

Multi-Objective Self-Organizing Migrating Algorithm: Sensitivity on Controlling Parameters

Petr KADLEC, Zbyněk RAIDA, Jiří DŘÍNOVSKÝ

Dept. of Radio Electronics, Brno University of Technology, Purkyňova 118, 612 00 Brno, Czech Republic

kadlecp@feec.vutbr.cz, raida@feec.vutbr.cz, drino@feec.vutbr.cz

Abstract. *In this paper, we investigate the sensitivity of a novel Multi-Objective Self-Organizing Migrating Algorithm (MOSOMA) on setting its control parameters. Usually, efficiency and accuracy of searching for a solution depends on the settings of a used stochastic algorithm, because multi-objective optimization problems are highly non-linear.*

In the paper, the sensitivity analysis is performed exploiting a large number of benchmark problems having different properties (the number of optimized parameters, the shape of a Pareto front, etc.). The quality of solutions revealed by MOSOMA is evaluated in terms of a generational distance, a spread and a hyper-volume error. Recommendations for proper settings of the algorithm are derived: These recommendations should help a user to set the algorithm for any multi-objective task without prior knowledge about the solved problem.

Keywords

MOSOMA, sensitivity, control parameters, multi-objective optimization.

1. Introduction

Stochastic optimizers became a very efficient tool for the design of various electromagnetic structures. Usually, the design of these structures considers more than one objective. Therefore, efficient multi-objective optimizers have been derived from single-objective ones: e.g. NSGA-II (elitist Non-dominated Sorting Genetic Algorithm) [1] for genetic algorithms, MOPSO (Multi-Objective Particle Swarm Optimization) [2] for the particle swarm intelligence, GDE3 (Generalized Differential Evolution) [3] for the differential evolution etc.

Since multi-objective optimization problems are highly non-linear, accuracy and efficiency of their solution strongly depends on settings of a used algorithm [4]. From the viewpoint of a user, a proper setting of the parameters controlling the optimizer is a rather difficult task. On one hand, the settings have to be very robust so that the optimizer can achieve the global optimum with high probabilit-

ity. On the other hand, the parameters should be chosen to ensure high efficiency of the optimization process. Therefore, many researchers put their efforts to reveal sensitivity rules of multi-objective algorithms and to formulate recommendations for proper setting of optimizers.

The influence of control parameters on the GDE3 algorithm is studied in [5]. The tuning of multi-objective PSO-based optimizer parameters is described in [6]. In [7], the author tries to fix crucial parameters of NSGA-II so that the user does not need to take care of it.

This paper tries to follow up the article [8] where a novel Multi-Objective Self-Organizing Migrating Algorithm (MOSOMA) was introduced. As described in [8] and [9], run of MOSOMA is controlled by few parameters set by a user. These parameters have to be set properly to make run of the algorithm efficient. In this paper, we describe the sensitivity of MOSOMA to the setting of its control parameters. The control parameters of MOSOMA are tuned on a test suite of various benchmark problems. The quality of the computed solution is evaluated by the spread, the generational distance and the hyper-volume error metrics.

In the paper, we briefly explain the most important principles of MOSOMA first. Then, we describe the setup of experiments: benchmark problems and evaluation metrics are introduced here. Results of experiments for each investigated control parameter are summarized consequently. Some values are recommended for each of control parameters. The correctness of the recommended values of individual control parameters is verified on two two-objective problems from the field of electromagnetics. Pareto fronts obtained by MOSOMA are compared with other benchmark algorithms for those EM problems. Last experiments present comparison of MOSOMA results with other benchmark optimizers when the recommended settings cannot be used due to an excessive increase of computation time.

2. MOSOMA

MOSOMA is based on the cooperation of a group of agents as described in a single-objective Self-Organizing Migrating Algorithm published in [10]. Each agent corre-

sponds to a vector of state variables of the optimized structure. For each agent, we evaluate objective functions. The agents can update their positions in the decision space in so called migration loops. The direction of the migration is determined by information obtained from the objective space. The values of objective functions are shared by the whole group of agents. The pseudocode of MOSOMA is depicted in Fig. 1.

First, agents from the initial population $Q(1)$ are randomly distributed over the whole decision space according to [8]:

$$x_{q,n} = x_{n,\min} + \text{rnd}_{q,n} (x_{n,\max} - x_{n,\min}) \quad (1)$$

where $x_{q,n}$ denotes the n -th variable of the q -th agent, $\langle x_{n,\min}, x_{n,\max} \rangle$ denotes the feasible interval for the n -th variable, and $\text{rnd}_{q,n}$ is a random number from the interval $\langle 0; 1 \rangle$ with a uniform distribution of probability.

Start

Define initial population $Q(1)$

Compute objective functions in tmp

Find external archive EXT

While $i < I \mid FFC < N_{f,\max} \mid |EXT(i)| < N_{ex,\max}$

For $q = 1 : |Q(i-1)|$

\mathbf{x}_q migrates to all members of $EXT(i-1)$

Compute objective functions in tmp

End

Find $EXT(i)$ from $\text{tmp} \cup EXT(i-1)$

While $|EXT(i)| < N_{ex,\min}$

Find advancing front and crowding distance

Fill $EXT(i)$ with best agents from advancing front

End

$i++$

End

Choose final set P from $EXT(i)$

End

Fig. 1. Pseudocode of MOSOMA [9].

Then, the non-dominated sorting [4] according to the values of the objective function f_m is made for the entire group of agents. The non-dominated sorting uses a concept of dominance which can be found in [4].

The non-dominated solutions build so called advancing fronts as can be seen in Fig. 2. Here, the group of 12 individuals is sorted into three fronts. The solutions from the first front dominate every solution from the advancing fronts, the solutions from the third front are dominated also by the solutions from the second front. Within each front, the less crowded solutions are preferred; therefore, e.g. the solution \mathbf{x}_6 is better than the solution \mathbf{x}_{11} in the first front.

The non-dominated solutions of the first order are saved into an external archive (EXT). If the number of non-dominated solutions in the first front is lower than

the minimal size of the external archive $N_{ex,\min}$, then the EXT is filled in with the best solutions from advancing fronts [4]. The less crowded solutions are preferred.

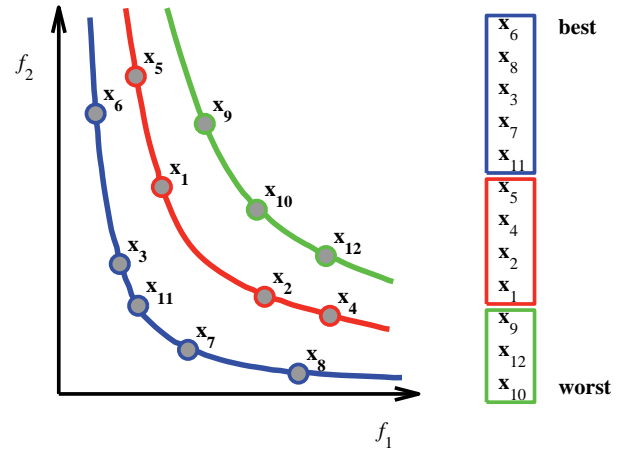


Fig. 2. Non-dominated sorting to advancing fronts.

In each migration loop, T selected agents migrate towards the members of current EXT . The migration proceeds in steps. Each agent \mathbf{x}_p migrating towards the agent \mathbf{x}_q from EXT visits temporary places tmp [9]:

$$\begin{aligned} \text{tmp}_{p,s}(i) &= \mathbf{x}_p(i-1) \\ &+ (\mathbf{x}_q(i-1) - \mathbf{x}_p(i-1)) \cdot \frac{s}{ST} \cdot PL \cdot \text{PRTV}_{p,s} \end{aligned} \quad (2)$$

where i denotes the current migration loop, ST is the number of steps during one migration (s goes from 1 to ST), PL stands for the normalized length of the migration (the distance between the p -th and q -th agent multiplied by PL) and PRTV denotes a perturbation vector. This vector has the same length as the vector \mathbf{x} and has to be generated for every migration according to the equation:

$$\text{PRTV}(n) = \begin{cases} 1 & \text{if } \text{rnd}(n) > PR \\ 0 & \text{if } \text{rnd}(n) \leq PR \end{cases} \quad (3)$$

where PR stands for the probability of perturbation. Perturbation is introduced here to prevent the algorithm from freezing in a local optimum. Perturbation behaves similarly to mutation in genetic algorithms.

