SEECCM III
3rd South-East European Conference on Computational Mechanics
An ECCOMAS and IACM Special Interest Conference
M. Papadrakakis, M. Kojic, I. Tuncer (eds.)
Kos Island, Greece, 12–14 June 2013

MULTI-OBJECTIVE SHAPE DESIGN OPTIMIZATION INTO AN XFEM FRAMEWORK

Manolis Georgioudakis, Nikos D. Lagaros*, Manolis Papadrakakis

Institute of Structural Analysis & Seismic Research,
National Technical University Athens
Zografou Campus, Athens 15780, Greece
{geoem, nlagaros, mpapadra}@central.ntua.gr

Keywords: extended finite element method, fracture mechanics, crack propagation, multi-objective shape optimization, metaheuristics.

Abstract. *Almost every real world problem involves simultaneous optimization of several incommensurable and often competing objectives which constitutes a multi-objective optimization problem. In multi-objective optimization problems the optimal solution is not unique as in single-objective optimization problems. In this paper, a multi-objective shape design optimization is implemented within the context of Extended Finite Element Method (XFEM). Two objective functions dealing with maximum service life (represented by maximizing the number of fatigue cycles), and minimum weight subjected to specified minimum service life and an upper limit on volume are considered. Structural performance entities are selected as constraints and geometrical data are employed as shape design variables. Nature inspired optimization techniques have been proven to be quantitatively appealing, since they have been proved to be robust and efficient even for the most complex problems examined. These methods, also known as metaheuristics, are used for combinatorial optimization problems. Special techniques with parallel processing for solving nested XFEM propagation problem are considered, in order to reduce the total computational cost.*

1 INTRODUCTION

Gradient-based optimizers capture very fast the right path to the nearest optimum, irrespective if it is a local or a global optimum but it cannot assure that the global optimum can be found. On the other hand metaheuristics, due to their random search, are being considered more robust in terms of global convergence; they may suffer, however, from a slow rate of convergence towards the global optimum. When metaheuristics are adopted to perform the optimization, the solution of the finite element equations is of paramount importance since more than 95% of the total computing time is spent for the solution of the finite element equilibrium equations [1]. A second characteristic is that in place of a single design point metaheuristics work simultaneously with a population of design points in the space of design variables. This allows for a natural implementation of the evolution procedure in parallel computer environments.

In single-objective optimization problems the optimal solution is usually clearly defined since it is the minimum or maximum value of the objective function. This does not hold in real world problems where multiple and conflicting objectives frequently exist. Instead of a single optimal solution, there is usually a set of alternative solutions, generally denoted as the set of Pareto optimal solutions. These solutions are optimal in the wider sense since no other solution in the search space is superior to them when all objectives are considered. In the absence of preference information, none of the corresponding trade-offs can be said to be better than the others. On the other hand, the search space can be too large and too complex, which is the usual case of real world problems, hence the implementation of gradient based optimizers for this type of problems becomes even more cumber-some. Thus, efficient optimization strategies are required able to deal with the presence of multiple objectives and the complexity of the search space. Metaheuristics and in particular EA have several characteristics that are desirable for this kind of problems and most frequently outperform the deterministic optimizers such as gradient based optimization algorithms. In sizing optimization the aim is to minimize the objective function which is usually the weight or the cost of the structure under certain restrictions imposed by the design codes when the characteristics of the cross-sections of the members are under investigation.

The aim of this paper is to couple a multi-objective shape design optimization within context of Extended Finite Element Method (XFEM) [2]. To this purpose, two objective functions dealing with (a) maximum service life (by maximizing the number of fatigue cycles), and (b) minimum weight subjected to specified minimum service life and an upper limit on volume. Metaheuristic optimization methods and in particular algorithms based on evolution strategies are implemented for the solution of the problem at hand. After a brief review of XFEM procedure, the problem formulation and the optimization algorithm used is described.

2 X-FEM CONCEPTS IN BRIEF

Since its first introduced by Moës *et al.* [2], eXtended Finite Element Element Method (XFEM) has gained a significant interest of researchers and became a powerful tool to simulate crack propagation phenomena [3]. Based on the concept of *partition of unity*, X-FEM allows special local enrichment functions to be incorporated into a standard finite element approximation, while preserving the classical displacement variational settings and meshing concepts. Hence, mesh does not need to conform to the problem geometry due to discontinuous enrichment functions and remeshing techniques are not required during crack propagation.

