

Performance optimization methods for switched-capacitor biquadratic filters

Jiri Nahlik^{*}, Jiri Hospodka^{*}, Pavel Sovka^{*},
Brendan Mullane^{**}, Ondrej Subrt^{*}

A class of computer-aided optimization methods based on Differential Evolution (DE), Particle Swarm Optimization (PSO) and Nelder Mead algorithms applied to a switched-capacitor (SC) filter circuit design are investigated. Comparisons of these algorithms applied to a 4th order biquadratic two-channel filter bank CMOS design on 0.35 μm technology are made. The frequency responses of the biquadratic filters must match ideal responses in a finite number of iterations with a limited number of "particles". The original and derived methods are evaluated on the base of their convergence progress and their reliability over different starting populations. An optimal design approach based on combining algorithms is derived as a more suitable and more reliable method for SC circuit optimization.

Keywords: optimization, DE, PSO, simplex, filter, SC circuit

1 State of the art

Although densely integrated digital circuit design on chip is prevalent today, analogue blocks are still needed. These analogue circuits include oscillators, filters, ADC and DAC converters etc. One of the most utilized analogue blocks are filters where there exists a number of analogue design implementation techniques. The Switched Capacitor technique (SC) is more popular compared to other design methods for on-chip filter implementation because of the advantage of simple tuning and accuracy of initial filter settings [1].

Although the switched capacitor circuits offer interesting advantages the non-ideality and parasitics of real circuit parts (*eg* switch resistance, DC gain of OpAmp, charge injections) have a direct influence on the ideal behaviour of the circuit (*eg* transfer characteristic in the case of filters). The non-ideality effects have been studied for different circuit topologies [2, 3], however, there are no universal analytical methods for suppressing the non-ideality effects because they are topology dependent.

As computing performance has increased, numerical optimization algorithms have become more useful for suppressing the non-ideality effects. The speed of different optimization algorithms is usually compared for the number of iterations (evaluations of optimized setup) needed to find global extreme of mathematical functions [4,5] since this number is not dependent on the performance of the used computer.

For circuit design, optimization time is additionally influenced by the analysis time component. Therefore the optimization algorithms are predominantly performed by focusing on specific parts of the analogue circuits (*eg* OpAmp design [6-8]), where analysis takes considerable time. Unlike to other analysis methods [9,10] the analysis of nonlinear discrete time analogue circuits must

be performed in time domain. Unfortunately, this approach is compute-intensive and thus time consuming. Therefore designers prefer to use circuits whose parameters are not so much dependent on capacitor value accuracy. For effective computer-aided optimization of such circuits, the designer needs to choose a robust algorithm that converges to the global optimum in the low number of analysis cycles.

Typically used heuristic optimizations algorithms are inspired by nature. Three of the most popular optimization methods are the Differential Evolution (DE) [11], Particle Swarm Optimization (PSO) [12] and Nelder Mead Simplex [13] algorithms. Although DE, PSO and NM method principles are easy to understand, these algorithms are quite robust and therefore seem to be a good choice for optimizing SC circuits.

This paper investigates and addresses the effectivity of different optimization algorithms (mainly DE and PSO based) applied to a SC filter circuit design that is a 4th order two-channel filter bank [14] in 0.35 μm CMOS technology. This design was the focus of our previous work [15]. The developed algorithm in this paper focuses on the speed and robustness of the optimization method applied to optimization of the frequency responses of the Fleischer-Laker biquadratic SC filters [1]. The optimization method was performed using the *Maple*TM program together with PraCAn package [16] and linked to a WinSpice simulator [17].

We present the SC circuit design used for these optimisation methods introduce the principle of the DE, PSO and Nelder Mead Simplex algorithms. Further details four algorithms derived from these base DE, PSO and NM methods are given and the metrics to compare the method performance are presented together with the performance results.

^{*} Department of Circuit Theory, Czech Technical University in Prague-FEE, Technická 2, 166 27 Prague, Czech republic, ^{**} Circuit and Systems Research Centre, University of Limerick, Limerick, Ireland, nahlijir(hospodka,sovka,subrto)@fel.cvut.cz, brendan.mullane@ul.ie

