

Paper III

Design of coupled SAW filters using
genetic and evolution algorithms

J. Meltaus, P. Hämäläinen,
and V. P. Plessky

© 2006 J. Meltaus, P. Hämäläinen, V. P. Plessky.

Report TKK-F-A847
Submitted to IEEE Transactions on Evolutionary
Computation



Design of Coupled SAW Filters Using Genetic and Evolution Algorithms

J. Meltaus, P. Hämäläinen, and V. P. Plessky

Abstract— We study two stochastic optimization algorithms, a genetic algorithm and an evolution strategy, in the context of surface acoustic wave (SAW) filter design. The main reason for employing stochastic methods is to avoid converging to a local minimum of the objective function. As an example we apply these methods to the design of longitudinally coupled SAW filters. As a test structure, we use a symmetric filter structure with 14 simultaneously optimized parameters. We discuss the algorithms and the effect of the main parameters governing the optimization in each case, giving recommendations for parameter values when possible, and compare the performance of the algorithms when applied to the test structure. We demonstrate that the evolution strategy results in more efficient search of combination of variables than the corresponding genetic algorithm.

Index Terms— Optimization Methods, Genetic Algorithms, Evolutionary Optimization, Covariance Analysis, Surface Acoustic Wave Filters, Bandpass Filters.

I. INTRODUCTION

MODERN surface acoustic wave (SAW) devices set high demands on the optimization methods employed in the design process. For instance, in longitudinally coupled resonator filters (CRF), the number of the parameters to be optimized simultaneously can exceed twenty, and the objective function (deviation from the desired characteristics) can be multi-modal, i.e., contain several local minima. In gradient-based methods typically used in optimization, the final result can depend strongly on the initial guess of the parameters. Furthermore, the gradient method can converge to a local minimum, and often there is no way of determining whether the solution is the global minimum or not, or how good the quality of the final solution is compared to the global minimum. Performing several optimization runs with different starting points can help find a better solution, but this kind of manual iteration is often time-consuming. Stochastic optimization methods, such as evolutionary algorithms (EA), for example genetic algorithms (GA) and evolution strategies (ES), depend on randomness in the search process. The algorithms aim at searching over the whole parameter space, starting from a random situation and converging towards the

global minimum. Each iteration step is based on the solutions created in the previous step, but the method of proceeding from one step to the next is in some way arbitrary. The random element increases the probability of finding the global minimum. Even though there is no guarantee of converging to the global minimum, the final solution is not dependent on the initial guess of parameters. Therefore, the problem of converging to a low-quality local optimum near the starting point of the optimization process, often associated with gradient optimization methods, can be overcome. Since no gradients or Hessians need to be calculated, stochastic algorithms can be computationally less demanding than gradient methods. It also makes them well-suited for use with various other simulation and optimization methods, such as finite-element method (FEM) and combined SAW and package simulation. Moreover, they often outperform gradient methods if the objective function has a complicated shape, for example includes several close-lying local minima or points that are not differentiable. A gradient optimization step can be subsequently carried out for the best solution found by the stochastic algorithm to ensure that the solution is at the bottom of the found minimum (hybrid optimization).

A multitude of optimization algorithms have been applied to the design of SAW devices [1]. Previous work suggests that stochastic methods can be advantageous for different types of design problems encountered in SAW device design. Evolutionary algorithms have been successfully employed in designing SAW transducers [2, 3] and correlation devices [4]. In the context with coupled resonator filters and transversal filters, several types of biological and other evolutionary algorithms have been studied [5-10]. The results indicate that improvements in terms of, e.g., robustness and speed can be obtained as compared to conventional (gradient-based) methods.

In this paper, we study two stochastic algorithms: a genetic algorithm and a correlated evolution strategy. The operation of algorithms is illustrated with a simple test function. As the actual test structure, we use a longitudinally coupled multi-transducer SAW filter with 14 simultaneously optimized parameters. We discuss the governing parameters of optimization in each case, giving recommendations for their numeric values if possible. A comparison of the performance of the methods for the optimization of the test structure is carried out.

J. Meltaus is with Helsinki University of Technology, P.O.Box 3500, 02015 TKK, Finland (Johanna.meltaus@tkk.fi).

P. Hämäläinen is with Helsinki University of Technology, Finland.

V. P. Plessky is with Helsinki University of Technology, Finland, and with GVR Trade SA, Switzerland.



Fig. 1. Schematic of the test filter structure. The filter has 5 main transducers with short transducer sections replacing the gaps between them. The structure is symmetric such that it has 14 simultaneously optimized parameters as indicated in the figure.

II. TEST STRUCTURE

The structure studied in this paper is a multi-transducer longitudinally-coupled resonator filter (CRF), schematically shown in Fig. 1. It consists of five main interdigital transducers (IDT) and 8 short transducer sections between the main IDTs, replacing the gaps between the transducers. The structure is symmetric such that the total number of device parameters to be optimized is 14, as indicated in Fig. 1: aperture W , reflector finger number N_g and period p_g , gap between the reflector and the first transducer, and transducer finger numbers N_t and periods p_t . In principle, the numbers of fingers in the short transducers need not be constant.