2.1 Basic Formulation

For solid mechanics problems, the equation to be solved is usually on the displacement **u** of the body. The discretization of displacement field in order to model crack surfaces and crack tips, then reads:

$$\mathbf{u}^{h}(\mathbf{x}) = \mathbf{u}(\mathbf{x}) + \mathbf{u}^{H}(\mathbf{x}) + \mathbf{u}^{tip}(\mathbf{x}) \tag{1}$$

where the approximate displacement function $\mathbf{u}^h$ can be expressed in terms of the standard **u**, crack-split $\mathbf{u}^H$ and crack-tip $\mathbf{u}^{tip}$, or more explicitly:

$$\mathbf{u}^{h} = \underbrace{\sum_{i\in I}\mathbf{u}_i N_i}_{\text{standard part}} + \underbrace{\sum_{j\in J}\mathbf{b}_j N_j H\left(f(\mathbf{x})\right) + \sum_{k\in K} N_k \left(\sum_{l=1}^{4}\mathbf{c}_k^l F_l(\mathbf{x})\right)}_{\text{enriched part}} \tag{2}$$

In (2) the last two terms on the right part, are the terms associated to enrichment functions. The function H (Heaviside function) is the jump function and is used to introduce discontinuity in crack faces. It has the following formulation:

$$H(\xi) = \operatorname{sign}(\xi) = \begin{cases} -1 & \forall\ \xi > 0 \\ 1 & \forall\ \xi < 0 \end{cases} \tag{3}$$

Also, f(x) showing the side of the crack where **x** is located and can be the signed distance function to the crack, *N* are the standard shape functions and $\mathbf{b}_j$, $\mathbf{c}_k$ are the vectors of additional degrees of nodal freedom for modeling crack faces and crack tips respectively. I is the set of all nodes in the mesh, J is the set of nodes in the mesh whose shape function support is completely cut by the crack and K is the set of nodes enriched by the crack tip enrichment functions which are inside a specific zone around crack tip. F_l are tip enrichment functions (branch functions) used to increases the accuracy of the numerical solution around crack tip and their formulation is dependent on the nature of the problem to be solved. In this study, crack propagation problem was solved within the linear elastic fracture mechanics framework (LEFM), in which the size of the plastic zone around the crack tip is very small compared to the structure size. So, these functions are chosen based on the asymptotic behavior of the displacement field at the crack tip:

$$F_l(r,\theta) = \left\{\sqrt{r}\sin\frac{\theta}{2}, \sqrt{r}\cos\frac{\theta}{2}, \sqrt{r}\sin\theta\sin\frac{\theta}{2}, \sqrt{r}\sin\theta\cos\frac{\theta}{2},\right\} \tag{4}$$

where (r,θ) is a polar co-ordinate system with its origin at the crack tip and θ=0 tangent to the crack at its tip.

2.2 Fracture parameters calculation

The introduction of the discrete approximation (2) into the principle of virtual work leads to a system of linear equations. The stress intensity factors (SIFs) K_I and K_{II} for modes I and II, respectively, are computed using the domain form of the J-integral as described in [4]. Once SIFs are obtained, fracture parameters such as ΔN and θ_c can be easily computed from (5, 7):

$$\Delta N = \frac{\Delta\alpha}{C(\Delta K_{eq})^m} \tag{5}$$

where ΔN is the total cycles for crack to grows by length $\Delta\alpha$. Usually the crack growth $\Delta\alpha$ is very small and in this study is predetermined equal to a/10, where α is the initial crack length. C and m are material constants. For general mixed mode loading, ΔK_{eq} is given by:

$$\Delta K_{eq} = \sqrt{\Delta K_I^2 + \Delta K_{II}^2} \tag{6}$$

where $\Delta K = K_{max} - K_{min}$ is the SIF range. The direction in which the crack will propagate from its current tip, θ_c, is obtained using the maximum hoop stress criteria [2]. The angle θ_c depends on the stress intensity factors, K_I and K_{II}, and is given by:

$$\theta_c = 2\arctan\frac{1}{4}\left(\frac{K_I}{K_{II}} \pm \sqrt{\left(\frac{K_I}{K_{II}}\right)^2 + 8}\right), \quad -\pi < \theta_c < \pi \tag{7}$$

