Computer Assisted Mechanics and Engineering Sciences, **17**: 41–50, 2010.

Multiobjective evolutionary optimization of MEMS structures

Adam Długosz
Department of Strength of Materials and Computational Mechanics
Silesian University of Technology
ul. Konarskiego 18A, 44-100 Gliwice, Poland
e-mail: adam.dlugosz@polsl.pl

(Received in the final form November 12, 2010)

The paper is devoted to the shape optimization of piezoelectric and electro-thermo-mechanical devices by the use of multiobjective evolutionary algorithm. In this paper, special implementation of multiobjective evolutionary algorithm is applied (MOOPTIM). Several test problems are solved in order to test efficiency of the algorithm. The results are compared with the Non-Dominated Sorting Genetic Algorithm (NSGA-II). The objective function values are calculated for each chromosome in every generation by solving a boundary value problem for the piezoelectricity and electro-thermal-mechanical analysis. In order to solve the boundary value problems, the finite element method is used. Different functionals based on the results derived from coupled field analyses are formulated. The aim of the multiobjective problem is to determine the specific dimensions of the optimized structures. Numerical examples for multiobjective shape optimization are enclosed.

Keywords: multiobjective optimization, evolutionary algorithm, piezoelectricity, electro-thermo-mechanical analysis, coupled problems, finite element method, MEMS

1. Introduction

The aim of this work is an optimal design of the micro-electro-mechanical systems (MEMS). For such systems, necessity of taking into consideration interaction between different physical fields leads to the formulation of many special criteria. Generally, optimization should take into account electrical, thermal and mechanical properties of the system [7]. Important information for designers are also related to the costs or volume of the structures. Taking into account the foregoing aspects, the application of multicriteria optimization is indispensable. Performing single objective optimization with respect to one arbitrarily chosen criterion while other criteria are treated as constraints, or using weighted sum method, can be inefficient and inadequate.

Evolutionary Algorithms (EAs) become a powerful optimization tools for many different engineering disciplines [11]. Many features of EAs, such as: great probability of finding global optimum, no need of using gradient of fitness function, easy implementation, flexible operators, etc., make them very popular comparing to other optimization techniques. EAs seem to be an excellent tool for finding a set of Pareto optimal solutions, because they works on a population of individuals. In comparison with the other techniques, EAs give a proper balance between two conflicting aspects needed in successful optimization: exploitation near the likely optimum and the exploration of the search space. For the single-objective optimization problem, these two aspects are mainly considered, whereas for the multi-objective optimization, well spread of Pareto optimal solutions is also essential.

Two models of MEMS structures are considered in the present work. It requires solving a boundary value problem for the piezoelectricity and electrical-thermal-mechanical analysis. Finite Element

Method (FEM) [1, 13] is used to obtain solutions for above mentioned problems. Functionals are calculated on the basis of a nodal results for electrical, thermal and mechanical quantities.

To obtain Pareto optimal solutions an own implementation of the multiobjective evolutionary algorithm (MOOPTIM) is used. It allows designers to easily make a choice from the set of non-dominated solutions according to their preferences and system requirements. In order to check effectiveness of MOOPTIM, the algorithm was examined on several test functions. The results are compared with those obtained by using NSGAII. Solutions of shape optimization of MEMS devices are also compared for both algorithms.

2. Multiobjective optimization

2.1. Description of the multiobjective optimization problem

The process of finding a vector of decision variables that satisfies some restrictions and optimizes the vector of functionals is called multiobjective optimization (MOO). A MOO problem is formulated as follows

$$\text{find:} \quad \mathbf{x} = [x_1, x_2, \ldots, x_k]^{\mathrm{T}},$$

$$\text{minimize or maximize:} \quad f(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_n(\mathbf{x})]^{\mathrm{T}},$$

$$\begin{aligned} \text{subject to:} \quad & g_i(\mathbf{x}) \geqslant 0, \qquad i = 1, 2, \ldots, m, \\ & h_i(\mathbf{x}) = 0, \qquad i = 1, 2, \ldots, p, \\ & x_i^L \leqslant x_i \leqslant x_i^R, \end{aligned}$$

where k is the number of design variables, n is the number of objective functions, m is the number of inequality constraints, p is the number of equality constraints, x_i^L and x_i^R are minimum and maximum acceptable values for the variable x_i.

Multi-objective optimization deals with multiple conflicting objectives and usually, the optimal solution for one of the objectives is not necessarily the optimum for any of the other objectives. For such a case, instead of one optimal solution, many solutions are incomparable and optimal. These solutions are called Pareto-optimal ones.