The objective functions are computed for each temporary position tmp . Then, a new external archive $EXT(i)$ is determined by non-dominated sorting the union between $EXT(i-1)$ and all $f_m(\text{tmp})$. In one migration loop, the number of $FFC(i)$ is given by the size of the previous external archive $EXT(i-1)$, the number of migrating agents T and the number of steps ST :

$$FFC(i) = |EXT(i-1)| \cdot T \cdot ST. \quad (4)$$

When a new external archive is determined, stopping conditions are checked. If any of the stopping conditions becomes fulfilled, the final set P (with defined size) is

chosen from the current *EXT* so that the members of *P* are distributed as uniformly as possible [9].

3. Experimental Settings

The multi-objective optimization is assumed to achieve two goals at the same time:

1. The element of the computed non-dominated set *P* has to lie in the minimal distance from the true Pareto front of the problem.
2. Elements of the set *P* have to be distributed uniformly. This ensures that the whole Pareto front has been found.

The quality of achievements can be expressed by means of metrics applied for benchmark problems with known Pareto fronts. This section describes used metrics, benchmark problems and general settings of MOSOMA for the analysis of the sensitivity on values of individual control parameters.

3.1 Evaluation Metrics

3.1.1 Generational Distance

The generational distance (*GD*) was introduced by Veldhuizen in [11]. This metric evaluates the quality of the computed non-dominated set from the viewpoint of accuracy. *GD* measures the Euclidian distance between *P* members and a set of 500 uniformly spread true Pareto front members *P**. The *GD* metric is defined by [11]:

$$GD = \frac{\sqrt{\sum_{p=1}^{|P|} d_p^2}}{|P|} \tag{5}$$

where *d_p* stands for the minimal Euclidian distance measured in the objective space between the *p*-th solution from the computed *P* and the corresponding member of the true Pareto front *P** [4]:

$$d_p = \min_{k=1}^{|P^*|} \sqrt{\sum_{m=1}^M (f_m(p) - f_m^*(k))^2} \tag{6}$$

Here, *k* denotes the index of the solution in the set *P** which has the minimal distance to the *p*-th member of the set *P* and *M* stands for the number of objective functions. The lower value of the *GD* metric achieved, the more accurate the solution. For the ideal solution, the *GD* reaches zero.

3.1.2 Spread

The spread (Δ) was introduced by Deb in [1]. This metric expresses the quality of the spread of the computed set *P* and can be evaluated in the objective space. The spread metric measures the ratio between the sum of de-

viations from average distances among neighboring Pareto-optimal solutions and the sum of all distances. The set *P* has to be sorted such a way so that the neighboring solutions can be found. Therefore, such a defined metric can be computed only for two-objective problems.

The way of also evaluating a spread for problems with more than two objectives was proposed in [9]. The procedure is based on searching for the so-called minimum spanning tree (*MST*) of the set *P*. Simply said, *MST* connects all the nodes in the set so that the sum of connections is the shortest one. Connections between any two nodes have to be available, and cycles in the tree have to be prohibited.

The described approach can be applied to problems with an arbitrary number of objectives (including two). The multi-objective spread Δ_M can be computed using equation [9]:

$$\Delta_M = \frac{\sum_{m=1}^M d_{e,m} + \sum_{p=1}^{|MST|} |d_p - d_{avg}|}{\sum_{m=1}^M d_{e,m} + |MST| \cdot d_{avg}} \tag{7}$$

where *d_p* denotes the Euclidian distance between the *p*-th and (*p* + 1)-th solution from *P*, *d_{e,m}* denotes the distance from computed extreme solutions to the true ones and *d_{avg}* is the average distance among all computed solutions.

If computed solutions are spread ideally on the true Pareto front, the spread metric becomes zero. Commonly used algorithms usually achieve a spread between 0.1 and 0.7 [4].

3.1.3 Hyper-volume

The hyper-volume metric *HV* evaluates the multi-objective optimizers from the viewpoint of the spread and the accuracy at the same time. *HV* measures the volume in the objective space, which is covered by the solutions from the found non-dominated set *P* [11]. Here, the metric was introduced as hyperarea, but it is usually called hyper-volume in most references [4]. For each solution from *P*, the hypercube volume *v_p* between the *p*-th solution and a reference point *W* is computed. Then, the hyper-volume can be defined [11]:

$$HV = volume \left(\bigcup_{p=1}^{|P|} v_p \right) \tag{8}$$

Here, *v_p* denotes the hypervolume between *p*-th point from the non-dominated set *P* and the reference point *W*. Total hypervolume *HV* is the union of individual hypervolumes *v_p*. The position of the reference point can be simply defined by a vector composed of the maximal values of the extreme solutions of the true Pareto front for particular objective. For example in Fig. 2, the reference point would be defined *W* = {*f*₁(**x**₈); *f*₂(**x**₆)} considering that **x**₆ and **x**₈ are the extreme true Pareto-optimal solution.

Obviously, values of the *HV* metric are significantly influenced by the magnitudes of individual objectives. Therefore, the relative hyper-volume metric *HVR* was proposed in [11]. The hyper-volume of the computed set *P* is normalized with the hyper-volume size of the true Pareto front (the same set of 500 uniformly spread solutions, as in the case of *GD*, is used) [11]:

$$HVR = \frac{HV(P)}{HV(P^*)}. \tag{9}$$

The relative hyper-volume metric can increase with the increasing accuracy of the computed solution *P*. The relative hyper-volume metric equals one for the ideal distribution of the computed Pareto optimal set *P*. Usually, the relative hyper-volume error is given by:

$$\text{error}_{HV} = |1 - HVR|. \tag{10}$$

Similarly to the previously described metrics, solutions with a better spread and accuracy can reach a smaller value of the hyper-volume error.

3.2 Benchmark Problems

The benchmark suite should be large enough to cover different types of problems. Therefore, we have chosen 9 different unconstrained problems having a different number of decision space variables, different number of objective functions and different shapes of Pareto fronts. All fitness functions are formulated as minimization problems.

abbrev.	ref.	<i>M</i>	<i>N</i>	<i>PF</i>
DTLZ1	[12]	3	5	flat
DTLZ2	[12]	3	5	concave
FON	[13]	2	10	concave
GSA2	[14]	3	2	convex
SCH1	[15]	2	1	convex
UF8	[16]	3	10	concave
TP1	[17]	3	2	convex
ZDT1	[18]	2	30	convex
ZDT2	[18]	2	30	concave

Tab. 1. The MOSOMA settings for the sensitivity analysis of the *FFC* parameter.

Nine multi-objective benchmark problems were employed for purposes of MOSOMA sensitivity analysis. To keep the reasonable extent of the paper, we put here only references to the benchmark problem used, their abbreviations, number of objective functions *M*, number of variables *N* and type of Pareto front *PF*.

Performance of MOSOSMA has been tested on a large suite of benchmark problems having various numbers of decision space variables and objectives in [8] and [9]. The results of various convergence metrics were compared with commonly used algorithms NSGA-II [1] and SPEA2 [19]. All algorithms were allowed to evaluate objective functions same times to maintain fairness of

those comparisons. MOSOMA achieved at least comparable results in accuracy (generational distance metric) and significantly better results in uniform coverage of the true Pareto front (spread metric).

4. Sensitivity Analysis

The sensitivity analysis of the MOSOMA convergence was performed for seven control parameters:

1. The total number of computations of the fitness function *FFC*,
2. The minimal size of the external archive $N_{ex,min}$,
3. The size of the initial population *P*(1),
4. The path length *PL*,
5. The probability of perturbation *PR*,
6. The number of steps *ST*,
7. The number of migrating agents *T*.

Since each benchmark problem from the test suite has a different number of decision space variables, some control parameters are normalized to the number of the decision space variables *N* (control parameters *FFC*, *P*(1) and *T*) or the initial population size *P*(1) ($N_{ex,min}$). During the sensitivity evaluation of a selected parameter, other parameters remain constant. Constant values of parameters are summarized in Tab. 2. These settings were chosen according to the results of previously performed tests [8] and [9].

