Constrained Globalized Nelder–Mead Method for Simultaneous Structural and Manufacturing Optimization of a Composite Bracket

HOSSEIN GHIASI,* DAMIANO PASINI AND LARRY LESSARD

McGill University, Department of Mechanical Eng., Macdonald Eng. Bldg
817 Sherbrooke St. West, Montreal, QC, Canada, H3A 2K6

ABSTRACT: The optimized design of composite structures is a difficult task. It requires optimizing simultaneously both structural and manufacturing objectives. The objectives do not have closed form solutions and have multiple local optima that calls for a global search. This paper improves the global search method called GBNM [1], which is based on several restarts of the Nelder–Mead method. Two issues are addressed here. First, the restart procedure is improved by using a one-dimensional probability function and a weighted selection procedure. Second, nonlinear constraints are included by projecting the infeasible points onto the nonlinear constraints. The improved procedure is applied to four mathematical test functions. Numerical results show the proposed approach is more efficient in terms of computational time and probability of finding the global minimum. The improved GBNM is then applied to the simultaneous structural and manufacturing design of a Z-shaped composite bracket. The results are compared to those obtained with the genetic algorithm.

KEY WORDS: composite materials, global optimization, Nelder–Mead method, simultaneous optimization.

INTRODUCTION

COMPOSITE MATERIALS OPEN a new window in engineering by providing excellent mechanical properties. However, this feature is accompanied by the complexity of the design problem, especially when the manufacturing aspects are involved in the structural design. The large number of design variables including continuous and discrete variables, lack of closed form solutions, the difficulties in simulation and analysis, as well as multiplicity of the local optima make the design and optimization of composite structures a difficult task. Lack of closed form solutions rules out the use of gradient-based optimization methods, because no gradient information is available. Additionally, multiplicity of local optima calls for a global optimization.

High computational cost is the main drawback to a global optimization. Several research papers are devoted to speeding up a global search, either by adding problem specific knowledge to the search, or by embedding an efficient local algorithm into a global one. One of these attempts by Luersen et al. [1] is devoted to combine the Nelder–Mead method (i.e., a local optimizer) to a random search (i.e., the simplest global optimizer).

The Nelder–Mead (N–M) sequential optimization method, proposed in 1965 [2], is the most popular direct method for local optimization of unconstrained problems. It has been shown to be effective in practice by producing a rapid initial decrease in function values [3]. Recently, this popular algorithm has been used in combination with global search algorithms such as random search [4], genetic algorithm [5], and Tabu search [6].

A common scenario in engineering design is that the global optimum should be found with a limited number of analyses. However, it is uncertain if it is possible to perform as many analyses as needed for an evolutionary algorithm to converge. In such a case, a local-global search that can be terminated in a short time is the safest strategy [1]. Luersen et al. [1] introduced such a local-global search based on several restarts of the N–M method. The procedure, called globalized bounded Nelder–Mead method (GBNM), repeatedly restarts a local search by the N–M method using a probability function. The probability function keeps a memory of past local searches and pushes the local search to the regions far from already-known solutions.

This article extends efforts by Luersen to make such a local-global search more efficient and to apply it to simultaneous structural and manufacturing design of a composite part. It is shown that the restart procedure used by Luersen is computationally time consuming and is not always successful in finding the optimum solution. Here, a new restart procedure is introduced to improve the probability of finding the global minimum and to reduce the computational time. In addition, in this article, a backtracking procedure is used to incorporate nonlinear constraints into the design problem.

The improvements on the restart procedure and the nonlinear constraint handling method are tested on four mathematical functions with 12 variables and different number of local minima. The results show that the new restart procedure is more efficient than Luersen restart procedure, both in terms of probability of finding the global minimum and of computational time. Then the improved GBNM is applied to simultaneous structural and manufacturing design of a composite bracket.

This article is divided into two main parts: (a) improvement and test of the optimization procedure; (b) application of the improved method to simultaneous structural and manufacturing optimization of a Z-shaped composite bracket.

OPTIMIZATION PROCEDURE

A global search can be performed by repeatedly restarting a local optimizer. To avoid finding the same local optima, the new initial points should be different and preferably far from previous initial points and already known local solutions. Luersen introduced a probabilistic restart procedure that gives points far from previous local optima and previous initial points more chance to be selected as initial point for the next local search. The restart procedure is shown in Figure 1.