2 Filter transfer functions

The performance of the optimization method is compared on a design of biquads for a 4th order two-channel filter bank [15]. The 3rd order two-channel filter bank has been designed in [18], with implementation aspects discussed in [19]. As the filter order increases, the filter transition band becomes narrower and proper placing of filter poles and zeros becomes crucial for good functionality of the filter bank circuit. The designed filter bank consists of a 4th order low-pass filter H_{LP} and a 4th order high pass filter H_{HP} . The transfer functions were factorized into two biquadratic functions for cascade implementation. The poles and zeros implemented by one biquad were chosen as far from each other as possible in order to reduce the sensitivity of implemented function to component inaccuracy. The low-pass $H_{LP}(z)$ and high-pass $H_{HP}(z)$ transfer functions have been factorized into two functions of the form

$$H_{LP}(z) = H_{LP1}(z)H_{LP2}(z) = \quad (1)$$

$$= \frac{0.409z^2 + 0.336z + 0.409}{z^2 - 0.0122z + 0.092} \times \frac{0.409z^2 + 0.698z + 0.409}{z^2 - 0.0082z + 0.629},$$

$$H_{HP}(z) = H_{HP1}(z)H_{HP2}(z) = \quad (2)$$

$$\frac{0.415z^2 - 0.710z + 0.415}{z^2 - 0.0082z + 0.629} \times \frac{0.415z^2 - 0.341z + 0.415}{z^2 - 0.0122z + 0.092}.$$

3 Biquad

Since the proposed filter transfer functions contain all the powers of z , the Fleischer-Laker biquad topology [1] was chosen for the implementation. This topology is stray insensitive and employs only two OpAmps.

Although some circuit part properties effects on block functions can be predicted [3] such as finite OpAmp gain or bandwidth, other effects such as the nonlinear large signal behaviour also influence the resulting function and are topology dependent. Therefore simulations were performed with our designed folded cascode OpAmp [15,20]. Due to low offset OpAmp design ($V_{off} = 58 \mu V$) the capacitances of input differential pair is not negligible and affects the biquad transfer characteristic.

Due to the complexity of the used part models, the biquads had to be simulated by transient analysis. The required frequency responses were subsequently obtained by means of Discrete Fourier Transform (DFT) [21]. The simulation and DFT computations were powered by a WinSpice Program [17]. For SC biquad simulations transistor level of OpAmps model and behaviour model of switches were used. The switch model consisted of on-state resistance, off-state resistance and an overlapping capacitor which simulated charge injection. The analysis of the biquad frequency response is time consuming - one analysis takes 30 seconds on 3.3 GHz Quad-Core PC.

4 Optimization methods

Numerous optimization methods exist and each method is tuned to different optimization tasks. Therefore it is a very difficult task to choose the best method. The criteria for performance comparing of the optimization method is typically the number of algorithm iterations needed to find the global extreme of mathematical functions. There exist many of the different testing functions which usually have a lot of local extremes but only one global extreme [4]. Then the algorithm author has to choose suitable functions for his algorithm (*eg* the testing functions in [5] are very usual).

The differential evolution algorithm (DE), particle swarm optimization (PSO) algorithm, Nelder-Mead simplex algorithm and their combined performance are compared in order to find a universal robust method (and its settings) for optimizing SC filter circuits. The goal is to find an algorithm or a hybrid algorithm of the DE, PSO or NM methods which is more robust and resistant to getting locked in a local optimum or which features a lower number of simulations than these methods standalone.

4.1 DE algorithm

The DE algorithm is a "population based" algorithm where the new population of vectors is generated from the previous population by addition of the weighted difference of two different vectors to the third vector [11].

The objective function (OF), which covers all of the requirements to the optimized object, is subsequently evaluated for this new vector. If the value of the OF for the new vector is better than the OF value of the previous one, the new vector will replace the previous one in the new population. If the OF is worse, the old vector is only copied to the new population. Storn and Price [11] suggested two schemes for generating the new vectors. The 2nd scheme of DE is utilized in this work.

In that DE scheme, the weighted difference is added on the line between the best found vector by the whole population \mathbf{X}_{BG} and the current position of vector \mathbf{X}_G

$$\mathbf{X}_{G+1} = \mathbf{X}_G + \lambda(\mathbf{X}_{BG} - \mathbf{X}_G) + F(\mathbf{X}_{2G} - \mathbf{X}_{3G}), \quad (3)$$

where λ is a parameter which controls position on the line between \mathbf{X}_G and \mathbf{X}_{BG} .

The vectors \mathbf{X}_{2G} and \mathbf{X}_{3G} are randomly chosen vectors, and are different from \mathbf{X}_G received in (old) population G . The parameter F is chosen as a positive real constant (usually in range $\langle 0; 2 \rangle$).

