MATHEMATICAL OPTIMIZATION TECHNIQUES

S. Russenschuck
CERN, 1211 Geneva 23, Switzerland

Abstract
From the beginning the ROXIE program was structured such that mathematical optimization techniques can be applied to the design of the superconducting magnets. With the concept of features it is possible to create the complex coil assemblies in 2 and 3 dimensions with only a small number of engineering data which can then be addressed as design variables of the optimization problem. In this chapter some background information on the application of mathematical optimization techniques is given.

1 Historical overview

Mathematical optimization including numerical techniques such as linear and nonlinear programming, integer programming, network flow theory and dynamic optimization has its origin in operations research developed in world war II, e.g., Morse and Kimball 1950 [45]. Most of the real-world optimization problems involve multiple conflicting objectives which should be considered simultaneously, so-called vector-optimization problems. The solution process for vector-optimization problems is threefold, based on decision-making methods, methods to treat nonlinear constraints and optimization algorithms to minimize the objective function.

Methods for decision-making, based on the optimality criterion by Pareto in 1896 [48], have been introduced and applied to a wide range of problems in economics by Marglin 1966 [42], Geoffrion 1968 [18] and Fandel 1972 [12]. The theory of nonlinear programming with constraints is based on the optimality criterion by Kuhn and Tucker, 1951 [37]. Methods for the treatment of nonlinear constraints have been developed by Zoutdendijk 1960 [70], Fiacco and McCormick 1968 [13] and Rockafellar 1973 [54] among others. Numerous optimization algorithms both using deterministic and stochastic elements have been developed in the sixties and covered in the books by Wilde 1964 [67], Rosenbrock 1966 [55], Himmelblau 1972 [25], Brent 1973 [5], and Schwefel 1977 [62]. Researchers tend to come back to genetic and evolutionary algorithms recently as they are suited for parallel processing, finding global optima, and are reported to be suitable for a large number of design variables Fogel 1994 [15], Holland 1992 [26].

Mathematical optimization techniques have been applied to computational electromagnetics already for decades. Halbach 1967 [23] introduced a method for optimizing coil arrangements and pole shapes of magnets by means of finite element (FE) field calculation. Armstrong, Fan, Simkin and Trowbridge 1982 [2] combined optimization algorithms with the volume integral method for the pole profile optimization of a H-magnet. Girdinio, Molfino, Molinari and Viviani 1983 [20] optimized a profile of an electrode. These attempts tended to be application-specific, however. Only since the late 80 th, have numerical field calculation packages for both 2d and 3d applications been placed in an optimization environment. Reasons for this delay have included constraints in computing power, problems with discontinuities and nondifferentiabilities in the objective function arising from FE meshes, accuracy of the field solution and software implementation problems. A small selection of papers can be found in the references.

The variety of methods applied shows that no general method exists to solve nonlinear optimization problems in computational electromagnetics in the same way that the simplex algorithm exists to

solve linear problems. There are many different applications in computational electromagnetics and each one requires its own particular procedure. Some optimization procedures are described in the following sections that have been proven efficient for problems in computational electromagnetics and are provided for general use in the ROXIE program.

2 Pareto-optimality

Most of the real-world optimization problems involve multiple conflicting objectives that must be mutually reconciled. Characteristic for these so-called vector-optimization problems is the appearance of an objective-conflict where the individual solutions for each single objective function differ and no solution exists where all the objectives reach their individual minimum.

A vector-optimization problem in a standardized mathematical form reads:

$$"min"\vec{F}(\vec{X}) = "min"(f_1(\vec{X}), f_2(\vec{X}), .., f_K(\vec{X})) \tag{1}$$

$\vec{F} : R^n \rightarrow R^K , g_i, h_j : R^n \rightarrow R$ subject to

$$g_i(\vec{X}) \leq 0 \qquad (i = 1, 2, ..., m) \tag{2}$$

$$h_j(\vec{X}) = 0 \qquad (j = 1, 2, ..., p) \tag{3}$$

$$x_{ll} \leq x_l \leq x_{lu} \qquad (l = 1, 2, ..., n) \tag{4}$$

with the design variable vector $\vec{X} = (x_1, x_2, ...x_n)$ and the in general nonlinear objective functions f_k arranged in the vector $\vec{F}(\vec{X})$. The x_{ll} and x_{lu} are the lower respectively upper bounds for the design variables. For the definition of the optimal solution of the vector-optimization problem we apply the optimality criterion by Pareto originally introduced for problems in economics Pareto [48], Stadler [65]. A Pareto-optimal solution $\vec{X}^*$ is given when there exists **no** solution $\vec{X}$ in the feasible domain $M = \{\vec{X} \in R^n \mid g_i(\vec{X}) \leq 0;\ h_j(\vec{X}) = 0;\ x_{ll} \leq x_l \leq x_{lu} \forall\, i = 1, ..m;\ j = 1, ..p;\ l = 1, ..n\}$ for which

$$f_k(\vec{X}) \leq f_k(\vec{X}^*) \quad \forall\, k \in [1, K] \tag{5}$$

$$f_k(\vec{X}) < f_k(\vec{X}^*) \quad for\, at\, least\, one\, k \in [1, K] \tag{6}$$

A design where the improvement of one objective causes the degradation of at least one other objective, is an element of the Pareto-optimal solution set. It is clear that this definition yields a set of solutions rather than one unique solution. Fig. 1. shows a geometric interpretation of Pareto-optimal solutions for two conflicting objectives.