2.3 Level Set Method for modeling crack

Level Set Method (LSM) offers an elegant way of modeling discontinuities. Is a numerical technique for tracking the motion of interfaces and has been successfully applied for modeling cracks. The key point in modeling of crack and any discontinuity using level set method is to represent the discontinuity as a zero level set function. For the modeling of crack we define the level set function as a signed distance function. As the crack is a discontinuity which does not divide the domain into two distinct parts completely, rather a portion of the domain is divided, hence to fully characterize a crack we define two level set functions (i) a normal level set function φ and a tangential level set function ψ (see Figure 1). Both the two level set functions are defined as a signed distance functions. According to Figure 1, φ is the distance of points from the tangent line to the crack face (l_{sn}) and ψ is the distance from the line perpendicular to the crack at the crack tip (l_{st}). The crack is then defined as the set of the points for which $l_{sn} = 0$ and $l_{st} \leq 0$ and the crack tip is defined the point for which $l_{sn} = l_{st} = 0$. Such representation of crack is suitable for coupling LSM with XFEM, where enrichment functions are used, since it makes it also easy to obtain polar coordinates of points with respect to crack tip according to:

$$r = \sqrt{l_{st}^2 + l_{sn}^2} \qquad \theta = \arctan\left(\frac{l_{st}}{l_{sn}}\right) \tag{8}$$

Also, level sets are used for selection of enriched nodes. To determine whether a node lies above or below the crack, one simply needs to retrieve sign of ψ at that point. So, if the crack cuts through an element, then $\varphi \leq 0$ and $\psi_{min}\psi_{max} \leq 0$, where ψ_{min} and ψ_{max} are the minimum and maximum values of ψ at the nodes of this element. Through the XFEM process the new crack tip is computed and the a new crack segment is added to the current crack path taking into account only the level set function update. Thus, no need for re-meshing is required.

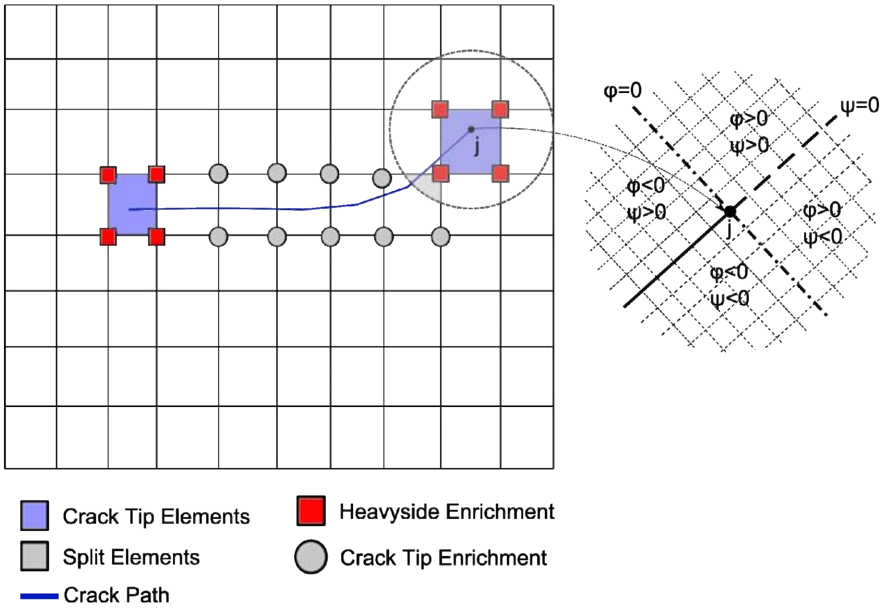


Figure 1: Nodal enrichment and level set functions in crack tip for the node j.

