Self Adaptive Fuzzy Sets in Multi Objective Optimization using Genetic Algorithms

Ch. Magele, G. Fürntratt, B. Brandstätter, K.R. Richter
Institute for Fundamentals and Theory in Electrical Engineering, Technical University of Graz

***Abstract*-Optimization in electrical engineering has attracted an increasing attention over the last few years. Various strategies to solve electromagnetic optimization problems have been introduced, amongst them stochastic, deterministic and hybrid ones. Most of today's real world applications, however, involve multiple conflicting objectives which should be considered simultaneously. The aim of this paper is to introduce self adaptive fuzzy sets to treat vector optimization problems.**

I. Introduction

Stochastic methods like Evolution Strategies, Genetic Algorithms or Simulated Annealing and first and second order deterministic methods have successfully been applied to the optimization of electromagnetic devices. Today rather complicated arrangements are chosen to be optimized, while in the beginning of optimization in electrical engineering mainly simple problems like the shape of a shimming device which should produce a homogeneous field were investigated [1]. Characteristic of these so called vector optimization problems (VOP) is the appearance of a conflict between the individual solutions for each single part of the objective function. Generally no solution exists where all the different objectives can reach their individual minimum. The goal of the optimization process is then to find the optimal compromise. A common way to treat such kind of problems is the transformation of the VOP to a scalar optimization problem (SOP) by means of weighted sums of the different objectives. A more promising way, however, seems to be the introduction of a fuzzy modeling of the objective function. Both methods will be compared using the TEAM workshop problem 22 [2,3] with respect to the convergence stability and convergence speed.

II. SMES Configuration

SMES systems are devices which store significant amounts of energy in magnetic fields in a fairly simple and economical way. A double solenoidal coil configuration, which has a smaller magnetic stray field than a single solenoid if both coils are powered with currents in opposite directions, was chosen.

This arrangement is an excellent example for multiobjective optimization problems. Two demands were implemented: the magnetic stray field should be minimized without reducing the stored energy value which was required to be 180MJ. The optimization process was carried out varying the design parameters shown in Fig.1.

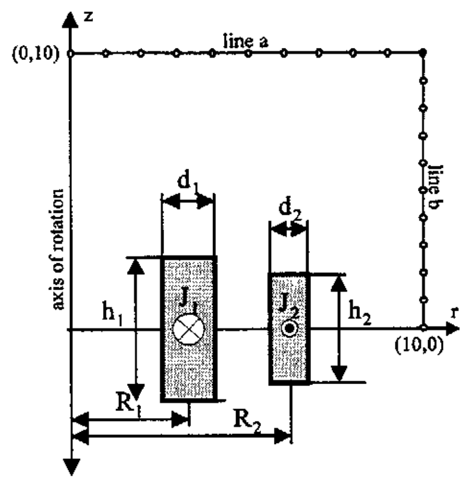

Fig.1. SMES arrangement

The box constraints of the eight degrees of freedom are given in Table I.

TABLE I - Limits of the Optimization Parameters for the Active Shielding SMES

	R_1 [m]	R_2 [m]	h_1 [m]	h_2 [m]	d_1 [m]	d_2 [m]	J_1 [MA/m^2]	J_2 [MA/m^2]
min.	1.0	1.8	0.2	0.2	0.1	0.1	10.0	-30.0
max.	4.00	5.0	3.6	3.6	0.8	0.8	30.0	-10.0

Beside these box constraints there exist some physical constraints for the design parameters too. Firstly the solenoids should not overlap each other and secondly the superconducting material should not violate the „quench" condition that sets up a relation between the current density and the maximum value of the magnetic induction |**B**| within the coils as shown in Fig. 2.

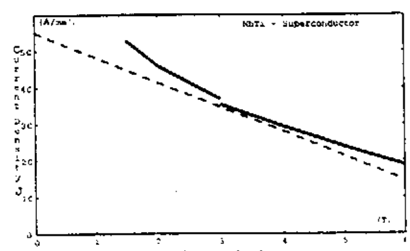

Fig. 2. Properties of the Superconductor

Manuscript received September 23rd, 1996
This work was supported in part by the „Fonds zur Förderung der Wissenschaften", Vienna under the Grant No. SFB 316

The critical curve was approximated by a straight line:

$$|\mathbf{J}| = \left(-6.4|\mathbf{B}| + 54.0\right). \quad \left[\mathrm{MA/m^2}\right] \qquad (1)$$

Configurations violating the above conditions were treated by adding a penalty term to the objective function.