3 Methods of decision-making

Applying mathematical optimization routines requires a decision-making method that guarantees a solution from the Pareto-optimal solution set. Below some methods are described that have been applied to computational electromagnetics, author's papers [56], [57]. A comprehensive overview can be found in Cohon [7].

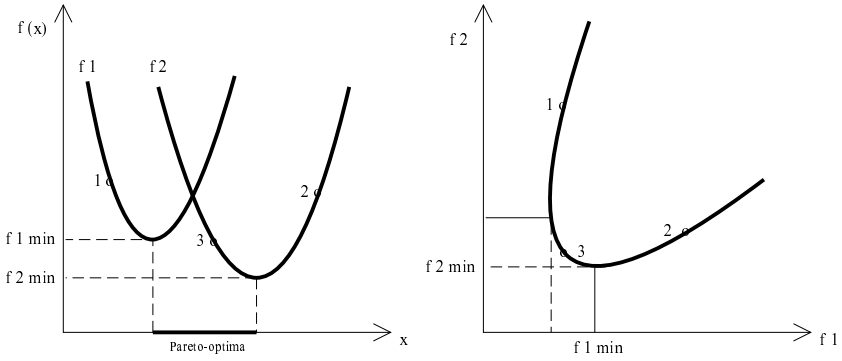

Fig. 1: Pareto-optimal solutions. Point 1 and 2 are not pareto-optimal because one objective can always be improved without deteriorating the other

3.1 Objective weighting

The objective weighting function, Kuhn and Tucker [37] is the sum of the weighted objectives and results in the minimization problem:

$$min\{u(\vec{F}(\vec{X})) = \sum_{k=1}^{K} t_k \cdot f_k(\vec{X}) \mid \vec{X} \in M\} \tag{7}$$

with the weighting factors t_k representing the users preference. For convex optimization problems where for all $k \in [1, K]$ and $\vec{X}_1, \vec{X}_2 \in M$, $0 \leq \nu \leq 1$ yields $f_k(\nu\vec{X}_1 + (1-\nu)\vec{X}_2) \leq \nu f_k(\vec{X}_1) + (1-\nu) f_k(\vec{X}_2)$ it can be proved indirectly, Fandel [12], that eq. (7) is a minimization problem with a unique Pareto-optimal solution. The problem is to find the appropriate weighting factors in particular when the objectives have different numerical values and sensitivity. Using objective weighting results therefore in an iterative solution process where a number of optimizations have to be performed with updated weighting factors.

3.2 Distance function

The problem of choosing the weighting factors appropriately also occurs when the distance function method, Charnes and Cooper [6], is applied. Most common is a least squares objective function. The f_k^* are the requirements for the optimum design. The minimization problem reads for $\vec{X} \in M$ and the norm $\parallel \vec{x} \parallel_p = (\mid x_1 \mid^p + ... + \mid x_n \mid^p)^{1/p}$

$$min \parallel \vec{F}^*(\vec{X}) - \vec{F}(\vec{X}) \parallel_2^2 = min \parallel \vec{z}(\vec{X}) \parallel_2^2 =$$

$$min \sum_{k=1}^{K} t_k (f_k^*(\vec{X}) - f_k(\vec{X}))^2. \tag{8}$$

For convex functions and for f_k^* taken as the minimal individual solutions it can be proved, in the same manner as for the objective weighting function, that (8) has an unique Pareto-optimal solution. The disadvantage of least squares objective functions with the Euclidean norm $\parallel \, . \, \parallel_2$ is the low sensitivity for residuals smaller than one. Therefore sufficiently high weighting factors t_k have to be introduced. If the absolute value norm is applied, the disadvantage is the nondifferentiable objective function in the optimum.