2.2. MultiObjective OPTIMization tool – MOOPTIM

A requirement for EAs in multi-objective optimization can be generally formulated as follows: emphasize non-dominated solutions for progressing towards the Pareto-optimal front, emphasize the less crowded solutions for maintaining a good diversity among the obtained solutions and emphasize elites to provide a faster convergence near the Pareto-optimal front.

Good MOO algorithm should guarantee the above conditions without specifying any additional parameters, except those typical for EA (population size, number of generations, probabilities of evolutionary operators, etc.). Among many different types of multiobjective genetic and evolutionary algorithms *Strength Pareto Evolutionary Algorithm* and *Non-Dominated Sorting Genetic Algorithm* are the most popular and fulfil the above-mentioned expectations. Consecutive versions of such algorithms SPEA2 and NSGAII have many practical applications in different engineering disciplines.

In this work, our own implementation of the MultiObjective OPTIMization tool based on evolutionary algorithm (MOOPTIM) is used for optimization. Some specific methods implemented in NSGAII are applied in MOOPTIM.

2.2.1. Nondomination clasification

For each individual, the objective functions values are calculated. The classification of the individuals is performed by using nondominated sorting procedure [3–5]. The population is sorted on the basis of nondomination of the individuals. Each chromosome is assigned a rank equal to its nondomination level (solutions of the first front represent the best level, solutions of the second front represent the next best level, and so on).

2.2.2. Diversity preservation

In order to preserve diversity in the population, a crowding coefficient, proposed in [5], is calculated for each solution. Sorting procedure according to each objective function value is performed for the solutions on all fronts. The boundary solutions for each objective function have assigned infinite distance. This ensures survival of the extreme solutions. All other solutions are assigned a distance value equal to the absolute normalized difference between the function values of two neighboring solutions. The total value of the coefficient is the sum of all distances in n-dimensional objective space.

2.2.3. Core of the algorithm

The algorithm consists of two parts: an initialization and a main loop. Figure 1 shows the flowchart of the multiobjective evolutionary algorithm MOOPTIM. In the initialization step, besides determining all settings of the algorithm, populations Q_i and P_i are generated and the fitness functions are evaluated for population Q_i. In the main loop, after evaluation of the fitness functions for P_i and checking the stop conditions, populations Q_i and P_i are combined. Selection is performed on the set R_i, which is two times bigger than P_i. The individuals from the population R_i are put to P_{i+1} on the basis of nondomination level (*membership to the certain front*). If the solutions belong to the same front, solutions with a bigger value of crowding coefficient are selected to P_{i+1}. Individuals from P_{i+1} are copied to Q_{i+1} and then evolutionary operators change the population P_{i+1}. Two types of mutation are used: *uniform* and *Gaussian*, and two types of crossover: *simple* and *arithmetic*. It can

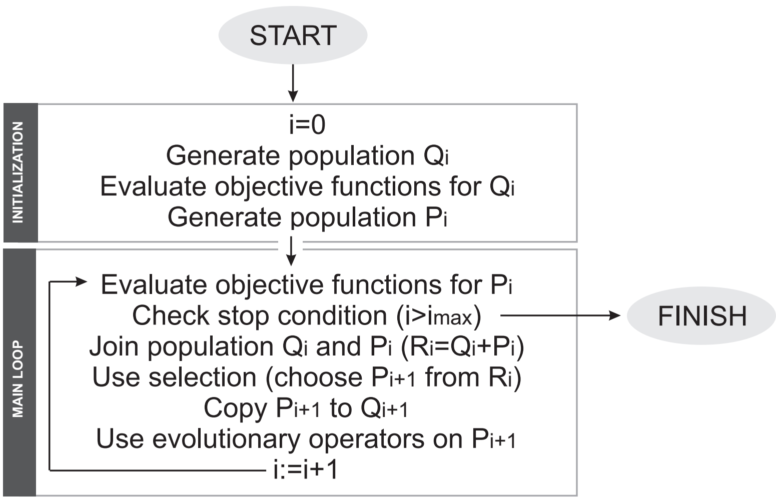

Fig. 1. The flowchart of the multiobjective evolutionary algorithm *MOOPTIM*

be noticed that both algorithms (MOOPTIM and NSGAII) have common features. Compared to the NSGAII, the proposed implementation has more evolutionary operators. It should be emphasized that Gaussian mutation has significant influence on the effectives of searching by the algorithm.