III. GENETIC ALGORITHM

In nature, evolution can be taken as an example for a very efficient adaptation process of living organisms to their environment. Thus it is very promising to use nature analogous problem solving strategies to achieve an optimal adaptation of the given system [4]. The Genetic Algorithm (Fig. 3), as a result of that, is a machine-based optimization routine which connects evolutionary learning to genetic laws.

To incorporate problem specific knowledge [5] the chromosome representation chosen is natural and therefore the parameters were set up as a vector of eight floating-point numbers.

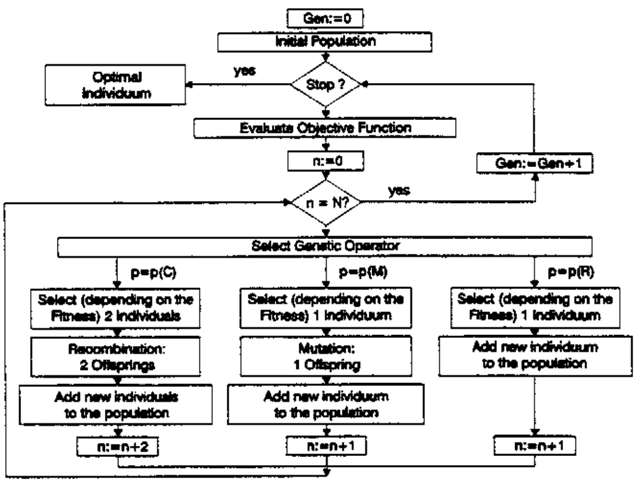

Fig. 3 Flow chart of the Genetic Algorithm

The initialization procedure creates at random a population of solutions. At first, an individual is produced randomly, and its feasibility is checked with respect to the constraints. If it is not feasible, the algorithm adds a penalty term to its fitness value. The process is repeated until the number of the individuals in the population equals the specified population size of 21.

The selection is a process to choose some individuals of high fitness for „breeding". In this paper, the commonly used roulette wheel selection is applied.

The operators used for genotypic represented chromosomes are quite different from the classical ones. However, because of the intuitive similarities, we will divide them into the standard classes, mutation and crossover.

Arithmetical crossover is defined as a linear combination of two vectors. If s_v^t and s_w^t are to be crossed, the resulting offsprings are

$$\left.\begin{aligned} s_v^{t+1} &= a s_w^t + (1-a) s_v^t \\ s_w^{t+1} &= a s_v^t + (1-a) s_w^t \end{aligned}\right\}. \qquad (2)$$

This operator uses a parameter a which is selected at random out of an interval from 0.8 to 1.4 .

The mutation tries to produce an offspring in the most promising direction. A mutation direction is given by the difference vector between the best and the second best individual. A vector with normaly distributed elements randomly selected around this difference vector is added to the parent vector under investigation.

The genetic operators are iteratively applied corresponding to their probabilities, as follows:

1. Crossover is performed with a probability of 53%.
2. Mutation is carried out with a probability of 7%.
3. The rest of the individuals are reproduced.

The algorithm stops if no improvement is obtained in the course of 500 iterations.

IV. DEFINITION OF THE OBJECTIVE FUNCTION

A. *Weighted sum of the individual objectives*

A common possibility to define the objective function *OF* of a multi objective optimization problem is the use of a weighted sum of the individual objectives.

$$OF = \frac{B_{stray}^2}{B_{norm}^2} + \frac{\left|Energy - E_{ref}\right|}{E_{ref}} \qquad (3)$$

where E_{ref}=180 MJ, B_{norm}=200 μT and B_{stray} is defined as:

$$B_{stray}^2 = \frac{\sum_{i=1}^{22}\left|\mathbf{B}_{stray_i}\right|^2}{22}. \qquad (4)$$

The value B_{norm} has to be introduced to keep the two terms of the sum at the same order of magnitude. Although this approach can yield the optimal solution with a reasonable computational effort, it needs an initial tuning phase to determine well suited weights. The convergence stability and the convergence speed are very sensitive to the choice of these weighting factors. Setting the normalization parameter B_{norm} to 1mT, which is 5 times the 200 μT only, leads to a non converging situation, as shown in Figures 4 and 5 for the evolution of the strayfield and the energy in the due course of the optimization process. It can be seen very well, that the resulting energy does not meet the required 180MJ in the least, while the strayfield reaches very low level. The bad choice of the weights degenerates the multi objective optimization problem to a single objective optimization problem in terms of the stray field.