3.3 Constraint formulation

The problem with the weighting factors can be overcome by defining the problem in the constraint formulation, Marglin [42]. Only one of the objectives is minimized and the others are considered by constraints. The resulting optimization problem reads:

$$min\, f_i(\vec{X}) \tag{9}$$

$$f_k(\vec{X}) - r_k \leq 0 \tag{10}$$

$\forall k = 1, K;\ k \neq i$ and additional constraints, eq. (2)-(4). The r_k represent the minimum request value specified by the user for the k-th objective. Combining (10) and (2) and, because they can be treated separately, omitting the bounds for the design variables (4) yields in a vector notation $\vec{g}', \vec{c} \in R^{m+K-1}, \vec{h}', \vec{d} \in R^p$:

$$min\, f_i(\vec{X}) \tag{11}$$

$$\vec{g}'(\vec{X}) - \vec{c} \leq \vec{0} \tag{12}$$

$$\vec{h}'(\vec{X}) - \vec{d} = \vec{0} \tag{13}$$

3.4 Sensitivity analysis

The constraint formulation has the advantage that a sensitivity analysis can be performed using the necessary optimality conditions at the optimum point $\vec{X}^*$ which read, see Luenberger [38]:

$$\nabla_{\vec{X}} L = \nabla_{\vec{X}} f_i(\vec{X}^*) + \vec{\alpha} \nabla_{\vec{X}} \vec{g}'(\vec{X}^*) + \vec{\beta} \nabla_{\vec{X}} \vec{h}'(\vec{X}^*) = \vec{0} \tag{14}$$

$$\vec{g}'(\vec{X}^*) - \vec{c} = \vec{0} \tag{15}$$

$$\vec{h}'(\vec{X}^*) - \vec{d} = \vec{0} \tag{16}$$

$$\vec{\alpha} > \vec{0} \tag{17}$$

The $\vec{\alpha}, \vec{\beta}$ are the vectors of the corresponding Lagrange multipliers. Equations (14) - (17) are the Kuhn-Tucker equations. The gradient of the Lagrange function has to be zero, and the Lagrange multipliers of the active inequality constraints have to take values greater than zero, otherwise it would be possible to decrease the value of a constraint without increasing the objective function, which is of course not characteristic for an optimal point. By means of the corresponding Lagrange function L it can also be proved that (11) - (13) is a minimization problem with a unique Pareto-optimal solution if all constraints are active. A non-active constraint would be equivalent to a zero weight in the weighting function.

The Lagrange-multipliers are estimated by solving the linear equation system (14) by means of the variational problem

$$min_{\vec{\alpha},\vec{\beta}} \parallel \nabla L \parallel = min_{\vec{\alpha},\vec{\beta}} \parallel \nabla_{\vec{X}} f(\vec{X}^*) + \mathbf{A}\vec{\alpha} + \mathbf{B}\vec{\beta} \parallel \tag{18}$$

with the gradients of the constraints arranged in the matrices $\mathbf{A}$ and $\mathbf{B}$. The Lagrange multipliers are a measure of the price which has to be paid when the constraint is decreased. Mathematically this relationship is expressed by [38]

$$\nabla_{\mathbf{c}} f(\vec{X}^*) = -\vec{\alpha}, \tag{19}$$

$$\nabla_{\mathbf{d}} f(\vec{X}^*) = -\vec{\beta}. \tag{20}$$

3.5 Payoff table

A tool which provides the decision maker with a lot of information about the hidden resources of a design is the payoff-table. To create this table K individual optimization problems are solved to find the best solution for each of the K objectives (X^i being the minimizer of the problem $min f_i(\vec{X})$.)

$f_1(\vec{X}^1)$	$f_2(\vec{X}^1)$	.....	$f_K(\vec{X}^1)$	$\vec{X}^1$
$f_1(\vec{X}^2)$	$f_2(\vec{X}^2)$	.....	$f_K(\vec{X}^2)$	$\vec{X}^2$
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
$f_1(\vec{X}^K)$	$f_2(\vec{X}^K)$	.....	$f_K(\vec{X}^K)$	$\vec{X}^K$

Table 1: Payoff table for K objectives

Best compromise solutions can then be found by minimizing the distance from the in general non-feasible "perfect" solution on the diagonal of the payoff-table, cf. Fig. 1. By applying different norms e.g. the L_1, L_2 and L_∞ norm the optimal compromise solutions can be found. The payoff-table can also help to set up constraint problems with Pareto-optimal solutions (i.e. finding feasible solutions for constraint problems with active constraints).

3.6 Fuzzy sets

Considering the often imprecise nature of judgements in multiobjective programming problems, the fuzzy set, Bellman and Zadeh [3] approach looks promising. A fuzzy subset A of X is defined by its membership function

$$\mu_A : X \rightarrow [0, 1] \tag{21}$$

which assigns to each element $x \in X$ a real number $\mu_A(x)$ in the interval [0,1] where the value of $\mu_A(x)$ represents the degree of membership of x in A. The idea behind the fuzzy set theory is therefore not whether an element is in the subset or not, but if it is more or less a member of the subset. A constraint which may be expressed by, e.g., "the value has to be considerably larger than 10" could be associated with the membership function

$$\mu_A(x) = \begin{cases} 0 & x < 10 \\ 1 - \frac{1}{1+(0.1(x-10))^2} & x \geq 10 \end{cases} \tag{22}$$