3 THE STRUCTURAL OPTIMIZATION PROBLEM

Structural optimization problems are characterized by various objective and constraint functions that are generally non-linear functions of the design variables. These functions are usually implicit, discontinuous and non-convex. The mathematical formulation of structural optimization problems with respect to the design variables, the objective and constraint functions depend on the type of the application. However, all optimization problems can be expressed in standard mathematical terms as a non-linear programming problem (NLP), which in general form can be stated as follows:

$$\begin{aligned} &\min && F(\mathbf{s}) \\ &\text{subject to} && g_j(\mathbf{s}) \le 0 \quad j=1,...,k \\ & && s_i^{low} \le s_i \le s_i^{up}, \quad i=1,...,n \end{aligned} \tag{9}$$

where **s** is the vector of design variables, F(**s**) is the objective function to be minimized, $g_j(\boldsymbol{s})$ are the behavioural constraints, s_i^{low} and s_i^{up} are the lower and the upper bounds of the ith design variable.

There are mainly three classes of structural optimization problems: sizing, shape and topology or layout. Initially structural optimization was focused on sizing optimization, such as optimizing cross sectional areas of truss and frame structures, or the thickness of plates and shells. The next step was to consider finding optimum boundaries of a structure, and therefore to optimize its shape. In the former case the structural domain is fixed, while in the latter case it is not fixed but it has a predefined topology. In both cases a non-optimal starting topology can lead to sub-optimal results. To overcome this deficiency structural topology optimization needs to be employed, which allows the designer to optimize the layout or the topology of a structure by detecting and removing the low-stressed material in the structure which is not used effectively.

In structural shape optimization problems the aim is to improve the performance of the structure by modifying its boundaries. This can be numerically achieved by minimizing an objective function subjected to certain constraints [5]. All functions are related to the design variables, which are some of the coordinates of the key points in the boundary of the structure. The shape optimization methodology proceeds with the following steps: (i) At the outset of the optimization, the geometry of the structure under investigation has to be defined. The boundaries of the structure are modelled using cubic B-splines that, in turn, are defined by a set of key points. Some of the coordinates of these key points will be the design variables which may or may not be independent to each other. (ii) An automatic mesh generator is used to create a valid and complete finite element model. A finite element analysis is then carried out and the displacements and stresses are evaluated. In order to increase the accuracy of the analysis an h-type adaptivity analysis may be incorporated in this stage. (iii) If a gradient-based optimizer is used then the sensitivities of the constraints and the objective function to small changes of the design variables are computed either with the finite difference, or with the semi-analytical method. (iv) The optimization problem is solved; the design variables are being optimized and the new shape of the structure is defined. If the convergence criteria for the optimization algorithm are satisfied, then the optimum solution has been found and the process is terminated, else a new geometry is defined and the whole process is repeated from step (ii).

4 MULTI-OBJECTIVE STRUCTURAL OPTIMIZATION

In practical applications of structural optimization the material weight or the structural cost rarely gives a representative measure of the performance of the structure. In fact, several conflicting and incommensurable criteria usually exist in real-life design problems that have to be dealt with simultaneously. This situation forces the designer to look for a good compromise design between the conflicting requirements. This kind of problems is called optimization problems with many objectives. The consideration of multi-objective optimization in its present sense originated towards the end of the 19th century when Pareto presented the optimality concept in economic problems with several competing criteria (Pareto, 1897). Since then, although many techniques have been developed in order to deal with multi-objective optimization problems the corresponding applications were confined strictly to mathematical functions. The first applications in the field of structural optimization with multiple objectives appeared at the end of the seventies.

4.1 Criteria and conflict

The designer looking for the optimum design of a structure is faced with the question of selecting the most suitable criteria for measuring the economy, the strength, the serviceability or any other factor that affects the performance of a structure. Any quantity that has a direct influence on the performance of the structure can be considered as a criterion. On the other hand, those quantities that must satisfy some imposed only requirements are not criteria but they can be treated as constraints. Most of the structural optimization problems are treated with one single-objective usually the weight of the structure, subjected to some strength constraints. These constraints are set as equality or inequality constraints using some upper and lower limits. When there is a difficulty in selecting these limits, then these parameters are better treated as criteria.