As the objective function to be minimized, we use the weighed sum of the deviations of the filter response from the desired characteristics, such as magnitude response and standing wave ratio. In the simulation of the filter structure we employ the coupling-of-modes (COM) model (see, e.g., [11]) and the transfer matrix method (see, e.g., [12]). The response is very sensitive to certain parameters, for example to the periods of the short transducer sections (p_{t2} , p_{t3} , p_{t5} , and p_{t6} in Fig. 1). Fig. 2 illustrates the behavior of the objective function when two parameters are varied while other parameters are kept constant, showing the complicated nature of the objective function, in particular with respect to the periods (Fig. 2(a) and Fig. 2(d)). The dependence on aperture W and on the number of fingers N_t is less pronounced.

As the calculation of the frequency response of the filter involves cascading the filter elements by multiplying their transfer matrices, the computation of the objective function is rather a heavy task. With a 2.6 GHz Pentium 4 computer, one objective function evaluation with 901 frequency points takes approximately 2 seconds. Some of the computational advantage gained from the use stochastic algorithms is thus lost, as the major part of the computational time required by the procedure is taken by the function evaluation. On the other hand, since the optimization problem is complicated and the gradient methods easily converge to a local minimum, the use of a stochastic algorithm benefits the design process.

III. OPTIMIZATION METHODS

In this paper, we typically use the term “solution” for the studied set of optimized parameters, but in context with GAs, they are also termed “genes” or “individuals”. The simultaneously-existing group of solutions is called a “population” or a “generation” in this paper; these terms are widely used in genetic optimization. Several excellent

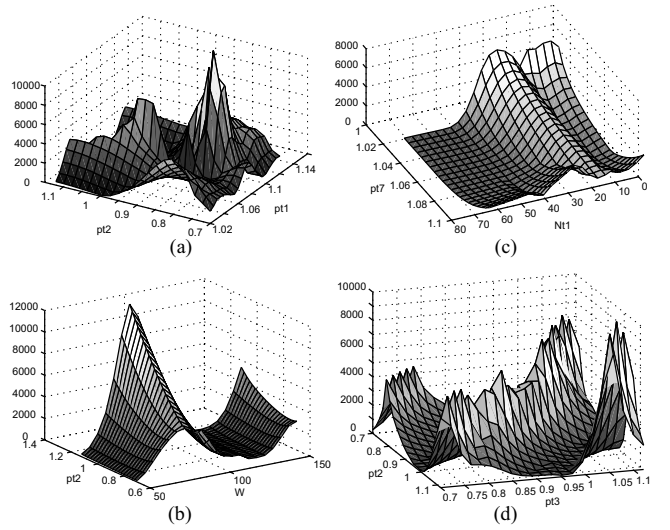


Fig. 2. Dependence of the objective function of the test structure on some of the device parameters. Two parameters are varied while others are kept constant. The figures illustrate the strong dependence of the objective function shape on certain combinations of parameters, for example pitches p_{t1} and p_{t2} of the first main transducer and the first short transducer (a). The shape of the objective function for this combination is very complicated with local minima and sharp corners.

textbooks dealing with evolutionary algorithms are available, e.g., [13-16]. For a review on evolutionary optimization methods, see, e.g., [17].

A. Genetic Algorithm

Genetic optimization strives for imitating the natural evolution process. From an initial pool of solutions, only some get to produce offspring for the next generation. While the best individuals have a large probability of getting selected for recombination, in principle every individual has a non-zero selection probability. Typically, a genetic algorithm comprises the steps of creating the population, selection, recombining genes, and mutation. Mutation is used to introduce random variation in the population. Different types of GAs have been previously studied in the design of SAW transducers [2] and filters [3,6,8].

The individuals in the first generation are often selected randomly and uniformly from the whole parameter space. There are numerous ways of selecting the individuals for recombination; in the ultimate case, only the two best individuals are selected. For recombining the genes, the binary crossover method that simulates the operation of biological DNA is often employed: the bit strings representing the parent individuals are cut at the same, random, crossover site, and the ends of the strings are switched, resulting in two offspring. This is essentially the same as choosing an offspring parameter randomly between the parent parameters (blend crossover), or in a range that exceeds beyond the parent parameters, depending on the binary representation. A modification of the blend crossover method is also used in the GA studied in this paper.

In the GA employed in this work, each optimized parameter

value is assigned limits a_{\max} and a_{\min} . The first generation of N individuals is created by randomly and uniformly selecting parameter values between these limits. The individuals (parameter vectors) are sorted according to their fitness F . If the objective function is to be minimized, the fitness can be defined, e.g., as the inverse of the objective function value. For each individual x_i , we then determine a recombination probability defined as

$$p(\mathbf{x}_i) = \frac{[F(\mathbf{x}_i)]^w}{\sum_{n=1}^N [F(\mathbf{x}_n)]^w}, \quad (1)$$

where w acts as a weighting parameter. Increasing w increases the relative recombination probabilities of the individuals with highest fitness values. In recombination, each offspring parameter y_k is selected randomly between the limits

$$y_{kMax} = \frac{1}{2} [(x_{ik} + x_{jk}) + S|x_{ik} - x_{jk}|], \quad (2)$$

$$y_{kMin} = \frac{1}{2} [(x_{ik} + x_{jk}) - S|x_{ik} - x_{jk}|]$$

where x_{ik} and x_{jk} denote the corresponding parameters of the parents and S is a stretching coefficient. When $S=1$, the offspring parameter is selected between the parent parameters. Every parameter of each offspring is mutated with a probability M . If a parameter is mutated, a new value is randomly selected from the range $[a_{\min}, a_{\max}]$. The number of offspring is governed by recombination rate p ; all in all, $n=p*N$ offspring are produced. Then the fitness values of the new individuals are calculated, and all the $n+N$ individuals are sorted according to their fitness. The N best individuals comprise the next generation. The parameters appearing in the algorithm are the generation size N , the probability scaling exponential w , the number of the offspring n , the stretch coefficient S , and the mutation probability M .