Bellman and Zadeh [3] introduced three basic concepts: Fuzzy goal, fuzzy constraint and fuzzy decision. Let be $G1, G2, ..Gm$ the m fuzzy goals represented by their membership functions $\mu_{G1}, ..\mu_{Gm}$ and $C1, C2, ..Cm$ the m fuzzy constraints represented by their membership functions $\mu_{C1}, ..\mu_{Cm}$ then the fuzzy decision is the element with the maximum degree of membership of the intersection of the fuzzy goals and constraints.

$$max_x \mu_D = max_x \, min(\mu_{G1}, .., \mu_{Gm}, \mu_{C1}, .., \mu_{Cm}) \tag{23}$$

There are two drawbacks to the method, however. The first is the appropriate choice of membership functions to be associated with fuzzy statements like small, big, very big, lower, considerably lower, etc. The second is the flat function topology for areas with violated constraints where the membership function of the intersection is zero thus making the application of stochastic optimization algorithms necessary.

4 Solution methods

4.1 Bounds for design variables

As the design variables of the optimization problem can usually only be varied between upper and lower bounds, a modified objective function is applied

$$p(\vec{X}) = \begin{cases} f(\vec{X}) & \text{no bound violated} \\ f(\vec{X}^*) + r(\vec{X}) & \text{bound violated} \end{cases} \tag{24}$$

with $\vec{X}^* = (x_1, x_2, ..., x_l^*, ...x_n)$ and $x_l^* = x_{lu} \;\; if \;\; x_l > x_{lu}$ if (upper bound violated) and $x_l^* = x_{ll} \;\; if \;\; x_l < x_{ll}$ (lower bound violated). The added penalty term reads:

$$r(\vec{X}) = \sum_l r_l \begin{cases} (x_l - x_{lu})^2 & \text{if } x_l > x_{lu} \\ (x_{ll} - x_l)^2 & \text{if } x_l < x_{ll} \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

with sufficiently high penalty parameters r_l. The advantage of this procedure is that the violation of the bounds is checked before a function evaluation is carried out (violated bounds may even cause an abort of the numerical field calculation routines), and that the existing algorithms for unconstrained minimization can be applied without modifications.

4.2 Treatment of nonlinear constraints

With the constraint formulation the problem of the treatment of the nonlinear constraints arises. One method is the transformation of the constrained problem into an unconstrained problem by means of a **penalty function** method. The main idea is to add a penalty term to the objective function which depends on the degree to which the constraints are violated and vanishes if the constraints are satisfied. The optimization problem (11)-(13) transformed into the penalty function reads:

$$p(\vec{X}) = f_i(\vec{X}) \; + \sum_k p_k \cdot max^2(0., g_k(\vec{X}) - c_k) + \sum_j q_j (h_j(\vec{X}) - d_j)^2 \tag{26}$$

In order to prove the feasibility of the result the penalty factors p_k, q_j have to be chosen infinite. Large penalty factors, however, lead to ill-conditioned problems. Replacing the square term in (26) by the modulus results in the **exact penalty** method. Here the weighting factor can be finite ($p_k > \mid \alpha_k \mid, q_j > \mid \beta_j \mid$),

the α and β being the Lagrange Multipliers eq. (19,20), to guarantee feasibility of the solution. The disadvantage is the nondifferentiability of the objective function in the optimum, the same problem as when applying the distance function method with the L_1 norm.

It was therefore proposed to solve a sequence of unconstrained minimization problems with increasing penalty factors. This is called the **sequential unconstraint minimization technique, SUMT** Fiacco and McCormick [13]. However, as the objective function is discontinuous from step to step, the optimization algorithm has to be restarted with updated determination criteria. In practical applications it seems more reliable to adjust the penalty factor once and keep these values for the next runs. Using finite penalty factors includes a certain fuzziness into the solution, the constraints will not exactly be fulfilled.

If the penalty term is not added to the objective function but to the Lagrangian function, the constrained problem (11)-(13) can be solved by minimizing the **augmented Lagrangian function** Rockafellar [54]:

$$L_R = f_i(\vec{X}) + \sum_j \beta_j (h_j(\vec{X}) - d_j) + \sum_j q_j (h_j(\vec{X}) - d_j)^2 +$$

$$\frac{1}{4} \sum_k \frac{1}{p_k} \cdot max^2(0., \alpha_k + 2 \cdot p_k (g_k(\vec{X}) - c_k) - \alpha_k^2) \tag{27}$$

This leads to an iterative procedure with

1. Estimation of a suitable penalty factor and estimation of the Lagrange multiplier by minimizing (18).
2. Minimization of the augmented Lagrangian function (27) in $\vec{X}$.
3. Updating the Lagrange multipliers by $\alpha_k^{k+1} = max(0, \alpha_k^k + 2p_k(g_k(\vec{X}) - c_k))$ and $\beta_j^{k+1} = \beta_j^k + 2q_j(h_j(\vec{X}) - d_j)$ and returning to the second step.

