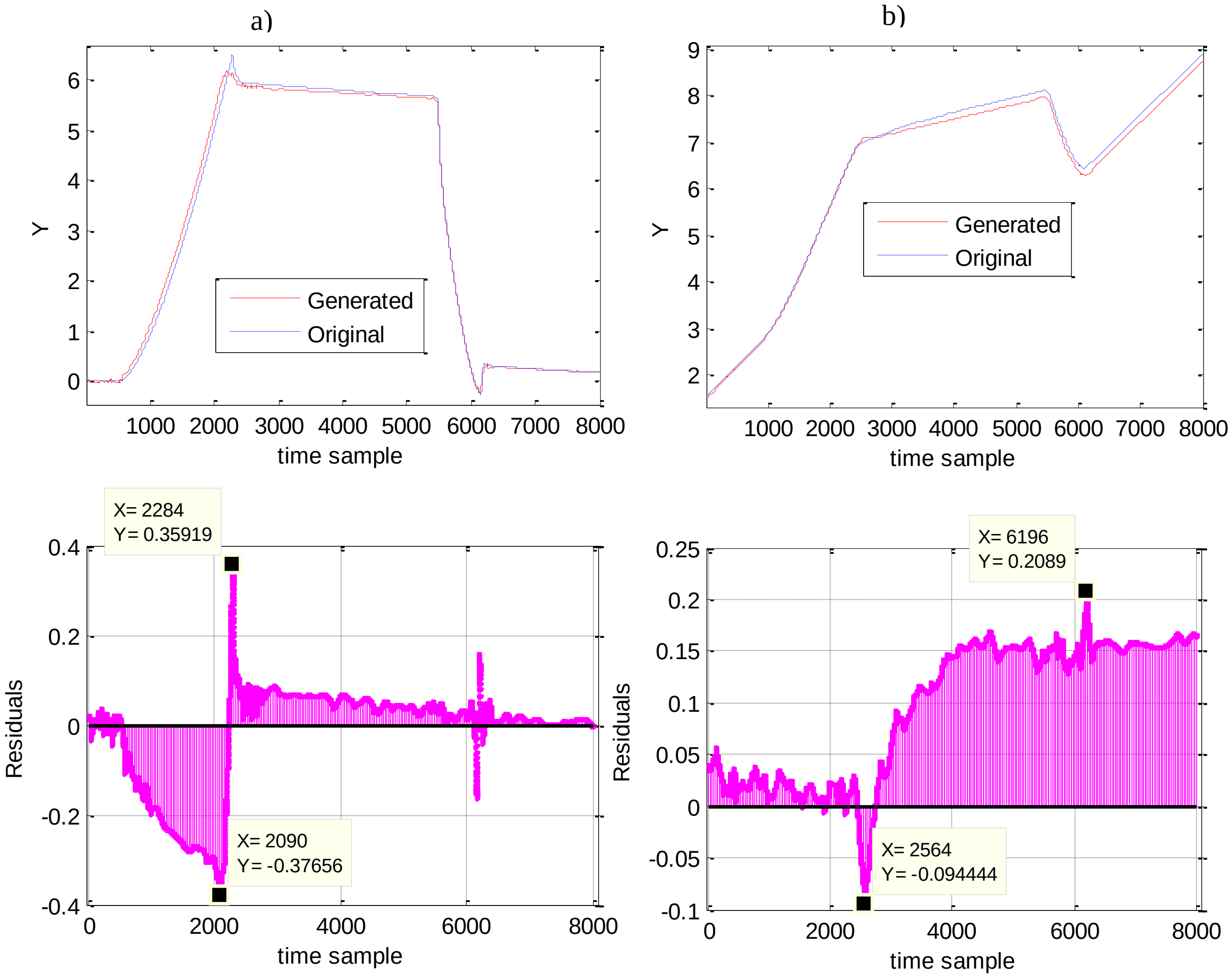


Fig. 52 Comparison between the predicted and original waveforms with highest *mse* (lowest accuracy):
a) 2nd family – waveform 184 (generated vs. original waveform – top; residuals - bottom);
b) 3rd family – waveform 57 (generated vs. original waveform – top; residuals - bottom).

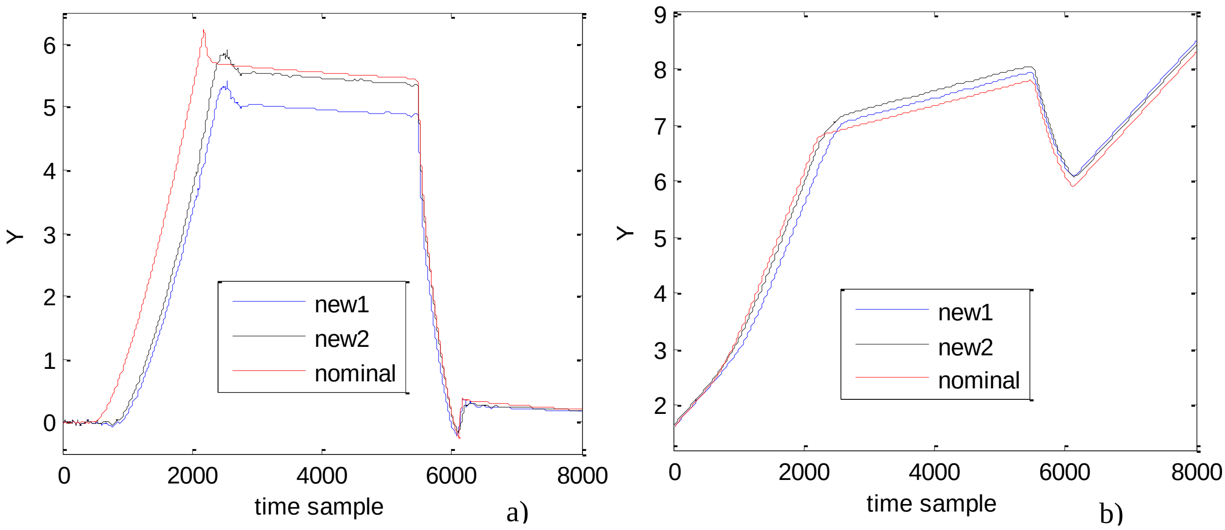


Fig. 53 Two new generated waveforms and the nominal waveform:
a) 2nd waveform family; b) 3rd waveform family.

For the 3rd family some differences appear mostly in the second and third regions, starting from sample 2450 (Fig 52b) top). The residuals are mainly positive, with a maximum magnitude of 0.2089 for time sample 6196, where the sample value should be 6.5320 and it is 6.3231. For this 3rd family, the resulting errors are mainly due to the fact that inside of this family there is an important dispersion from one waveform to another (see also Fig 39b)).

Fig. 53 presents two completely new waveforms for two new parameters combination, different from the ones used to build and test the metamodel. For both 2nd and 3rd families our metamodels generate trusted waveform as they share the same characteristics with the nominal waveforms and the other waveforms in each family (see also Fig 50).

Discussion

A synthesis of the accuracy of our OWTMs, measured by *mse* between original and predicted waveforms, in the unity-based normalized version is illustrated in Table 16. As a whole, the accuracy is similar for all families, both mean and maximum values of *mse* having the same magnitude order (10^{-5} for mean value, respectively 10^{-4} for the maximum value). The minimum *mse* values are similar for the 1st and 3rd families (order of magnitude 10^{-6}) and one magnitude order greater for the 2nd family (10^{-5}). According with the numerical values, it appears that the 2nd family is a little bit more difficult to be predicted, presenting the largest *mse* (7.0648×10^{-5} mean value, 2.0054×10^{-5} minimum value, and 3.3616×10^{-4} maximum value). Anyhow, it results that all metamodels generate accurate results, which validates both the metamodel structure and the development procedure.

Table 16 Synthesis of *mse* of the generated waveforms with the metamodels in all three waveform families.

Waveform family	Mean squared error (*mse*)		
	Mean value	Minimum value	Maximum value
1st family	3.1837×10^{-5}	3.0273×10^{-6}	2.8418×10^{-4}
2nd family	7.0648×10^{-5}	2.0054×10^{-5}	3.3616×10^{-4}
3rd family	2.2924×10^{-5}	3.1278×10^{-6}	3.0750×10^{-4}

From the point of view of computational effort it is important to know the necessary hardware resources and the CPU time. The development of above presented metamodels was realized on a general purpose computer (i5-4460 CPU @ 3.2GHz, 8GbRAM, 64-bit operating system). The time necessary for metamodel development (considering that the necessary numerical data – waveform families - is available) was maximum 18 min for the 1st family. For the other two families, the time was slightly shorter (11 min and 30 s for 2nd family and 14 min for the 3rd family).

Considering the fact that the metamodel should be developed only once for every waveform family, a time of maximum 18 min is fully convenient, from the practical point of view. Indeed, if the number of available waveforms in the family increases, or if the number of time samples increases, the development time increases as well, but it remains acceptable, in terms of tens of minutes.

To generate a new waveform for a new parameter combination, our metamodel does the job in less than 1s (0.947s). This means that even in a case of performing extensive waveform generations, the necessary time remains a practical one; for example to generate 1000 new waveforms, it takes less then 1000s, meaning less than 17min.

The metamodels presented here are made for cost-effective, but reliable replacements of complex systems (ECU) that empower the design team to perform extensive analyses of system response, under a theoretically infinite number of possible combinations of input parameters. This is very useful, because multiple simulations of a detailed model of the system lead to a very long simulation time, which is usually prohibitive.

Since the running time of our metamodel takes less than 1 s, the metamodel can be further used for efficient generation of not-yet simulated output waveforms, for any possible combination of dependent parameters, offering the possibility to cover the entire design space.

The metamodels satisfy the main required key points:

- accuracy – they are capable of generating the system response over the design space, the maximum *mse* being 3.3616×10^{-4} for the 2nd family;

- efficiency – the computational effort required for constructing the metamodel is a fully affordable one: a maximum of 18 min for the metamodel for 1st family on a general purpose computer (i5-4460 CPU @ 3.2GHz, 8GbRAM, 64-bit operating system);
- simplicity – the user can easily run the metamodel in less than 1s, on a general purpose computer: the user should only provide the combination of input parameters and the metamodel will generate the output waveform in a vector (sample by sample) and as a plot.

If, in some special cases, a higher exactitude of generated waveform is necessary, it is possible to achieve this using the proposed method. There are two main action directions. The first one consists in increasing the number (or ratio) of selected coefficients for the wavelet transform in the GA optimization, to capture more information about the waveform family. Of course this leads to a certain increase of the metamodel's complexity, by increasing the number of ANN neurons in the output level. This may also require a larger data training set. In fact, this is the second course of action, to increase the available data for metamodel development, especially for the ANN learning process. In this case, the developer should also consider the necessary resources for obtaining more numerical data by performing costly experiments and/or measurements.