One important basic property in the multi-criterion formulation is the conflict that may or may not exist between the criteria. Only those quantities that are competing should be treated as independent criteria whereas the others can be combined into a single criterion to represent the whole group. The local conflict between two criteria can be defined as follows: The functions f_i and f_j are called locally collinear with no conflict at point s if there is c > 0 such that

$\nabla f_i(s)=c\nabla f_j(s)$. Otherwise, the functions are called locally conflicting at s. According to this definition any two criteria are locally conflicting at a point of the design space if their maximum improvement is achieved in different directions. The global conflict between two criteria can be defined as follows: The functions f_i and f_j are called globally conflicting in the feasible region $\mathcal{F}$ of the design space when the two optimization problems $\min_{s\in\mathcal{F}} f_i(s)$ and $\min_{s\in\mathcal{F}} f_j(s)$ have different optimal solutions.

4.2 Formulation of a multiple objective optimization problem

In formulating an optimization problem the choice of the design variables, criteria and constraints represents undoubtedly the most important decision made by the engineer. In general the mathematical formulation of a multi-objective problem includes a set of n design variables, a set of m objective functions and a set of k constraint functions and can be defined as follows:

$$\begin{aligned} &\min_{\mathbf{s}\in\mathcal{F}} && [f_1(\mathbf{s}),f_2(\mathbf{s}),\ldots,f_m(\mathbf{s})]^T \\ &\text{subject to} && g_j(\mathbf{s}) \le 0 \quad j=1,...,k \\ & && s_i \in R^d, \quad i=1,...,n \end{aligned} \tag{10}$$

where the vector $\mathbf{s} = [s_1\, s_2\, ...\, s_n]^T$ represents a design variable vector and $\mathcal{F}$ is the feasible set in design space R^n. It is defined as the set of design variables that satisfy the constraint functions g(s) in the form:

$$\mathcal{F} = \left\{ \mathbf{s} \in R^n \mid \; g(\mathbf{s}) \le 0 \right\} \tag{11}$$

Usually there exists no unique point which would give an optimum for all m criteria simultaneously. Thus the common optimality condition used in single-objective optimization must be replaced by a new concept the so called Pareto optimum: A design vector $s^* \in \mathcal{F}$ is Pareto optimal for the problem of Eq. (10) if and only if there exists no other design vector $s \in \mathcal{F}$ such that:

$$f_i(\mathbf{s}) \le \; f_i(\mathbf{s}^*) \text{ for } i = 1,2,...,m \tag{12}$$

with $f_j(s) < f_j(s^*)$ for at least one objective j. The solutions of optimization problems with multiple objectives constitute the set of the Pareto optimum solutions. The problem of Eq. (10) can be regarded, as being solved after the set of Pareto optimal solutions has been determined. In practical applications, however, the designer seeks for a unique final solution. Thus a compromise should be made among the available Pareto optimal solutions.

5 METAHEURISTICS

Nature has been solving various problems over millions or even billions of years. Only the best and robust solutions remain based on the principle of the survival of the fittest. Similarly, heuristic algorithms use the trial-and-error, learning and adaptation to solve problems. Modern metaheuristic algorithms are almost guaranteed to an efficient performance for a wide range of combinatorial optimization problems. The main aim of research in optimization and algorithm development is to design and/or choose the most suitable and efficient algorithms for a given optimization problem. In this section metaheuristics based on evolution strategies implemented for solving single-objective and multi-objective optimization problems are presented.

Several methods have been proposed in the past for treating structural multi-objective optimization problems [6, 7]. In this study two algorithms are used in order to handle the two-objective optimization problem at hand. The first one is based on the nondomination sort genetic algorithm (NSGA) developed by Deb *et al.* [9] while the second one is based on the strength Pareto evolutionary algorithm (SPEA) developed by Zitzler *et al.* [9]. The evolution strategies method has been proved very efficient for solving single objective structural optimization problems [10, 11], therefore ES method is combined with the philosophies of the first two multi-objective optimization methods (NSGA and SPEA). The resulting multi-objective optimization algorithms are denoted as NSES(μ+/,λ) and SPES(μ+/,λ).