B. Evolution Strategy

Evolution strategies consider the optimized parameters as traits affected by a number of genes. Such traits, e.g., weight and height, have often been found to be normally distributed in biological populations. The notation used in the context of ESs is $(\mu/\rho, \lambda)$ - or $(\mu/\rho+\lambda)$ -selection, where μ is the total number of parents, λ is the number of offspring, and ρ is the number of parents that generate one offspring. If the new generation is selected from the offspring only, the notation is “,”; if the parents are retained in the selection, it is “+” [15]. New individuals can be sampled around a number of parents having the best fitness, or the whole population can be used as a distribution. Evolution strategies in SAW design have been studied, e.g., in [4].

In this paper, we use a modified $(\mu/\mu, \lambda)$ -selection, that is, the whole population is used to create offspring. The method is similar to the GA, but instead of producing offspring by recombination, we directly compute the mean and covariance of the offspring and then sample the offspring from the corresponding multivariate normal distribution

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)}, \quad (3)$$

where x is the vector of optimized parameters, μ is the mean of the distribution, and Σ is the covariance matrix. Covariance analysis is used in correlated evolutionary optimization, see, e.g., [18].

Consider that we have generated a random population of parameter vectors x_i . We first need to obtain the expectation of the offspring parameters y_k as a function of the parent parameters x_{ik} and x_{jk} , that is

$$\mu_k = E[y_k] = \int f(y_k) y_k dy_k, \quad (4)$$

where $f(y_k)$ is the probability density function of y_k . We start with a simple case of genetic optimization where y_k is uniformly distributed between x_{ik} and x_{jk} , that is,

$$y_k(c_k, x_{ik}, x_{jk}) = c_k x_{ik} + (1 - c_k) x_{jk}, \quad (5)$$

where c_k is uniformly distributed in the range $0 \dots 1$. The expectation now becomes

$$\mu_k = \sum_{i=1}^N \sum_{j=1}^N \int_{-\infty}^{\infty} f(c_k, x_{ik}, x_{jk}) [c_k x_{ik} + (1 - c_k) x_{jk}] dc_k, \quad (6)$$

where $f(c_k, x_{ik}, x_{jk})$ is the joint probability density function c_k , x_{ik} and x_{jk} . If we allow self-recombination so that the parents are selected independent of each other, we can use $f(c_k, x_{ik}, x_{jk}) = f(c_k)p(x_{ik})p(x_{jk})$, since the probability of selecting parameter x_{ik} for recombination equals the probability of selecting the corresponding parent vector x_i . Substituting Eq.(1) and rearranging summation order, we obtain the expected offspring as a fitness-weighted mean of the parents:

$$\mu_k = \frac{\sum_{i=1}^N [F(\mathbf{x}_i)]^w x_{ik}}{\sum_{i=1}^N [F(\mathbf{x}_i)]^w}, \quad (7)$$

where F is the fitness and w is a weighing parameter that can be selected freely.

The element σ_{kl} of the covariance matrix of the parameters y_k and y_l is given by

$$\sigma_{kl} = \int \int (y_k - \mu_k)(y_l - \mu_l) f(y_k, y_l) dy_k dy_l, \quad (8)$$

where $f(y_k, y_l)$ is the joint probability density function of y_k and y_l . In the recombination, each offspring parameter depends only on the corresponding parent parameters and not on each other. Thus, we can use $f(y_k, y_l) = f(y_k)f(y_l)$. Substituting Eqs. (1) and (5) and rearranging the summation order, we obtain the covariances of the offspring parameters as fitness-weighted covariances of the parent parameters:

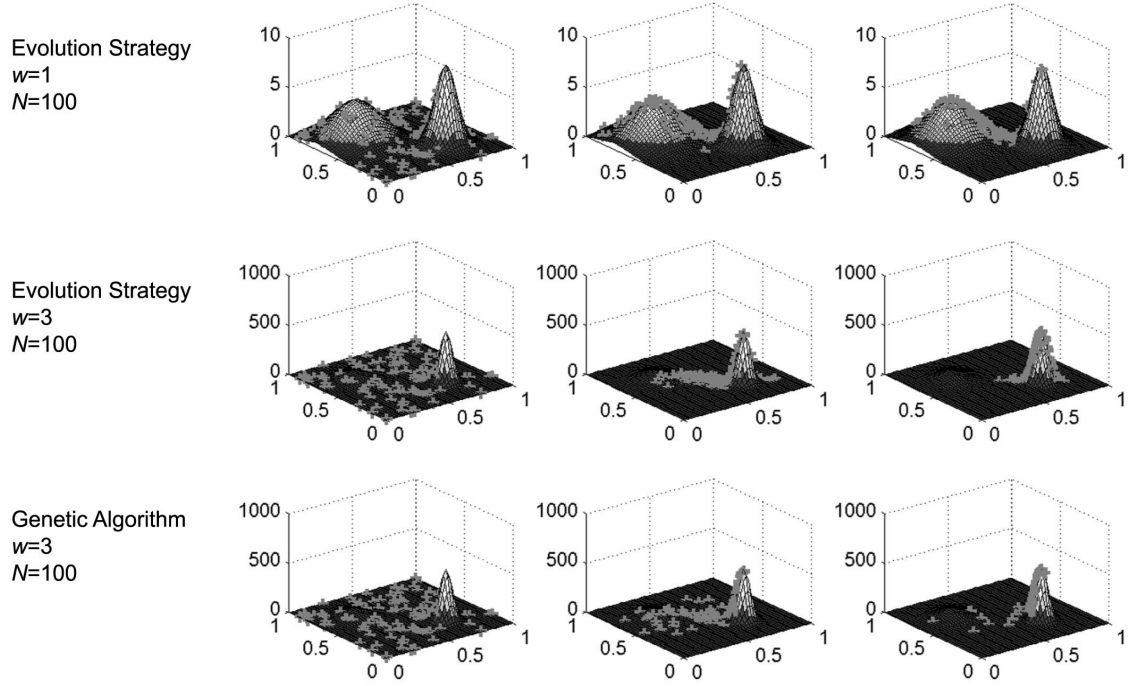


Fig. 3. The first three generations of evolution strategy (ES, top and middle images) and genetic algorithm (GA, bottom image) for a two-dimensional test function. Fitness function weighing parameter $w=1$ for the top images and $w=3$ for the middle and bottom images, pronouncing the highest peak in the fitness function surface. The ES produces samples that are concentrated on the axis between the two peaks, whereas the GA yields more scattered solutions.

$$\sigma_{kl} = \frac{S_n \sum_{i=1}^N [F(\mathbf{x}_i)]^w (x_{ik} - \mu_k)(x_{il} - \mu_l)}{\sum_{i=1}^N [F(\mathbf{x}_i)]^w}, \quad (9)$$

where $S_n=0.5$ by default, but it can also be changed to adjust convergence, similar to the stretching coefficient S in genetic optimization. In effect, the offspring statistics equal parent statistics with fitness as an additional weighing term.

For each generation, we first compute the mean and covariance and then sample the offspring from the corresponding normal distribution. The size of the first generation should be large enough to allow sufficient sampling of the search space, but in the following generations, the number of solutions can be reduced in order to accelerate the computation.

Using a normal distribution as a model may seem unfounded when the fitness function has several local maxima. However, when w is increased, individuals with high fitness are weighed more in the summation and μ approaches the individual with the best fitness. The highest peak of the effective fitness function $[F(x)]_e$ becomes more pronounced and the function better resembles the Gaussian model with only one peak. If the generation size N also approaches the number of possible individuals (depending on computation precision), μ approaches the global maximum of the fitness function.

Fig. 3 compares the convergence of the ES and the GA in a

case of a simple two-dimensional test function with two peaks. Results of the ES with two weighing parameter values, $w=1$ and $w=3$, are plotted in top and middle figure. The bottom figure is for GA with $w=3$. The two topmost figures show that the ES concentrates the search on the axis between the two maxima, whereas the GA yields more scattered solutions. Moreover, it is clearly seen in the middle figure that the ES approaches the final solution from both sides of the peak, whereas the GA can only obtain solutions that are for the most part situated between the two maxima – this is because in recombination, the offspring parameter values are selected from between the values of the two parents. In Fig. 4, the second generation of the ES for $w=1$ is shown. The black lines show the principal axes of variance of the normal distribution from which the solutions are sampled.

Statistically, the ES algorithm is not equal to the GA it was derived from. A normal distribution is fully defined by its mean and variance, but the offspring created by actual recombination have a more complex distribution with higher-order statistics. However, as can be seen in Fig. 3, the simpler normal distribution can be more optimal. The GA produces more scattered offspring, whereas the normally distributed samples are better focused on areas of high fitness. The mean of the normal distribution is far from optimum at first, but then gravitates toward the most significant fitness function peak.

An important characteristic of the ES studied here is that the significance of fitness function peaks can be adjusted with the

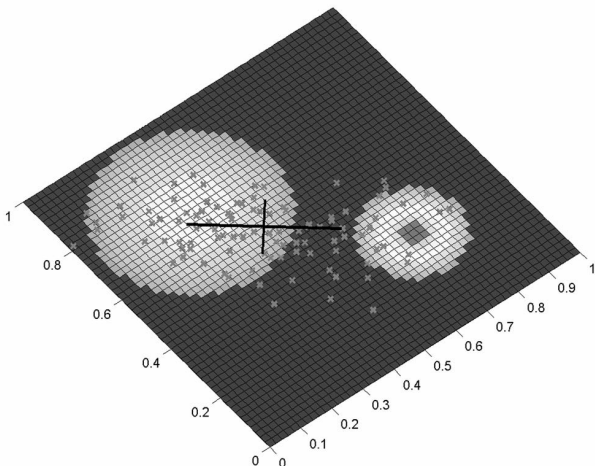


Fig. 4: Example of evolution strategy with weighing parameter $w=1$, second generation. The objective function is the same as in Fig. 3. The black lines indicate the principal axes of variance of the normal distribution. The principal search direction is oriented between the two peaks of the fitness function.