4 Other applications of CI techniques

Beside the previously presented areas of interest, since the Ph.D. completion in 2002 my research and didactic interest referring to applications of CI techniques in electronics also includes the following:

- Pattern recognition using fuzzy logic and artificial neuronal network [Olt16a], [Olt16b]:
    - iris flower classification using fuzzy subtractive clustering and fuzzy C-means algorithms;
    - iris flower recognition using T-S fuzzy logic systems and a decision function, with a correct recognition rate of 98%;
    - iris flower recognition using ANN, with a correct recognition rate of 98.7%;
    - capital letter recognition using ANN, with a correct recognition rate of 100% in the case of noise free letters, 96.15% for letters affect by low noise, 80,77% for letters affected by medium noise, and 53.85% for letters affected by high noise;
    - capital letter recognition using T-S FLS, with a correct recognition rate of 100% in the case of noise free letters, but only 53.85% for letters affect by low noise;
    - handwritten number recognition using ANN with a correct recognition rate of 94.10%.
    - isolated word recognition (monolocutor, 4 words) using ANN; the maximum correct recognition rate (100%) was systematically obtained by combining two types of feature for the audio signals (energy of wavelet decomposition coefficients and average values of Mel cepstral coefficients);
    - isolated word recognition (monolocutor, 4 words) using a 1st order T-S FLS and a decision function; the maximum correct recognition rate (100%) was obtained using a fuzzy subtractive clustering algorithm to generate the initial fuzzy system and a supervised training to obtain the final FLS.
- Alarming events detection based on audio signal recognition using ANN [Olt15b]. Two ANNs act as pattern recognition systems on two levels. On the first level the system detects only if the event is a dangerous one (alarm on) or a normal one (alarm off). On the second level, if it is the case, the system identifies exactly the nature of events in four classes: chainsaw, gunshot, human voice or tractors. The system uses a set of features of the audio signals associated with the events, as input data for the two ANNs. The experimental results prove that the proposed system is a very reliable one, presenting the maximum possible correct recognition rate (100%) on the first level and very high correct recognition rates on the second level: 99.50% across all data set (training, validation and testing) and 95.0% in the independent testing data subset.

- Temperature control system using a fuzzy logic system as controller [Olt16a], [Olt16b]. The fuzzy logic controller runs on the computer in Simulink, and it controls the external real thermal process, using the computer sound card to carry out the communication with the process.
- Intelligent decision making system for optimal hydroelectric energy generation allocation for the first hydropowerplant in cascaded hydropowerplants using fuzzy logic systems [Olt09b].

Academic and professional achievements

This section briefly reviews the main milestones related to the evolution of my career, from the academic and professional point of view from the moment I received the academic title of Doctor (Ph.D.) in the field of Electronics and Telecommunications, in 2002.

Academic achievements

In 2003, one year after the completion of my doctoral studies, I was promoted to associate professor, and four years later, in 2007, to professor in the department of Bases of Electronics, both positions being occupied after successfully passing the exams to fill these teaching positions.

From the beginning of my didactic activity, I taught laboratory, seminar and project classes, and later I gave lectures in the electronic devices and circuits area, in Romanian, for the students in Electronics and Telecommunications. In 2002 I started to teach Electronic Devices and Circuits in English.

After being promoted to the associate professor position, in 2003, I became in charge for another subject, Design of Electronic Devices and Circuits, in Romanian and I also started to teach lab sessions for a new subject, Fuzzy Systems at postgraduate level. In 2005 I was in charge for the subject Fundamentals of Electronic Circuits, in English, taught for the students in Automation and Applied Informatics. Starting at that moment I reorganised and constantly updated the content of this subject.

Starting with 2007, when I became a professor, I introduced and fully developed the content of a new subject, Computational Intelligence Techniques in Electronics, for the curriculum of a master program. Two years later, in 2009, I took charge for a new subject for the 4th year students, Fuzzy Logic Systems; my main concern was to update, reorganise, and develop its content.

I am currently in charge of the following subjects:

- Electronic Devices and Fundamental Electronic Circuits, in Romanian, bachelor level, field of study: Electronics and Telecommunications Engineering, and Engineering and Management;
- Electronic Devices and Fundamental Electronic Circuits, in English, bachelor level, field of study: Electronics and Telecommunication Engineering;
- Fuzzy Logic Systems, in Romanian and English, bachelor level, specialisation: Applied Electronics;
- Computational Intelligence Techniques in Electronics, in Romanian, master level, specialisation: Integrated Circuits and Systems;
- Fundamentals of Electronic Circuits, in English, bachelor level, specialization: Automation and Applied Informatics.

During my activity, both before and after I received the Ph.D. degree, I was permanently preoccupied with the improvement of my teaching skills, as well as with the development and improvement of the contents for the subjects I was in charge of.

To facilitate the students' access to all teaching resources, I created and constantly updated web pages dedicated to each subject. Each web page is populated with lecture presentations, lab / seminar manuals, example of problems, sample exam subjects, and other useful information.

I was widely concerned to provide my students with supporting materials, as they represent very important learning resources. Accordingly, I elaborated and published a series of books and application manuals:

- 3 books as single-author: "Electronic Devices and Circuits. Electronic Devices", in Romanian (2003, republished in 2004) [Olt04b]; "Electronic Devices" in English (2006) [Olt06c];

"Electronic Circuits", in Romanian (2007) [Olt07b];
- 7 application manuals as single-author or co-author: "Laboratory Manual for Electronic Devices and Circuits. Electronic Devices", in English (2004); "Laboratory Manual for Electronic Devices and Circuits. Electronic Circuits", in English (2004); "Fundamental Electronic Circuits", laboratory manual in Romanian, (2009); "Fundamental Electronic Circuits", laboratory manual in English, (2009); "Electronic Devices", laboratory manual in Romanian (2010); "Electronic Devices", laboratory manual in English (2010); "Manual for Practical Training. Guide to obtain a practice/work placement", in Romanian (2010).

Even if they weren't published in printed volumes, support materials for course and lab activities have been developed and published in electronic format for the subjects of Fuzzy Logic Systems and Computational Intelligence Techniques in Electronics. Given the advanced nature of these subjects and the fact that they are correlated with my main research interests, they have benefited from constant updates. Many results obtained in the research field were transformed and adjusted, so that they became lab/course topics.

I directly contributed to the development of the laboratory of Electronic Devices and Circuits, including the use of funds derived from my research grants, as director or member of the research team. In 2005, I initiated an action for replacing the circuit experiment boards with new, professionally made ones, more reliable and more attractive to students.

Based on my published work, the electronic resources, the equipment and experimental boards available in the lab, I can safely state that all the subjects I am in charge of have the necessary teaching resources for an effective teaching and learning environment.

Given that I was always concerned about the educational aspects, I contributed to the development of several new methods and support materials for teaching/learning activities:

- An intuitive method to deal with the amplification of a zero dc level variable signal using an op-amp operated from a single power supply [Olt04c].
- A web-based Digital Electronics Interactive Course (DEIC). DEIC transmits knowledge to the students by means of static text, figures, tables, and by interactive simulation of some digital circuits. It also offers different kinds of interactive tests with prompt feedback for self-assessment [Olt06d].
- A method that clearly explains why and how the circuit with transistor in common-source connection is capable to perform multiple applications, highlighting the intrinsic operation of the circuit and multiple ways to interpret it. This way, the emphasis is transferred on the understanding the intrinsic operation of a circuit, which opens the perspective for multiple applications, as opposed to the traditional approach, where the emphasis is placed on circuit applications[Olt11].

Moreover, I was involved in the development and deployment of the undergraduate program in Economic Engineering in the Electrical, Electronic and Power Field. I was the initiator of two new subjects in the curriculum, namely Quality Management and Intelligent Decision Support Systems (proposed in 2008). I was and currently am the president of the first graduation committee at this specialisation (2013), and I was in charge of defining the structure and the evaluation process of the diploma projects.

Since 2002 I was constantly appointed as member in various graduation committees, for short-cycle study, bachelor or master level.

My main scientific results, since 2002, can be synthetized in figures, as follows:

- Publication of 57 scientific papers;
- Publication of 2 scientific books, in Romanian publishing houses (2007)[Olt07a], [Fes07];
- Publication of a chapter in a book, in an international publishing house (Springer International Publishing, 2015)[Iva15];
- 54 citations in books, journals and volumes of scientific conferences indexed in ISI / international data bases;

- Director to 2 national research grants;
- Member of the research team of 7 national research grants;
- Member of the team of 2 national agreements, third-party type;
- Reviewer to 6 magazines and scientific conferences indexed in ISI (over 25 reviewed articles);
- Reviewer to 4 magazines / international conferences (over 20 reviewed articles);
- Member of the Evaluation Committee of the scientific papers presented in the student competition "Student Symposium on Electronics and Telecommunications - SSET, (2010 - 2012);
- President of the evaluation committee of the scientific papers presented at the student competition SSET, for the sections "Master IET" and "Bachelor IM" , since 2014;
- IEEE member, since 2000;
- IEEE Society member: Intelligent Computation Society (5 years), Electron Devices Society (2 years), Circuits and System Society (2 years), Computer Society (2 years).

Professional achievements

Besides didactic and research activities, I have always been an active member in the area of institutional development, both at department, faculty, and university levels.

I have also occupied a series of administrative positions, as follows:

- Vice dean of the Faculty of Electronics, Telecommunications and Information Technology, since 2012;
- Scientific secretary of the Faculty of Electronics, Telecommunications and Information Technology, 2008 - 2012;
- Member of the Board of the Bases of Electronics Department, since 2005;
- Member of the Quality Assurance Department of the Technical University of Cluj-Napoca, and responsible with quality assurance in the faculty, 2005-2008;
- Member of the Board of the faculty, since 2000;
- Auditor of CCSTTII of the university for research activity evaluation, 2005 and 2006;
- President of the Committee for Ph.D. Thesis Analysis and Defense, 11 candidates (2009 - 2011);
- Member of the Examining Committee to fill teaching positions (over 10 committees, one for a professor position with University POLITEHNICA of Bucharest), and auxiliary position (4 committees);

I have been and currently am involved in the representation of the university and faculty in national structures through some positions:

- Expert of the Romanian Agency for Quality Assurance In Higher Education (ARACIS - in Romanian) for university programs in the field of Electronics Engineering and Telecommunications, since 2011 (evaluation of 13 programs);
- Expert of the National Council of Scientific Research in Higher Education (CNCSIS - in Romanian), for proposal of research projects 2004 – 2008 (evaluation of 11 project proposals).