5.1 Nondominated Sorting Evolution Strategies (NSES)

The main part of the NSES algorithm is the Fast-Nondomination-Sort procedure according to which a population is sorted in non-dominated fronts and it is based on the work by Deb *et al.* [8]. This algorithm identifies nondominated individuals in the population, at each generation, to form Pareto fronts, based on the concept of nondominance. After this step, the basic operators of ES are implemented. In the ranking procedure, the nondominated individuals in the current population are first identified. Then, these individuals are assumed to constitute the first nondominated front assigning a large dummy fitness value to each one. All these solutions have an equal reproductive potential. In order to maintain population diversity, these nondominated solutions are then shared with their dummy fitness value. Afterward, the individuals of the first front are ignored temporarily, and the rest of the population is processed in the same way to identify individuals for the second nondominated front. They are also assigned a dummy fitness value, which is a little smaller than the worst shared fitness value observed in the solutions of the first nondominated front. This process continues until the whole population is classified into nondominated fronts. Since the nondominated fronts are defined, the population is then reproduced according to the dummy fitness value.

5.2 Strength Pareto Evolution Strategies (SPES)

The basic option of SPES(μ+/,λ) algorithm was proposed in [9] as an approach that incorporates several of the desirable features of other well-known multiobjective evolutionary algorithms. SPES(μ+/,λ) implements elitism through the maintenance of an external set of best solutions found during the whole iteration loop. Elitism, when applied by an evolutionary algorithm, guarantees that the solutions with higher fitness will not be eliminated during the execution of the optimization algorithm. The nondominated solutions in the external set are used to determine the fitness of the current population (set of solutions) and also take part in the selection process for reproduction. In SPES(μ+/,λ), the fitness of a solution in the population depends on the best solutions in the external set but is independent of the number of solutions. This solution dominates, or is dominated, within the population. The most important aspects of this algorithm are the fitness assignments and the clustering procedure. In each iteration, a population of individuals $\mathbf{B}_p^{(g)}$ is obtained, and the nondominated solutions of this population are copied to $\mathbf{A}^{(g)}$ (external population). Next, the solutions of $\mathbf{A}^{(g)}$ that are dominated by other solutions are eliminated, obtaining the front of Pareto of $\mathbf{A}^{(g)}$. In SPES(μ+/,λ), the number of externally stored nondominated solutions is limited to λ. If the number of solutions of the Pareto front is greater than λ, it is necessary to reduce the external population by some means of clustering.

6 NUMERICAL EXAMPLES

For numerical purposes in order to illustrate the efficiency of coupling the aforementioned methods and for testing the efficiency of the ES-based metaheuristics employed for solving the multi-objective optimization problem, we used the example of Figure 2 that shows crack growth inside a fillet in a structural member. The detailed configuration of the problem can be found in [3]. Four design variables were selected, while Table 1 shows the upper, lower bounds as well as the increment in each step of optimization process. For the multi-objective problem the two objectives considered are: (a) maximum service life (by maximizing the number of fatigue cycles), and (b) minimum weight subjected to specified minimum service life and an upper limit on volume. Two types of constraints are considered: (i) stress and (ii) displacement constraints.

Design Variable	**Upper Bound**	**Lower Bound**	**Step**
b_1	100.0	50.0	1.0
b_2	100.0	50.0	1.0
r_3	30.0	10.0	1.0
t_4	7.0	3.0	0.5

Table 1: Design variables with upper, lower and step. All dimensions in mm.