The N–M method compares values of the objective function at a set of $n+1$ points called a *simplex* where n is the number of design variables. Simplex vertices are changed
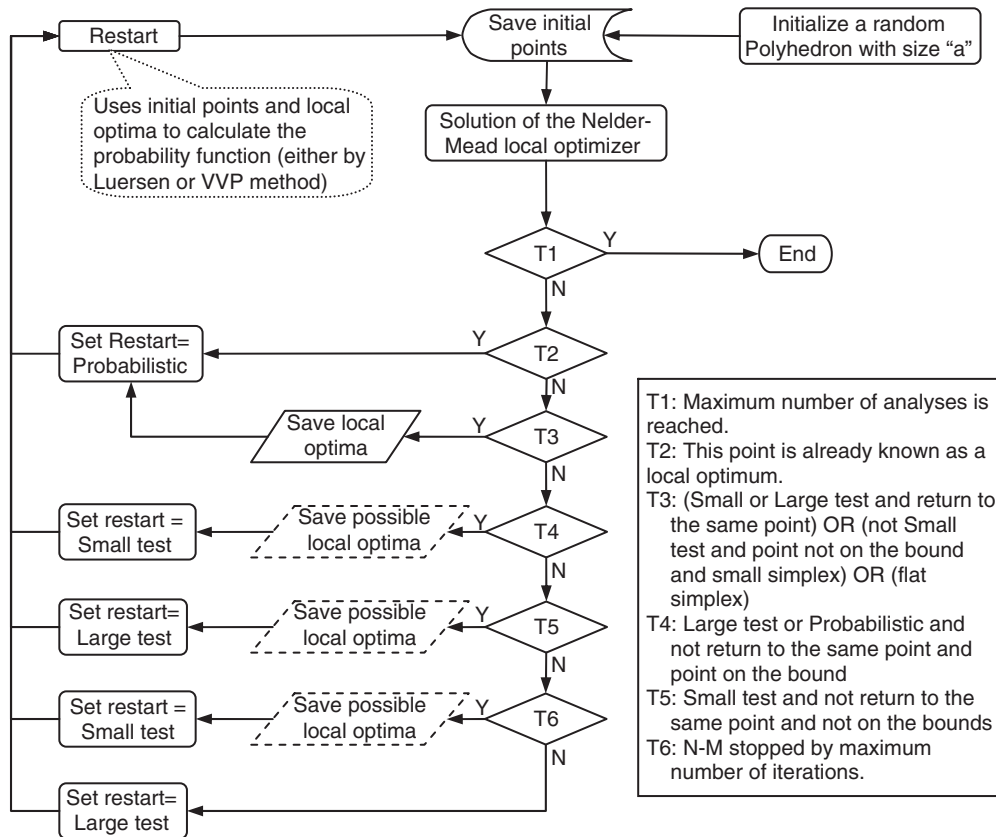
Figure 1. *Restart and convergence tests linking in GBNM.*

through reflection, expansion, contraction, or shrinkage. The process is continued until the simplex converges to a local optimum. Thus, the local optimum found is dependent on the initial simplex [7].

The following sections propose two improvements to the Luersen's GBNM. The first deals with the restart procedure. The second addresses the issue of how to include nonlinear constraints in the optimization algorithm.

Improved Restart Procedure

Luersen used a multi-dimensional probability (MDP) density to assign the sampling probability of a point. Luersen restart procedure selects N_r points randomly, and takes the point where the solution has the maximum probability to occur. The probability distribution achieved by this procedure is not equivalent to a normal distribution. It is strongly dependent on the number of points (N_r) and is independent of the probability density. The variance of the MDP does not affect the results. Luersen restart is computationally expensive because of the computational time needed for MDP.

Here, we propose an adaptive probability density to replace the MDP, called the variable variance probability (VVP). The new probability function is based on the minimum distance to the points already sampled and represented as:

$$\phi(x) = \frac{1}{\sqrt{2\pi}\sigma}(1 - e^{-(d_{\min}^2/2\sigma^2)}) \tag{1}$$

$$d_{\min} = \min_{i=1,\,...,\,m}\left\{ d_i = \sqrt{\sum_{k=1}^{n}\left(\frac{x_{k,\,i} - x_k}{x_{ku} - x_{kl}}\right)^2} \right\} \tag{2}$$

where $\phi(x)$ is the sampling probability of a point x, n is the number of design variables, x_i is a point previously sampled, and m is the number of points already sampled. Length d_i is the non-dimensional distance between point x and point x_i. The variance of the normal probability density, updated in each restart, is given by:

$$\sigma = \frac{1}{3\sqrt[n]{m}}. \tag{3}$$

The variance is gradually decreased when the number of sampled points is increased.