4.2 PSO algorithm

The particle swarm optimization (PSO) algorithm is based on preserving agent history in memory and sharing of this information among agents [12]. Each agent is moving in the optimization space with various speed and direction during the optimization process. Each algorithm iteration the agent speed and direction of movement is changed on the base of best agent position, found in previous iterations, and the best found position of the whole

swarm (population). The movement of particles can be described by the formulae

$$\mathbf{X}_i[n] = \mathbf{X}_i[n-1] + \mathbf{v}_i[n] \quad (4)$$

$$\mathbf{v}_i[n] = \mathbf{v}_i[n-1] + C_1 r_1 (\mathbf{X}_{iB} - \mathbf{X}_i[n-1]) + C_2 r_2 (\mathbf{X}_{GB} - \mathbf{X}_i[n-1]), \quad (5)$$

where $\mathbf{X}_i[n]$ is the position of agent i in iteration n , vector $\mathbf{v}_i[n]$ is agents speed (each parameter means change of one particle parameters), \mathbf{X}_{iB} is the best position found by agent \mathbf{X}_i in previous iterations. \mathbf{X}_{GB} is the best position found by whole swarm, C_1 and C_2 are chosen constants, r_1 and r_2 are random numbers in range $(0; 1)$.

4.3 Nelder-Mead simplex algorithm

The Nelder-Mead algorithm (NM) is a local search algorithm based on a modification of simplex [13]. The NM algorithm modifies only one agent (exclude the last rare step). The NM algorithm for n -dimensional optimization task can be described as follows:

- 1) at first, the $n + 1$ agents are randomly generated and evaluated by the objective function
- 2) then, the "centroid" is calculated from n best agents \mathbf{X}_i by the formula $\mathbf{C} = \frac{1}{n} \sum_{i=1}^n \mathbf{X}_i$
- 3) the worst of the $n + 1$ agent is reflected (turned) over the centroid point with some multiplicative constant
- 4) on the base of reflected testing agent objective function the testing agent can be moved either further from the centroid or closer to the centroid or flipped back on the line between original worst agent and centroid
- 5) if no one of tried testing agent position OF is better than original worst agent OF . The n agents with worse OF than the best agent are shrunk to the best agent.

The steps 2-5 are repeated for each iteration of the algorithm until the point where the stop conditions are met, usually the optimization is done or the algorithm stuck.

5 Tested proposed hybrid algorithms

The SC circuit simulation takes a lot of time and for SC circuit optimization. It is desired to use as small a number of testing vectors (agents or particles) as possible. Unfortunately, that impacts the number of iterations needed to find the global optimum. A compromise between the number of testing vectors and simulation time of the whole population has to be chosen (45 agents in our case).

Using a small number of vectors in the DE algorithm leads to a lower number of possible combinations. Thus this algorithm has difficulty to find a better solution. Also with the low number of used vectors, the vectors are quickly contracted to the best one in a few iterations and the method needs to be restarted.

On the other hand, PSO is a very useful algorithm to find the global or approximate local optimum in fewer iterations. Unfortunately, as the particles best positions get close to the optimum, the speed of the optimization process slows down (the particles circle in a spiral around the optimum). Naturally, the speed depends on the optimization parameters settings (constants $C1$ and $C2$). To speed up the optimization, the DE and PSO algorithms were combined together into the following three variants, introduced in [22].

5.1 DE-PSO1 algorithm

The DE-PSO1 method is primarily based on the DE algorithm. The main idea is to move particles to other positions if they have not been adjusted by the DE method within a few algorithm iterations (in our case 3 iterations). The points are moved by the PSO algorithm as this algorithm guarantees movement within the limited area itself. For details see [22].

5.2 DE-PSO2 algorithm

The DE-PSO2 algorithm is a variant of the DE-PSO1 algorithm and is also based on DE. The algorithm also runs the PSO algorithm if no better solution is found by DE method within the last 3 iterations as for the DE-PSO1 algorithm.

The difference between DE-PSO1 and DE-PSO2 is how the results are used from the PSO algorithm. In this algorithm approach, the DE method combines only the best results of each particles remembered in the PSO part of the algorithm (whereas in the DE-PSO1 method, the DE algorithm always combines current particle positions no matter what method was obtained). This modification can help in the situation when PSO is activated more times in a row to find the better solution. The agent position for DE algorithm part is fixed and the agents are not pushed far from the possible solution by this combination.