<i>FFC</i> / <i>N</i> (-)	$N_{ex,min}/P(1)$ (-)	<i>P</i> (1)/ <i>N</i> (-)	<i>PL</i> (-)	<i>PR</i> (-)	<i>ST</i> (-)	<i>T</i> / <i>N</i> (-)
12000	2/3	10	1.5	0.15	4	8

Tab. 2. The MOSOMA settings for the sensitivity analysis of the *FFC* parameter.

For the settings of each parameter, MOSOMA was run 50 times to search for 50 Pareto optimal solutions. Some statistics were computed for the generational distance, the spread and the hyper-volume error metric.

Totally, MOSOMA was run approximately 50000-times (50 repetitions × 7 parameters × 16 values per parameter × 9 test problems). In order to keep the paper to a reasonable extent, we aren't publishing the results of the investigated sensitivity for every test problem and every metric. For every parameter, the least sensitive test problem, a typical one and the most sensitive one were chosen. The sensitivity is evaluated only for one chosen metric in Fig. 3. - 9. We have tried to publish here results for metrics that should be the most influenced by the settings of a specific parameter.

The results are presented in the form of standard boxplot graphs for the investigated metric and parameter. Here, median of the data is highlighted with the central mark. Edges of the box are the 25th and 75th percentiles. These boxplots bring information about typical values of metrics for particular MOSOMA settings. For each parameter,

recommendations for its proper setting are derived according to the results of each metric and each test problem. The borders of recommended intervals are chosen so, that all metrics remain within this interval constant and all the metrics median values are sufficiently low ($GD < 2.0 \cdot 10^{-2}$, $\Delta_M < 3.0 \cdot 10^{-1}$, $errorHV < 2 \cdot 10^{-1}$).

4.1 Fitness Function Computation

Usually, the fitness function computation is the most time-consuming part of the whole process of the optimization. The number of computations of the objective functions in the migration loop changes with the size of the external archive when running MOSOMA. Therefore, formulating some limits for FFC is necessary. Obviously, FFC grows with increasing accuracy of the optimization. Unfortunately, the computational time grows.

For the experiments, the total number of computations of the fitness functions was changed within the interval $FFC/N \in \{1000; 20000\}$. Other parameters were set as indicated in Tab. 2.

Fig. 3 depicts the results of the generational distance metric for three test problems: GSA2, DTLZ2 and ZDT2. In accordance with predictions, the accuracy of the optimizer increases with the growth of FFC . The ZDT2 problem is strongly sensitive to the FFC parameter. The accuracy seems to remain the same from a certain value of FFC . This is not true for the ZDT2 problem, where several local optima can be assigned incorrectly as a result [18]. Therefore, we recommend to choose the parameter FFC from the interval $FFC/N \in \{8000; 15000\}$ considering the time devoted for the evaluation of objectives.

4.2 EXT Minimal Size

The minimal size of the external archive $N_{ex,min}$ influences the speed of the convergence of algorithms primarily. If MOSOMA is not able to find the sufficient number of non-dominated solutions of the first order during the first migration loop, we have to fill in the external archive with the solutions from advancing fronts. The less-crowded solutions are preferred in this case to preserve the diversity among the members of EXT . Obviously, the highest value of the EXT minimal size is limited by the size of the initial population.

The minimal size of EXT influences the diversity among the computed non-dominated sets. Therefore, a spread metric was chosen in Fig. 4 to show the sensitivity of MOSOMA on the $N_{ex,min}$ parameter. The influence of the EXT minimal size is more significant for more complex problems as indicated by the most sensitive results for the ZDT1 test problem.

MOSOMA is almost insensitive to the EXT minimal size in the case of less-complicated problems as shown by the results for FON and TP1. Usually, the quality of the spread slightly decreases with the $N_{ex,min}$ parameter. In

this case, a lot of evaluations of objectives are devoted for migration towards solutions which can lie a far distance from the true Pareto front, while the region of the true Pareto front is not revealed sufficiently.

The optimal value of the EXT minimal size can be somewhere within the borders of the recommended interval $N_{ex,min}/P(1) \in \{0.3; 0.6\}$.

4.3 Initial Population Size

The parameter $Q(1)$ defines the number of agents randomly generated at the beginning of a MOSOMA run. At first sight, a larger value of $Q(1)$ might seem to be better to research the whole decision space. On the other hand, a too large $Q(1)$ slows down the algorithm during the first migration loops. Therefore, some trade-off has to be chosen taking into account the number of state variables and the maximal FFC given by (4). The $Q(1)$ parameter directly influences the ability to achieve the true Pareto front. A larger $Q(1)$ ensures that agents can reach the region of the global optimum during the first migration loops. Then, this region can be researched adequately in the rest of the optimization process to provide a very good accuracy and the spread of the non-dominated set.

The sensitivity of MOSOMA on the $Q(1)$ parameter is depicted in Fig. 5. Here, the hyper-volume error measured for the different values of the $Q(1)$ parameter can be seen. The hyper-volume error seems to decrease slightly with the growth of the $Q(1)$ but the decrease is not so strong. After the detailed study of all tests, we can recommend to choose the size of the initial population from the interval $Q(1)/N \in \{5; 12\}$.

4.4 Path Length

The path length defines the length of the migration between two agents as a multiple of their Euclidian distance in the decision space. The path length is strictly connected to the ST parameter (the number of steps). Both parameters have to be set such a way so that the agents do not visit the previously researched positions. Therefore, the following condition has to be met:

$$s \frac{PL}{ST} \neq 1 \text{ for } \forall s \in \{1, 2, \dots, ST\}. \quad (11)$$

The sensitivity of MOSOMA to the path length (PL) is depicted in Fig. 6. The generational distance versus the path length is given here. The accuracy of MOSOMA grows with the increase of PL , especially for more complex problems (as in the case of DTLZ2 and ZDT1 problems). The searching range of MOSOMA is enlarged with a higher value of PL , which avoids the algorithm to freeze in the local optimum and increases the diversity among the non-dominated set.

The recommended interval for the path length is $PL \in \{1.2; 1.7\}$.

4.5 Number of Steps

During each iteration loop, each agent from the set T migrates towards the members of the external archive. Every migration proceeds in the predefined number of steps ST . Again, a larger value of ST causes an increase of computational demands, but the decision space is researched more carefully. The parameter ST has to be chosen considering the value of PL together with the condition in (11).

Fig. 7 depicts the variation of the hyper-volume error with the variation of the parameter ST . The hyper-volume error increases with a growing number of steps. This is probably caused by the fact, that regions wide from the Pareto-optimal solutions are researched in detail during first migration loops. Then, only few FFC remain for detailed research of Pareto-optimal regions during further migration loops. Therefore, we recommend to set the lower values of the ST in the interval $ST \in \{2; 5\}$.

4.6 Probability of Perturbation

The probability of perturbation was introduced in the original single-objective SOMA to prevent the algorithm from freezing in the local optimum. If the agent remains in the local optimum, the direction of this migration can be arbitrarily changed with a certain probability, defined by the parameter PR . Obviously, a higher value of the probability of perturbation decreases the speed of

MOSOMA to move agents in the decision space more chaotically, which reduces the convergence of MOSOMA.

Fig. 8 shows that the spread metric of the computed non-dominated set grows slowly with the increasing probability of the perturbation for all the test problems.

Therefore, we recommend lower values of PR . This parameter should be chosen within the interval $PR \in \{0.1; 0.3\}$.

4.7 Number of Migrating Agents

The number of migrating agents T is limited by the size of the initial population. With a higher number of T , the research of the decision space is more precise on the one hand, but the speed of MOSOMA decreases on the other hand.

The hyper-volume error metric covering both the accuracy and the spread has been chosen for the evaluation of the sensitivity of MOSOMA on the parameter T . Results are depicted in Fig. 9. MOSOMA seems to be almost insensitive on the settings of T for all types of problems.

The recommended interval is $T/N \in \{5; 10\}$.

Par.	FFC/N (-)	$N_{ex,min}/P(1)$ (-)	$P(1)/N$ (-)	PL (-)	PR (-)	ST (-)	T/N (-)
Min	8000	0.3	5	1.2	0.1	2	5
Max	15000	0.6	12	1.7	0.3	5	10

Tab. 3. Recommended intervals for MOSOMA control parameters.