It was observed in previous investigations, in which the Gaussian mutation was applied to many tests and practical problems for single-objective optimization [2]. As a consequence of applying this operator, besides probability, the range of the Gaussian mutation have to be specified. The other difference between these algorithms is related to the formation of population P_{i+1}. There is no binary tournament selection operator in MOOPTIM. As mentioned before, solutions are selected only on the basis of nondomination level and crowding coefficient.

2.3. MOOPTIM – NSGAII comparison

MOOPTIM was used to solve the following benchmark problems: SCH, ZDT3, ZDT4 and ZDT6 [4, 15]. In real-world engineering optimization tasks, the solutions of the boundary-value problem may sometimes fail, e.g. finite element mesh can not be created or may be inappropriate due to the wrong values of design variables. Incorrect solutions should not be considered and should be eliminated from the population. In such cases the death penalty method is applied which leads to constrained optimization. For this reason examination of the efficiency of MOOPTIM for a constrained test problem is very important. Three test problems, namely CONSTR, SRN, TNK, are considered [5]. Due to the similarity of MOOPTIM to NSGAII, comparison is performed only for these two algorithms.

As mentioned before, for the multiobjective optimization, diversity in this set is essential besides the convergence to the Pareto-optimal set. It is impossible to measure these two tasks only by one performance metric. In the present paper, the comparison was performed by using the following indexes, proposed in [5]: *convergence metric* – measures the distance to the known set of Pareto-optimal solutions, *diversity metric* – measures span and spread of founded set of solutions. The values proceed to zero show better performance for both metrics.

2.3.1. Unconstrained optimization

Four different functions are used for test bechmark problem. All functions are minimized. The number of variables, the variables bounds, the objective functions and the optimal solutions for the unconstrained optimization are collected in Table 1. All tests are run with the population size

Table 1. Unconstrained test problems

Problem	k	Variable bounds	Objective functions	Optimal solutions
SCH	1	$\left[-10^3, 10^3\right]$	$f_1(x) = x^2$	$x \in [0, 2]$
			$f_2(x) = (x-2)^2$	
ZDT3	30	$x_i \in [0, 1]$	$f_1(x) = x_1$	$x_1 \in [0, 1]$
		$i = 1, \ldots, k$	$f_2(x) = g(x)\left(1 - \sqrt{\frac{x_1}{g(x)}} - \frac{x_1}{g(x)}\sin(10\pi x_1)\right)$	$x_i = 0$
			$g(x) = 1 + 9\left(\sum\limits_{i=2}^{k} x_i\right) / (k-1)$	$i = 2, \ldots, k$
ZDT4	10	$x_1 \in [0, 1]$	$f_1(x) = x_1$	$x_1 \in [0, 1]$
		$x_i \in [-5, 5]$	$f_2(x) = g(x)\left(1 - \sqrt{x_1/g(x)}\right)$	$x_i = 0$
		$i = 2, \ldots, k$	$g(x) = 1 + 10(k-1)\sum\limits_{i=2}^{k}\left(x_i^2 - 10\cos(4\pi x_i)\right)$	$i = 2, \ldots, k$
ZDT6	10	$x_i \in [0, 1]$	$f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$	$x_1 \in [0, 1]$
		$i = 1, \ldots, k$	$f_2(x) = g(x)\left(1 - (f_1(x)/g(x))^2\right)$	$x_i = 0$
			$g(x) = 1 + 9\left(\left(\sum\limits_{i=2}^{k} x_i\right) / (k-1)\right)^{0.25}$	$i = 2, \ldots, k$

100 and for 250 iterations for both algorithms. For NSGAII crossover probability is set to 0.9, mutation probability 0.1 as suggested in the papers [4, 5]. The probabilities of arithmetic crossover, simple crossover and uniform mutation were set to 0.1 for MOOPTIM. The probability of Gaussian mutation is 0.7 and range of Gaussian mutation is 0.2. Thirty runs of the algorithms for each benchmark optimization problem are performed. The average values and variance of the convergence and diversity metric are collected in Table 2. Results for SCH test problem are comparable, for ZDT3