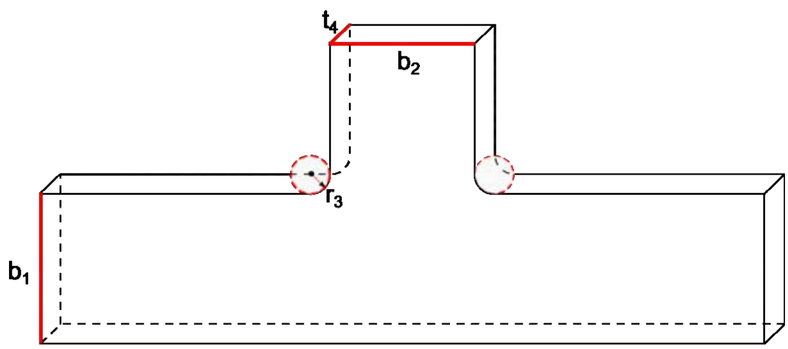
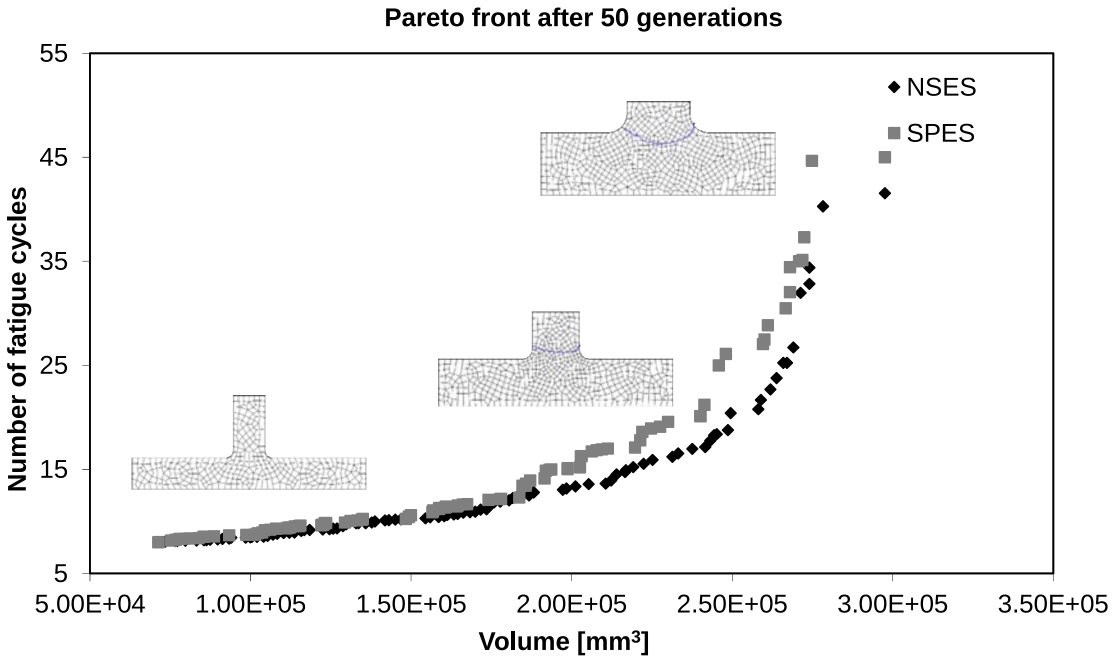
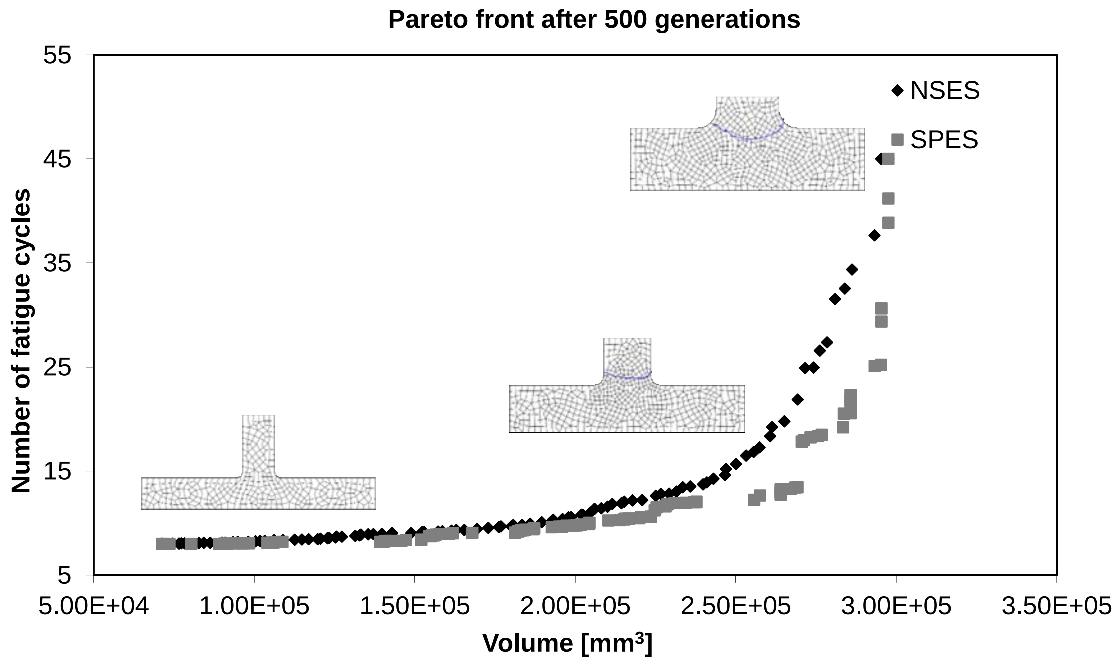


Figure 2: Selected design variables of specimen.