Although this procedure leads to well-conditioned function topologies, the convergence of the procedure is very dependent on the accuracy with which step 2 is performed as step 3 assumes $\nabla L_R = 0$ which is only true if the minimum of step 2 is found.

4.3 Algorithms for the minimization of scalar unconstraint objective functions

The objective weighting function, the distance function and the fuzzy set decision allow the immediate application of an algorithm for finding the minimum value of an unconstrained objective function. It is most important to find the suitable minimization method to fit the method of decision-making and the treatment of the nonlinear constraints. The special problem in optimization in electromagnetism is the time consuming evaluation of the objective function (electromagnetic field) using the finite-element method. The advantages of stochastic algorithms are the possible treatment of problems with a high number of design variables, the possibility to treat non convex, and discrete problems and the ease of use. The deterministic algorithms converge much faster (usually around 200 function evaluations) if the search is started from a carefully chosen design (we recall that in literature, e.g. Schwefel [62], Himmelblau [25] for test examples usually more than 1000 function evaluations are carried out).

Some of the algorithms frequently used for our optimization problems in the design of superconducting magnets are described below.

4.3.1 Deterministic methods

The optimization algorithm **EXTREM by Jacob** [31] consists of one-dimensional minimizations by means of Powell extrapolations [51] in a main search direction (user supplied) and an orthogonal direction evaluated by Gram-Schmidt orthogonalization. After these one-dimensional searches have been carried out (end of a search step) the main search direction is updated by a vector pointing from the initial outline to the minimum of the search step. The user has to supply an initial step-size which is taken to $(x_{lu} - x_{ll})/10$. This user-friendly algorithm is suitable for practically all applications including unconstrained scalar functions, distance functions, penalty functions, and augmented Lagrange functions. Besides the initial step size there are no user supplied parameters which could influence the convergence rate.

The **Levenberg-Marquard algorithm** was originally developed for nonlinear regression problems using least squares objective functions. It can efficiently be applied to the minimization of the distance function. Assuming the objective function in the vector notation by $min\, z(\vec{X}) = min\, \vec{F}(\vec{X})^T \vec{F}(\vec{X})$ with the residuals f_i arranged in the vector $\vec{F}(\vec{X}) = (f_1(\vec{X}), f_2(\vec{X}), ..., f_k(\vec{X}))$ and the Jacobi Matrix $\mathbf{J}(\vec{X})$ of $z(\vec{X})$ we get

$$\nabla z(\vec{X}) = 2 \cdot \mathbf{J}(\vec{X})^T \vec{F}(\vec{X}) \tag{28}$$

$$\nabla^2 z(\vec{X}) = 2 \cdot \mathbf{J}(\vec{X})^T \mathbf{J}(\vec{X}) + 2\frac{\partial \mathbf{J}(\vec{X})}{\partial \vec{X}} \vec{F}(\vec{X}) \tag{29}$$

Using a quadratic approximation of $z(\vec{X})$ and neglecting the term $2\frac{\partial \mathbf{J}(\vec{X})}{\partial \vec{X}} \vec{F}(\vec{X})$ stepsize and direction is given by:

$$\Delta \vec{X} = -\frac{1}{2}[\mathbf{J}(\vec{X})^T \mathbf{J}(\vec{X}) + \lambda \mathbf{I}]^{-1} \cdot \mathbf{J}(\vec{X})^T \vec{F}(\vec{X}) \tag{30}$$

$\lambda \mathbf{I}$ can be regarded as an approximation for the neglected term. With a high λ the algorithm starts in a Gauss-Newton direction and λ is decreased in the optimization procedure because the neglected term gets less and less important with smaller and smaller residuals.

The **Davidon-Fletcher-Powell algorithm** uses a quadratic approximation of the objective function where the step-size and direction is given by $\Delta \vec{X} = -\mathbf{H}^{-1} \nabla f(\vec{X})$. The Hessian Matrix $\mathbf{H}$ does not have to be calculated in each point but is updated iteratively beginning with the unity Matrix $\mathbf{I}$. The derivatives of the objective function have to be approximated by differential quotients though which makes the convergence behavior dependent upon the function topology. In recent publications an efficient method for the calculation of the derivatives in FE solutions has been proposed by Gitosusastro et. al. [21] and Park et. al. [49]. The idea is as follows: The objective function can be expressed as $f(\vec{X}, \vec{A})$ with the vector potential at the nodes of the finite element mesh. Then we get for a design variable x the total derivative

$$\frac{df}{dx} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial A_1} \cdot \frac{\partial A_1}{\partial x} + \frac{\partial f}{\partial A_2} \cdot \frac{\partial A_2}{\partial x} + ... =$$

$$\frac{\partial f}{\partial x} + \nabla_{\vec{A}} f \cdot \frac{\partial \vec{A}}{\partial x} \tag{31}$$

f is an explicit function of x and $\vec{A}$ and therefore the terms $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial A_i}$ are known. Differentiating the systems equation $\{K\} \cdot \vec{A} = \vec{Q}$ yields:

$$\{K\} \cdot \frac{\partial \vec{A}}{\partial x} + \frac{\partial \{K\}}{\partial x} \cdot \vec{A} = \frac{\partial \vec{Q}}{\partial x} \tag{32}$$