5.3 PSO-DEm algorithm

The PSO-DEm main algorithm is PSO which is assisted by the DE algorithm. Firstly, an iteration of the PSO algorithm is performed for the whole population. Compared to the PSO method described above, the three last best results of each agent are kept in memory for further processing by the DE method (PSO algorithm still uses only the very last best position). Then Y mod X vectors are chosen, where Y is the number of vectors in the population and X is the number of vector parameters (eg capacitors values which need to be optimized in our case). The DE method is subsequently performed for these vectors corresponding to formula

$$\mathbf{X}_{G+1} = \mathbf{X}_G + \lambda (\mathbf{X}_{BG} - \mathbf{X}_G) + F (\mathbf{X}'_2 - \mathbf{X}'_3), \quad (6)$$

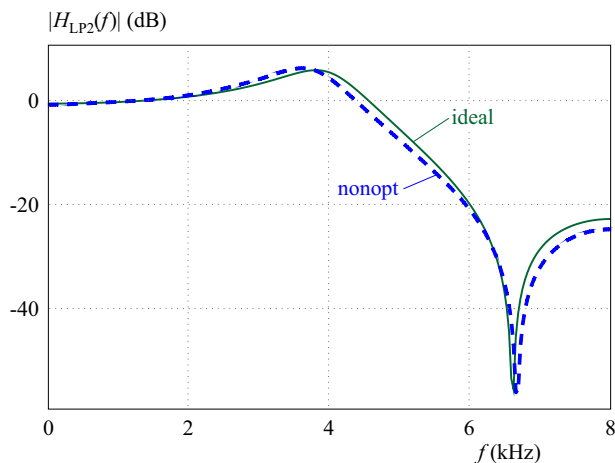
where \mathbf{X}'_2 and \mathbf{X}'_3 are randomly chosen best positions of agents \mathbf{X}_2 and \mathbf{X}_3 from their last 3 best positions. The meaning of the other symbols are similar to that

Table 1. Average number of simulation N and success rate SR of different optimization algorithms with applied limit of 100 iterations without change

	H_{LP1}		H_{LP2}		H_{HP1}		H_{HP2}	
	N (-)	SR (%)	N (-)	SR (%)	N (-)	SR (%)	N (-)	SR (%)
DE	1333	100	2172	67	1811	67	1292	100
DE-PSO1	6146	100	10299	100	5939	100	3976	100
DE-PSO2	1512	100	2725	100	2027	100	1417	100
DESX	6509	20	7770	0	5810	0	5872	50
PSO	38058	10	84942	0	47677	0	32127	0
PSO-DEm	48754	0	77855	0	55755	0	37140	0
NM simplex	3143	100	5371	60	3495	80	2370	100

Table 2. Average number of simulation N and success rate SR of different optimization algorithms without limit of 100 iterations without change

	H_{LP1}		H_{LP2}		H_{HP1}		H_{HP2}	
	N	SR (%)	N	SR (%)	N	SR (%)	N	SR (%)
PSO	43030	80	91614	20	56229	100	43375	100
PSO-DEm	58411	100	93500	10	71059	80	44178	100
NM-simplex	3143	100	9812	70	5006	100	2370	100

**Fig. 1.** Comparison of an ideal and non-optimized frequency response of low-pass biquad H_{LP2}

in formula (3). If the DE method finds a better agent position, this position is only written to the memory of best results (DE does not change current agent position). The algorithm principle can be seen in flowchart in [22].

5.4 DESX algorithm

This newly proposed hybrid algorithm [23] combines the NM Simplex algorithm and the DE algorithm. The main idea of algorithm DESX is to reduce computing time of NM Simplex method by parallelizing of simplex transformation for all agents and substitution of shrinking part of NM algorithm by DE approach.

The original NM method works with $n + 1$ agents, where n is the number of optimized variables. Due to using the DE method in DESX hybrid algorithm we recommend to use the number of agents m in population at least $5n$. Also, contrary to the NM algorithm the new agent positions are evaluated for the whole population, and the centroid is calculated by the different formula. The DESX algorithm can be described as follows:

- 1) at first, the agents are randomly generated and evaluated by the objective function
- 2) then the centroid is calculated from the all m agents \mathbf{X}_i by the formula

$$\mathbf{C} = \frac{\sum_{i=1}^m (1 + OF_{\max} - OF_i) \mathbf{X}_i}{\sum_{i=1}^m (1 + OF_{\max} - OF_i)},$$

where OF_i is evaluated objective function of agent X_i and OF_{\max} is evaluated objective function for the agent with worst OF in iteration.