Equation (3) provides points located in the one-third of the middle of the line connecting two previously sampled points about 65% chance of selection. This property will be preserved all along the restart process. In contrast, the MDP tends to a uniform distribution when the number of sampled points is increased. Figure 2 compares the probability of sampling a new point using Luersen and VVP restarts. When the number of previously sampled points are increased, Luersen restart gives almost no chance of selection to the points between these points (Figure 2(c)), but it is not the case with VVP restart (Figure 2(d)). Thus, no part of the design field is ignored with VVP. The numerical results show the VVP is more efficient when the objective functions have several local optima clustered around the global one.

The restart procedure is not only to assign a probability density, but also includes a selection procedure to pick a new point based on assigned probability. The VVP restart uses a selection procedure different than that of Luersen restart. N_r points are randomly selected to create a selection pool, which is a set of points whereby each has a number of copies proportional to its probability value. A new point is randomly selected from this pool. In this procedure, the probability of sampling a new point is not affected by the number of selected points, N_r.

Numerous probability functions exist; therefore another probability function could work better, but by keeping the normal distribution and changing N_r we do not expect considerable improvement.

Non-linear Constraints

N–M method is originally introduced for unconstrained optimization, but the variables in an engineering problem are usually constrained not only by upper and lower bounds (i.e., box constraints) but also by nonlinear constraints. GBNM uses a projection
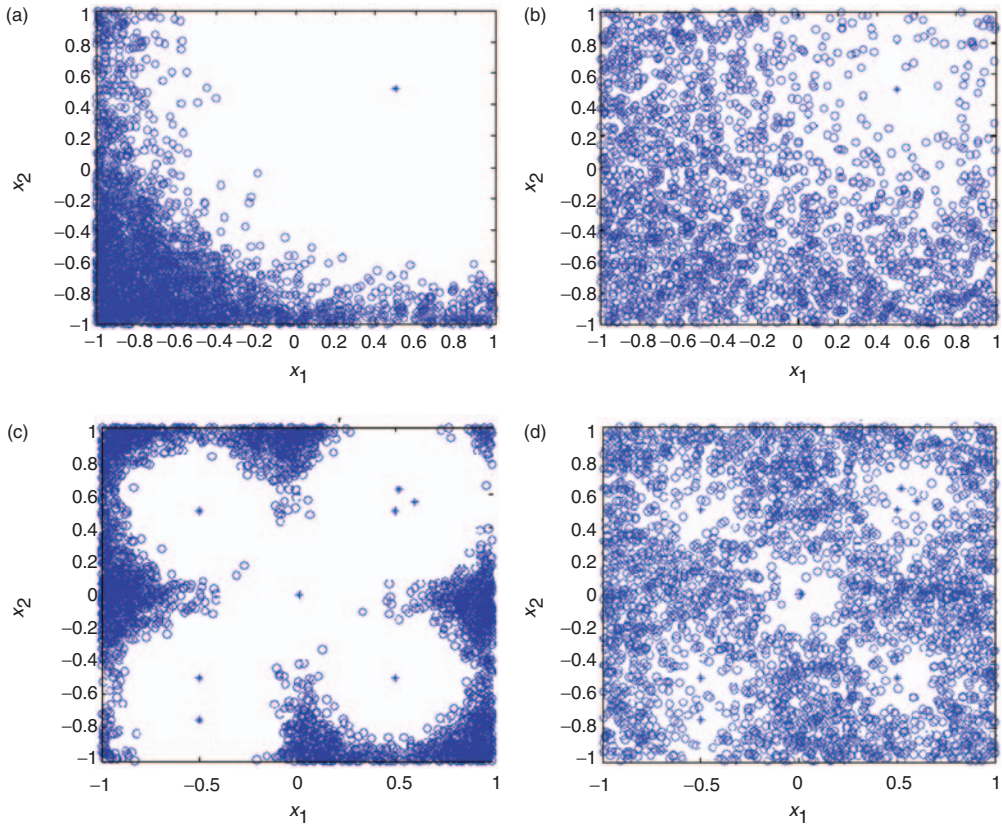
Figure 2. *Sampling 1000 points in a 2D design space using the Luersen ($N_r = 10$) and the VVP restarts (a) Luersen restart with one point (b) VVP restart with one point; (c) Luersen restart with eight points (d) VVP restart with eight points.*