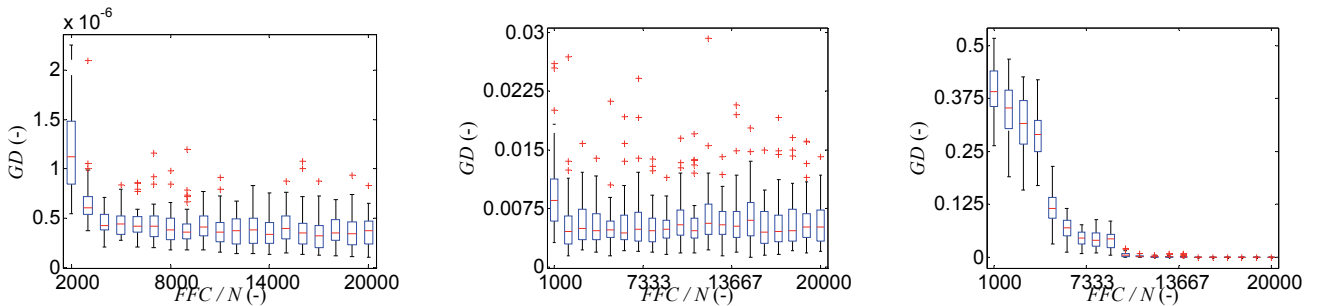


Fig. 3. Sensitivity of MOSOMA on FFC (the number of fitness functions computations) measured by generational distance metric: test problem GSA2 (left), DTLZ2 (middle) and ZDT2 (right).

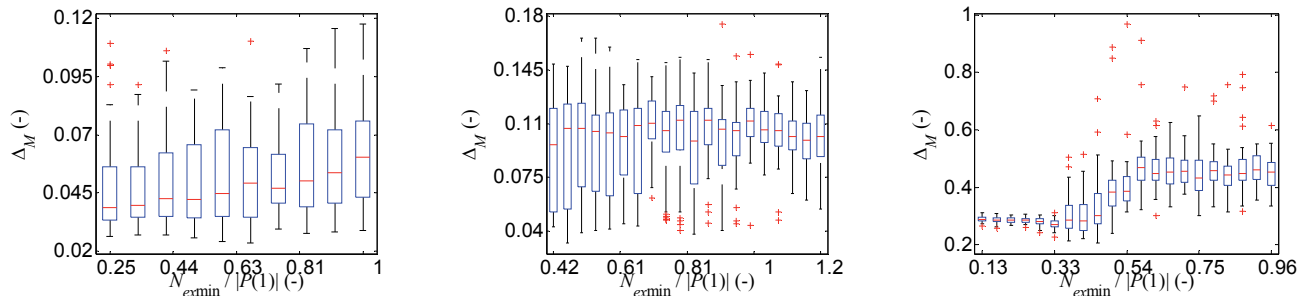


Fig. 4. Sensitivity of MOSOMA on $N_{ex,min}$ (external archive minimal size) measured by spread metric: test problem FON (left), TP1 (middle) and ZDT1 (right).

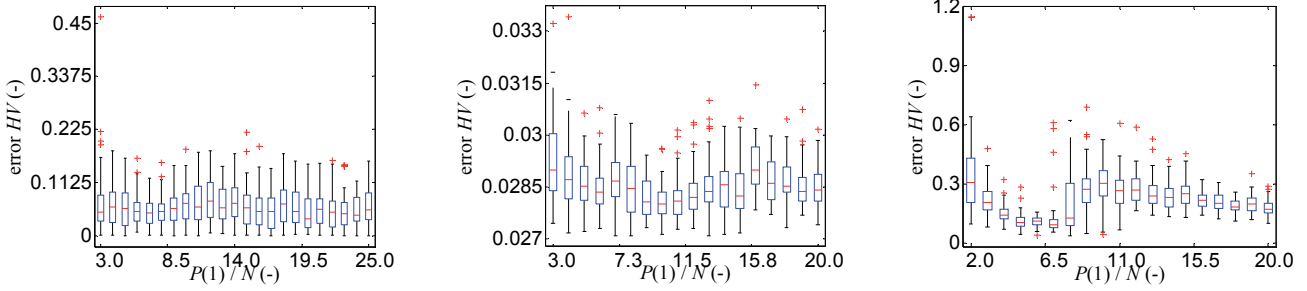


Fig. 5. Sensitivity of MOSOMA on $P(1)$ (initial population size) measured by hyper-volume error metric: test problem TP1 (left), FON (middle) and DTLZ2 (right).

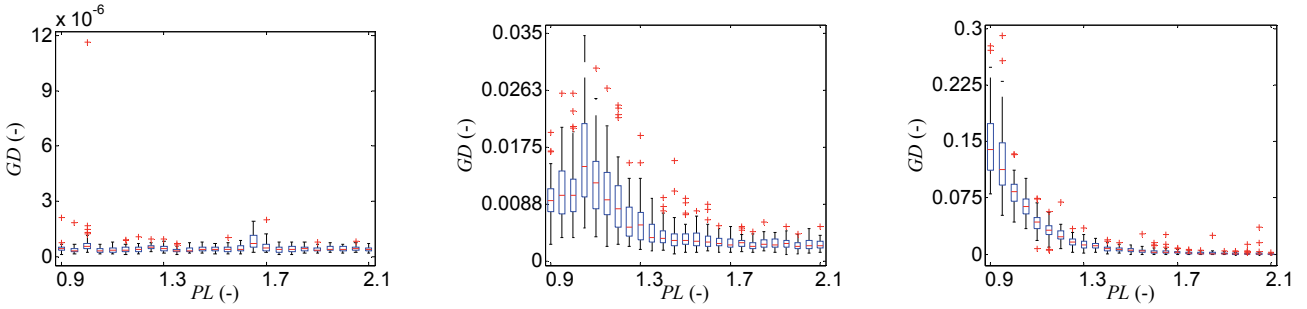


Fig. 6. Sensitivity of MOSOMA on PL (path length) measured by generational distance metric: test problem GSA2 (left), DTLZ2 (middle) and ZDT1 (right).

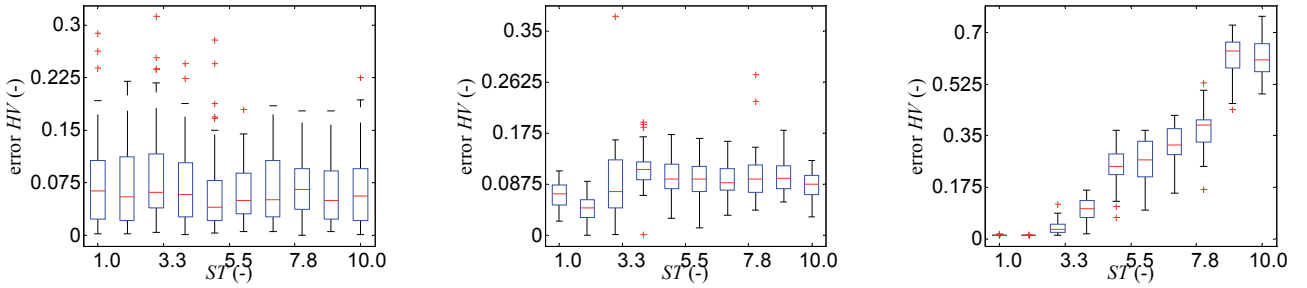


Fig. 7. Sensitivity of MOSOMA on ST (number of steps during one migration) measured by hyper-volume error metric: test problem GSA2 (left), UF8 (middle) and ZDT1 (right).

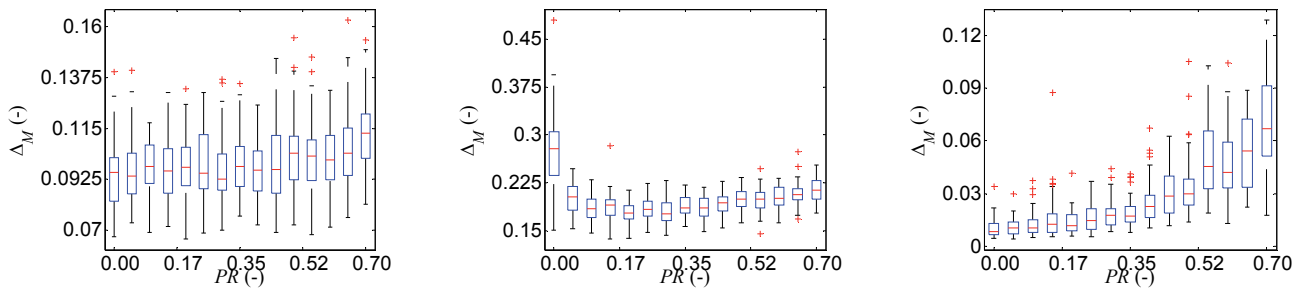


Fig. 8. Sensitivity of MOSOMA on PR (probability of perturbation) measured by spread metric: test problem GSA2 (left), ZDT2 (middle) and SCH1 (right).