Table 2. Results for unconstrained optimization

Convergence metric	SCH	ZDT3	ZDT4	ZDT6
MOOPTIM (avg)	0.008165	0.051932	**0.014609**	**0.0030099**
MOOPTIM (var)	3.7e-07	0.000444	**6.0e-05**	**4.4e-08**
NSGAII (avg)	**0.007925**	**0.0060867**	0.757388	0.077317
NSGAII (var)	**1.7e-07**	**1.5e-05**	0.398811	5.5e-05
Diversity metric	SCH	ZDT3	ZDT4	ZDT6
MOOPTIM (avg)	**0.470163**	0.881946	**0.375466**	0.720222
MOOPTIM (var)	**0.001602**	0.001697	**0.003964**	0.001618
NSGAII (avg)	0.491478	**0.626831**	0.788826	**0.592125**
NSGAII (var)	0.002238	**0.001671**	0.025975	**0.000827**

problem NSGAII reaches better convergence and diversity metric. MOOPTIM outperforms NSGAII for ZDT4 and ZDT6 test problems, although NSGAII achieves slightly better diversity metric for ZDT6.

2.3.2. Constrained optimization

For constrained optimization three test problems are chosen: CONSTR, SRN and TNK. All functions are minimized. The number of variables, the variables bounds, objective functions and constraints are collected in Table 3. All tests are run with the population size 100 and for 200 iterations.

Table 3. Constrained test problems

Problem	k	Variable bounds	Objective functions	Constraints
CONSTR	2	$x_1 \in [0.1, 2.0]$	$f_1(x) = x_1$	$g_1(x) = x_2 + 9x_1 \geqslant 6$
		$x_2 \in [0, 5]$	$f_2(x) = (1 + x_2)/x_1$	$g_2(x) = -x_2 + 9x_1 \geqslant 1$
SRN	2	$x_i \in [-20, 20]$	$f_1(x) = (x_1 - 2)^2 + (x_2 - 1)^2 + 2$	$g_1(x) = x_1^2 + x_2^2 \leqslant 225$
		$i = 1, 2$	$f_2(x) = 9x_1 - (x_2 - 1)^2$	$g_2(x) = -x_1 - 3x_2 \leqslant -10$
TNK	2	$x_i \in [0, \pi]$	$f_1(x) = x_1$	$g_1(x) = -x_1^2 - x_2^2 + 1 + 0.1\cos(16\arctan(x_1/x_2)) \leqslant 0$
		$i = 1, 2$	$f_2(x) = x_2$	$g_2(x) = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leqslant 0.5$

Probabilities of the operators for both algorithms are identical as in unconstrained optimization. Figure 2 shows Pareto-optimal solutions obtained by the use MOOPTIM and NSGAII for tests CONSTR and SRN. The obtained results are also comparable for the TNK test problem.

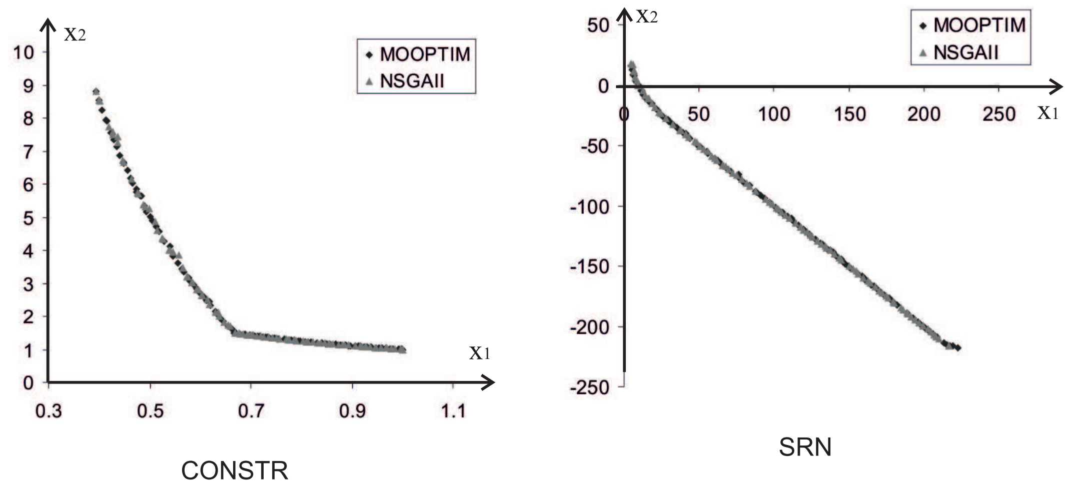

Fig. 2. MOOPTIM-NSGAII comparison for the constrained test problems

3. Multiobjective optimization of MEMS structures

3.1. Boundary-value problem

MOOPTIM is applied to the shape optimization of MEMS structures by the minimization or maximization of an appropriate functions. In the present work two types of boundary value problem are considered:

- electrical-thermal-mechanical analysis,
- piezoelectricity.