procedure on the box constrained variables. Projection of variables is mathematically specified by:

$$x_i = \begin{cases} x_i^{\text{lower bount}} & \text{if } x_i < x_i^{\text{lower Bound}} \\ x_i^{\text{upper bound}} & \text{if } x_i > x_i^{\text{upper bound}} \end{cases}. \tag{4}$$

GBNM as used by Luersen cannot deal with nonlinear constraints; however, the variables in a composite optimization problem are often constrained by nonlinear functions, e.g., strength constraints. Identical to the projection of box-constrained variables, a projection procedure is used for nonlinear constraints. The projection of nonlinear constraint includes a backtracking procedure illustrated in Figure 3. When a new point generated by reflection, contraction, or expansion violates one of the nonlinear constraints, the new point is moved toward the original feasible point such that the distance between these two points is reduced by a factor of $\alpha \in (0, 1)$. The procedure is terminated when a feasible point is found or a predetermined number of trials is reached. If the procedure does not find a new feasible point in a predetermined
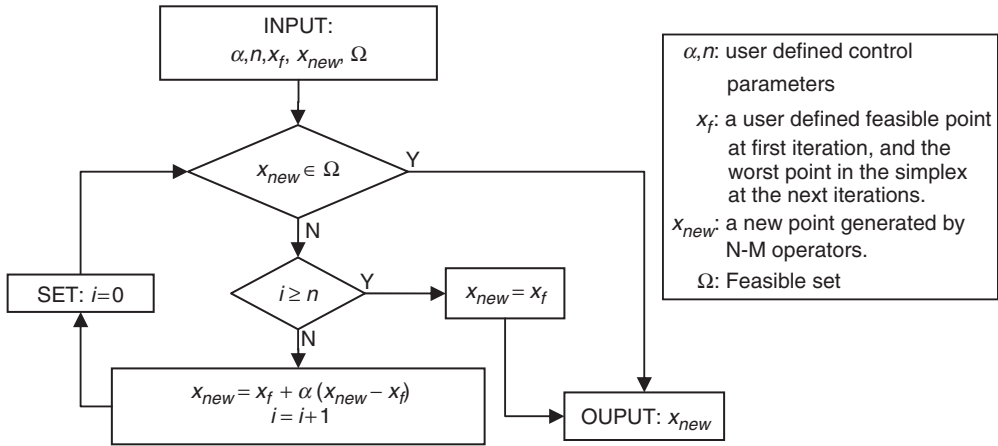
Figure 3. Backtracking procedure projects an infeasible point to the corresponding nonlinear constraint.

number of trials, the original point is kept and the new point is dismissed. This procedure guarantees the feasibility of the final solution.

Mathematical Test Functions

Four mathematical test functions are used in this section to assess the improvement suggested in the optimization procedure. The test functions are of two types. The first are those having a periodic term and, as a result, having many local optima all over the field and clustered around the global one. The second are those that do not have an alternating term. In this case, the number of local minima is smaller and arbitrarily distributed in the design space.

FUNCTIONS WITH PERIODIC TERM
Griewank's Test Function
The first test function is the Grienwank's test function [1] with n design variables and several local minima. The global minimum of the function is -1 achieved at $x_i = 0$, $i = 1, \ldots, n$ and is surrounded by several local minima. Figure 4 shows this function in two dimensions. The function is defined by:

$$A_1(x) = \sum_{i=1}^{n} \frac{x_i^2}{400n} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right); \quad x_i \in [-1000, 1000]; \quad i = 1, \ldots, n. \tag{5}$$