As the solution for $\vec{A}$ is already known from the FE solution we have a linear equation system for $\frac{\partial \vec{A}}{\partial x}$ without another FE calculation as $\frac{\partial \{K\}}{\partial x}$ and $\frac{\partial \vec{Q}}{\partial x}$ can be computed directly from the finite element mesh.

Even if these procedures are not available in the FE package applied, the Davidon- Fletcher- Powell algorithm is very well suited to check the optimality conditions by means of a Lagrange-Multiplier estimation, minimizing eq. 14. As here the "design variables" are the Langrange - multipliers $\vec{\alpha}, \vec{\beta}$ the derivatives can be approximated with a high accuracy thus resulting in a good convergence behavior.

4.3.2 Stochastic methods

Genetic algorithms are specially suited to solve discrete problems as each trial solution is coded as a vector (chromosome) $\vec{X}$ with elements being described as genes. Holland [26] suggested that the chromosomes should be represented by binary strings. From two sets of chromosomes offspring are produced by the use of genetic operators such as crossover and mutation. Crossover means the interchanging of sections of the chromosomes between two parent chromosomes. The position where the chromosomes are split into two sections is chosen randomly. Mutation means that with a certain likelihood single bits in the chromosomes are flipped. The third operator is the random selection where the chances of selection is proportional to the individuals fitness (objective function value). Therefore, even the weakest individual has a chance of being selected. The principle and usage of this algorithm is explained in detail in the next chapter.

Evolution strategies go back to the work by Schwefel [62] Rechenberg [53] and Fogel [16]. The methods are based on the link between reproductive populations rather than genetic links. The representation of the individuals is done with floating point vectors $\vec{\nu} = (\vec{X}, \vec{\sigma})$ where $\vec{\sigma}$ is a vector of standard deviations (in accordance with biological facts that smaller changes occur more often than larger ones). From a population of μ parents (multi-membered evolution $(\mu+1)$ strategy) the offspring is created by adding a Gaussian random variable to each component of $\vec{X}_i$ with a mean zero and a standard deviation σ_i.

$$\vec{X}^{n+1} = \vec{X}^n + N(0, \vec{\sigma}) \tag{33}$$

From the offspring μ vectors $\vec{X}_{i=1,..\mu}$ that represent the lowest objective function value (for minimization) are chosen as parents of the next generation. Different to the genetic approach, least fit individuals are immediatly removed from the population. The remaining ones have all the same mating probabilities.

In the multi-membered evolution strategy we find a similarity to the crossover in genetic algorithms, here called recombination, where some elements of the design variable vector are swapped between two members of the population. An extension is the $(\mu + \lambda)$ strategy where μ elter produce λ offspring, the elter survive and compete with the offspring. If the elter are completely replaced in each generations we have the so-called (μ, λ) strategy. The problem with the evolution strategy is the choice of number of parents and offspring and the adjustment of the search step size.

The **simulated annealing** Kirkpatrick et. al. [34] simulates the slow cool-down of thermodynamic systems in order to achieve its lowest energy state. Starting from a given search point $\vec{X}$, new design variable vectors $\vec{X}_i$, are created applying random moves along each coordinate direction. Let be $\Delta f = f(\vec{X}_i) - f(\vec{X})$ then the new point is accepted if $\Delta f < 0$ else it is accepted with a probability

of $p = e^{-\Delta f/T}$ with a certain temperature T. For a high temperature basically all the new trails are excepted, whereas for T=0 only new points with lower function values are accepted not yielding a global optimum (comparable to a rapidly cooled material which does not show the crystalline state of lowest energy but glass-like intrusions). The problem here is the choice of the starting temperature and the cool-down process.