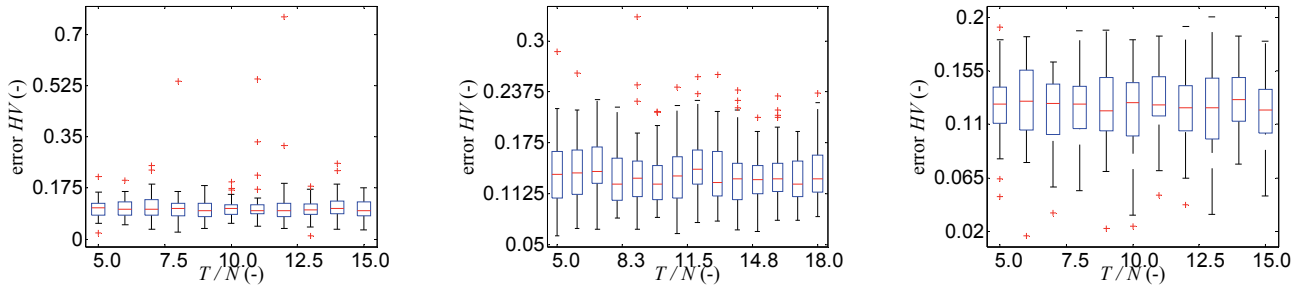


Fig. 9. Sensitivity of MOSOMA on T (number of migrating agents) measured by hyper-volume error metric: test problem UF8 (left), DTLZ2 (middle) and ZDT1 (right).

5. EM Problems

In this section, we are going to demonstrate that recommended values of control parameters are derived reasonably. Two multi-objective problems from the field of electromagnetics introduced in [8] are solved here: the design of a partially filled waveguide and the order reduction of a Debye model. Results of MOSOMA with different settings randomly chosen from the recommended intervals are presented here.

Two multi-objective problems from [8] are completed by a real-life EM optimization which requires an extremely high CPU power to evaluate objective functions. The design of a Yagi-Uda filtenna shows the behavior of MOSOMA in case when an adequate number of FFC cannot be performed.

Pareto fronts found by MOSOMA are compared with results of commonly used optimizers NSGA-II [1] and MOPSO [20] for all the examined problems.

5.1 Partially Filled Waveguide

The cut-off frequency of a waveguide dominant mode can be decreased if a proper dielectric layer is placed in [21]. The model of a partially filled waveguide is depicted in Fig. 10. Here, the symbol a denotes the width of the waveguide, d_1 and d_2 stand for the height of the dielectric layer and vacuum ($\epsilon_2 = 1$) inside the waveguide, respectively. Relative permittivity of the dielectric layer is denoted by ϵ_1 . We assume non-magnetic materials in the waveguide (relative permeability remains one $\mu_1 = \mu_2 = 1$).

Our goal is to determine the “cheapest” dielectric layer for every possible cut-off frequency of the R100 waveguide ($a = 22.86$ mm, $d_1 + d_2 = 10.16$ mm). Therefore, two objectives were formulated in [8]:

$$\begin{aligned} f_1 &= f_c \\ f_2 &= \epsilon_1 d_1 \end{aligned} \quad (12)$$

The first objective minimizes the cut-off frequency f_c of the dominant mode. The second objective minimizes the expected production cost. The price of the used dielectric

material is assumed to increase with the increasing relative permittivity and the height of the layer.

We would like to make a note that both the relative permittivity and the height of the layer are discrete parameters in real life. This is caused by the fact that dielectric substrates are manufactured in a limited number of combinations of relative permittivity and heights.

The state parameters can be chosen from the following intervals: $\epsilon_1 \in \langle 1; 10 \rangle$ and $d_1 \in \langle 0; 10.16 \rangle$ mm.

The decision space with the highlighted Pareto-optimal solutions is depicted in Fig. 11. Two types of Pareto-optimal solutions can be found here: the waveguide is fully filled by the dielectric material ($d_1 = 10.16$ mm) and the waveguide is partially filled by the layer having the relative permittivity approximately $\epsilon_1 \approx 2$. The Pareto front is fully completed by the waveguide filled by vacuum only ($d_1 = 0$ mm).

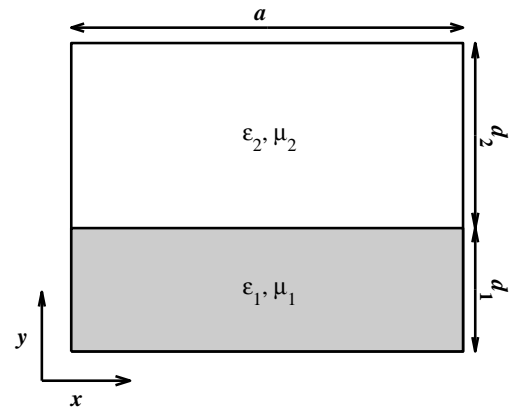


Fig. 10. Partially filled waveguide [8].

MOSOMA was run several times with different settings of control parameters. Three randomly chosen settings are summarized in Tab. 4. In Fig. 12, we compare Pareto fronts consisting of 25 elements, which were computed by MOSOMA with the corresponding settings, with the true Pareto front of the problem. Obviously, MOSOMA was able to find the Pareto front with a very good accuracy and an excellent spread. The stability of the results can be seen in the detailed plot of the Pareto front, especially.

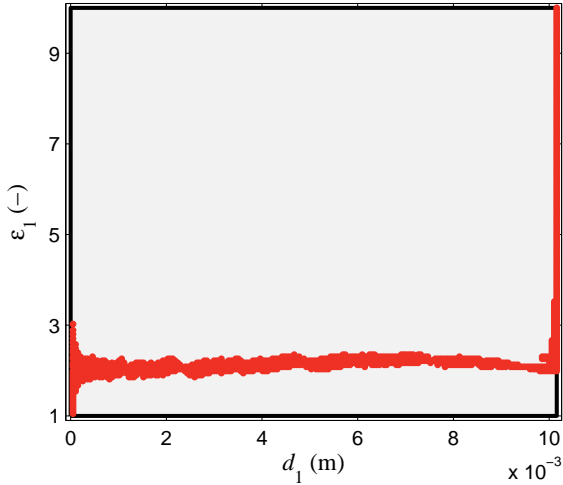


Fig. 11. Decision space of the partially filled waveguide problem: Pareto-optimal solutions are depicted in red.

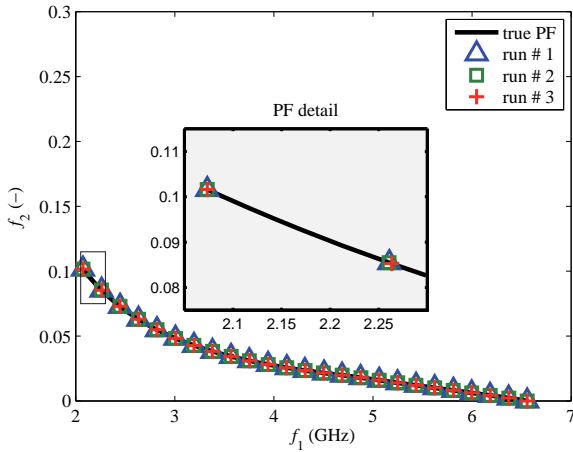


Fig. 12. Pareto front members obtained by three independent runs of MOSOMA with randomly chosen settings.