Ackley's Test Function
The second test function is the Ackley's test function [8] with n variables and several local optima described by:

$$A_2(x) = -20\, e^{-1/5\sqrt{1/n \sum_{i=1}^{n} x_i^2}} - e^{-1/n \sum_{i=1}^{n} \cos(2\pi x_i)}, \quad x_i \in [-15, 30], \tag{6}$$
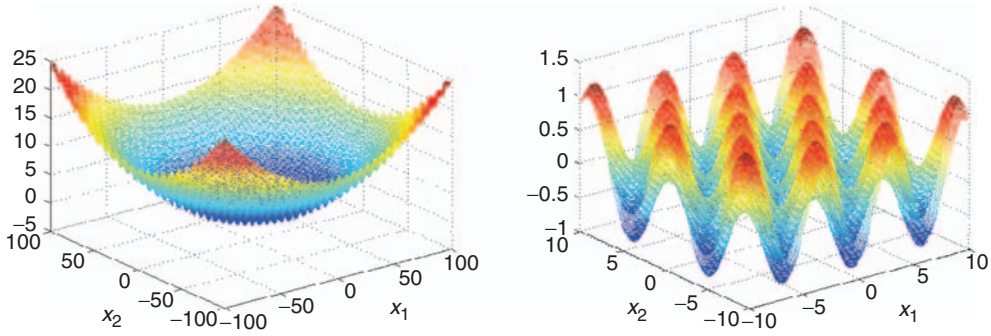$$i = 1, \ldots, n.$$

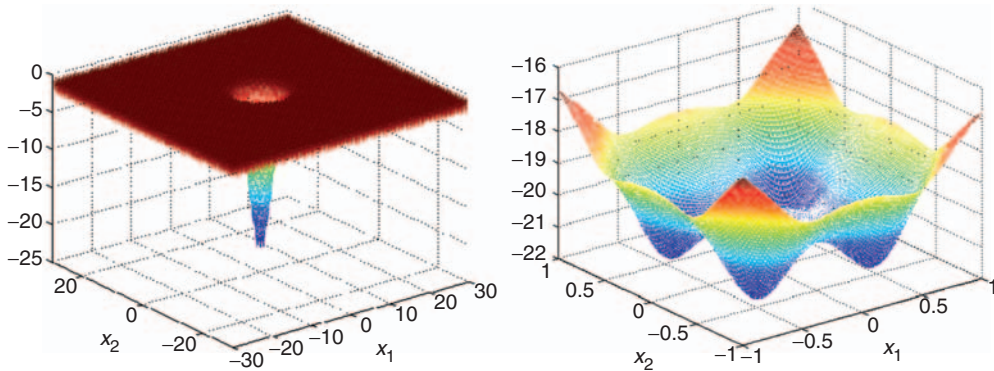**Figure 4.** Griewank's test function with two design variables.

Figure 5. Ackley's test function with two design variables.

Ackley's test function in two dimensions is depicted in Figure 5. It has several local and n^2 global minima. The global minimum of the function is -21.8072.

FUNCTIONS WITH NON-ALTERNATING TERMS
Six-hump Camelback

Six-hump camelback test function [1] with two variables is illustrated in Figure 6. It has six local minima and is defined by:

$$cb(x) = 4x_1^2 - 2.1x_1^4 + \tfrac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4, \quad x_1, x_2 \in [-2, 2]. \tag{7}$$

The global minimum of the function is -1.03163 obtained at $x = (0.0898, -0.7126)$ and $x = (-0.0898, -0.7126)$

To obtain a test function with *2n* variables and *6ⁿ* local minima we consider:

$$B_1(x) = \sum_{i=1}^{n} cb(x_{2i-1}, x_{2i}), \quad x_i \in [-2, 2], \quad i = 1, \ldots, n. \tag{8}$$
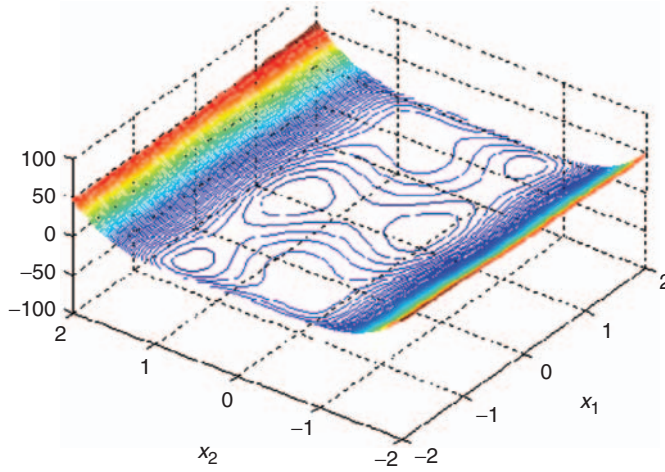
Figure 6. *Six-hump camelback test function with two design variables.*