As expert in projects funded by the European Social Fund - Sectorial Operational Programme - Human Resources Development, I have participated in the development of human and material resources of the faculty and the university, as follow:

- "Partnership for the practical placement of students (P3S)", POSDRU/ 22/ 2.1/ G/ 24450, 2009 – 2011;
- "Practical training - the first step in your career", POSDRU/90/2.1/S/61811, 2011 -2012;
- "Lifelong learning and training for higher education teachers in the technical sciences and engineering fields" – DIDATEC, POSDRU/ /87/1.3/S/60891, 2010-2013.

I have also facilitated a series of collaborations with the industrial environment:

- Initiation and coordination of an institutional collaboration between the faculty and Robert Bosch Cluj-Napoca, with the main objective of improving and sustaining the academic curriculum. Issues like increasing the study program adaptability to labour market needs, and facilitating the students' transition from education to the working life environment were also on the table. This collaboration was mainly intended for the Economic Engineering in the Electrical, Electronic and Power Field study program (2014 - 2015);
- Development and implementation of a partnership with Samsung Electronics Romania, which resulted in the equipment of a nationwide innovation laboratory within faculty premises, the "Tech Institute - Samsung Innovation Lab". The laboratory is equipped with modern, all-in-one monitors connected to cloud, specially dedicated to Android programming, interactive whiteboard with multiple functions, Smart TV, and mobile devices to test the applications implemented by students (2014-2015);
- Event coordinator for the inauguration of the 'Samsung Innovation Lab ", in the presence of the Minister for Higher Education, representatives of Samsung and local and central media. The event also included the conference "Education in the digital age: How to prepare young people for future careers?", held by prof. dr. Peck Cho, distinguished professor at the University Dongguk and member of the Academy of Technical Sciences of Seoul, South Korea (2014).
- Active member of the faculty/university team, in the discussions with representative companies, for the identification and implementation of some cooperation mechanisms for practical training, education, research and development: Bosch Romania Cluj-Napoca (2014, 2015), EBS (2015, 2016), AROBS (2015), Continental Sibiu (2016), et al.

Other activities related with institutional development:

- Maintenance and update of the faculty web page, 2008-2014;
- Development, maintenance and update to the new version of the faculty web page, since 2014;
- Coordination of the process of student evaluation of teaching activities (EADS - in Romanian) at faculty level, and the elaboration of the whole procedure to be applied at university level, 2005-2008;
- Coordination of all activities related to graduation exams, and the elaboration of regulations and related documents, since 2012;
- Co-organizer of the event "Student Symposium on Electronics and Telecommunications - SSET", since 2012;
- Co-organizer of the event "La multi ani, ETTI" dedicated to celebrating the 25^{th} anniversary of the faculty founding, 2015.

Management of scientific and academic activities

My capacity to coordinate research teams, organize and manage teaching and research activities is highlighted by the numerous activities in which I was involved, as well as their results.

As far as the teaching component is concerned, all the activities in which I was involved (giving lectures, teaching application classes, preparing and disseminating support materials) were very well organized and managed. These conclusions result from the students' positive direct feedback (during classes), as well as from the student evaluation of teaching activities (EADS – in Romanian) questionnaires. In this regard, I can mention:

- Within EADS, the subjects I teach have always obtained a score located in the upper half, above the average for the year of study, on a scale from 1 (minimum) to 5 (maximum): Electronic Devices (4.42, 3.72, 3.89), Fundamental Electronic Circuits: (4.10, 4.33, 4.89, 4.29, Fuzzy Logic Systems: (4.19, 4.02, 4.69), Computational Intelligence Techniques in Electronics: 4.28;
- In 2009, the 4th year graduating students from the English study line appointed me as "Cel mai iubit cadru didactic", a tradition used in our faculty to designate the most appreciated professor;
- In 2014 I was awarded with the title of "Profesor Bologna" by the ANOSR organization (the nationwide organization of students from Romania), while being previously nominated by the students in my faculty.

Every academic year, bachelor and master level students ask me to be the scientific supervisor of their graduation projects, and I gladly accept. During the past 10 years (2006-2015), the successful collaboration with students has resulted in me supervising over 30 diploma projects (bachelor level) and 7 master theses (master level). Some of these projects and theses were included among the best ones by the graduation committees, as follows:

- "Best diploma project", awarded by each graduation committee:
  - Ciortea Florin Teodor, Sorting System based on Fuzzy Logic, diploma project, 2010;
  - Tarta Luminita Ioana, Smart House Prototype, diploma project, 2013.
- "High rated graduation theses", at faculty level:
  - Ciortea Florin Teodor, Sorting System based on Fuzzy Logic, diploma project, 2010;
  - Caucean Sergiu Razvan, Nonlinear System Identification using Fuzzy Systems, diploma project, 2010;
  - Rodila Alexandru, Speed Control of a DC Motor using Fuzzy Logic, diploma project, 2010;
  - Tarta Luminita Ioana, Smart House Prototype, diploma project, 2013;
  - Abrudan Adriana-Florentina, Control System for Water Temperature, diploma project, 2013;
  - Morari Florina, Isolated Words Recognition using Artificial Neural Network, diploma project, 2013;
  - Vele Carmen-Aira, Control System for Water Purity, diploma project, 2013;
  - Prodan Alex, Prediction of Waveforms under the Variation of Input Parameters using Neural Networks, master thesis, 2013;

I have also obtained remarkable results as the coordinator of students for scientific papers presented at the "Student Symposium on Electronics and Telecommunications - SSET":

- Paul Farago, A Genetic Algorithm-based Approach for Passive Filter Design, 1st prize, section: “Doctor ETTI”, 2010;
- Alex Prodan, Prediction of Waveforms under the Variation of Input Factors using Neural Network, 3rd prize, section “Master/Doctor ETTI”, 2013;
- Marius Nechiti, Laser Microphone, 2nd prize, section “Student IET”, 2014.

The ability to coordinate research teams and organize research activities result from the activities and outcomes of the research contracts for which I was responsible, as well as through the collaborations I had, especially with doctoral students.

Therefore, as project manager I organized and coordinated activities for:

- AT-type CNCSIS research grant "Development of a Computer Aided Design Tool for Optimizing Analog Circuit Design using Fuzzy Techniques". The project was successfully carried out over a period of two years (2001-2002) with funding reapplication for the second year. In the frame of this project I coordinated a team composed by 3 teaching staff member (and Ph.D. students at that time) and one student.
- A-type CNCSIS research grant “Analog Modules in Nanometric Technology - Development of Some Functional Models using Fuzzy Techniques”. The project was successfully carried out for two years (2005 - 2006) with funding reapplication for the second year. In this project I coordinated a team composed by 6 teaching staff member (3 of them were Ph.D. students at that time) and two students.

I have also coordinated a team of 4 members, in the frame of the third-party contract "The Design of DIGITAL ASIC Circuits for Image Processing Applications", beneficiary: SC Pitagorasic srl, Brasov", 2006.

I have been involved in doctoral programs in the Electronics and Telecommunications Engineering domain:

- I participated 3 times as a member of the Committee for Ph.D. Thesis Analysis and Defense, for the following candidates: Claudia Moisa (cas. Farago), 2015; Laura Nicoleta-Ivanciu, 2014; Levente Szalontai, 2012.
- I was a member of the evaluation committee for research reports, in more than 25 committees;
- I have been member of the committees for guidance and control for more than 10 doctoral students that have prof. Dana Marina Țopa, Ph. D. or prof. Sorin Hintea, Ph.D., as scientific advisers.

Through the years, I worked with and advised several PhD students that were carrying out their doctoral research activity. The results of these collaborations were used both in their doctoral theses and also in the development and publication of scientific papers. Only the doctoral theses related with the computational intelligence field are mentioned here:

- “Contributions to the Design of Integrated Circuits and Systems using Computational Intelligence Techniques’, Claudia Moisa (cas. Farago), 2015. We coauthored 3 papers: An Electronically Tunable Transconductance Amplifier for Use in Auditory Prostheses, 2015 [Far15a]; An evolutionary approach for nominal design and yield enhancement of analog amplifiers, 2015 [Far15b]; An Evolutionary FPAA Reconfiguration Algorithm Based on the Open Traveling Salesman Problem Routine, 2015 [Far15c].
- “Contributions to the Design of Telecommunications Receivers using Computational Intelligence Techniques”, Laura-Nicoleta Ivanciu, 2014. We co-authored 3 papers on the topic of the thesis: Parameter Distribution in Receiver Chains: a Hybrid Fuzzy-Genetic Algorithm Approach, 2013 [Iva13a]; A TSK Fuzzy Approach to Channel Estimation for 802.11a WLANs, 2013 [Iva13b]; Design Illustration of a Symmetric OTA using Multiobjective Genetic Algorithms, 2011 [Iva11]. We also co-authored a chapter (Design of Telecommunication

Receivers Using Computational Intelligence Techniques) in the book published by Springer International Publishing, 2015 [Iva15].

- "Synthesis of Some Multivalent Logic Circuits", Emilia Mocean (căs. Şipoş), 2012. We co-authored 3 papers on the topic of the thesis: Simulink Library of Basic Building Blocks for Ternary Logic, 2009 [Sip09]; Towards Reconfigurable Circuits Based on Ternary Controlled Analog Multiplexers/Demultiplexers, 2008 [Sip08a]; A Method to Design Ternary Multiplexers Controlled by Ternary Signals Based on SUS-LOC, 2008 [Sip08b];
- “Advanced Design Techniques for Some Analog Reconfigurable Circuits”, Paul Farago, 2012. We co-authored 2 papers on the topic of the thesis: “A Programable Gain Amplifier for Automated Gain Control in Auditory Prostheses, 2011 [Hin11]; A Double-Layer Genetic Algorithm for Gm-C Filter Design, 2010 [Far10];

Career development plan

My entire activity within the faculty of Electronics, Telecommunications and Information Technology has been and will be oriented towards fulfilling the mission and objectives of the faculty, as follows:

The mission of the faculty: "To contribute through advanced research to the knowledge development in the fields of electronics, telecommunications and information technology and to prepare specialists able to develop, design, implement and operate electronic and telecommunication systems with applications in various fields of industry and of everyday life".

Objectives of the faculty:

- Improving the quality of study programs and improving the professional performance of students and graduates;
- Developing the performance in scientific research.

My career development plan relies on all components specific to a successful academic activity (didactic, scientific and institutional), in various ways: every component with its own specific approach, specific objectives and actions, but also an integrated approach generating synergy to produce combined results that are expected to be more valuable than the ones generated by each component alone.