The Pareto front chosen from 10 independent MOSOMA runs with settings #1 from Tab. 4 is compared with results obtained by algorithms NSGA-II and MOPSO in Fig. 13. The best result of ten runs of each algorithm is depicted here. All the algorithms were set so that the objective functions were evaluated 20000 times in maximum to keep the comparison fair.

run	FFC/N (-)	$N_{ex,min}/P(1)$ (-)	$P(1)/N$ (-)	PL (-)	PR (-)	ST (-)	T/N (-)
1	10 000	0.50	10	1.5	0.2	3	8
2	7 000	0.56	8	1.3	0.3	2	7
3	9 000	0.42	6	1.4	0.1	4	9

Tab. 4. MOSOMA settings for individual runs of partially filled waveguide design.

Parameters of the NSGA-II algorithm were set in the following way: the population consisted of 25 individuals; the optimization comprised 800 generations; probability of the crossover was 0.9 and probability of the mutation was 0.1.

Parameters of the MOPSO algorithm were set in the following way: the population was created by 25 individuals; the optimization comprised 800 generations; the

inertia weight decreased from 0.9 in the first iteration to 0.4 in the last iteration; and current speed vector was computed using the coefficients $c_1 = c_2 = 1.494$.

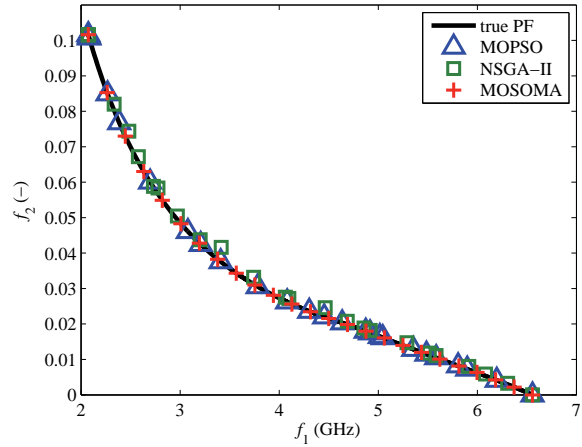


Fig. 13. Pareto front elements for partially filled waveguide problem obtained by the following algorithms: MOPSO (blue triangle), NSGA-II (green square) and MOSOMA (red cross).

Obviously, the Pareto-front elements computed by MOSOMA and MOPSO catch accurately the true Pareto front. On the other hand, some Pareto-front elements computed by NSGA-II do not catch the true Pareto front.

MOSOMA outperforms both MOPSO and NSGA-II in the spread of the computed set: the length of the curve between two neighboring elements of the MOSOMA set remains constant. On the contrary, both the NSGA-II algorithm and the MOPSO algorithm overcrowds some regions of the true Pareto front by the solutions and do not cover other parts of the true Pareto front by the solutions.

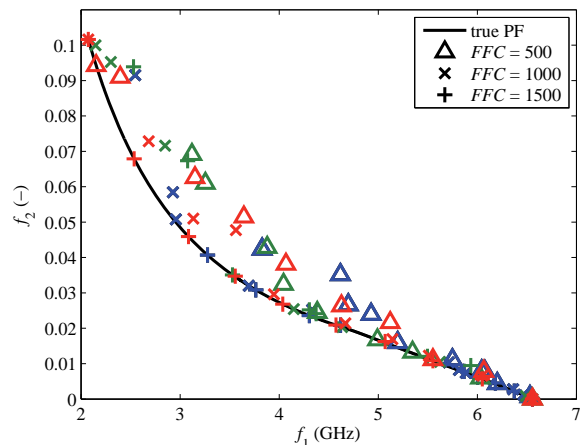


Fig. 14. Elements of the non-dominated sets computed by MOPSO (blue), NSGA-II (green) and MOSOMA (red) after 500, 1 000 and 1 500 evaluations of the objective function.

Fig. 13 demonstrates that all the used algorithms are more or less effective with their robust settings. Dealing with the efficiency, Fig. 14 shows the evolution of the non-dominated set computed by particular optimizers in 500, 1 000 and 1 500 FFC. The best result of ten independent

runs for every optimizer was chosen. The settings of optimizers were identical to tests presented in Fig. 13; only the size of the non-dominated set was reduced to 10 to keep Fig. 14 readable.

Obviously, all the optimizers are able to reach the true Pareto front after 1500 *FFC*. MOPSO seems to be the most efficient algorithm because all the elements of the non-dominated set are located almost on the true Pareto front just after 1000 *FFC*.

Efficiency of the NSGA-II algorithms seems to be the lowest one because some of the elements of the non-dominated set are located far away from the true Pareto front even after 1500 *FFC*. The evolution of the non-dominated set computed by MOSOMA shows that MOSOMA is able to produce the non-dominated set with good spread very efficiently after reaching the true Pareto front region.

5.2 Debye Model Order Reduction

Dispersive materials can be modeled using the Debye model if a time domain technique is employed. Then, the frequency dependent relative permittivity of the dispersive material is described by [22]:

$$\varepsilon_r(j\omega) = \varepsilon_\infty + \sum_{n=1}^N \frac{\varepsilon_n - \varepsilon_\infty}{1 + j\omega\tau_n} \quad (13)$$

where ε_r denotes the relative permittivity of the material, j stands for the imaginary unit, the angular frequency is denoted by ω , N is the order of the Debye model, ε_∞ is the relative permittivity for the infinite frequency and ε_n and τ_n are the static permittivity and the relaxation time of the n -th pole, respectively.

In some solvers, the first-order Debye model is implemented only. Then, the materials described by higher order Debye models have to be replaced by the model with the reduced order. The order reduction can be defined as a two-objective problem:

$$\begin{aligned} f_1 &= \sum_{p=1}^P \left| \operatorname{Re}(\varepsilon_{r,\text{first}}(p) - \varepsilon_{r,\text{third}}(p)) \right| \\ f_2 &= \sum_{p=1}^P \left| \operatorname{Im}(\varepsilon_{r,\text{first}}(p) - \varepsilon_{r,\text{third}}(p)) \right| \end{aligned} \quad (14)$$

where P stands for the number of frequency steps used for the comparison of the first-order Debye model and the third-order Debye model of the relative permittivity $\varepsilon_{r,\text{first}}$ and $\varepsilon_{r,\text{third}}$.

When reducing the order of Debye model of the Ecosorbe LS22 material, three parameters have to be determined: the relative permittivity for the infinite frequency $\varepsilon_\infty \in \langle 1; 100 \rangle$, the relaxation time $\tau_1 \in \langle 1 \cdot 10^{-12}; 1 \cdot 10^{-9} \rangle$ s and the static permittivity $\varepsilon_1 \in \langle 1; 30 \rangle$.

The frequency band from 0.3 GHz to 6.0 GHz was divided into $P = 359$ samples. Again, three randomly cho-

sen MOSOMA settings from the intervals recommended in Sec. 4 of this paper were used to solve the optimization task (see Tab. 5).

run	FFC/N (-)	$N_{ex,\min}/P(1)$ (-)	$P(1)/N$ (-)	PL (-)	PR (-)	ST (-)	T/N (-)
1	10000	0.50	10	1.5	0.2	3	6
2	12000	0.58	8	1.4	0.3	3	9
3	8000	0.42	11	1.3	0.1	5	7

Tab. 5. MOSOMA settings for individual runs of order reduction of Debye model.

Fig. 15 depicts 25 Pareto front solutions of three independent runs of MOSOMA. Our algorithm was able to find the Pareto front with very good accuracy. Considering the previous example, the individual Pareto-optimal sets differ in their spread because MOSOMA has problems to find the extreme solutions of the Pareto front. This problem is probably caused by a higher dimensionality of the problem and a higher sensitivity of objective functions on the variation of input variables.

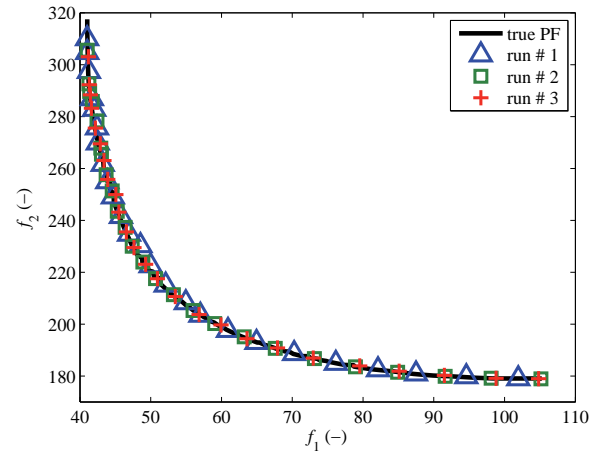


Fig. 15. Pareto front of order reduction of Debye model for three runs of MOSOMA with randomly chosen settings.