parameter w . This feature can be used to obtain more robust solutions. The top and middle images in Fig. 3 compare cases of $w=1$ and $w=3$ for the ES. For $w=1$, the broader peak area gets weighed more, but for $w=3$, the global but narrower maximum is found. Choosing a large w increases the probability of finding the global optimum. On the other hand, for practical devices it can be advantageous to use a shallower but broader local maximum, as the performance of the device is then less dependent on the fabrication tolerances. It should also be noted that if the search space is insufficiently sampled by the first generation, heavy weighing can accelerate the convergence on the highest peak found by the first generation, even if it is not the global extremum.

For multi-modal objective functions that are typical to SAW design problems, the concept of niching in order to find multiple peaks would be advantageous.

In practice, samples from a multivariate normal distribution can be generated via Cholesky decomposition of the covariance matrix [19]. The Cholesky decomposition yields a lower-triangular matrix A for which $AA^T = \Sigma$. If y' is a vector of univariate normal draws, the corresponding multivariate normal offspring vector can be obtained as $y = \mu + Ay'$. The computational complexity of Cholesky decomposition is $O(m^3)$, where m is the number of optimized parameters. In our case, this is negligible - computing the Cholesky takes less time than one fitness function evaluation.

IV. SIMULATION RESULTS

The convergence of an optimization algorithm is governed by the free parameters appearing in the algorithm, such as the size of one generation in the GA. This makes the comparison of different algorithms difficult, as the outcome for one algorithm may strongly depend on the selection of parameters. Moreover, the nature of the optimization problem affects the optimum choice of parameters.

TABLE I
LIMITS OF THE OPTIMIZED FILTER PARAMETERS USED IN THE OPTIMIZATION RUNS

Parameter	Min	Max
Aperture (W , μm)	25	150
Number of fingers in reflectors (N_g)	31	101
Reflector period (p_g , μm)	1.07	1.15
Number of fingers in IDTs (N_{t1} , N_{t4} , N_{t7} , N_{t10} , N_{t13})	5	71
IDT period ($pt1$, $pt4$, $pt7$, $pt10$, $pt13$, μm)	1.02	1.12
Short IDT period ($pt2$, $pt3$, $pt5$, $pt6$, $pt8$, $pt9$, $pt11$, $pt12$, μm)	0.7	1.12
Gaps (relative to adjacent pitches: $\text{Gap}(\mu\text{m}) = \text{Gap}(\text{rel}) * (\text{pt1} + \text{pt2}) / 2$)	0	2

In this work, we have used the same test filter structure with both algorithms to first determine the best combination of optimization parameters for each method. To study how the convergence depends on the parameters in the algorithm, we employed different parameter values in the optimization of the filter structure shown in Fig. 1, focusing on the parameters that have the most significant effect on the performance of the algorithm [20]. Only one parameter was changed while others were kept constant. We then used these parameters to compare the performance of the algorithms.

Although the optimal parameter values are found for one test structure (5-IDT SAW filter), the results should be valid for other types of CRF structure as well.

Selecting the feasible range of the optimized device parameters appropriately is important for the performance of the algorithms, as it determines the size of the parameter space and the number of minima in the objective function. The proper range is problem-specific and depends also on the structural limitations imposed on the final device, such as the maximum size of the structure. The parameter ranges used in this work are shown in Table I.

A. Parameters in Genetic Algorithm

The parameters appearing in the GA are generation size N , recombination rate p , probability scaling parameter w , stretch parameter S and mutation probability M . Of these parameters, we varied N , p , and w , keeping S and M constant. The constant parameter values were $N=1000$, $p=20\%$, $w=2.5$, $S=3$, and $M=0.15\%$. A more detailed study is presented in [20].

Generation Size. We studied convergence for $N=20$, $N=50$, $N=100$, $N=500$, and $N=1000$. As expected, a larger population yields a better solution. For the multi-IDT CRF studied here, increasing the generation size even above $N=1000$ improved the final result. However, a very large generation size renders the run time impractically long.

Recombination probability distribution. Parameter w is used to weigh the fitness of the individuals in order to increase the recombination probability of the best individuals as defined in Eq. (1). A large w increases the recombination probability of the best individuals in the population. However, increasing w over the optimum value decreases the quality of the final

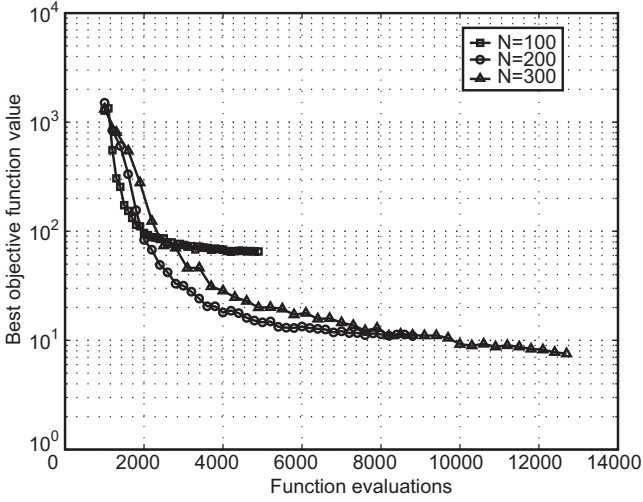


Fig. 5. Convergence of evolution strategy for population size $N=100$, $N=200$, and $N=300$, optimizing the test filter structure. Curves are averages of 5 runs. Growing the population over the optimum value increases the computational cost of the procedure, in effect slowing the convergence.

solution, as the probability of converging to a local minimum increases. The best value for the power w , at least for the problem studied in this paper, is between 2 and 3.