- 3) all agents are reflected over the centroid on the same manner as in NM algorithm (reflection, expansion, contraction)
- 4) if the reflection, contraction or expansion of agent does not find a position with better OF than the original position of the agent had, the agent is moved to another position by adding the weighted difference of another two randomly chosen agents (DE part of the algorithm).
- 5) if the OF of any agent does not match to optimization stop conditions repeat point 2-5.

6 SC filter optimization

Each biquad frequency response should be as close to the ideal response as possible (especially in the cascade structure which is used).

Fig. 1 shows the difference between the ideal amplitude frequency response and real response of designed low-pass biquad H_{LP2} with designed values of capacitors. The ideal and real characteristics are different by approximately 0.7 dB on average (15 dB in the worst case). The goal of the optimization process is to find new capacitor values to minimize the difference between the ideal

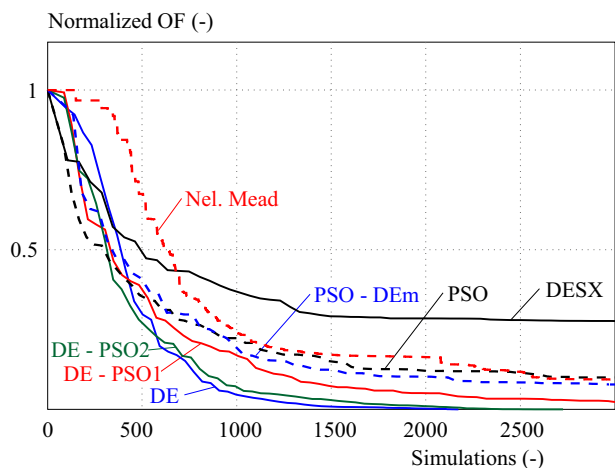


Fig. 2. Optimization convergence progress for H_{LP2}

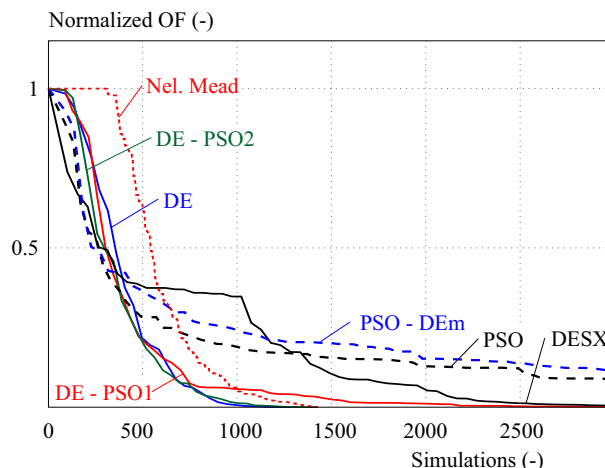


Fig. 3. Optimization convergence progress for H_{HP2}

and magnitude frequency response of real circuit with designed OpAmp and switch model. Two boundaries - B_H and B_L were defined for this purpose. The boundaries are created by adding (or subtracting) a small constant D to ideal magnitude frequency response (in our case D was set to 10 mdB which leads to a filter bank design error under 0.5% without considering capacitor mismatch). The optimized magnitude frequency response should be

$$|H_{Opt}(j\omega)| \in \langle B_L(j\omega), B_H(j\omega) \rangle,$$

where

$$B_H(j\omega) = |H_{Opt}(j\omega)| + D,$$

$$B_L(j\omega) = |H_{Opt}(j\omega)| - D.$$

The above relation should be fulfilled over the complete frequency range 0-8 kHz (especially near the zeros and poles). The required relation is checked by the Objective Function (OF) which expresses a distance (difference) of biquad magnitude frequency response from the gap defined by the boundaries. The distance is checked in 250 equidistant frequencies. The OF value is zero if the magnitude frequency responses in the all considered frequencies lie within the boundaries. The OF value *vs* number of simulations is used for comparison of different methods.

To compare the optimization method the number of iterations needed to optimize the circuit, reliability (robustness) of algorithm and progress of optimization during iterations (or time) is used. As the results may depend on the initial population, the results are averaged from optimizations with different initial populations and the optimization success rate is compared. For progress comparison, the first 3000 evaluations of SC filters (simulations) is used as it shows to be sufficient to optimize the SC biquad in the reasonable time (approximately 15 hours in our case).

As it has been written, the optimization algorithm progress is important, especially for SC circuit simulation, where the analysis takes a time. Therefore, the tested algorithms are stopped when no better solution is found for

100 iterations (only for NM simplex algorithm 400 iterations were used as it is not parallelized algorithm). This number of iterations equals about 6 hours of simulations in our test conditions.