Both problems are described by the appropriate partial differential equations. These equations with arbitrary geometries and boundary conditions are usually solved by numerical methods. Finite element method is used to solve both boundary-value problems in the paper.

Coupled electrical-thermal-mechanical analysis (Joule-mechanical) combines electrical-thermal analysis (joule heating) with thermo-mechanical problem. The problem is weakly coupled, which requires solving electrical, thermal and mechanical analysis separately [10]. Coupling is carried out by transferring loads between the considered analysis and by using staggered procedures. Matrix equations of electrical-thermal-mechanical problem can be expressed as follows:

$$\mathbf{K_E V} = \mathbf{I}, \tag{1}$$
$$\mathbf{K_T T} = \mathbf{Q} + \mathbf{Q_E}, \tag{2}$$
$$\mathbf{K_M u} = \mathbf{F} + \mathbf{F_T}, \tag{3}$$

where $\mathbf{K_E}$ is the electrical conductivity matrix, $\mathbf{K_T}$ is the thermal conductivity matrix, $\mathbf{K_M}$ is the stiffness matrix, $\mathbf{Q_E}$ is the heat generation vector due to current flow, $\mathbf{F_T}$ is the force due to thermal strain vector, $\mathbf{V}$, $\mathbf{T}$, $\mathbf{u}$, are the nodal vector of voltage, temperature, displacements, respectively, $\mathbf{I}$, $\mathbf{Q}$, $\mathbf{F}$, are the nodal vector of current, heat fluxes and applied forces, respectively.

The electrical, thermal and mechanical boundary conditions for the equations (1), (2) and (3) take the form:

$$\Gamma_\phi : \phi = \bar{\phi}, \qquad \Gamma_\omega : \omega = \bar{\omega}, \tag{4}$$
$$\Gamma_T : T = \bar{T}, \qquad \Gamma_q : q = \bar{q}, \qquad \Gamma_c \colon q = \alpha(T - T^\infty), \tag{5}$$
$$\Gamma_t : t = \bar{t}, \qquad \Gamma_u : u = \bar{u}, \tag{6}$$

where ϕ denotes the electric potential, ω is the charge flux density, T, q, α, T^{∞} denote the temperatures, the heat fluxes, the heat transfer coefficients and the ambient temperatures respectively, t denotes the tractions and u denotes the displacements.

Γ_{ϕ}, Γ_{ω}, Γ_T, Γ_q, Γ_c, Γ_t and Γ_u denote parts of the boundary where the potentials, the charge flux densities, the temperatures, the heat fluxes, the heat transfer coefficients, the ambient temperatures, the tractions and the displacements are specified. These parts of the boundaries fulfil the following relations

$$\Gamma = \Gamma_{\phi} \cup \Gamma_{\omega} = \Gamma_T \cup \Gamma_q \cup \Gamma_c = \Gamma_t \cup \Gamma_u,$$
$$\Gamma_{\phi} \cap \Gamma_{\omega} = \varnothing, \qquad \Gamma_T \cap \Gamma_q \cap \Gamma_c = \varnothing, \qquad \Gamma_t \cap \Gamma_u = \varnothing.$$

Piezoelectricity couples electrical and mechanical fields. The problem is solved by using strong coupling method [8, 12] . It requires the usage of coupled finite elements, which have all mechanical and electric degrees of freedom (displacements and electric potential). Coupled field equations of static piezoelectricity can be expressed as follows

$$\begin{bmatrix} \mathbf{K}_{uu} & \mathbf{K}_{u\varphi} \\ \mathbf{K}_{\varphi u} & \mathbf{K}_{\varphi\varphi} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{\Phi} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_u \\ \rho_{\varphi} \end{bmatrix}, \tag{7}$$

where $\mathbf{K}_{uu}$ is mechanical stiffness matrix, $\mathbf{K}_{u\varphi}$, $\mathbf{K}_{\varphi u}$ are piezoelectric stiffness matrices, $\mathbf{K}_{\varphi\varphi}$ is dielectric stiffness matrix, $\mathbf{F}_u$ is force vector and ρ_{φ} is charge flux vector. Equation (7) is completed with electrical (4) and mechanical boundary condition (6).