I have identified several major intervention directions that can be associated with general goals, as follows:

- G1 - Orientation towards a student-centred learning environment;
- G2 – Development of research-informed teaching;
- G3 - Modernization and internationalisation of subjects/study programs;
- G4 - Contribution to knowledge development through fundamental and applied research;
- G5 – Increasing the research visibility;
- G5 – Development of resources;
- G7 – Development of collaborations with the industrial environment.

Further on there is a descriptions of these general goals and probable modes of action for their implementation.

G1 - Orientation towards a student-centred learning environment

Achieving this goal is primarily based on the capitalization of the teaching experience I have gained in my career so far, in more than 20 years, as a solid foundation for orienting towards the creation of a student-centred teaching / learning environment.

The main activities I intend to implement, in order to achieve this goal are:

- Development and update of the content for all the subjects I am in charge of, and providing the students with all the necessary materials in electronic format, on-line and/or off-line. I will continue to maintain the existing web pages dedicated to each subject; each web page is populated with lecture presentations, software demonstrations, lab/seminar manuals, example of problems, sample exam subjects, and other useful information. For illustrative purposes, the web page for the subject "Fuzzy Logic Systems" can be freely accessed: http://www.bel.utcluj.ro/dce/didactic/sln/sln.htm).
  This way, students have the flexibility to learn anytime and anywhere, at their own pace, an important attribute of student-centred learning.

- Increasing the interactivity of my lectures by extended inductive teaching and learning methods as: inquiry-based learning, case-based instruction, problem-based learning, discovery learning and also by increasing the number of examples and case studies presented and actively discussed with my students. Using these methods leads to active learning and the focus of activity can be shifted from the teacher to the learners.
- Combining modern teaching methods (information and communications technology - ICT) with classical teaching methods. ICT provides several advantages for teaching and learning: high quality materials, easier explanation of complex aspects using multimedia, possibilities to create interactive classes for more attractive lectures, activation of student participation concentration and interest, finally conducting to a higher rate of understanding and retaining new knowledge.
- Guiding and counselling students, including during the weekly consultations, so that each student can receive personalized guidance (by suggesting additional sources of documentation or by direct discussions and answers).
- Increasing the quality of the teaching-learning process by valorising the feedback provided directly by the students but also the feedback provided by the institutional evaluation of teaching activities by the students.

G2 - Development of research-informed teaching

Research-informed teaching deals with the practice of connecting research activities and results with teaching and learning activities. Integrating some advanced concepts and methods in teaching curriculum subjects, will encourage my students to develop into accomplished and valuable graduates who can confidently take their place on the labour market. They will be well equipped either for an academic career or an industrial one, with a deep understanding of their potential, but also of their limitations.

I intend to develop the research-informed teaching by:

- Integrating the results of research in teaching on both levels, bachelor and master, for the specialized subjects: Fuzzy Logic Systems, bachelor, 4^{th} year and Computational Intelligence Techniques in Electronics, master, 1^{st} year).
- Coordinating students for diploma projects and dissertations theses on topics emerging from my research directions / interests.
- Involving and coordinating the students' participation in extracurricular activities with scientific character such as: including them in research and development projects and participation in student scientific competitions as SSET - Student's Symposium of Electronics and Telecommunications.

G3 - Modernization and internationalisation of subjects/study programs

The modernisation and internationalisation of the study programs is a must in the actual and future conditions of a global labour market, but also at a national level, considering the multitude of multinational companies acting in our country that ask for our well trained graduates. On the other hand, it became more and more important for our faculty /university to be present and to be integrated on the international education market with educational offers that attract students from other countries. Besides the subjects I am currently teaching in English at bachelor level (Electronic Devices and Fundamental Electronic Circuits), I plan to contribute further through the following activities:

- For the subject I tech at master level (Computational Intelligence Techniques in Electronics) I intend to transpose all the teaching materials in English, so that this subject becomes a valid option for international ERASMUS students. Furthermore, I will militate in favour of offering the entire master program in English, thus increasing the international visibility and internationalization of our study programs and faculty.

- Recalibrating the subject ”Fuzzy Logic Systems", offered to bachelor, 4th year, by emphasizing on the control systems based on fuzzy controllers part. Also, for the laboratory classes I intend to increase the proportion of practical experiments (hands-on) while limiting the number of computer simulation experiments. The idea is to implement control systems on development boards (embedded systems), including the concept of model-based design and implementation. This approach already is in progress, as I am supervising two students in Applied Electronics for their diploma projects, which consists of the development and implementation of two applications with fuzzy logic controllers on some development boards.

G4 - Contribution to knowledge development through fundamental and applied research

As it results from the above section “Scientific achievements” and also from my list of scientific publications (77 scientific papers, 3 books in the scientific domain, 2 research projects as project manager and 9 projects as a team member) during my career I accumulated a lot of experience in performing and coordinating research activities. All these achievements make for a successful future activity in research and development, both for the research activity itself as well as for proper management and coordination of all necessary components.

I will continue the research activities particularly in the field of applications of computational intelligence techniques in different areas of electronics. My further research directions are (but not limited to):

- Development and implementation of an automatic procedure for the generation and exhaustive analysis of possible waveforms, in various parts of an electronic system, depending on the combinations of some parameters, using cost effective metamodels (OWTM). This way, a wide range of possibilities becomes achievable to the designer: all design corners can be explored, possible worst-case situations can be investigated, extreme values of waveforms can be discovered, sensitivity analyses can be performed, and so on. To generate new parameter combinations for a well-controlled, systematic data collection, the DoE (design of the experiment) concept should be invoked. There are a lot of methods to be investigated, as for example: full factorial design, fractional factorial design, response surface design, D-optimal design, latin hypercube sample design, quasi random design or sequential optimal space. Two important DoE concepts, namely blocking and replications, can be used in order to improve the analysis of the electronic system performance in the case of variations of both parameters and measurements.
- The use of computational intelligence techniques in the development of systems for surveillance and public safety insurance. This research direction is consistent with research project "Interactive Real-Time Video Surveillance Services for Improved Public Safety in Crowded Urban Areas - INTEREVISS" where I am a member of the research team. The following aspects are considered:
    - Development of some fuzzy expert systems dedicated to the analysis of video sequences from the camera to identify the risk situations. The implementations can be made on two levels: at the first level the type of objects in the scene, and their movement/activity should be identified, while on the second level a decision should be made if there is an alarming situation or not.
    - Design and implementation of decision-support systems, for warning and alerting, based on fuzzy expert systems and fuzzy rules. The decision support system should be user-friendly with a graphical interface as the central element through which various areas under surveillance are monitored and managed. The graphical user interface will realize the information transfer between operational personnel and decision support system, giving to

the user the ability to monitor and perform a pertinent analysis of the situation and take the most appropriate decisions knowingly.
    - Implementation of a strategy based on computational intelligence techniques for people evacuation in case of large crowds and emergency situations. The strategy can be based on fuzzy systems to calculate some priorities of the evacuation routes, and on a genetic algorithm to optimize the evacuation strategy.
    - Implementation of some solutions for processing and intelligent analysis (based on artificial neural networks) of colour images and video sequences from the camera.
- Investigation of a new research direction referring to the optimization of fuzzy logic systems and artificial neural networks utilizing genetic algorithms. Two different aspects can be considered:
    - In control systems, some characteristics of the process under control can change in some way (with time, with environmental conditions). As a result an initially well-tuned controller is no longer acting optimally. Even a fuzzy controller cannot completely accommodate all the variations as long as their parameters are fixed. A possible solution is to develop fuzzy adaptive controllers (for example first order Takagi-Sugeno) whose parameters (position and/or shape of fuzzy sets) can be adapted using an off-line optimization involving a genetic algorithm. The parameters of the fuzzy sets will be genes of the chromosome, and the objective function should be formulated to obtain an optimal response of the control system.
    - In CI based pattern recognition systems or systems/functions modeling, the classic supervised learning methods based on numeric data can be replaced by genetic algorithm optimisation in the quest of the best structure and parameters of the involved artificial neural network or fuzzy logic system.

After obtaining the status of Ph.D. adviser, I will advise doctoral students primarily in the research areas listed above, but I'll show full openness to other adjacent research directions, if favourable conditions should emerge.

As Ph.D. adviser, being up to date with recent advances in my current research field is even more important. The voluntary participation in scientific workshops, seminars, round-tables needs to become mandatory, provided that I intend to offer my Ph.D. students consistent feedback and support, in their research activities.

G5 – Increasing the research visibility

From the point of view of research activity it is important for a researcher and for his/her results to gain some national and international visibility. Being visible is one important condition for being part of cooperative research and development projects.

My approach to strengthen and enhance the impact of my research activity and results, as well as to contribute to the increase of visibility for the faculty/university, consists in achieving the following actions:

- To continue to publish scientific papers in selected journals and conferences, especially the one that are indexed in the ISI Web of Science data base or other international data bases (as IEEE Explore, Scopus, SpringerLink, etc.). So far, I have already published 22 papers indexed in the ISI data base and 34 papers in international data bases. Also, I plan to regularly publish in the journal edited by our faculty, Acta Technica Napocensis - Electronics and Telecommunications.
- To continue and extend my activity as member in scientific/programme committees and as reviewer for journals and conferences registered in the ISI database and other international databases. Over the years I have been involved with 10 journals and conferences (more than 45 reviewed papers), as for example The International Journal for Computation and Mathematics in Electrical and Electronic Engineering - COMPEL, Acta Technica Napocensis – Electronics and Telecommunications, UK Workshop on Computational Intelligence - UKCI, etc.

Also, for promoting the research results in the field of applications of computational intelligence techniques I intend to propose and chair a special session to a conference in the field, as for example UK Workshop on Computational Intelligence – UKCI.

- To participate in social networking sites for scientists and researchers to share papers, ask and answer questions, and find collaborators. Currently, I am active in the largest academic social network, ResearchGate, where I accumulated more that 1k reads in the last year.

As a result of these actions I expect an increase of my international visibility, including the number of citations. Currently, I have 29 citations in books, journals and volumes of scientific conferences indexed in ISI data base and another 27 citations in journals and volumes of scientific conferences indexed in international data bases, with citation indices (h-index) of 6 (Google Scholar), 4 (Scopus), and 3 (ISI).

G6 - Development of resources

For a successful activity, one very important premise is the existence of necessary resources. Human and material resources are two most important ingredients in the progress and development of any domain, including education and research.

Currently I am a member of the research structure "Integrated Circuits and System Group" from the Bases of Electronics Department. The existence of a research team is, in my opinion, the basic condition for successful research and development activities.

In the future I will focus on the development of a research team, which will consist in both Ph.D. students (who bring a dynamic and fresh approach), as well as a number of teaching staff members (who ensure stability and continuity).