For the multi-objective optimization methods the simple yet effective, multiple linear segment penalty function is used in this study for handling the constraints. According to this technique if no violation is detected, then no penalty is imposed on the objective function. If any of the constraints is violated, a penalty, relative to the maximum degree of constraints' violation, is applied to the objective functions, while the optimization schemes used are NSES(10+10) and SPES(10+10). The resultant Pareto front curve is depicted in Figure 3, with the volume of the structure and the number of fatigue cycles on the horizontal and vertical axis, respectively. The Pareto front curve shows a strong conflict between the two objective functions in question. In order to assess the two methods the Pareto front curves obtained after 50 and 500 generations are compared (see Figures 3(a) and 3(b), respectively). As it can be seen a good quality Pareto front curve is obtained for all three methods. As can be seen, 50 generations are sufficient to obtain a good quality Pareto front curve for both methods adopted.



(a)



(b)

Figure 3: Pareto front curves after: (a) 50 and (b) 500 generations

7 CONCLUSIONS

In this study successful implementation of metaheuristics is presented for solving multi-objective structural shape optimization problem in an extended finite element method framework. Multi-objective evolution strategies based optimization methods in particular the nondominated sorting evolution strategies and the strength Pareto evolution strategies methods are used.

Comparing the two algorithms it can be said that evolution strategies based algorithms can be considered as efficient tools for multi-objective design optimization of structural problems such as shape optimization. In terms of computational efficiency it appears that all three methods considered require similar computational effort with approximately the same number of generation steps. In both problems, a large number of solutions need to be found and evaluated in search of the optimum one. The metaheuristics employed in this study have been found efficient in finding an optimized solution, overcoming excessive computational effort, local optima while they are capable dealing with discrete variables when needed.

REFERENCES

[1] Papadrakakis, M., Tsompanakis, Y., Lagaros, N.D. (1999) Structural shape optimization using Evolution Strategies. *Engineering Optimization Journal*, **31**, 515-540.

[2] Moës N., Dolbow J., Belytschko T, A finite element method for crack growth without remeshing, *International Journal of Numerical Methods in Engineering*, **46**:131–150, 1999.

[3] Stolarska M., Chopp D.L., Moës N. and Belytschko T., Modelling crack growth by level sets in the extended finite element method. *International Journal of Numerical Methods in Engineering*, **51**:943–960, 2001.

[4] Moran B, Shih CF. Crack tip and associated domain integrals from momentum and energy balance. *Engineering Fracture Mechanics*, **27**:615–641, 1987.

[5] Ramm E., Bletzinger K.-U., Reitinger R., Maute K., (1994). The challenge of structural optimization, In *Advances in Structural Optimization*, Topping B.H.V., Papadrakakis M. (Eds), CIVIL-COMP Press, Edinburgh, 27-52.

[6] Coello Coello, C.A. (2000). An updated survey of GA-based multi-objective optimization techniques. *ACM Computing Surveys*; **32**(2): 109-143.

[7] Marler, R.T., Arora, J.S. (2004). Survey of multi-objective optimization methods for engineering, *Structural and Multidisciplinary Optimization*; **26**(6): 369-395.

[8] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*; **6**(2): 182-197.

[9] Zitzler E., Laumanns M., Thiele L. (2001). SPEA 2: improving the strength Pareto evolutionary algorithm. In: Giannakoglou K, Tsahalis D, Periaux J, Papailou P, Fogarty T. (Eds.), EUROGEN 2001, evolutionary methods for design, optimization and control with applications to industrial problems, Athens, Greece, 95-100.

[10] Bäck, T., Schwefel, H-P. (1993). An overview of evolutionary algorithms for parameter optimization, *Journal of Evolutionary Computation*; **1**(1): 1-23.

[11] Lagaros, N.D., Fragiadakis, M., Papadrakakis, M. (2004). Optimum design of shell structures with stiffening beams. *AIAA Journal*; **42**(1): 175-184.