Both boundary value problems are solved using FEM software ANSYS [6]. In order to evaluate fitness functions, several tasks are performed (Fig. 3). Genes in individuals are design variables, which are responsible for the generation of the geometry. On the basis of geometry, a finite element mesh is generated. Next, the boundary conditions are applied to appropriate parts of the model. All necessary settings of the analysis are specified before solving the direct problem. Fitness functions are calculated on the basis of post-files with the use of electric, thermal and mechanical quantities. All steps of computation are aided by ANSYS Parametric Design Language (APDL) [6].

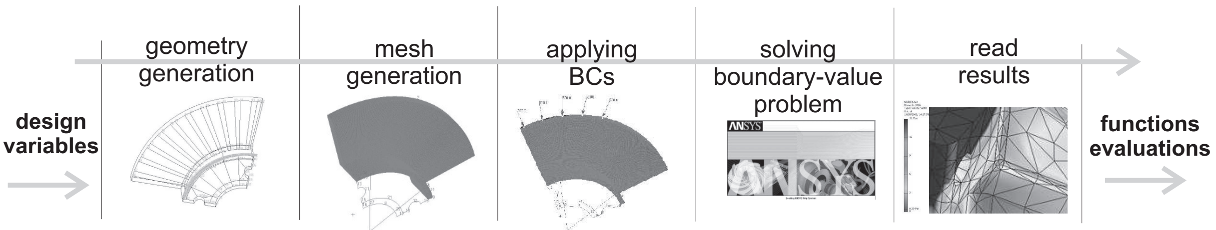

Fig. 3. Steps in calculation of the fitness functions

3.2. Fitness functionals

In the present paper, the following functionals are defined: the minimization of the maximum value of the equivalent von Mises stress, the maximization of the deflection of the structure, the minimization of volume of the structure

$$\min_x f_1 \stackrel{\text{def}}{=} \max\left(\sigma_{eq}\right), \tag{8}$$

$$\max_x f_2 \stackrel{\text{def}}{=} \max\left(u_i\right), \tag{9}$$

$$\min_x f_3 \stackrel{\text{def}}{=} \int_{\Omega} d\Omega. \tag{10}$$

3.3. Numerical examples

3.3.1. Example 1

The MEMS structure modelled as microelectrothermal actuator whose geometry is shown in Fig. 4. Coupled electrical-thermal-mechanical analysis is assumed. The structure is made of polycrystalline silicon with the following material properties: Young's modulus – 158×10^3 MPa, Poisson's ratio – 0.23, thermal expansion coefficient – $3.0 \times 10^{-6}\ \mathrm{K}^{-1}$, thermal conductivity – 140×10^8 pW/(μm K), resistivity – 3.3×10^{-11} T$\Omega\mu$m. The deflection of the actuator is produced when the electric potential difference is applied across the two electrical pads (EP1, EP2). It is possible due to high electrical resistivity and different thermal expansion between thin and wide arms. The device is subjected to the electrical, thermal and mechanical boundary conditions. The electric potential difference between electrical pads is 5 V, temperature of the pads is fixed at 300 K, pads are fixed in space for all degrees of freedom. The length of the actuator is 260 μm, electrical pad is 20 μm×20 μm. Six design variables are assumed. Limitations of the design variables are as follows: Z1, Z2, Z3 – 1.0÷3.0, Z4 – 12.0÷18.0, Z5 – 30.0÷100.0, Z6 – 2.0÷8.0. The multiobjective problem concerns determining

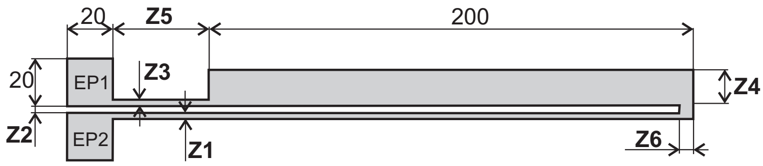

Fig. 4. Geometry and design variables of the thermal actuator

of the particular dimensions of the structures, considering different pairs of the proposed functionals (8), (9), (10). MOOPTIM and NSGAII is used for the optimization tasks. Size of the population and number of iterations are 50 for both algorithms. The further parameters of MOOPTIM and NSGAII are identical as in Section 2.3.1. Figure 5 shows the results of optimization.