Number of recombinations per generation. In every generation, we produce $p*N$ offspring. We studied values $p=10\%$, $p=20\%$, and $p=30\%$. Increasing p improves the final solution, but the effect slows down when $p>20\%$, as the tendency to find a local minimum increases. This is because the best individuals found in the beginning of the optimization tend to be emphasized when a large number of recombinations is made, and their properties soon become prominent in the population. A large p also adds to the computation time required by the routine.

B. Parameters in Evolution Strategy

The parameters in the ES are initial generation size N_0 , generation size N , fitness weighing parameter w and covariance coefficient S_n . We ran the optimization with $N=100$, 200, and 300 and $w=1$, 2, and 3. The constant parameters were $N_0=1000$, $w=2$, $N=200$, and $S_n=1$.

Weighing Parameter. The weighing parameter w is used as defined in Eqs. (7) and (9). The results show that weighing the fitness from $w=1$ to $w=2$ increases considerably the rate of convergence. However, increasing the weighing constant from $w=2$ to $w=3$ deteriorates the final solution, as the probability of converging to an inferior solution during the first steps of the optimization process increases. For this particular problem, we determined $w=2$ to be the optimum value for the weighing parameter.

Generation Size. The optimal generation size appeared to be $N=200$ for this particular problem. While $N=100$ converged fast, it did not reach as good a final value, and $N=300$ reached a value comparable with $N=200$ but converged a bit more slowly in the beginning (Fig. 5).

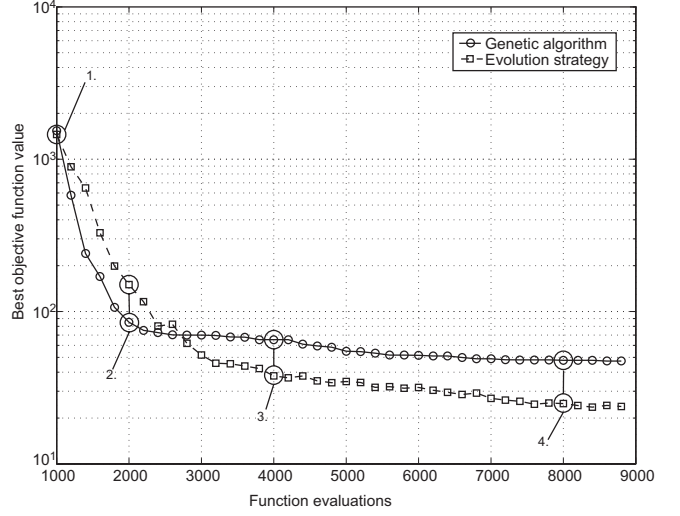


Fig. 6. Convergence of the studied optimization algorithms when used to optimize the test filter structure. Curves are averages of 5 runs. The best objective function value of a generation is plotted as a function of the target function evaluations performed. Frequency responses for typical runs at the points indicated by 1-4 are shown in Fig. 7.

C. Comparison of Optimization Methods

The performance of the studied methods depends strongly on the choice of parameters in each algorithm, and care must be exerted when comparing the results. To gain some insight on the relative performance of the methods, however, we made five independent runs with each algorithm and studied the average convergence of the best solution found in each step. The parameters used for the runs are listed in Table II.

Fig. 6 shows the values of the best solutions of the above-described optimization methods as the average of 5 runs. To facilitate the comparison of the computation time, the convergence is plotted as a function of the number of objective function evaluations, which require the major part of the computation time. The average of the best values of the population was calculated for every generation.

From Fig. 6, it can be concluded that GA and ES give comparable performance. The ES tends to find a better solution, although the GA converges faster in the beginning. However, both algorithms show rapid convergence in the beginning, and the improvement in the objective function value is small after 10-15 generations.