REFERENCES

[1] Appelbaum, J., Fuchs, E.F., White, J.C.: Optimization of three-phase induction motor design, IEEE Transactions on Energy Conversion, 1987

[2] Armstrong, A.G.A.M. , Fan, M.W., Simkin, J., Trowbridge, C.W.: Automated optimization of magnet design using the boundary integral method. IEEE-TMAG, 1982

[3] Bellman, R.E., Zadeh, L.A.: Decision making in a fuzzy environment, Management Science, 1970

[4] Bertsekas, D.P.: Constrained optimization and Lagrange multiplier methods, Academic Press, 1982

[5] Brent, R.P.: Algorithms for minimization without derivatives, Prentice Hall, 1973

[6] Charnes, A., Cooper, W.: Management Models and Industrial Applications of Linear Programming, Wiley, 1961

[7] Cohon, J.L.: Multiobjective Programming and Planning, Academic Press, New York, 1978

[8] Davidon, W.C.: Variance algorithm for minimization, Computer Journal, 1968

[9] Di Barba, P. Navarra, P., Savini, A., Sikora, R.: Optimum design of iron-core electromagnets. IEEE-TMAG, 1990

[10] Dughiero, F., Guarnieri, M., Lupi, S. An optimization procedure for electromagnetic confinement and levitation systems. IEEE-TMAG ,1993

[11] Dyck, D.N., Lowther, D.A.: The optimization of electromagnetic devices using sensitivity information. Proceedings of the 6th International IGTE Symposium, Graz Technical University, 1994

[12] Fandel, G.: Optimale Entscheidung bei mehrfacher Zielsetzung, Springer, Berlin 1972

[13] Fiacco, A.V., McCormick, G.P.: Sequential unconstrained minimization techniques, Wiley, 1968

[14] Fletcher, R., Powell, M.J.D.: A rapidly convergent descent method for minimization, Computer Journal, Vol. 6, 1963

[15] Fogel D.B.: An Introduction to Simulated Evolutionary Optimization, IEEE-Transactions on Neural Networks, 1994.

[16] Fogel L.J.: "Autonomous automata", Industrial Research, 1962

[17] Fletcher, R., Reeves, C.M.: Function minimization by conjugate gradients, Computer Journal, Vol. 7, 1964

[18] Geoffrion, A.M.: Proper Efficiency and the Theory of Vector Maximization, Journal of Mathematical Analysis and Applications, 1968

[19] Gill, P.E., Murray, W., Wright, M.H.: Practical optimization, Academic Press, 1981

[20] Girdinio, P., Molfino, P., Molinari, G., Viviani, A. A package for computer aided design for power electrical engineering. IEEE-TMAG, 1983

[21] Gitosusastro, S., Coulomb, J.L., Sabonnadiere, J.C. Performance derivative calculations and optimization process. IEEE-TMAG, 1989

[22] Gottvald, A.: Comparative analysis of optimization methods for magnetostatics, IEEE-TMAG, 1988

[23] Halbach, K.: A Program for Inversion of System Analysis and its Application to the Design of Magnets, Proceedings of the International Conference on Magnet Technology (MT2), The Rutherford Laboratory, 1967

[24] Hestenes, M.R.: The conjugate gradient method for solving linear systems, Proceedings of the Symposium on Applied Mathematics, Vol. VI, McGraw Hill, 1956

[25] Himmelblau, D.M.: Applied nonlinear Programming, McGraw-Hill, 1972

[26] Holland, J.H.: Gentic algorithms, Scientific American, 1992

[27] Holsinger R.F., Iselin Ch.: The CERN-POISSON Program Package User Guide, Cern Computer Center, Program Library, 1984

[28] Hooke, R., Jeeves, T.A.: Direct search solution of numerical and statistical problems, Journal of the Association for Computing Machinery, Vol. 8, 1961

[29] Hoole, S.R.H. Optimal design, inverse problems and parallel computers. IEEE-TMAG, 1991

[30] Ishiyama, A., Yokoi, T., Takamori, S., Onuki, T. An optimal design technique for MRI superconducting magnets using mathematical programming method. IEEE-TMAG, 1988

[31] Jacob, H.G.: Rechnergestützte Optimierung statischer und dynamischer Systeme, Springer, 1982.

[32] Kasper, M.: Optimization of FEM Models by Stochastic Methods, International Journal of Applied Electromagnetics in Materials 4, Elsevier, 1993

[33] Kickert, J.M.: Fuzzy theories on decision-making, Frontiers in Systems Research, Martinius Nijhoff Social Science Division, Leiden, 1978

[34] Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing, Science 220, 1983

[35] Krebs, M.: Vergleich unterschiedlicher Methoden zur Behandlung nichtlinearer Restriktionen in Optimierungsproblemen, Diplomarbeit am Institut fuer Elektrische Energiewandlung, Technische Hochschule Darmstadt, Germany, 1990.