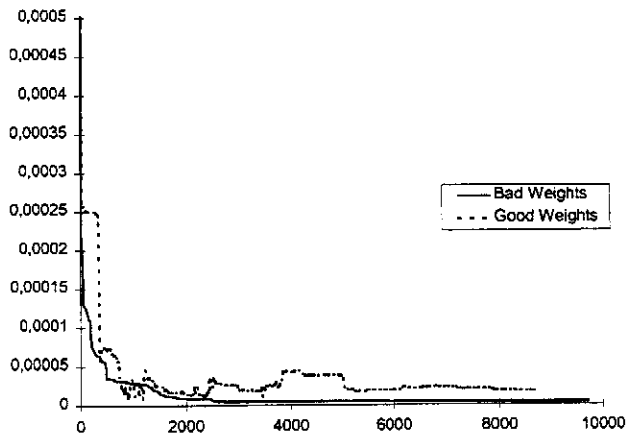

Fig. 4. Stray field versus function calls; weighted sum

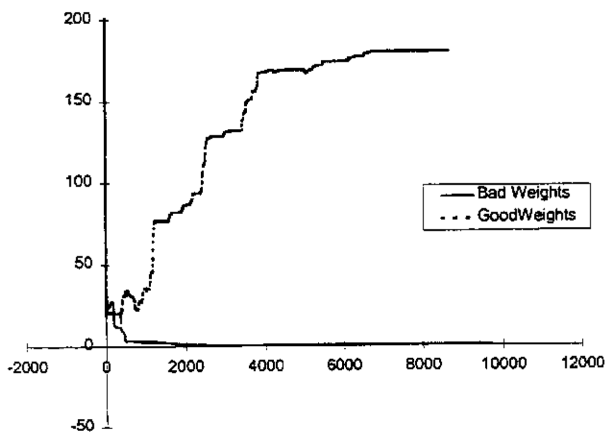

Fig. 5. Energy versus function calls; weighted sum

B. Static fuzzy definition of the objective function

Another more promising and intuitive attempt is the introduction of a fuzzy definition of the objectives [6]. Requirements like „best fit" („Energy should be sufficiently near to 180MJ") or „minimal" and „maximal" objectives („Stray field should be as small as possible") can easily be modeled by bell shaped convex fuzzy sets as indicated in Fig. 6 and 7 [7]. For bell shaped convex fuzzy sets two functions are defined (one for each side) which are given by:

$$\mu_A(x) = \begin{cases} L(e^{-l(x-m)^2}) & when \ x \le m \\ R(e^{-r(x-m)^2}) & when \ x > m \end{cases} \qquad (5)$$

The designer defines the 90% acceptance parameters $x_{left90\%}$ and $x_{right90\%}$ and can calculate the constants l and r by:

$$\left. \begin{aligned} l &= \ln(0.9) / -x_{left90\%}^2 \\ r &= \ln(0.9) / -x_{right90\%}^2 \end{aligned} \right\} . \qquad (6)$$

One important feature of describing the objectives in such a way is the implicit normalization of the values of the individual objectives.

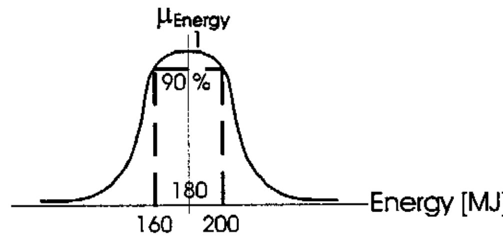

Fig. 6. Fuzzy modeling of the energy objective

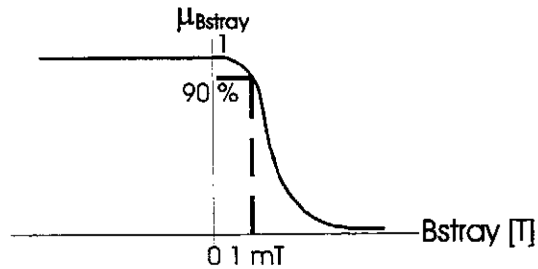

Fig. 7. Fuzzy modeling of the stray field objective

The two fuzzy sets are combined by means of a product rule to arrive at a scalar optimization problem.