7 Results

The optimization methods mentioned above in paragraphs 4 and 5, were tested on the real design of four SC biquadratic filters with different transfer functions according to paragraph 2 and 3. Each algorithm was also tested for at least 10 different initial populations to include/exclude dependency on initial population and results were averaged for each biquad. The objective function progress during the optimization was checked. If the objective function is not changed (improved) for 100 iterations the algorithm is stopped (only for NM simplex algorithm the limit of 400 iterations was used). The number of needed simulations for OF evaluation is used for method comparison. This number is further averaged and results as a parameter N is listed in Tab. 1 for each biquad transfer functions H_{XXY} . In addition, a parameter labelled SR is also listed in the Tab. 1. SR means a success rate and is defined by the ratio of successful optimization runs to all optimization runs (expressed as a percentage).

The DE algorithm seems to be the fastest tested algorithm which is able to optimize the SC biquad in the lowest number of used simulations (in average). Unfortunately, the DE algorithm success rate is only about 67% for H_{LP2} and H_{HP1} . Although the 1650 needed simulation (in average for all biquads) takes only 8 hours, the success rates below 70% can be critical in real optimization tasks, especially in comparison with optimization time.

On the other hand, the DE-PSO2 hybrid algorithm finished the optimization in comparable number of simulations (*eg* only 1.25 times more simulations for the worst case - biquad H_{LP2}) with the success rate of 100% for all biquads optimization test.

The results for DE-PSO1 method show also good ability to optimize designed biquad SC filters. On the other

hand, this method needs more iteration than its modified version DE-PSO2.

Other algorithm results show that the PSO and Nelder-Mead simplex based algorithms tend to optimize circuit quite slow and their update rate of the objective function is decreasing with the number of iteration (see their progress in Fig. 2 and Fig. 3). Therefore, their success rate is low. If the update condition of 100 iterations is not applied the success rate of PSO based method increase (see Tab. 2).

Equally important optimization method characteristic is the dependence of optimization convergence on time, which can be seen on objective function value progress. The convergence progress for the first 3000 of evaluation of objective function (simulations) for the H_{LP2} can be seen in Fig. 2. The convergence progress for H_{HP1} and H_{LP1} biquad looks similar to H_{LP2} , therefore they are not depicted. Figure 2 shows the convergence rapid slowing down for PSO and simplex based methods. The convergence progress of simplex based method DESX is better only for biquad H_{HP2} , see Fig. 3, which has also better success rate than other methods. During test the DESX algorithm optimization has shown a tendency of DESX to shrink agents together in the low number of algorithm iterations (dependent on driving constant settings). That probably leads to the low success rate of this method. Therefore, a proper setting of driving constants for this method should be inspected (different from NM simplex method which was used in this work).

The NM simplex algorithm also has shown its ability to optimize the SC biquad in the lower number of simulations than 10000 with sufficient success rate with exception of biquad H_{LP2} (especially if no change condition is not applied). Unfortunately, this method is hard to parallelize, thus the time the optimizing time is longer.

8 Conclusion

This paper presents a comparison of optimization method performance for SC filter optimization. For this purpose, the design of SC biquads for a 4th order two-channel filter bank was used. SC biquads were implemented by Fleischer-Laker biquads in 0.35 μm CMOS technology.

The comparison was performed for three popular optimization algorithms (Differential Evolution, Particle Swarm Optimization, and Nelder-Mead simplex algorithm) and their proposed combinations. The goal of the optimization method was to minimize the difference of magnitude frequency responses of the designed biquads from ideal responses (with a tolerance of 10 m dB).

Although the DE method has shown to be the fastest algorithm, its success rate for the different initial population was about 67% for H_{LP2} and H_{HP1} biquad. Such success rate together with the time-intensive simulations is not sufficient for design optimization with limited time. Conversely, the proposed DE-based hybrid method DE-PSO2 offer comparable results with a higher success rate

and only hundreds of simulations more than the original algorithm. Because of these reasons the use DE method stand-alone cannot be recommended for real SC filter optimization.

The PSO algorithm based methods showed their ability to scan optimization space roughly in a reduced time with a low number of agents, see comparable convergence speed in first iterations in Fig. 2 and Fig. 3. Unfortunately, the progress speed of optimization decreases with decreasing distance from the expected optimum and optimization time becomes up to 40 times longer than for DE method.

The Nelder Mead simplex method has also shown its good progress during the optimization process. Although the resulting number of needed simulations was only 2-3 times bigger than for DE method, required optimization time is longer as this algorithm is hard to parallelize.