Anyway, it is obvious that it is not at all an easy task to find candidates able and motivated to face the challenges of an intensive research activity lasting at least three years, of writing and publishing scientific articles, of elaborating and defending a Ph.D. thesis. One way to approach this issue is to start the preparation of human resources early. There must be an ongoing concern to attract students, even at bachelor level, towards some extracurricular activities, as research and developments. They should be gradually integrated in such activities by their inclusion in research and development teams or by sustaining and coordinating them for the participation in student scientific competitions as SSET or Tudor Tanăsescu). The students shall be promoted along their entire professional education program, by advising them for diploma project, by working with them in the research and design activity (ACP in Romanian) at master level, and then coordinating their dissertation, all of these on research-oriented topics.

During doctoral studies, it is important for the doctoral candidates to follow a judicious planning of activities: documentation, research, development, implementation, testing, writing, publishing, etc. Increased attention will be given to the dissemination of the research results by their publication in journals and at conferences. An important contribution to the professional training of PhD students can be brought by their participation in scientific conferences by offering them opportunities to directly interact with other researcher and experts in the field, to make new contacts, to sustain their ideas or to hear and debate on new ideas and approaches.

Supplementary efforts will be needed to identify and attract additional research funding to sustain the Ph.D. students and their doctoral research activities. These funds can originate either by winning/ participating in national research grants, bilateral or international research projects, or by carrying out research in collaboration with prestigious companies. In the latter case, we can develop research directions supported by companies, with positive impact on both parties.

G7 – Development of collaborations with the industrial environment

One of my priorities will be the continuous development of the industry collaboration, action that targets the institutional development and is based on the fusion of all components of the academic activity.

As a consequence of the expressions of interest from several major companies (Continental Sibiu, EBS Cluj-Napoca, Robert Bosch Cluj-Napoca) I believe that there is a favourable environment and we can strive to initiate and sustain a series of win-win activities, such as:

- Obtaining some agreements (collaboration contracts) for the development / implementation / testing of products (third-party type agreements) that should engage both students and teaching staff, with beneficial effects both on the students training and on developing human and material resources in the faculty.
- Updating, developing, or recalibrating of some study subjects (especially at master level) or even the curriculum, that shall develop students' skills required by the labour market. For these actions, representatives of companies can be actively involved as consultants. Several subjects can be taught by partially using some resources provided by companies (lectures/applications taught by specialists from companies, support materials and equipment coming from the companies, working visits in companies, etc.).
- Obtaining scholarships or other forms of support to students at all levels (undergraduate, master, doctorate) granted by companies, so that the students are able to complete their entire professional education, before being directly attracted into the labour market.
- Topics for graduation theses provided/suggested by companies, at both bachelor and master levels, topics on which students will have a company specialist as co-adviser.
- Upgrading and equipping some teaching and research spaces with necessary means for teaching, learning and research activities.
- Organising practical training for students, in real working environments.
- Organising training sessions, in partnership, for the companies' personnel or for the teaching staff in faculty.
- Ensuring technology transfer and technological innovation by creating conditions for the possibility of participating in joint scientific research project.

I strongly believe all the above mentioned actions can be successfully implemented. We can greatly benefit from the previously gained experience, since similar activities were carried out or are in progress in our faculty.

The aforementioned goals outline my career development strategy. The proposed activities and actions can ensure the achievement of the goals; however, they will be re-analyzed and adjusted, with respect to the future state.

Continuous education is a lifelong commitment, in my opinion, and thus I intend to spare no efforts in upgrading my knowledge, keeping in touch with novel scientific breakthroughs and contributing to the advance of our faculty, in both teaching and research activities.

References

1 [Agu94] M. A. Aguirre, J. Chavez, A. Torralba, L. G. Franquelo, Analog Design Optimization by means of Tabu Search Approach, Proc of the IEEE Int. Symp. of Circuits and Systems ISCAS'94, London May 1994, Vol 1, pp. 375-378

2 [Alp03] Alpaydin, G., Balkir, S., Dundar, G., An Evolutionary Approach to Automatic Synthesis of High-Performance Analog Integrated Circuits, IEEE Transaction on Evolutionary Computation, Vol. 7, No.3, June 2003, pp. 240- 252, 2003;

3 [Bag07] M. S. Baghini, R. Kanphade, D. G. Wakade, P. Gawande, M. Chhangani, M. Patil, GP-based Design and Optimization of a Floating Voltage Source for Low-Power and Highly Tunable OTA Applications, *WSEAS Transactions on Circuits And Systems*, Issue 10, Volume 6, ISSN: 1109-2734, October 2007, pp. 577- 582.

4 [Bal04] S. Balkir, G. Dundar, G. Alpaydin, Evolution Based Synthesis of Analog Integrated Circuits and Systems, IEEE NASA/DoD Conference on Evolution Hardware, EH'04, ISBN 0-7695-2145-2/04, 2004. pp. 26-29;

5 [Bao08] Bao, Z., Watanabe, T.: A New Approach for Circuit Design Optimization using Genetic Algorithm. In: International SoC Design Conference, Busan, 2008, pp. 383- 386;

6 [Bar05] M. Barros, G. Neves, J. Guilherme, N. Horta, An evolutionary optimization kernel with adaptive parameters applied to analog circuit design, International Symposium on Signals, Circuits and Systems, ISSCS 2005, Volume 2, 14-15 July 2005, pp. 545 – 548.

7 [Bar06] M. Barros, J. Guilherme, N. Horta, GA-SVM Optimization Kernel Applied to Analog IC Design Automation, Proc. of 13th IEEE International Conference on Electronics, Circuits and Systems, ICECS06, 2006, pp. 486-489

8 [Ber03] F. De Bernardis, M. I. Jordan, A. Sangiovanni Vincentelly, Support Vector Machines for Analog Circuit Performance Representation, DAC 2003, Anaheim, California, USA, 1-58113-688-9/03/0006, 2003, pp. 964-969.

9 [Bin07] D. M. Binkley, Tradeoffs and Optimization in Analog CMOS Design, 14th International Conference on Mixed Design of Integrated Circuits and Systems, MIXDES07, 21-23 June 2007, pp. 47 – 60.

10 [Che95] Chen, W.-K., Editor, Synthesis and Design of Nonlinear Circuits, The Circuit and Filters Handbook, CRC Press, pp. 935-945, 1995.

11 [Coi95] Coifman RR, Donoho DL, Stanford University Department of Statistics. Translation invariant de-noising. 1995; Report No.: 475.

12 [Dau93] Daubechies I. Ten lectures on wavelets. J Acoust Soc Am. 1993; 93(3):1671.

13 [DoE] Design of Experiments, Statistics and Machine Learning Toolbox. The MathWorks, Inc. Available: http://www.mathworks.com/help/stats/design-of-experiments-1.html. Accessed 2015 Jul 9.

14 [Dud01] R. O. Duda, P. E. Hart, and D. G. Stork, Pattern Classification, John Wiley & Sons, Inc., Second Edition, 2001

15 [Eec06] T. Eeckelaert, R. Schoofs, G. Gielen, M. Steyaert, W. Sansen, Hierarchical Bottom-up Analog Optimization Methodology Validated by a Delta-Sigma A/D Converter Design for the 802.11a/b/g standard, Design Automation Conference, 43rd ACM/IEEE, 2006, pp. 25-30.

16 [Eri08] Eriksson L. Design of experiments. Umeå, Sweden: Umetrics Academy; 2008.

17 [Fak15] Mourad Fakhfakh, Esteban Tlelo-Cuautle, Patrick Siarry - editors, Computational Intelligence in Analog and Mixed-Signal (AMS) and Radio-

Frequency (RF) Circuit Design, ISBN 978-3-319-19871-2, Springer International Publishing, 2015, 491pp;

18 **[Far10]** Farago, P., Hintea, S., **Oltean, G.,** Festila, Lelia, A Double-Layer Genetic Algorithm for Gm-C Filter Design, 14rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, KES2010, Cardiff, UK, September 8-10, 2010, Pages: 623-632; Publisher: Springer-Verlag Berlin, LNAI 6279, ISBN 0302-9743, ISBN-13 978-3-642-15383-9 DOI: 10.1007/978-3-642-15384-6;

19 **[Far14]** Farago, P., **Oltean, G**., Ivanciu, Laura, Neural networks and wavelet transform in waveform approximation, IEEE 14th International Workshop on Computational Intelligence (UKCI2014), September 8-10, 2014, Bradford, UK, DOI: 10.1109/UKCI.2014.6930164, pp. 1- 8;

20 **[Far15a]** Farago, P., Farago, Claudia, **Oltean, G**., Hintea, S., An Electronically Tunable Transconductance Amplifier for Use in Auditory Prostheses, Advances in Electrical and Computer Engineering, ISSN: 1582-7445, e-ISSN: 1844-7600, Volume 15, Issue 4, DOI: 10.4316/AECE.2015.04013, 2015, pp. 95 -100;

21 **[Far15b]** Farago, Claudia, **Oltean, G**., Hintea, S., Farago, P., An evolutionary approach for nominal design and yield enhancement of analog amplifiers, IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI2015), 21-23 May, 2015, Timisoara, Romania, DOI: 10.1109/SACI.2015.7208210, pp. 265 - 270;

22 **[Far15c]** Farago, Claudia, Farago, P., Hintea, S., **Oltean, G**., An Evolutionary FPAA Reconfiguration Algorithm Based on the Open Traveling Salesman Problem Routine, Acta Technica Napocensis. Electronics and Telecommunications, Nr. 1/2015, ISSN 1221-6542, Vol.56, pp. 21 - 28;

23 [Far95] M. Fares, B. Kaminska, FPAD: A Fuzzy Nonlinear Programming Approach to Analog Circuit Design, IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 14, no.7, July 1995, pp. 785-793.

24 [Fen10] Gang Fen, Analysis and Synthesis of Fuzzy Control Systems. A model-based Approach, CRC Press, Taylor and Francis Group, 2010.

25 [Fer05] Ferrari S, Stengel R. Smooth Function Approximation Using Neural Networks. IEEE Trans Neural Netw. 2005;16(1):24-38.

26 **[Fes07]** Festila, Lelia, Gordan, Mihaela, Cirlugea, Mihela, **Oltean, G**., Hintea, S., Csipkes, G., Rus, C., Groza, R., Szolga, L., Circuite analogice pentru clasificatoare masini cu vectori suport Editura Casa Cartii de Stiinta, Cluj-Napoca, ISBN 978-973-133-107-2, 168 pag., 2007.

27 [Fie14] Field S, Galley C, Hesthaven J, Kaye J, Tiglio M. Fast Prediction and Evaluation of Gravitational Waveforms Using Surrogate Models. Phys Rev X. 2014; 4(3).