Even if the acceptance parameters were changed by a factor of 100, the optimization procedure still converged to the global minimum of the problem, revealing a slightly worse convergence speed. Fig. 8 and 9 show the progress of both the strayfield and the energy in the respective cases.

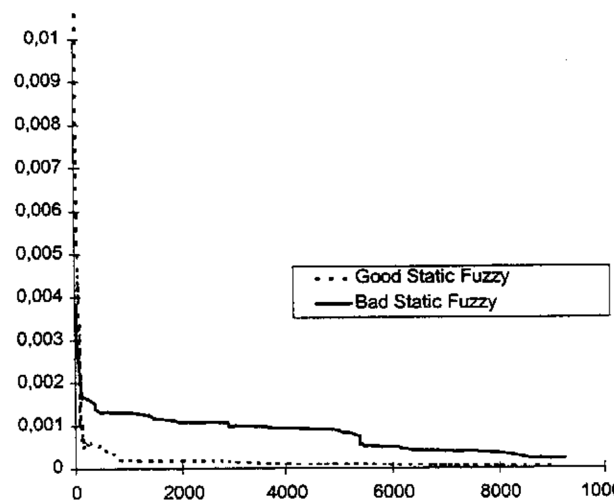

Fig. 8. Stray field versus function calls; static fuzzy sets

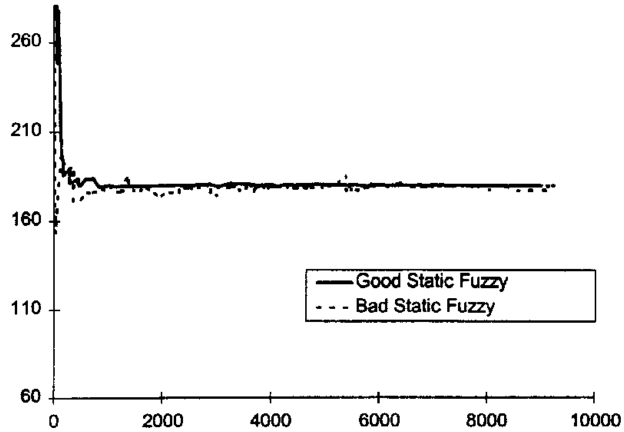

Fig. 9. Energy versus function calls; static fuzzy sets

C. *Self adaptive fuzzy definition of the objective function*

Although the convergence speed and stability of the applied optimization algorithm proved to be very insensitive to the choice of the acceptance parameters, an additional step towards something like a „plug and play" optimization algorithm was done by introducing „self adaptive" membership functions which adapt themselves to the local properties of the optimization path in the multidimensional parameter space.

From the first few function calls (e.g. the first generation of the genetic algorithm) initial acceptance parameters are evaluated. Then an adaptation algorithm was implemented to either widen or shrink the membership functions in the due course of the optimization process (Fig. 10).