Fig. 16 shows the comparison of Pareto fronts computed by MOSOMA, MOPSO and NSGA-II; the best result of 10 independent runs of the algorithms is presented.

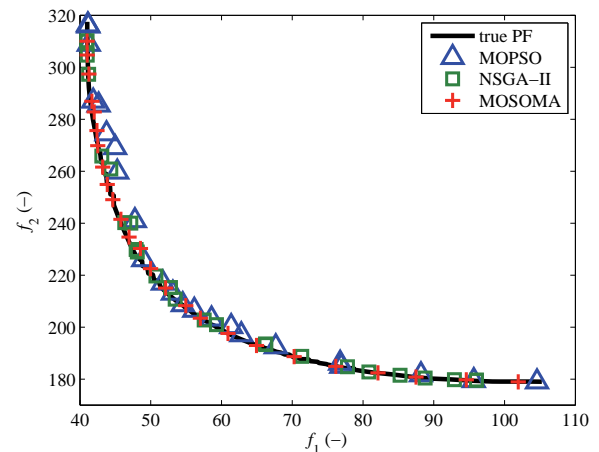


Fig. 16. Pareto sets obtained by MOPSO, NSGA-II and MOSOMA; order reduction of Debye model.

MOSOMA was set according to #1 from Tab. 5 (objective functions could be computed 30000-times in maximum). Settings of other two algorithms were identical with the previous example; only the number of generations was increased to 1200 to keep the comparisons fair.

Some non-dominated elements of the set computed by MOPSO are located in a longer distance from the true Pareto front. Elements of the set computed by NSGA-II caught the Pareto-front accurately but were non-uniformly spread on the true Pareto front. MOSOMA had a problem to reveal the extreme Pareto front solutions but almost the whole true Pareto front was covered with a good accuracy and uniformity.

5.3 Yagi-Uda Filtenna Design

The multi-objective design of the Yagi-Uda filtenna was introduced in [23]. The filtenna is an antenna which can provide prescribed frequency filtering for the predefined direction of the radiation or the reception. Elements of the Yagi-Uda filtenna are designed to irradiate the energy in the main-lobe direction in the pass band and to the backward directions in the stop band.

For the design of the filtenna, two objectives can be formulated. First, the gain of the antenna in the main lobe direction should be relatively high with minor variations for the pass band. Second, the gain of the antenna in the main lobe direction should sharply dive out for the stop band. These two goals can be formulated as the minimization of the following objective functions F_1 and F_2 [23]:

$$F_1 = -\frac{1}{P} \sum_{p=1}^P G(f_p, \theta_0) \tag{15}$$

$$F_2 = \frac{1}{S} \sum_{s=1}^S G(f_s, \theta_0)$$

where P and S denote the number of frequency samples in the pass band f_p and the stop band f_s , respectively. The symbol G denotes the gain and θ_0 stands for the main lobe direction. In order to ensure the proper impedance matching of the designed antenna, two constraint functions have to be formulated [23]:

$$\begin{aligned} |50 - \text{Re}(Z_{in})| < 5 \\ |\text{Im}(Z_{in})| < 10 \end{aligned} \tag{16}$$

where Z_{in} denotes the input impedance of the antenna.

The standard configuration of the Yagi-Uda antenna [24] with one driven element, one reflector and five directors is considered for this experiment. The length of individual elements d and the spacing between neighboring elements s are the parameters computed by the optimizer. Hence, the decision space has $N=13$ dimensions. The length of individual elements can vary within $0.3\lambda < d_n < 0.7\lambda$ and spacing of elements can vary within $0.2\lambda < s_n < 0.6\lambda$. Here, the symbol λ denotes the wave-

length for the central frequency of the pass band f_p . The frequency band considered was from 24 GHz to 36 GHz with the pass band f_p from 28 GHz to 32 GHz.

The values of the gain and the input impedance which are necessary for the evaluation of objective functions and constraint functions are computed by the moment-method software 4NEC2 [25]. The analysis of an antenna configuration takes two seconds, approximately. Therefore, the number of FFC was limited to 10000. So, the whole optimization process takes approximately 5.5 hours. All the algorithms MOPSO, NSGA-II and MOSOMA were run five-times and the best results are presented here.

Settings of MOPSO and NSGA-II are identical to the previous examples; only the number of generations was reduced to 200 and number of individuals was set to 50. For MOSOMA, we set the total number of evaluation of objectives to $FFC = 10000$, we considered the size of the initial population $P(1) = 25$, the length of the path $PL = 1.3$, probability of perturbation $PR = 0.1$, the number of steps $ST = 3$ and the number of migrating agents $T = 20$. Parameters $P(1)$ and T were not set according to the recommended values given in Tab. 3: we decreased them to lower the number of FFC . If the recommended values would be used, MOSOMA would spent too much of the available FFC for exploring the initial populations.

All the algorithms started with the same random initial population to enhance the fairness of the comparison.

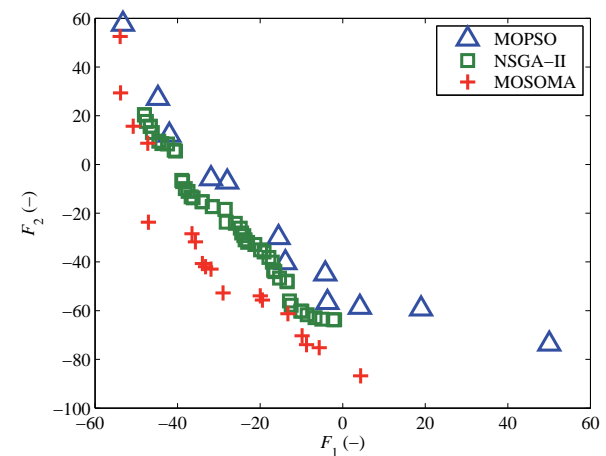


Fig. 17. Pareto fronts computed by MOPSO, NSGA-II and MOSOMA: design of Yagi-Uda filtenna.

Fig. 17 shows that all the optimizers face a problem to find the desired number of non-dominated solutions. The set computed by NSGA-II contains 46 elements; MOSOMA computed 19 non-dominated solutions and MOPSO produced only 12 elements. The Pareto set of the lowest quality was computed by MOPSO: all the computed elements were dominated by the solutions computed by NSGA-II and MOSOMA. The Pareto set computed by NSGA-II was closer to the best Pareto set computed by MOSOMA compared to the MOPSO set. Still, all elements of the MOPSO set are dominated by the solutions computed by MOSOMA.

6. Conclusions

The paper deals with control parameters of the novel Multi-Objective Self-Organizing Migrating Algorithm. The sensitivity of the algorithm on its own settings has been evaluated on a large test suite of benchmark problems with known true Pareto fronts. The quality of computed solutions was measured by the spread, generational distance and hyper-volume error.

The sensitivity analysis has shown that results of the optimization are almost insensitive on some parameters (the number of migrating agents or the size of the initial population). On the other hand, the convergence of MOSOMA is slightly influenced by some parameters (the probability of perturbation or the path length). The simulation can become instable with a more complex problem having a large number of decision space variables and local optima (the ZDT problems family).

According to the sensitivity analysis, some recommendations for setting the control parameters correctly were made. All the evaluated metrics of MOSOMA results are stable within those intervals. Choosing parameters according to the recommended intervals should ensure that MOSOMA reveals the true Pareto front with satisfactory accuracy and uniformity of spread.

The correctness of recommended intervals for MOSOMA control parameters derived in Tab. 3 was verified by three two-objective electromagnetic problems: the design of a partially filled waveguide, the reduction of the order of the Debye model, and the design of a Yagi-Uda filtenna. The MOSOMA was shown to produce stable solutions in terms of the accuracy of the Pareto front and the spread of elements of the Pareto front if control parameters are chosen from the recommended intervals.

Pareto fronts computed by MOSOMA were compared with solutions produced by commonly used algorithms MOPSO and NSGA-II. MOSOMA outperforms both the algorithms particularly in spread of the computed non-dominated set. In case of the Yagi-Uda filtenna design, optimizers could not perform an adequate number of evaluations of objective functions due to an excessively increasing computational time. Still, MOSOMA was able to find the non-dominated set with the highest quality.