TABLE II
PARAMETERS USED FOR COMPARISON OF OPTIMIZATION ALGORITHMS:
GENETIC ALGORITHM (GA) AND EVOLUTION STRATEGY (ES)

Parameter	GA	ES
Initial population N_0	1000	1000
Recombination rate p	20%	
Population size N		200
Weighing parameter w	2.5	2
Stretch parameter S	3	1
Mutation probability M	0.15%	

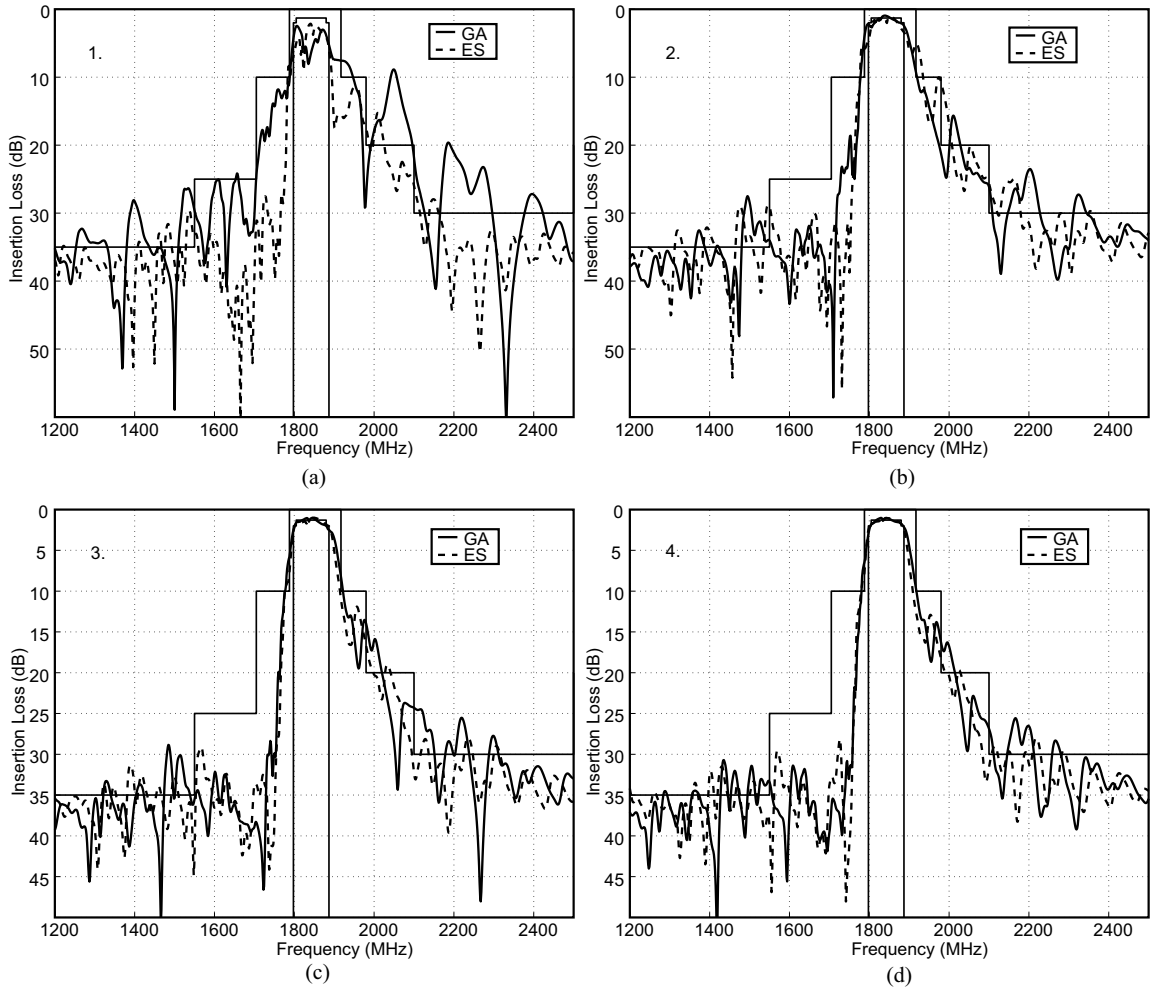


Fig. 7. Simulated frequency responses of the test filter structure of the best solutions for a typical run of the genetic algorithm (GA) and evolution strategy (ES) in four stages of optimization: (a) after the first generation, (b) after 6 generations (2000 objective function evaluations), (c) after 16 generations (4000 objective function evaluations), (d) after 36 generations (8000 objective function evaluations). The specification shown in the figures is the one used in the optimization process.

Fig. 7 shows frequency responses of a typical run (one that converges near the average value shown in Fig. 6) of the GA and the ES. Responses are plotted in four stages of the optimization, indicated by 1-4 in Fig. 6. The specification shown in Fig. 7 was used in the optimization process. As Fig. 7(d) shows, already after 6 generations, the response resembles the final result. The final responses are close to each other, but the ES arrives at a solution that satisfies the specified suppressions better.

V. DISCUSSION

The design problem of multi-transducer SAW filter is one that can benefit from the use of stochastic evolutionary optimization methods. The objective function shape is complicated and depends more strongly on some parameter combinations than on others, and typically contains several local minima. Our study shows that, in principle, both genetic optimization and the correlated $(\mu/\mu, \lambda)$ -evolution strategy give good results with this problem, and a satisfactory response is achieved already after a reasonable number of

function evaluations.

This type of problem may benefit more from the ES, as the covariance analysis focuses the search to the direction of those variable combinations that are significant for the target function. However, the advantage over the GA may be in other optimization problems than SAW design. If the evaluation of objective function is not computationally heavy, computing the Cholesky decomposition of the covariance matrix can dominate the overall computational cost.