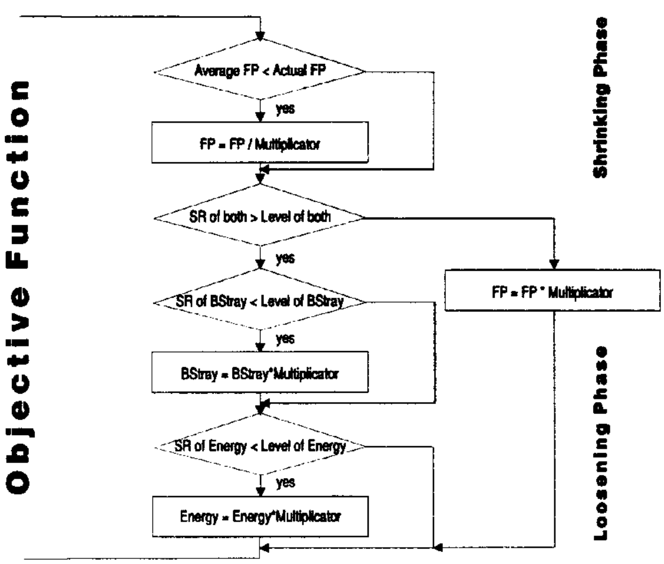

Fig.10. Flow chart of the adaptation of the fuzzy sets

The number of improvements of both the energy and the strayfield and the mean value of the product of the acceptance parameters FP are monitored during a certain number of function calls (e.g. one generation of the genetic algorithm). Then an adaptation step follows. If the mean value of the product of the acceptance parameters is smaller than the current product, the parameters are reduced by dividing them by a factor of 1.5 („shrinking phase"). If the individual success rates (SR) of the objectives are smaller than a specified value (Level), which was chosen to be 0.56, the acceptance parameters are increased by multiplying them by the above factor.

Fig. 11 and 12 show the progress of the strayfield and the energy when applying the self adaptive fuzzy sets.

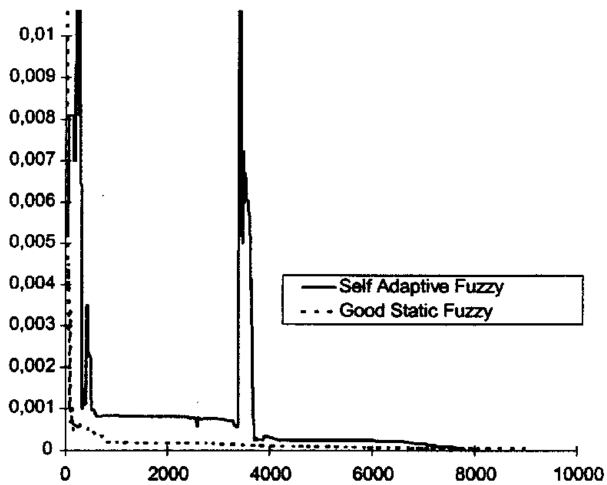

Fig. 11. Stray field versus function calls; self adaptive fuzzy sets

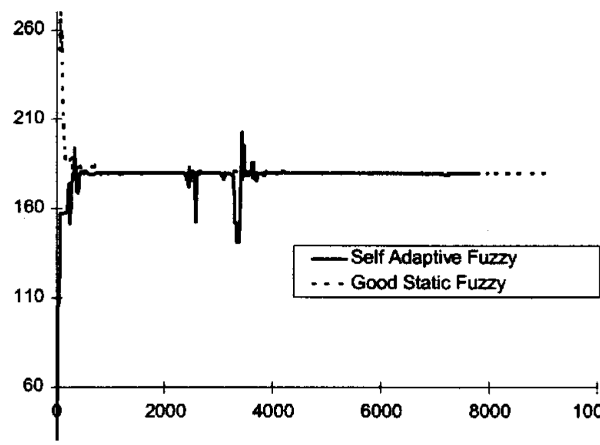

Fig. 12. Energy versus function calls; self adaptive fuzzy sets

Fig. 13 shows the values of the acceptance parameter (90% value) of the energy in the due course of the optimization process. As it can be seen in Fig. 13, this parameter reaches very high values after some 3000 function calls. Looking at that specific optimization run in more detail, it can be recognized that at this stage of the procedure the strategy ran the risk of being trapped in a local minimum. Therefore the enlarging of the acceptance parameters appears to be something like a „destabilization phase“ or „disaster“ just in time to overcome the local minimum. But it should be noted here that the „environment“ (e.g. the objective function) is changed and that the optimization strategy itself remains unchanged.