Acknowledgements

Research described in this paper was financially supported by the Czech Science Foundation under grant no. P102/12/1274.

Support of the WICOMT project, the registration number CZ.1.07/2.3.00/20.0007, is also gratefully acknowledged.

Optimization routines were run on computational facilities of the SIX Research Center, the registration number CZ.1.05/2.1.00/03.0072.

References

- [1] DEB, K., PRATAP, A., AGARWAL, S., MEYARIVAN, T. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002, vol. 6, no. 2, p. 182 - 197.
- [2] COELLO COELLO, C. A., LECHUGA, M. S. MOPSO: A proposal for Multiple Objective Particle Swarm Optimization. In *IEEE Congress on Evolutionary Computation*. Hawaii (USA), 2002, p. 825 - 830.
- [3] KUKKONEN, S., LAMPINEN, J. GDE3: The third evolution step of Generalized Differential Evolution. In *IEEE Congress on Evolutionary Computation*. Edinburgh (Scotland), 2005, p. 443 - 450.
- [4] DEB, K. *Multi-Objective Optimization using Evolutionary Algorithms*. Chichester (UK): Wiley, 2001.
- [5] KUKKONEN, S., LAMPINEN, J. An empirical study of control parameters for the third version of Generalized Differential Evolution (GDE3). In *Proceedings of IEEE Congress on Evolutionary Computation*, Vancouver (Canada), 2006, p. 2002 - 2009.
- [6] ZHENG, B., LI, Z., FENG, X. An exploratory study of sorting particle swarm optimizer for multiobjective optimization. In *Fourth International Conference on Bio-Inspired Computing BIC-TA '09*. Hangzhou (China), 2009, p. 112 - 119.
- [7] KHOA, D. T. Elitist non-dominated sorting GA-II (NSGA-II) as a parameter-less multi-objective genetic algorithm. In *Proceedings of IEEE SoutheastCon*. Fort Lauderdale (Florida, USA), 2005, p. 359 - 367.
- [8] KADLEC, P., RAIDA, Z. A novel Multi-Objective Self-Organizing Migrating Algorithm. *Radioengineering*, 2011, vol. 20, no. 4, p. 804 - 816, 2011.
- [9] KADLEC, P., RAIDA, Z. Self-Organizing Migrating Algorithm for optimization with general number of objectives. In *Proceedings of 22nd International Conference Radioelektronika 2012*. Brno (Czech Republic), 2012, p. 111 - 115.
- [10] ZELINKA, I., LAMPINEN, J. SOMA - Self organizing migrating algorithm. In *Proceedings of 6th MENDEL International Conference on Soft Computing*. Brno (Czech Republic), 2000, p. 76 - 83.
- [11] VELDHUIZEN, D. V. Multiobjective evolutionary algorithms: classifications, analyses and new innovations. *Ph.D. Thesis*, Dayton, OH, Air Force Institute of Technology. Technical report No. AFIT/DS/ENG/99-01, 1999, 270 p.
- [12] DEB, K., THIELE, L., LAUMANN, M., ZITZLER, E. *Scalable Test Problems for Evolutionary Multi-Objective Optimization*. Kanpur (India): Kanpur Genetic Algorithms Lab. (KanGAL), Indian Institute of Technology, 2001. KanGAL Report 2001001.
- [13] FONSECA, C. M., FLEMING, P. J. An overview of evolutionary algorithms in multi-objective optimization. *Evolutionary Computation Journal*, 1995, vol. 3, no. 1, p. 416 - 423.
- [14] MAO, J., HIRASAWA, K., HU, J., MURATA, J. Genetic symbiosis algorithm for multiobjective optimization problem. In *Proceedings of the International Workshop on Robot and Human Interactive Communication*. Osaka (Japan), 2000, pp. 137 - 142.
- [15] SCHAFFER, J. D. Some experiments in machine learning using vector evaluated genetic algorithms. *Ph.D. Thesis*, Nashville, Tennessee, Vanderbilt University.
- [16] LI, H., ZHANG, Q. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2009, vol. 13, no. 2, p. 284 - 302.
- [17] JIN, Y., OKABE, T., SENDHOFF, B. Solving three-objective optimization problems using evolutionary dynamic weighted aggregation: Results and analysis. In *Genetic and Evolutionary*

- Computation – GECCO 2003*. Berlin: Springer, p. 203 – 214, 2003.
- [18] ZITZLER, E., DEB, K., THIELE, L. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation Journal*, 2000, vol. 8, no. 2, p. 125 – 148.
- [19] ZITZLER, E., LAUMANN, M., THIELE, L. SPEA2: Improving the strength Pareto evolutionary algorithm. *Technical Report 103*, Zürich, Switzerland: Computer Engineering and Networks Laboratory (TIK), Swiss Federal Inst. of Technology (ETH), 2001.
- [20] NANBO, J., RAHMAT-SAMII, Y. Advances in Particle Swarm Optimization for antenna designs: Real-number, binary, single-objective and multi-objective implementations. *IEEE Transactions on Antennas and Propagation*, vol. 6, no. 2, p. 182 – 197, 2002.
- [21] HARRINGTON, R. F. *Time-harmonic Electromagnetic Fields*. New York (US): McGraw-Hill, 1961.
- [22] MARADEI, F. A frequency-dependent WETD formulation for dispersive materials. *IEEE Transactions on Magnetics*, 2001, vol. 37, no. 5, p. 3303 - 3306.
- [23] RAIDA, Z. et al. Communication subsystems for emerging wireless technologies. *Radioengineering*, 2012, vol. 21, no. 4, p. 1036 to 1049.
- [24] VENKATARAYALU, N. V., RAY, T. Optimum design of Yagi-Uda antennas using computational intelligence. *IEEE Transactions on Antennas and Propagation*, 2004, vol. 52, no. 7, p. 1811 - 1818.
- [25] RICHESON, P. D. NEC-2 Manual, Part III: User's Guide, available at <<http://www.nec2.org/other/nec2prt3.pdf>>.

About Authors...

Petr KADLEC was born in Brno, Czech Republic in 1985. He received his B.Sc., M.Sc. and Ph.D. degrees in Electronics and Communication in 2007, 2009 and 2012, respectively, all from the Brno University of Technology, Brno, Czech Republic. Currently, he is with the Dept. of Radio Electronics, Brno University of Technology, as a researcher. His research interests include numerical methods for electro-magnetic field computations and evolutionary algorithms for the optimization of the electro-magnetic components.

lutionary algorithms for the optimization of the electro-magnetic components.

Zbyněk RAIDA received Ing. (M.Sc.) and Dr. (Ph.D.) degrees from the Brno University of Technology in 1991 and 1994, respectively. Since 1993, he has been with the Dept. of Radio Electronics, FEEC BUT as an assistant professor (1993 to 1998), associate professor (1999 to 2003), and professor (since 2004). In 1997, he spent six months at the Laboratoire de Hyperfréquences, Université Catholique de Louvain, Belgium working on variational methods of numerical analysis of electromagnetic structures. Since 2006, he has been the head of the Dept. of Radio Electronics. Zbyněk Raida has been working together with his students and colleagues on numerical modeling and optimization of electromagnetic structures, exploitation of artificial neural networks for solving electromagnetic compatibility issues, and the design of special antennas. Prof. Raida is a member of the IEEE Microwave Theory and Techniques Society.

Jiří DŘÍNOVSKÝ was born in Litomyšl, Czech Republic, in 1979. He received the M.Sc. and Ph.D. degrees in Electronics and Communication from the Brno University of Technology, Czech Republic, in 2003 and 2007, respectively. His Ph.D. thesis was awarded by Emil Škoda Award in 2007. Since 2006 he has been an assistant professor in Electronics and Communication at the Dept. of Radio Electronics, Brno University of Technology. His research activities include selected topics of EMC, EMI measurements, and EMS testing. He is also interested in specialized problems of radiofrequency and microwave measurements. Since 2008, he has been leading the “Radioelectronic measurements” course in master degree study program and since 2009 he has been leading the Electromagnetic compatibility course in bachelor study program at the Faculty of Electrical Engineering and Communication, Brno University of Technology. He is a member of IEEE.