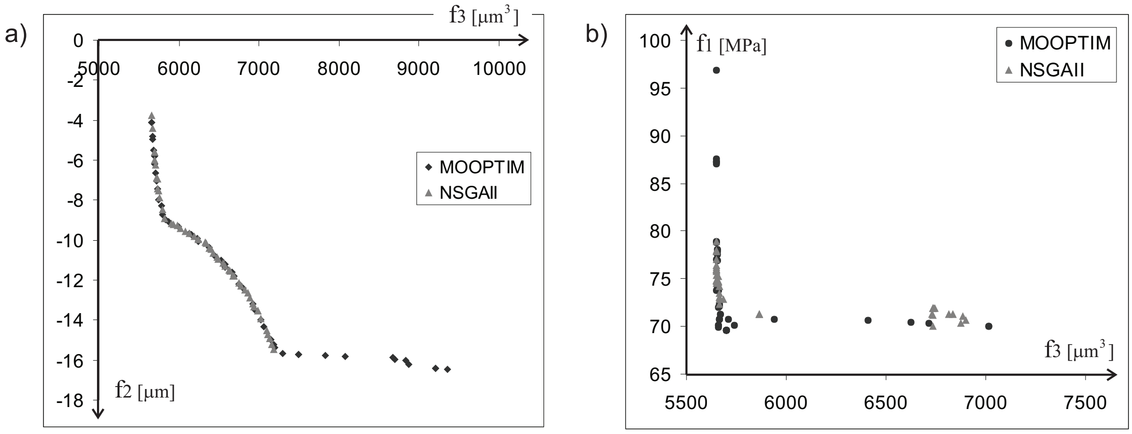

Fig. 5. Pareto-optimal solutions for the electrothermal actuator: (a) volume minimization (10) and deflection maximization (9), (b) volume minimization (10) and equivalent stress minimization (8)

3.3.2. Example 2

In the second optimization problem L-shaped piezoelectric structure is considered (Fig. 6). The length of the structure is 10 mm, whereas the thickness of the thin arm is equal to 1 mm. Left side of the structure (segment AF) is clamped. Electric potentials: 1000 V and 1000 V are applied on the segments AF and CD, respectively. Four design variables: vertical coordinate of point A (range 0÷6.0), vertical and horizontal coordinates of point B (ranges 1.0÷6.0 and 1.0÷5.0), horizontal coordinate of point C (range 5.0÷9.0) are assumed. The PZT-5 ceramic material is applied. In this example also both algorithms (MOOPTIM and NSGAII) are used for the optimization

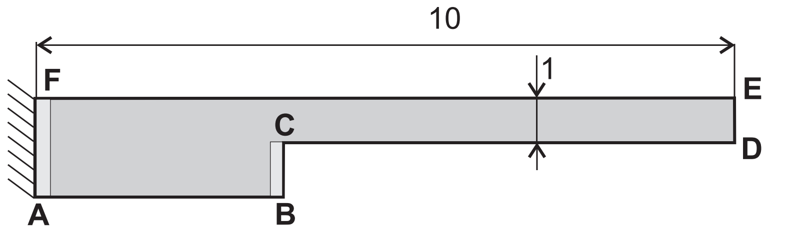

Fig. 6. Geometry of the optimized piezoelectric structure

tasks. Parameters of the algorithms are identical as in Example 1. Figure 7 presents the results of multiobjective optimization for different pairs of functionals.

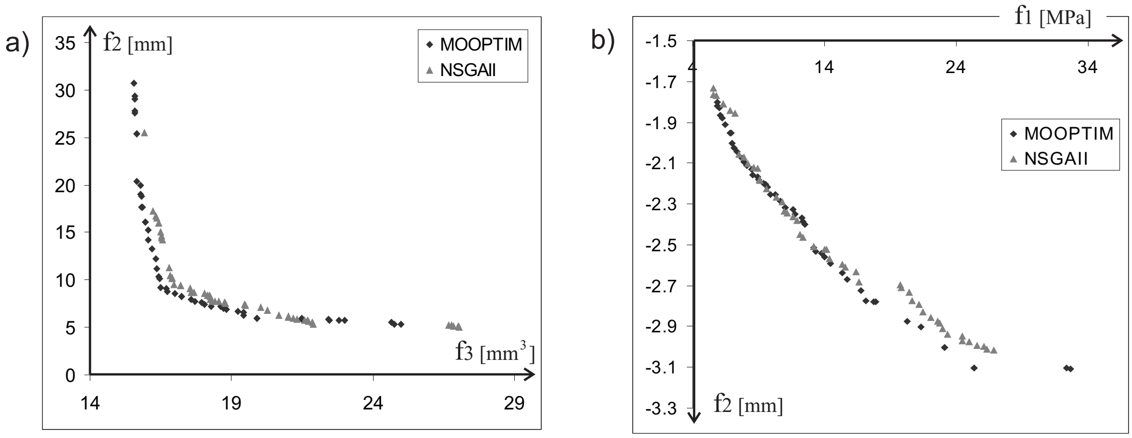

Fig. 7. Pareto-optimal solutions for the L-shaped piezoelectric structure: (a) volume minimization (10) and equivalent stress minimization (8), (b) equivalent stress minimization (8) and deflection maximization (9)