For a multi-modal objective function, the weakness of the ES is the unimodal distribution used in the offspring sampling. To find multiple solutions, a niching strategy should be incorporated in the optimization. In SAW design, multi-modal search can be used to find robust solutions. In practice it may not be convenient to include sensitivity analysis in the population-based ES, as it would increase the number of function evaluations. Therefore, it is preferable to obtain multiple solutions and perform the sensitivity analysis afterwards.

Stochastic optimization methods are easy to distribute to

several machines as the calculation of the solutions does not depend on the previous solutions. Ultimately, one computer can calculate the objective function value of one individual solution and pass that value to the central computer.

It should be noted here that the comparison of optimization methods is always strongly dependent on the choice of the parameters appearing in each algorithm, as well as on the features of the design problem they are applied to. Therefore, the results should be viewed as an illustration of the possibilities and limitations of each method, rather than a decisive order of performance.

ACKNOWLEDGMENT

J. Meltaus thanks the Finnish Cultural Foundation for supporting her research. The authors thank Prof. Kalyanmoy Deb of Indian Institute of Technology, Prof. Jarmo Alander of University of Vaasa, and Prof. Harri Ehtamo of Helsinki University of Technology for valuable discussions.

REFERENCES

- [1] C. C. W. Ruppel, A. A. Sachs, and F. J. Seifert, "A review of optimization algorithms for the design of SAW transducers", in Proc. of the 1991 IEEE Ultrasonics Symposium, pp. 73-83.
- [2] J. Döring and H.-J. Fröhlich, "SAW transducer optimization by the evolutionary computation", in Proc. of the 1994 IEEE Ultrasonics Symposium, pp. 237-240.
- [3] J. Franz, C. C. W. Ruppel, F. Seifert, and R. Weigel, "Hybrid optimization techniques for the design of SAW-filters", in Proc. of the 1997 IEEE Ultrasonics Symposium, pp. 33-36.
- [4] H. Hofmann, H. Stab, and S. Komarov, "Design of SAW correlation devices with evolutionary strategy methods (EVCOR)", in Proc. of the 1997 IEEE Ultrasonics Symposium, pp. 171-174.
- [5] K. Tagawa, M. Masuoka, and M. Tsukamoto, "Robust optimum design of SAW filters with the Taguchi method and a memetic algorithm", in Proc. of the 2005 IEEE Congress on Evolutionary Computation, pp. 2146-2153.
- [6] K. Tagawa, T. Yamamoto, T. Igaki, and S. Seki, "An Imanishian Genetic Algorithm for the Optimum Design of Surface Acoustic Wave Filter", in Proc. of the 2003 Congress on Evolutionary Computation, v. 4, pp. 2748-2755.
- [7] S. Goto and T. Kawakatsu, "Optimization of the SAW filter design by immune algorithm", in Proc. of the 2004 IEEE International Ultrasonics, Ferroelectrics and Frequency Control Joint 50th Anniversary Conference, pp. 600-603.
- [8] V. Prabhu, B. S. Panwar, and Priyanka, "Linkage Learning Genetic Algorithm for the Design of Withdrawal Weighted SAW Filter", in Proc. of the 2002 IEEE Ultrasonics Symposium, pp. 357-360.
- [9] K. Tagawa, K. Togunaka, H. Haneda, T. Igaki, and S. Seki, "Optimal design of three-IDT type SAW filter using local search", in Proc. of the 2002 IEEE 28th Annual Conference of the Industrial Electronics Society, pp. 2572-2577.
- [10] K. Tagawa, T. Ohtani, T. Igaki, and S. Seki, "Robust optimization design of SAW filters by using penalty function method", in Proc. of the 2004 IEEE International Conference on Industrial Technology, pp. 751-756.
- [11] V. P. Plessky and J. Koskela, "Coupling-of-modes analysis of SAW devices", *Int. J. High Speed El. and Syst.*, vol. 10, no. 4, pp. 867-947, Dec. 2000.
- [12] K.-Y. Hashimoto, *Surface Acoustic Wave Devices in Telecommunications: Modelling and Simulation*, Springer-Verlag, Berlin, 2000.
- [13] E. K. P. Chong and S. H. Zak, *An Introduction to Optimization*, J. Wiley & Sons, New York, 1996.
- [14] D. Quagliarella, J. Périaux, C. Poloni, and G. Winter (editors), *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science: Recent Advances and Industrial Applications*, J. Wiley & Sons, Chichester, 1998.
- [15] H.-G. Beyer, *The Theory of Evolution Strategies*, Springer-Verlag, Berlin, 2001.
- [16] H.-P. Schwefel, *Evolution and Optimum Seeking*, J. Wiley & Sons, Chichester, 1995.
- [17] D. B. Fogel, "An introduction to simulated evolutionary optimization", *IEEE Trans. Neural Networks* 5, no. 1, pp. 3-14, Jan 1994.
- [18] N. Hansen and A. Ostermeier, "Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation", in Proc. 1996 IEEE Conference on Evolutionary Computation, pp. 312-317.
- [19] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, "Bayesian Data Analysis", 2nd ed., Chapman & Hall/CRC (2004)
- [20] J. Meltaus, P. Hämmäläinen, V. P. Plessky, and M. M. Salomaa, "Genetic optimization algorithms in the design of coupled SAW filters", in Proc. of the 2004 IEEE Ultrasonics Symposium, pp. 1901-1904.