[36] Kuester, J., Mize, J.H.: Optimization techniques with Fortran, Mc Graw-Hill, 1973

[37] Kuhn, H.W., Tucker, A.W.: Nonlinear Programming, Proceedings of the 2nd Berkeley Symposium on Mathematical Statistics and Probability, University of California, Berkeley, 1951

[38] Luenberger, D.G.: Introduction to linear and nonlinear programming, Addison-Wesley, 1973

[39] Leboucher, L., Marty, P., Alemany, A., Masse, P. An inverse method in electromagnetism applied to the optimization of inductors. IEEE-TMAG, 1992

[40] Liu, J., Freeman, E.M. Yang, X., Sheng, J. Optimization of electrode shape using the boundary element method. IEEE-TMAG, 1990

[41] Manella A., Nervi M., Repetto M.: Response Surface Method in Magnetic Optimization, International Journal of Applied Eletromagnetics in Materials 4, Elsevier, 1993

[42] Marglin, S.A.: Objectives of Water-Resource Development in Maass, A. et al. Design of Water-Resource Systems, Cambridge, 1966

[43] Marquard, D.M.: An algorithm for least squares estimation of nonlinear parameters, Society for Industrial and Applied Mathematics, 1963

[44] Mohammed, O.A.; Park, D.C.; Uler, F.G.; Ziqiang, C.: Design optimization of electromagnetic devices using artificial neural networks. IEEE Transactions of Magnetics, 1992

[45] Morse P.M., Kimball G.E.: Methods of operations research, Wiley & Sons, New York, 1950.

[46] Nelder, J.A., Mead, R.: A simplex method for function minimization, Computer Journal, Vol. 7, 1964

[47] Ogle, M.D., D'Angelo, J. Design optimization method for a ferromagnetically self-shield MR magnet. IEEE-TMAG, 1991

[48] Pareto, V.: Cours d'Economie Politique, Pouge 1896 or translation by Schwier, A.S.: Manual of Political Economy, The Macmillan Press, 1971

[49] Park, I.-h., Lee, B.-t., Hahn, S.-Y.: Design sensitivity analysis for nonlinear magnetostatic problems using finite element method, IEEE-TMAG, 1992

[50] Perin, R.: The Superconducting Magnet System for the LHC, IEEE-TMAG, 1991

[51] Powell, M.J.D.: An efficient method for finding the minimum of a function of several variables without calculating derivatives, Computer Journal, 1964

[52] Preis, K., Biro, O., Friedrich, M. Gottvald, A., Magele, C. Comparison of different optimization strategies in the design of electromagnetic devices. IEEE-TMAG, 1991

[53] Rechenberg, I.: Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution, Frommann Verlag, 1973

[54] Rockafellar, R.T.: The multiplier method of Hestenes and Powell applied to convex programming, Journal of Optimization Theory and Applications, Vol 12, 1973

[55] Rosenbrock, H.H.: An automatic method for finding the greatest or least value of a function, Computer Journal, Vol. 3, 1960

[56] Russenschuck, S.: Mathematical optimization techniques for the design of permanent magnet synchronous machines based on numerical field calculation, IEEE-TMAG, Vol-Mag 26, 1990

[57] Russenschuck, S.: Application of Lagrange - multiplier estimation to the design optimization of permanent magnet synchronous machines, IEEE-TMAG, Vol-Mag 28, 1992

[58] Russenschuck, S., Tortschanoff, T.: Optimization of the Coil of the LHC main Dipole, IEEE TMAG, Vol-Mag 29, 1993

[59] Saldanha, R.R., Coulomb, J.L., Foggia, A., Sabonnadiere, J.C. A dual method for constrained optimization design in magnetostatic problems. IEEE-TMAG, 1991

[60] Sattler, H.J.: Ersatzprobleme für Vektoroptimierungsaufgaben und ihre Anwendung in der Strukturmechanik, Fortschritt-Berichte VDI, 1982

[61] Schaefer-Jotter, M., Mueller, W.: Optimization of electrotechnical devices using a numerical laboratory, IEEE-TMAG, 1990

[62] Schwefel, H.P.: Numerische Optimierung von Computer- Modellen mittels der Evolutionsstrategie, Birkhäuser Verlag, 1977

[63] Siebold, H. , Huebner, H., Soeldner, L., Reichert, T.: Performance and results of a computer program for optimizing magnets with iron. IEEE-TMAG, 1988

[64] Simkin, J., Trowbridge, C.W.: Optimization problems in electromagnetics. IEEE-TMAG, 1991

[65] Stadler, W.: Preference optimality and application of Pareto-optimality in Marzollo, Leitmann: Multicriterion Decision Making, CISM Courses and Lectures, Springer, 1975

[66] Takahashi, N., Nakata, T., Uchiyama, N.: Optimal design method of 3-D nonlinear magnetic circuit by using magnetization integral equation method, IEEE-TMAG, 1989

[67] Wilde, D.G.: Optimum seeking methods, Prentice Hall, 1964

[68] Weeber, K., Johnson, E., Sinniah, S., Holte, K., Sabonnadiere, J.C., Hoole, S.R.H.: Design sensitivity for skin effect and minimum volume optimization of grounded and ungrounded shields. IEEE-TMAG, 1992

[69] Yokoi, T., Ebihara, D.: An optimal design technique for high speed single-sided linear induction motors using mathematical programming method. IEEE-TMAG, 1989

[70] Zoutdendijk, G.: Methods of feasible directions: a study in linear and nonlinear programming, Elsevier, 1960