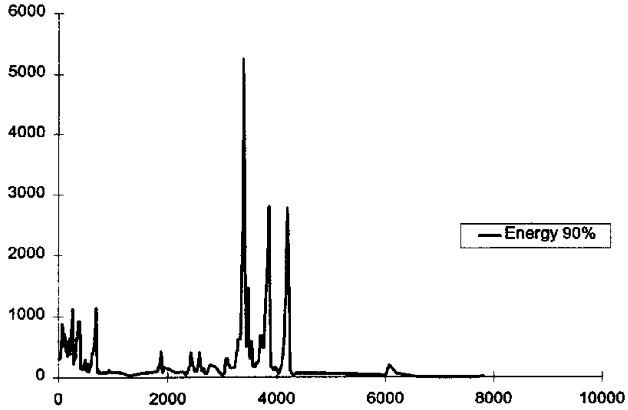

Fig. 13. Energy acceptance parameter versus function calls; self adaptive fuzzy sets

RESULTS

The different methods to define the objective function were applied to the TEAM workshop problem 22. At least ten runs where done with the weighted sum and the static and self adaptive fuzzy sets, respectively. Table. II summarizes the mean number of function calls used by the different methods to reach the global optimum. No convergence was achieved with the badly chosen weights.

TABLE II - Number of function calls

	Weighted Sum bad choice	Weighted Sum best choice	Static Fuzzy bad choice	Static Fuzzy best choice	Self adaptive Fuzzy bad choice
fu-calls	-	27112	13778	7652	7124

TABLE III - Optimal Solution

R_1 [m]	R_2 [m]	$h_1/2$ [m]	$h_2/2$ [m]	d_1 [m]	d_2 [m]	J_1 [MA/m^2]	J_2 [MA/m^2]	bstray [µT]	energy [MJ]
1.30147	1.8	1.13216	1.54212	0.57927	0.19588	16.4156	-18.925	9.306	179.99