The lowest optimization convergence showed proposed simplex based hybrid algorithm DESX, which is a parallelized simplex algorithm using DE approach as exploration part of the algorithm.

Therefore, we conclude that using DE-based hybrid algorithms with other methods (in our case PSO method) for SC filter optimization is recommended especially for a lower number of optimizing agents.

Future work involves analysis of performance dependency on driving constants of all algorithms, especially for DESX method.

Acknowledgement

The work has been supported by Internal CTU grant SGS17/183/OHK3/3T/13 Special Applications of Signal Processing. Maple is trademark of Waterloo Maple Inc. Maplesoft is a division of Waterloo Maple Inc. All other trademarks are property of their respective owners.

REFERENCES

- [1] P. V. A. Mohan, V. Ramachandran and M. N. S. Swamy, *Switched capacitor filters - theory, analysis and design*, Prentice Hall International, ISBN 0-13-879818-4, 1995.
- [2] B. J. Sheu and Hu Chenming, "Switch-induced error voltage on a switched capacitor," *Solid-State Circuits, IEEE Journal of*, vol.19, no.4, pp.519-525, Aug 1984 doi: 10.1109/JSSC.1984.1052176.
- [3] K. Martin and A. S. Sedra, "Effects of the op amp finite gain and bandwidth on the performance of switched-capacitor filters," *IEEE Transactions on Circuits and Systems*, vol.28, no.8, pp.822,829, Aug 1981 doi: 10.1109/TCS.1981.1085052.
- [4] M. Jamil and Xin-She Yang, "A literature survey of benchmark functions for global optimization problems," *Int. Journal of Mathematical Modelling and Numerical Optimisation*, Vol.4, No.2, pp.150-194, 2013, doi: 10.1504/IJMMNO.2013.055204 <http://arxiv.org/pdf/1308.4008.pdf>.
- [5] S. Yuhui and R.C. Eberhart, "Empirical study of particle swarm optimization", *CEC 99. Proceedings of the Congress on Evolutionary Computation*, vol.3, 1999, doi: 10.1109/CEC.1999.785511.
- [6] A. Jafari, S. Sadri and M. Zekri, "Design optimization of analog integrated circuits by using artificial neural networks", 2010

International Conference of Soft Computing and Pattern Recognition (SoCPaR), pp.385,388, 7-10 Dec.2010, doi: 10.1109/SOC-PAR.2010.5686736.

- [7] M. Barari, H. R. Karimi and F. Razaghian, "Analog Circuit Design Optimization Based on Evolutionary Algorithms", *Mathematical Problems in Engineering*, Article ID 593684, 12 pages, 2014. doi:10.1155/2014/593684.
- [8] T. Mallard, "An improved Nelder-Mead method for analog design optimisation applied to deep sub-micron technology", *25th IET Irish Signals Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CICT 2014)*, pp.164–168, 26-27 June 2013, doi: 10.1049/cp.2014. 0678.
- [9] C. F. Kurth and G. S. Moschytz, "Nodal analysis of switched-capacitor networks", *IEEE Transactions on Circuits and Systems*, Feb 1979, vol.26, no.2, pp.93,105, doi: 10.1109/TCS.1979. 1084613.
- [10] A. Opal and J. Vlach, "Analysis and sensitivity of periodically switched linear networks", *IEEE Transactions on Circuits and Systems*, Apr 1989, vol.36, no.4, pp.522,532, doi: 10.1109/31.92884.
- [11] R. Storn and K. Price, "Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces", *Technical report TR-95-012*, ICSI, 1995.
- [12] J. Kennedy and R. Eberhart, "Particle swarm optimization", *IEEE International Conference on Neural Networks*, pp.1942–1948, vol.4, Nov/Dec 1995 doi: 10.1109/ICNN.1995.488968.
- [13] J. A. Nelder and R. Mead, "A Simplex Method for Function Minimization", *Computer Journal*, Vol.7, No.4, pp.308–313,1965, doi:10.1093/comjnl/7.4.308.
- [14] P. P. Vaidyanathan, *Multirate Systems and Filter Bank*, Prentice-Hall, New Jersey, 1993.
- [15] J. Nahlik, B. Mullane, J. Hospodka, P. Sovka and D. O'Hare, "Optimized switched capacitor biquads for two-channel quadrature-mirror filter bank", *25th IET Irish Signals Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CICT 2014)*, pp.412–417, 26-27 June 2013, doi: 10.1049/cp.2014. 0724.
- [16] J. Bicak and J. Hospodka, "PrACAn-Maple Package for Symbolic Circuit Analysis", *Digital Technologies 2008*, EDIS Zilina University Publisher, Zilina, 2008, ISBN 978-80-8070-953-2.
- [17] M. Smith, WinSpice User's Manual [online] 2009 [cit.2012-10-06], URL: <http://www.winspice.com>, 2012.
- [18] J. Nahlik, J. Hospodka, P. Sovka and B. Psenicka, "Implementation of a two-channel maximally decimated filter bank using switched capacitor circuits", *Radioengineering*, 2013, vol.22, no.1, p.167-173. ISSN 1210-2512.
- [19] J. Nahlik, J. Hospodka, P. Sovka and B. Psenicka, "Switched capacitor circuit implementation of two-channel filter bank", *2013 International Conference on Applied Electronics (AE)*, Sept.2013, pp.1,4.
- [20] D. B. Ribner and M.A. Copeland, "Design techniques for cascaded CMOS op amps with improved PSRR and common-mode input range", *IEEE Journal of Solid-State Circuits*, Dec 1984, vol.19, no.6, pp.919,925, doi: 10.1109/JSSC.1984.1052246.
- [21] J. Bicak and J. Hospodka, "Frequency Response of Switched Circuits in SPICE", *ECCTD'03 Proceedings of the European Conference on Circuit Theory and Design 2003*, Monterey: IEEE Circuits and Systems Society, vol.3, pp.333–336, 2003, ISBN 83-88309-95-1.
- [22] J. Nahlik J, J. Hospodka, O. Šubrt and A. B. Chávez, "Optimization Methods for Switched Capacitor Circuits.", *International Conference on Applied Electronics. 2015*, Plzeň