28 [FLT] Fuzzy Logic Toolbox, MathWorks, Available from http://www.mathworks.com/help/fuzzy/index.html, [cited March 2016]

29 [Fu99] L. Fu, J. Fan, L. Yuanchun, T. Yantao, D.Yisong, Study and Application of a Constrained Multi-objective Optimization Algorithm, IEEE International Vehicle Electronics Conference, IVEC 1999, pp. 305-307.

30 [GA] Genetic Algorithm, Global Optimization Toolbox, MathWorks, Available from http://www.mathworks.com/help/gads/genetic-algorithm.html, [cited March 2016]

31 [Gar12] Garitselov O, Mohanty S, Kougianos E. A comparative study of metamodels for fast and accurate simulation of nano-CMOS circuits. IEEE Trans Semicond Manufact. 2012; 25(1):26-36

32 [Gie00] G.E. Gielen and R.A. Rutenbar, “Computer Aided Design of Analog and Mixed Signal Integrated Circuits”, *Proceedings of IEEE*, Vol.88, No.12, pp. 1825-1852, 2000.

33 [Gie90] R. L. Gieger, P. E. Allen, N. R. Strader, *VLSI Design Techniques for Analog and Digital Circuits*, McGraw-Hill Publishing Company, 1990.

34 [God97] Godjevac, J., Neuro-Fuzzy Controllers. Design and Application, Presses politechniques et universitaries romandes, Lausanne, 1997;

35 [Gre07] Greenwood, G.W., Tyrrell, A.M.: Introduction to Evolvable Hardware, IEEE Press Series on Computational Intelligence, Wiley&Sons Inc., 2007.

36 [Gyo11] R. S. Győrödi, Pattern Recognition Systems (in Romanian), available at http://rgyorodi.webhost.uoradea.ro/modules/SRF/, 2011

37 [Had11] M.L. Hadjili, K. Kara, Modelling and control using Takagi-Sugeno fuzzy models, Saudi International Electronics, Communications and Photonics Conference (SIECPC), 24-26 April, Riyad, 2011.

38 [Has89] M. Hashizume, H.Y. Kawai, K. Nii, T. Tamesada, Design Automation System for Analog Circuits Based on Fuzzy Logic, IEEE Custom Integrated Circuits Conference, 1989, pp. 461-464.

39 [Her01] M. del Mar Hershenson, S. P. Boyd, T. H. Lee, Optimal Design of a CMOS Op-amp via Geometric Programming, IEEE Trans. on Computer-Aided Design of Integrated Circuits and System, Vol. 20, No.1, January 2001, pp. 1-21.

40 [Her04] [29] M. del Mar Hershenson, CMOS analog circuit design via geometric programming, American Control Conference, Volume 4, Issue , 30 June-2 July, 2004, pp. 3266 – 3271.

41 **[Hin11]** Hintea, S, Farago, P., Roman M.N., **Oltean, G.**, Festila, Lelia, A Programable Gain Amplifier for Automated Gain Control in Auditory Prostheses, Journal of Medical and Biological Engineering, Vol 31, No3, 2011, pp. 185-192, ISSN: 1609-0985, doi: 10.5405/jmbe.740;

42 [Hua15] Huang Q, Li X, Yang F, Zeng X, Li X. SIPredict: efficient post-layout waveform prediction via system identification. In: ASP-DAC Proceedings of the 20th Asia and South Pacific Design Automation Conference, 2015 Jan 19–22; Chiba, Japan. IEEE; 2015. pp. 460–465.

43 **[Iva11]** Ivanciu, Laura, **Oltean, G**., Hintea, S., Design Illustration of a Symmetric OTA Using Multiobjective Genetic Algorithms, Knowledge-Based and Intelligent Information and Engineering Systems; Lecture Notes in Computer Science, 2011, Volume 6883/2011, pp. 443-452, ISBN 978-3-642-23853-6, Publisher Springer, DOI: 10.1007/978-3-642-23854-3_47;

44 **[Iva13a]** Ivanciu, Laura, **Oltean, G.**, Parameter Distribution in Receiver Chains: a Hybrid Fuzzy-Genetic Algorithm Approach, Acta Technica Napocensis. Electronics and Telecommunications, Nr. 4/2013, ISSN 1221-6542, Vol.54, pp. 28-33

45 **[Iva13b]** Ivanciu, Laura, Cremene Chira, Ligia, **Oltean, G.**, A TSK Fuzzy Approach to Channel Estimation for 802.11a WLANs, Springer LNCS book chapter, Emerging Trends in Computing, Informatics, Systems Sciences, and Engineering Lecture Notes in Electrical Engineering, Vol. 151, 2013, pp 831-840;

46 **[Iva15]** Ivanciu, Laura, **Oltean, G**., "Design of Telecommunication Receivers Using Computational Intelligence Techniques", book chapter (pp. 413-438) in Mourad Fakhfakh, Esteban Tlelo-Cuautle, Patrick Siarry - editors, Computational Intelligence in Analog and Mixed-Signal (AMS) and Radio-Frequency (RF) Circuit Design, ISBN 978-3-319-19871-2, Springer International Publishing, 2015, 491pp;

47 [Jaf10] Jafari, A., Zekri, M., Sadri, S., Mallahzadeh, A. R.: Design of Analog Integrated Circuits by Using Genetic Algorithm, In: Second International Conference on Computer Engineering and Applications, Bali Island, 2010, pp. 578-581;

48 [Jan93] Jang, R.J-S., ANFIS, Adaptive-Network-Based Fuzzy Inference System, IEEE Transaction on System, Man, and Cybernetics, Vol. 23, No. 3, pp. 665-685, 1993;

49 [Jiu10] Jiuchao L, Jingchang N, Jingmei Z. Study and simulation of RF power amplifier behavioral model based on RBF neural network. In: ICMMT 2010 Proceedings of the International Conference on Microwave and Millimeter Wave Technology, 2010 May 8–11; Chengdu, China. Curran Associates, Inc; 2010. p. 1465–1467.

50 [Kha10] Khashei M, Bijari M. An artificial neural network (p,d,q) model for timeseries forecasting. Expert Systems with Applications. 2010;37(1):479-489.

51 [Kia14] Kianifar MR, Campean F, Wood A. PermGA algorithm for a sequential optimal space filling DoE framework, Proceedings of 14th UK Workshop on Computational Intelligence, 2014 Sept 8–10; Bradford, UK. IEEE; 2014. p.1–8

52 [Kie04] T. Kiely, G. Gielen, Performance Modeling of Analog Integrated Circuits Using Least-Squares Support Vector Machines, Proc. of Design, Automation and Test in Europe Conference and Exhibition, Vol. 1, 2004, pp. 448-453.

53 [Koh90] H. Y. Koh, C. H. Sequin, P. R. Gray, OPASYN: A Compiler for CMOS Operational Amplifiers, *IEEE Transactions on Computer-Aided Design*, vol.9, no.2, February 1990, pp. 113-125.

54 [Kon07] Konar A. Computational intelligence: principles, techniques and applications. Berlin: Springer Science & Business Media; 2006.

55 [Kuo05] Kuo, C-C., Liu C-N. J., Accurate behavioral modeling approach for PLL designs with supply noise effects, Proc of the IEEE International Behavioral Modeling and Simulation Workshop, 2005, pp. 48 – 55.

56 [Lag08] I. Lagrat, A. El Ougli, I. Boumhidi, Optimal Adaptive Fuzzy Control for a Class of Unknown Nonlinear Systems, *WSEAS Transactions On Computer Research*, Issue 1, Vol. 3, ISSN: 1991-8755, Jan. 2008, pp. 89-98.

57 [Lis04] Lister, P., Bassett, M., Watten, P., Trignano V., and Jackson, B., A systemC based virtual prototyping methodology for embedded systems, IP SoC Design Forum & Exhibition IP-SOC, France, 2004, http://www.us.design-reuse.com/ipbasedsocdesign/slides_2004-sussex_01.html

58 [Lup06] Florin Lupşa, Modelarea functionala a unui amplificator cascoda cu tranzistoare MOS, in tehnologie nanometrica utilizand tehnici fuzzy, Proiect de diploma, Conducator ştiinţific: Conf.dr.ing. **Gabriel Oltean**, Facultatea de Electronică, Telecomunicaţii şi Tehnologia Informaţiei UTCN, 2006

59 [Mau93] P. C. Maulik, L. R. Carley, D. J. Allstot, Sizing of Cell-Level Analog Circuits Using Constrained Optimization Techniques, IEEE Journal of Solid State Circuits, vol. 28, no. 3, March 1993, pp. 233-240.

60 [Men95] J. M. Mendel, Fuzzy Logic Systems for Engineering: A Tutorial, *Proc. of the IEEE*, Vol.83, No. 3, March 1995, pp. 345-377.

61 [Mic96] C. Michael, H. Su, M. Ismail, A. Kankunnen, M. Valtonen, Statistical Techniques for the Computer-Aided Optimization of Analog Integrated Circuit, *IEEE Transactions on Circuits and Systems-1; Fundamental Theory and Applications*, vol. 43, no. 5, May 1996, pp. 410-413.

62 [Mog09] Moghbelli H, Rahideh A, Safavi A A. Using wavelet based neural networks for feedback signals estimation of a vector controlled induction motor drive. Advances in Neural Networks, L N in Comp Sci. 2009; 5551: 852-862.

63 [Mog09] Moghbelli H, Rahideh A, Safavi AA. Using wavelet based neural networks for feedback signals estimation of a vector controlled induction motor drive. Advances in Neural Networks, L N in Comp Sci. 2009; 5551: 852–862.

64 [Moh12] Mohanty S. Ultra-fast design exploration of nanoscale circuits through metamodeling, Invited Talk, Semiconductor Research Corportation (SRC), Texas Analog Center for Excellence (TxACE), 27 April, 2012.

65 [Moh14] Mohanty S, Kougianos E. Polynomial metamodel based fast optimization of nano-CMOS oscillator circuits. Analog Integr Circ Sig Process. 2014; 79(3):437-453.

66 [Mon12] Montano J, Leon C, Garcia A, Lopez A, Monedero I, Personal E. Random generation of arbitrary waveforms for emulating three-phase systems. IEEE T Ind Electron. 2012; 59(11):4032–4040.

67 [Naj04] Najibi, T., Enabling analog-IP reuse: relating requirements to reality, D&R SOC News Alert, 2004; http://www.us.design-reuse.com/articles/article7677.html7

68 [Nas08] Nason GP. Wavelet Methods in Statistics with R. New York: Springer-Verlag; 2008.