4. Conclusions and future tasks

In the present work the MOOPTIM algorithm has been used for multiobjective shape optimization of MEMS structures. The effectiveness of MOOPTIM has been compared to NSGAII on several benchmark test problems. The obtained results show the effectiveness of MOOPTIM for both unconstrained and constrained optimization tasks. For some test problems, MOOPTIM significantly outperforms NSGAII.

The application of the MOOPTIM to the real-world engineering problems, such as multiobjective optimization of MEMS structures, shows its usefulness. The results obtained by using MOOPTIM are slightly better comparing to the results obtained by using NSGAII. For these problems, besides the convergence, especially distribution of the Pareto-optimal solutions is more extensive (Fig. 5), (Fig. 7). The presented set of final solutions is obtained only for 2500 fitness functions evaluations, so it may not be global. The quality of the Pareto fronts can be improved by increasing the population size or by extending the optimization process.

The application of the FEM software requires evaluation in several steps for each single solution (modification of the geometry, creating finite element mesh, etc.). It can be a very-time consuming task, especially for more complicated geometry. Solution of the coupled problems such as electro-thermo-mechanical or piezoelectric analysis is more time-consuming compared to the single-field problem. To reduce the time of the optimization, parallel computation or approximative surrogate evaluations can be used. Apart from above techniques, authors considered the application of such approach also in grid environment [9].

Acknowledgments

The research is financed by the Polish science budget resources in the years 2008-2011 as a research project.

References

[1] G. Beer. Finite element, boundary element and coupled analysis of unbounded problems in elastostastics. *Int. J. Numer. Meth. Eng.*, **19**: 567–580, 1983.
[2] T. Burczyński, W. Kuś. Distributed evolutionary algorithm - tests and applications. In: *Proc. AI-METH 2002*, Gliwice, 2002.
[3] K. Deb. Multi-objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems. *Evolutionary Computation*, **7**(3): 205–230, 1999.
[4] K. Deb. Evolutionary Multi-Objective Optimization Without Additional Parameters. In: J. Kacprzyk, ed., *Studies in Computational Intelligence*, **54**, pp. 241–257, Springer-Verlag, Berlin-Heidelberg, 2007.
[5] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan. A Fast Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, **6**(2): 182–197, 2002.
[6] *Documentation for ANSYS*. ANSYS Inc., 2007.
[7] B. Fu, T. Hemsel, J. Wallaschek. Piezoelectric transducer design via multiobjective optimization. *Ultrasonics*, **44**: e747–e752, 2006.
[8] L. Gaul, M. Kogl. A boundary element method for transient piezoelectric analysis. *Engineering Analysis with Boundary Elements*, **24**: 591–598, 2000.
[9] W. Kuś. Grid-enabled evolutionary algorithm application in the mechanical optimization problems. *Engineering Applications of Artificial Intelligence*, **20**(5): 629–636, 2007.
[10] N. Maluf, K. Williams. *An introduction to microelectromechanical systems engineering*. Artech House Publishers, 2004.
[11] Z. Michalewicz. *Genetic algorithms + data structures = evolutionary programs*. Springer-Verlag, Berlin-New York, 1996.
[12] H.F. Tiersten. *Linear piezoelectric plate vibrations*. Plenum Press, New York, 1969.
[13] Y. Zhang, Q. Huang, R. Li, W. Li. Macro-modeling for polysilicon cascaded bent beam electrothermal microactuators. *Sensors and actuators A: Physical*, **128**(1): 165–175, 2006.
[14] O.C. Zienkiewicz, R.L. Taylor. *The Finite Element Method*. Butterworth-Heinemann, Oxford, 2000.
[15] E. Zitzler, M. Laumanns, L. Thiele. *SPEA2: Improving the Strength Pareto Evolutionary Algorithm*. TIK-Report 103, 2001.