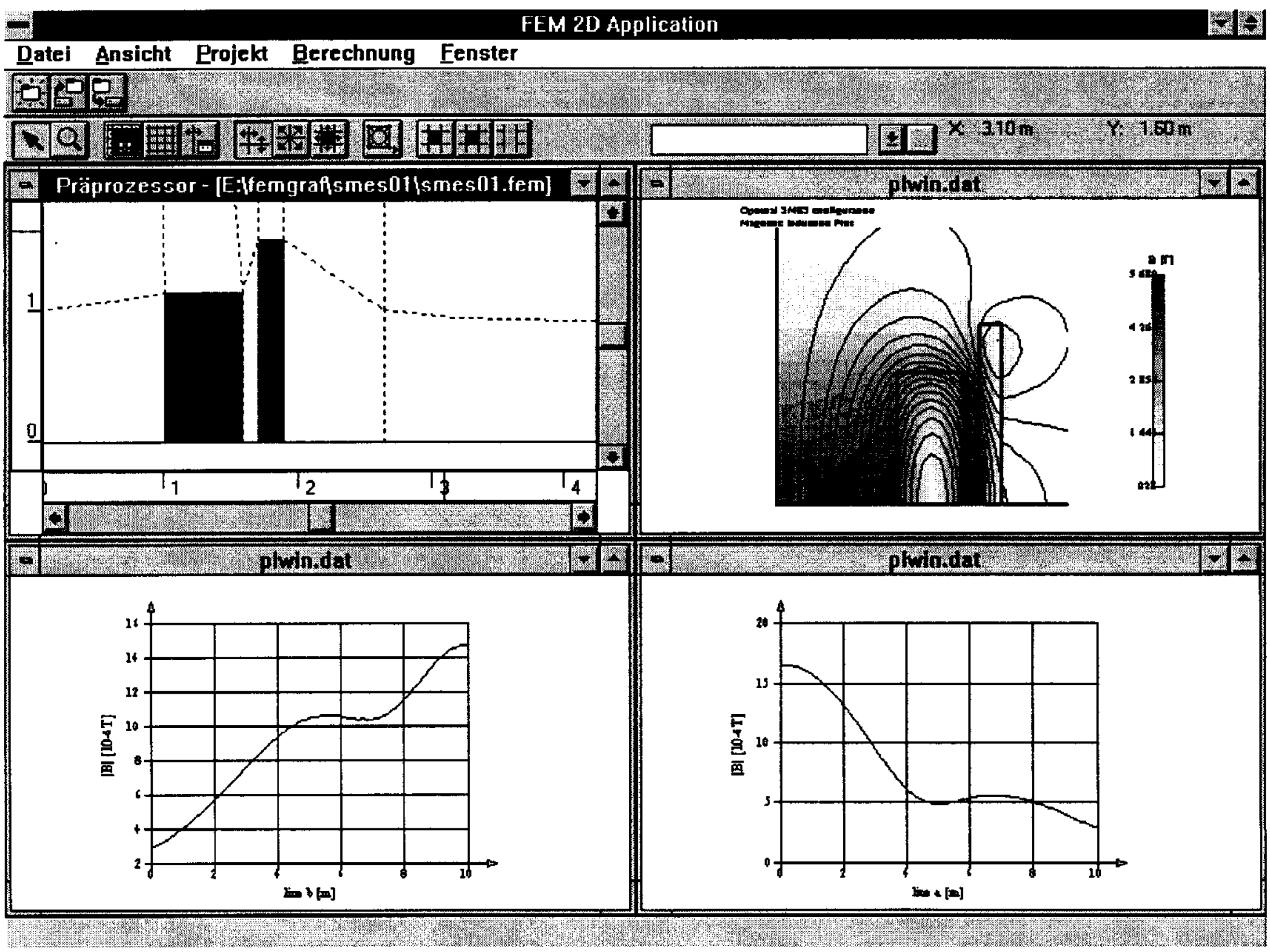

Fig. 14 Screen shot of different results of the optimal solution of the SMES configuration

As a stopping criterion of each run the bstray value should be less than 20 μT and the energy value should not differ more than 0.01MJ from the required 180 MJ.

Table. III summarizes the parameter values of the global optimum and the resulting solution of both objectives.

Fig. 14 shows a screen shot of the FEM software user interface available at the Institute for Fundamentals and Theory of Electrical Engineering at the Technical University of Graz. The upper left part presents the final geometry of the SMES configuration, the upper right part shows a plot of the absolute magnetic induction **B**, while the absolute value of the magnetic induction **B** versus the two lines a and b (Fig. 1) are plotted in the two lower pictures.

VI. Conclusion

A well chosen definition of the objective function is crucial to the convergence speed and stability of stochastic strategies used for multi objective optimization problems. The application of a weighted sum, which is commonly used, frequently needs a very cumbersome initial tuning phase to define useful weights.

Requirements like best fit objectives and minimal objectives can easily be modeled, implicitly normalized and merged by means of static fuzzy sets and suitable inference rules. The convergence stability and convergence speed are very insensitive to the actual choice of the fuzzy sets, defined by their respective acceptance parameters.

Using self adaptive fuzzy sets, which are comparable to the static ones with respect to the computational effort, new optimization problems can be handled without a priori knowledge of the specific problem, which could be termed as a „plug and play" optimization algorithm.

VII. References

[1] C.A. Magele, K. Preis, W. Renhart, R. Dyczij-Edlinger and K. R. Richter, "Higher order evolution strategies for the global optimization of electromagnetic devices", *IEEE Trans. Magn.*, Vol. 29, No. 2, Mar. 1993, pp 1775 - 1778

[2] P.Alotto, A.V. Kuntsevitch, Ch. Magele, G. Molinari, C. Paul, K. Preis, M. Repetto, K.R. Richter, "Multiobjective Optimization in Magnetostatics: A Proposal for Benchmark Problems", *IEEE Trans. on Magn*,Vol.32,pp1238-1241, 1996

[3] Ch. Magele et. all, "SMES Optimization Benchmark", Proceedings of the TEAM Workshop in the Sixth Round, March 20-21,1996 Okayama, Japan

[4] F.G. Ühler, O.A. Mohammed, C.S. Koh, "Utilizing Genetic Algorithms for the Optimal Design of ... ", *IEEE Trans. on Magn*,Vol.30, no. 6, pp 4296-4298, 1994

[5] Z. Michaliewicz: "Genetic Algorithms + Data Structures = Evolution Programs", Springer Verlag Berlin Heidelberg.

[6] M.Chiampi, C. Ragusa, M. Repetto, "Fuzzy Approach for Multiobjective Optimization in Magentics", *IEEE Trans. on Magn*,Vol.32, no. 3, pp 1234-1237, 1996

[7] R.E. Bellman, H.L. Zadeh, "Decision making in a fuzzy environment", Management Science, Vol. 17, Pg. 141-164, 1970.