69 [Neg09] Mircea Gh. Negoita, Sorin Hintea, Bio-Inspired Technologies for the Hardware of Adaptive Systems: Real-World Implementations and Applications, Studies in Computational Intelligence, Springer-Verlag Berlin Heidelberg, ISBN: 978-3-540-76994-1, 2009;

70 [Nic08] Nicosia, G., Rinaudo, S., Sciacca, E.: An Evolutionary Algorithm-based Approach to Robust Analog Circuit Design using Constrained Multi-objective Optimization, Knowledge-based Systems, Vol. 21, Iss. 3, 2008, pp. 175-183.

71 [Nig11] Nigam A, Tang Q, Zjajo A, Berkelaar M, van der Meijs N. Pseudo circuit model for representing uncertainty in waveforms. In: Lukasiewycz M, Chakraborty S, Milbredt P, editors. DATE 2011: Proceedings of Design, Automation & Test in Europe Conference & Exhibition; 2011 Mar 14–18; Grenoble, France.IEEE; 2011. p. 1–4.

72 [NNT] Neural Network Toolbox. The MathWorks, Inc. Available: http://www.mathworks.com/help/nnet/index.html. Accessed March 2016.

73 **[Olt02a] Oltean, G., Matematică nuanţată în analiza şi proiectarea unor circuite electronice, Teza de doctorat, Universitatea Tehnică din Cluj Napoca, 2002, 168pp.**

74 **[Olt02b] Oltean, G., Miron, C., Crasi, M., Carlugea, Mihaela, Evaluation of the Analog Circuits Performances using Fuzzy Models,Scientific Buletin of the Politechnica University of Timisoara, Romania, TRANSACTIONS on ELECTRONICS and COMMUNICATIONS, Timisoara, Sept. 2002, Vol I, ISSN 1224-6034, pag. 12-17;**

75 **[Olt02c] Oltean, G., Miron, C., Mocean, Emilia, Multiobjective Optimization Method for Analog Circuits Design based on Fuzzy Logic, Proccedings of the 9th IEEE International Conference on Electronics, Circuits and Systems - ICECS 2002, September 15-18, 2002, Dubrovnik, Croatia, ISBN 0-7803-7597-1, pag. 777-780**

76 **[Olt02d] Oltean, G., Miron, C., Analog Design: Multiobjective Optimization Method based on Fuzzy Logic, Acta Technica Napocensis. Electronics and Telecommunications, vol. , No. 1/2002, ISSN 1221-6542, pag. 21-26;**

77 **[Olt04a] Oltean, G., Multiobjective fuzzy optimization method, Scientific Buletin of the Politechnica University of Timisoara, Romania, TRANSACTIONS on ELECTRONICS and COMMUNICATIONS, Oct. 2004, Tom 49(63) Fascicola 1, ISSN 1224-6034, pag. 220-225;**

78 **[Olt04b] Oltean, G., Dispozitive si circuite electronice. Dispozitive electronice, Risoprint, Cluj-Napoca, ISBN 973-656-433-9, 316 pag., 2003, retiparita 2004**

79 **[Olt04c] Oltean, G., Sipos, Emilia, Oltean, Ioana, A new approach of op-amp amplifier biasing, Scientific Buletin of the Politechnica University of Timisoara, Romania, TRANSACTIONS on ELECTRONICS and COMMUNICATIONS, Timisoara, Oct. 2004, Tom 49(63) Fascicola 2, ISSN 1224-6034, pag. 328-331;**

80 **[Olt05a] Oltean, G., FADO - A CAD Tool for Analog Modules, Proc. of the International Conference on "Computer as a Tool" EUROCON2005,**

Belgrade, ISBN 1-4244-0050-3, IEEE catalog number: 05EX1255C, Serbia and Montenegro, 21-24 Nov., 2005, pp. 515-518.

81 **[Olt05b] Oltean, G., High Level Model for a Simple Transconductance Operational Amplifier, Proc. of the International Conference on "Computer as a Tool" EUROCON2005, Belgrade, ISBN 1-4244-0050-3, IEEE catalog number: 05EX1255C, Serbia and Montenegro, 21-24 Nov., 2005, pp. 1243-1246.**

82 **[Olt05c] Oltean, G., Sipos, Emilia, Csipkes, Doris, A Procedure to Model the Input-Output Behavior of Analog Modules, Procc. of the International Conference Interdisciplinarity in Engineering, Inter-Ing 2005, Tg. Mures, ISBN 973-7794-41-9, Romania, 10-11 Nov., 2005, pp. 703-708.**

83 **[Olt06a] Oltean, G., Sipos, Emilia, Simulink Implementation of Fuzzy Functional Models for Analog Modules, IEEE-TTTC International Conference on Automation, Quality and Testing, Robotics, AQTR 2006 (THETA 15), May 25 - 28, Cluj-Napoca, Romania, Tome I, ISBN 1-4244-0360-X, IEEE Catalog Number: 06EX1370, pp.90-95; DOI: 10.1109/AQTR.2006.254504**

84 **[Olt06b] Oltean, G., Sipos, Emilia, Simulink Models for Operational Transconductance Amplifiers Using Fuzzy Systems, Control Engineering and Applied Informatics Journal, Vol 8, No 4, 2006, ISSN 1454-8658, pp. 44-51;**

85 **[Olt06c] Oltean, G., Electronic Devices, Editura U.T. Pres, Cluj-Napoca, ISBN 973-662-220-7, 317 pag., 2006.**

86 **[Olt06d] Oltean, G., Oltean, Ioana, Hintea, S, DEIC: Web-Based Materials to Teach Digital Electronics, Scientific Buletin of the Politechnica University of Timisoara, Romania, TRANSACTIONS on ELECTRONICS and COMMUNICATIONS, Tom 51(65), Fascicola 1, ISSN 1583-3380, 2006, pp.69-73;**

87 **[Olt07a] Oltean, G, Şipoş, Emilia, Tehnici fuzzy in proiectarea si modelarea circuitelor analogice, Editura U.T. Pres, Cluj-Napoca, ISBN 978-973-662-302-8, 129 pag., 2007.**

88 **[Olt07b] Oltean, G., Circuite Electronice, UT Pres, Cluj-Napoca, ISBN 978-973-662-300-4, 203 pag., 2007.**

89 **[Olt08a] Oltean, G., Fuzzy Techniques in Analog Circuit Design, WSEAS Transactions on Circuits and Systems, Issue 5, Volume 7, May 2008, ISSN: 1109-2734. pp. 402-415.**

90 **[Olt08b] Oltean, G, Hintea, S., Sipos, Emilia, Computational Intelligence in Analog Circuits Design, Knowledge-Based Intelligent Information and Engineering Systems, Pt 3, Proceedings, Book Series: Lecture Notes in Artificial Intelligence, Volume: 5179, Pages: 384-391, KES2008, 3-5 September 2008, Zagreb, Croatia, Publisher: Springer-Verlag Berlin, Book Chapter, DOI: 10.1007/978-3-540-85567-5; ISSN: 0302-9743, ISBN: 978-3-540-85566-8**

91 **[Olt09a] Oltean, G, Hintea, S., Sipos, Emilia, Analog Circuit Design based on Computational Intelligence Techniques, Journal of Automation, Mobile Robotics & Intelligent Systems, ISSN 1897-8649, Vol. 3, No. 2, 2009, pp. 63-69;**

92 **[Olt09b] Oltean, G, Hintea, S., Sipos, Emilia, A Genetic Algorithm-based Multiobjective Optimization for Analog Circuit Design, 13rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, KES2009, 28-30 September 2009, Santiago, Chile, Pages: 507-514; Publisher: Springer-Verlag Berlin, LNAI 5712, ISBN 978-3-642-04591-2;**

93 **[Olt09b] Oltean, G., Gordan, Mihaela, Sipos, Emilia, Stãncel, E., Dancea, O., Fuzzy Logic Based Decision Making for Hydroelectric Energy Generation in a**

Cascaded Hydropowerplants, Acta Technica Napocensis. Electronics and Telecommunications, Nr. 1/2009, ISSN 1221-6542, Vol.50, pp. 42-48;

94 **[Olt11] Oltean, G., Miron, C., Sipos, Emilia, Ivanciu, Laura, A top-down approach to gaining an insight into transistor circuits in CS connection, International Journal of Electrical Engineering Education (IJEEE), ISSN: 0020-7209, Publisher: Manchester University Press, Vol. 48, No. 1, January 2011, pp. 10-16;**

95 **[Olt13] Oltean, G., Prodan, A., Rafaila, Monica, Ivanciu, Laura, Prediction of Waveforms under the Variation of Input Parameters using Neural Networks,Acta Technica Napocensis. Electronics and Telecommunications, Nr. 4/2013, ISSN 1221-6542, Vol.54, pp. 1-6;**

96 **[Olt15a] Oltean, G., Ivanciu, Laura-Nicoleta, Kirei, B.S., Signal approximation using GA guided wavelet decomposition, IEEE International Symposium on Signals, Circuits and Systems (ISSCS2015), 9-10 July, 2015, Iasi, Romania, DOI: 10.1109/ISSCS.2015.7203996, Print ISBN: 978-1-4673-7487-3, pp. 1- 4;**

97 **[Olt15b] Oltean, G., Grama, Lăcrimioara, Ivanciu, Laura, Rusu, C, Alarming Events Detection Based on Audio Signals Recognition, IEEE, 8th International Conference on Speech Technology and Human-Computer Dialogue (SpeD2015), 14-17 October 2015, Bucharest, Romania, DOI: 10.1109/SPED.2015.7343106, pp. 1- 8;**

98 **[Olt16] Oltean G, Ivanciu L-N, Computational Intelligence and Wavelet Transform Based Metamodel for Efficient Generation of Not-Yet Simulated Waveforms. PLoS ONE 11(1): e0146602., doi:10.1371/journal.pone.0146602, 2016.**

99 **[Olt16a] Oltean, G., Sisteme cu logica nuantata, http://www.bel.utcluj.ro/dce/didactic/sln/11_ModelareFuzzy.pdf, [cited March 2016]**

100 **[Olt16b] Oltean, G., Technici de inteligenta computationala in electronica, http://www.bel.utcluj.ro/dce/didactic/sln/11_ModelareFuzzy.pdf, [cited March 2016]**

101 [Oro113] Oros A, Kovacs I, Topa M, Buzo A, Rafaila M, Harrant M, et al. Data improvement in lab verification of smart power products using DoE,Proceedings of IEEE 19th International Symposium for Design and Technology in Electronic Packaging SITME,October 24–27; Galati, Romania. IEEE; 2013. p. 171–176.