pp.165–168, 8-9 September 2015, ISSN 1803-7232, ISBN 978-80-261-0385-1.

- [23] J. Nahlik, J. Hospodka and O. Šubrt, "SC Filter Optimization Performance by Hybrid Simplex Algorithm", *2018 15th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, Praha, pp.285-288, 2-5 July 2018, ISBN 978-1-5386-5152-0.

Received 5 August 2018

Jiří Náhlík was born in 1987. He graduated from the Czech Technical University in Prague in 2011. Since then, he has been pursuing his PhD study program at the Department of Circuit Theory at the same university. His research interests deal with the design of modern analog circuits, as well as their optimization.

Jiří Hospodka was born in 1967. He received his Masters and PhD degrees from the Czech Technical University in Prague in 1991 and 1995, respectively. Since 2007 he has been working as associate professor at the Department of Circuit Theory at the same university. Research interests: circuit theory, analog electronics, filter design, switched-capacitor, and switched current circuits.

Pavel Sovka received his Master and PhD degrees in electrical engineering at the Faculty of Electrical Engineering, the Czech Technical University in Prague, in 1981 and 1986, respectively. From 1985 to 1991 he worked in the Institute of Radioengineering and Electronics of the Czech Academy of Sciences, Prague. In 1991 he joined the Department of Circuit Theory, FEE CTU. He worked on the application of the adaptive systems to the noise and echo cancellation, speech analysis, changepoint detection and signal separation. At present, he is engaged in the research of biomedical signal processing and selective spectral transforms.

Brendan Mullane received his BEng degree in Electronic Engineering from the University of Limerick, Ireland in 1992 and also completed his PhD degree in System-on-Chip Data Converter Built-in-Self-Test in 2010 also from the University of Limerick. From 1992 to 1995 he worked with Alps Electric (Japan) as a Junior Design Engineer and with LSI Logic (Japan) from 1996 to 2002 as a Senior IC Designer. He is currently a Senior Research Fellow in the Circuits and Systems Research Centre (CSRC) at the University of Limerick, Ireland and has been the principal investigator on a number of significant research projects. He is author of one book chapter and more than 30 articles and holds four patents. His research interests include high performance, low-power VLSI circuits for signal processing, DSP/CPU and data converters (DAC, ADC) applications including digital assist analog IC calibration and Built-in-Self-Test (BIST) research activities.

Ondřej Šubrt was born in Hradec Králové on February 24, 1977. He works as Senior IC Designer engineer with ASI-Centrum Prague, a company of the Swatch Group. At present, he has also been appointed an Assoc Prof at the Faculty of Electrical Engineering, CTU Prague. His professional interests being integrated circuits design with emphasis to Non-volatile Memory design, compact models and Process Design Kit Development.