102 [Per10] Pereira-Arroyo, R., Nicaragua-Guzman, F., Chacon-Rodriguez, A.: Design of an Operational Transconductance Amplifier Applying Multiobjective Optimization, In: Argentine School of Micro-Nanoelectronics Technology and Applications, Montevideo, 2010, pp. 12-17;

103 [Phe00] R. Phelps, M. Krasnicki, R. A. Rutenbar, R. L. Carley, J. R. Hellums, Anaconda: Simulation-Based Synthesis of Analog Circuits via Stochastic Pattern Search., IEEE Tran. on Computer-Aided Design of Integrated Circuits and Systems, Vol. 19, No.6., June 2000, pp. 703-717.

104 [Pra04] K. Prakobwaitayakit, Analog circuit optimizer based on computational intelligence techniques, 47th Midwest Symposium on Circuits and Systems, MWSCAS, Volume 3, 25-28 July 2004, pp. iii - 291- 4.

105 [Pri15] Principe J, Chen B. Universal Approximation with Convex Optimization: Gimmick or Reality? [Discussion Forum]. IEEE Comput IntelMag. 2015; 10(2):68–77.

106 [Pro13] Alex Prodan, Prediction of waveforms under the variation of input factors using neural networks, Lucrare de disertatie, Conducator ştiinţific: Prof.dr.ing. **Gabriel Oltean**, Facultatea de Electronică, Telecomunicaţii şi Tehnologia Informaţiei, UTCN, 2006

107 [Raf09] Rafaila M, Decker C, Grimm C, Pelz G. New methods for system-level verification using SystemC-AMS extensions: application to an automotive ECU. In: Gremzow C, Moser N, editors. 12 Workshop Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen, 2009. p. 217–226.

108 [Raf10] M. Rafaila, C. Decker, C. Grimm, G. Pelz, Simulation-based “Sensitivity and Worst-Case Analyses of Automotive Electronics”, IEEE International Symposium on Design and Diagnostics of Electronic Circuits and Systems, 2010.

109 [Ren07] X. Ren, T. Kazmierski, Performance Modeling and Optimization of RF Circuits Using Support Vector Machines, Proc. of 14th International Conference on Mixed Design of Integrated Circuits and Systems, MIXDES07, 2007, pp. 317-321.

110 [Rut07] R. A. Rutenbar, G. G. E. Gielen, J. Roychowdhury, Hierarchical Modeling, Optimization, and Synthesis for System-level Analog and RF Designs, Proc. of the IEEE, Vol. 95, Issue 3, March 2007, pp. 640-669.

111 [Rut08] Rutkowski L. Computational intelligence: methods and techniques. Berlin: Springer Science & Business Media; 2008.

112 [San06] Sansen, W.M.C.: Analog Design Esentials, Springer, 2006;

113 [San14] Sang Y, Liu C, Wang Z, Wen J, Shang L. Energy-based wavelet de-noising of hydrologic time series. PLoS ONE. 2014; 9(10):e110733. doi: 10.1371/journal.pone.0110733 PMID: 25360533

114 [Sed11] Sedra, A.S. Smith, K.C.: Microelectronic Circuits, Oxford University Press, International Sixth Edition, 2011;

115 [Sha10] Shanlin K, Huanzhen Z, Yuzhe K. Application of signal processing and neural network for transient waveform recognition in power system. In: Proceedings of the Chinese Control and Decision Conference, 2010 May 26–28; Xuzhou, China. IEEE; 2010. p. 2481–2484.

116 [She88] B. J. Sheu, A. H. Fung, Y-H. Lai, A Knowledge-Based Approach to Analog IC Design, IEEE Transactions on Circuits and Systems, vol. 35, no.2, February 1988, pp. 256-258.

117 [Shi03] Shivraman SG, Nilesh PN, Swarnalatha P, Sundaresan R, Amol R (Veermata Jijabai Technological Institute, University Of Mumbai, India). Project report on speech compression using wavelets; 2003.

118 [Shr98] Shragowitz,E., Lee,J-Y. ,Kang,E.Q Application of Fuzzy Logic in Computer-Aided VLSI Design, IEEE Trans. on Fuzzy Systems, vol.6. No 1. Febr. 1998, pp.163-172;

119 [Sim01] Simpson T, Poplinski J, Koch P, Allen J. Metamodels for computer-based engineering design: survey and recommendations. Eng Comput. 2001; 17(2):129–150.

120 **[Sip08a]** Sipos, Emilia, Festila, Lelia, **Oltean, G**, Towards Reconfigurable Circuits Based on Ternary Controlled Analog Multiplexers/Demultiplexers, Knowledge-Based Intelligent Information and Engineering Systems, Pt 3, Proceedings, Book Series: Lecture Notes in Artificial Intelligence, Volume: 5179, Pages: 351-359, KES2008, 3-5 September 2008, Zagreb, Croatia, Publisher: Springer-Verlag Berlin, Book Chapter, DOI: 10.1007/978-3-540-85567-5, ISSN: 0302-9743, ISBN: 978-3-540-85566-8.;

121 **[Sip08b]** Sipos, Emilia, **Oltean, G**., Miron, C., A Method to Design Ternary Multiplexers Controlled by Ternary Signals Based on SUS-LOC, IEEE International Conference on Automation, Quality and Testing, Robotics, AQTR 2008 (THETA 16), May 22 - 25, Cluj-Napoca, Romania, ISBN 978-973-713-248-2;

122 **[Sip09]** Sipos, Emilia, **Oltean, G.**, Miron, C., Simulink Library of Basic Building Blocks for Ternary Logic, Acta Technica Napocensis. Electronics and Telecommunications, Nr. 2/2009, ISSN 1221-6542, Vol.50, pp. 31-36;

123 [Som07] A. Somani, P. P. Chakrabarti, A. Patra, An Evolutionary Algorithm-Based Approach to Automated Design of Analog and RF Circuits Using Adaptive Normalized Cost Functions, IEEE Transactions on Evolutionary Computation, Vol. 11, no. 3, 2007, pp. 336-353.

124 [Sur113] Suresh HN, Balasubramanyam V. Wavelet transforms and neural network approach for epileptical EEG. In: Kalra BM, Garg D, Prasad R, Kumar S, editors. IACC 2013 Proceedings of the IEEE 3rd International Advance Computing Conference, 2013 Feb 22–23; Ghaziabad, India. IEEE; 2013. p. 12–17.

125 [Swi93] K. Swings, W. Sansen, ARIADNE: A Constraint-Based Approach to Computer-Aided Synthesis and Modeling of Analog Integrated Circuits, Analog Integrated Circuits and Signal Processing, Kluwer Academic Publishers, Boston, 1993, pp. 37-55.

126 [Tah03] M. Taherzadeh-Sani, R. Lotfi, H. Zare-Hoseini, O. Shoaei, “Design optimization of analog integrated circuits using simulation-based genetic algorithm” International Symposium on Signals, Circuits and Systems, SCS2003, 10-11 July, Volume 1, 2003, pages 73-76.

127 [Tak04] K. Takemura, T. Koide, H.J Mattausch, T. Tsuji, “Analog-circuit-component optimization with genetic algorithm”, The 47th Midwest Symposium on Circuits and Systems Volume 1, 25-28 July, 2004, pages I - 489-92.

128 [Tak09] Takhti, M., Beirami, A., Shamsi, H.: Multi-Objective Design Automation of The Folded-Cascode OP-AMP Using NSGA-II Strategy, In: International Symposium of Signals, Circuits and Systems, Iasi, 2009, pp. 1-4;

129 [Tei15] Teixeira Júnior L, Souza R, Menezes M, Cassiano K, Pessanha J, Souza R. Artificial neural network and wavelet decomposition in the forecast of global horizontal solar radiation. Pesquisa Operacional. 2015; 35(1):73-90.

130 [Tei15] Teixeira Júnior L, Souza R, Menezes M, Cassiano K, Pessanha J, Souza R. Artificial neural network and wavelet decomposition in the forecast of global horizontal solar radiation. Pesquisa Operacional. 2015; 35(1):73–90.

131 [Tha11] Thames JL, Abler R, Schaefer D. Parametric optimization of artificial neural networks for signal approximation applications. In: Clincy VA, Hoganson KE, Garrido J, Dasigi V, editors. Proceedings of 49th ACM Southeast Conference, 2011 Mar 24–26; Kennesaw, GA, USA.2011. p. 209–215.

132 [Tor96] A. Torralba, J. Chavez, , L. G. Franquelo, FASY: A Fuzzy-Logic Based Tool for Analog Syhesis, IEEE Trans. Computer Aided Design, vol.15, no7, July 1996, pp. 705-715.

133 [Tor96b] A. Torralba, J. Chavez, L.G. Franquelo, A collection of Fuzzy Logic-Based Tools for the Automated Design, Modelling and Test of Analog Circuits, IEEE Micro, vol 16, no 4, Aug 1996, pp 61-68.

134 [Ven99] N. D. Venkata, B. L Evans, An automated Framework for Multicriteria Optimization of Analog Filter Design, IEEE Tran. on Circuits and System-II: Analog and Digital Signal Processing, Vol. 46, No. 8, August 1999, pp. 981-990.

135 [WT] Wavelet Toolbox. The MathWorks, Inc. Available: http://www.mathworks.com/help/wavelet/index.html. Accessed March 2016.

136 [Yas07] Yaser, M.A.K., Badar, K., Faisal, T.: Multiobjective Optimization Tool for a Free Structure Analog Circuits Design Using Genetic Algorithms and Incorporating Parasitics, Proc. of the Conference Companion on Genetic and Evolutionary computation ,ISBN 978-1-59593-698-1, 2007, pp. 2527-2534.

137 [Ye13] Ye H, Deng G, Mauger S, Hersbach A, Dawson P, Heasman J. A wavelet-based noise reduction algorithm and its clinical evaluation in cochlear implants. PLoS ONE. 2013; 8(9):e75662. doi: 10.1371/ journal.pone.0075662 PMID: 24086605

138 [Zai08] Zainuddin Z, Pauline O. Function approximation using artificial neural networks. WSEAS Trans Math. 2008 June; 7: 333-338.

139 [Zeb98] R. S. Zebulum, M. A. Pacheco, M. M. Velasco, A Multiobjective Optimization Applied to the Synthesis of CMOS Operational Amplifier, Proc. of the Intl. Conf. on Microelectronics and Packaging, Brazil, Aug 1998.

140 [Zha05] S. Zhao, L. Jiao, J. Zhao, Y. Wang, Evolutionary design of analog circuits with a uniform-design based multi-objective adaptive genetic algorithm, Conference on Evolvable Hardware 2005 NASA/DoD, 29 June-1 July 2005, pp. 26 – 29.

141 [Zit04] Zitzler, E., Laumanns, M., Bleuler, S.: A Tutorial on Evolutionary Multiobjective Optimization, Proceedings of the Workshop on Multiple Objective Metaheuristics, Springer-Verlag, 2004, pp. 3-38.