Combined Test Function

The next test function is a combination of Hartman test functions and Fletcher and Powell's helical valley and is given by:

$$F(x) = H_6(x_1, \ldots, x_6) + H_3(x_7, \ldots, x_9) + HV(x_{10}, \ldots, x_{12}), \quad x_i \in (0, 1);$$
$$i = 1, \ldots, 12 \tag{9}$$

where H_6 and H_3 are Hartman test functions with six and three variables respectively having six and four local minima [9], and HV is the Fletcher and Powell's helical valley [2] with three variables and one local minimum located at the end of a spiral valley. This function has 24 local optima and its global minimum is -7.18509 located on the boundary of the feasible region at: $x = (0.20, 0.15, 0.48, 0.28, 0.31, 0.66, 0.11, 0.56, 0.85, 1, 0, 0)$.

Optimization Set-up

The control parameters in optimization procedure are set to the recommended values in literature. Reflection, expansion, and contraction coefficients are taken to be 1, 2, and 0.5, respectively, as used by Nelder and Mead [2]. A shrink coefficient of 0.9 is used as recommended by [7] and [10]. The N–M method is terminated when the simplex is small or flat, or when the maximum number of iterations is reached. A simplex is small when:

$$\max_{i=1,\ldots,n} \left(|\frac{x_i^{k+1} - x_i^k}{x_i^u - x_i^l}| \right) \leq \varepsilon_1 \tag{10}$$

where k is the number of iterations, subscripts u and l represent the upper and lower bound on variables x_i, and ε_1 is a predetermined small number. Similarly, the simplex is flat when:

$$\left| f_H - f_L \right| \leq \varepsilon_2 \tag{11}$$

where f_H and f_L are the highest and lowest function values at the current simplex and ε_2 is a given small number.

Initial simplex of the N–M is a polyhedron with the edge size of 20% of the design space. The two small-test and large-tests restarts (see Figure 1), used to improve the convergence, restart the algorithm from the best point of the current simplex with a polyhedron of size 2% and 10% of the domain size. N_r in selection procedure is set to 10, and α and n in the backtracking procedure are set to be 0.9 and 10, respectively. Also, genetic algorithm with the population size of 20 and crossover and mutation fraction of 0.8 and 0.01 is used for comparison.

Numerical Results

RESTART PROCEDURES

Four test functions described earlier with 12 variables are minimized using GBNM with random restart, Luersen restart, and VVP restart. The optimization is conducted for 1000, 5000, and 10,000 iterations. A pool of solutions is needed to compare the results. Thus, each function is minimized 100 times and the average and the standard deviation of the solutions are compared. Results are compared to those obtained with the Genetic Algorithm (GA).

Figures 7–10 compare the average minimum found by each algorithm in 100 runs. The figures plot the average runtime for each algorithm. The performance of different restart procedures depends on the function type and number of iterations.

The random restart is not able to find an adequate minimum for a small number of iterations (i.e., 1000) and a function with many local minima (Figures 7 and 8); however if the objective function has only a few local minima, the performance of all restart procedures is similar (Figures 9 and 10).

After about 5000 iterations, all the restart procedures have the same performance, except on Ackley's test function (Figure 8). Ackley's test function has many local optima. Therefore, the local optimizer traps in a local optimum close to the initial simplex. The global minima of Ackley's test function are located at the end of a narrow hole (see Figure 5). If only a random restart is performed, the probability of having an initial simplex located sufficiently close to this hole (i.e., $x_i \in (-5, 5)$) is less than 10^{-8}. Thus it is rare that the random restart reaches the global minimum of this function.

With Luersen restart, the probability of having an initial simplex sufficiently close to the global minimum is even smaller than random restart, because Luersen restart pushes the new initial points away from previous ones (see Figure 2). Luersen restart finds the global minimum of Ackley's test function 8 times and random restart finds it 14 times out of 100 runs. With VVP restart, points located between known local minima still have the chance to be selected, whereas with Luersen restart if a local optimum around the global one is found, the probability of finding a new point in that region is almost zero (see Figure 2). Thus the VVP restart performs better than both Luersen and random restarts (Figure 8) by finding the global minimum 95 times out of 100 runs. Figure 11 shows the average probability of finding the global minimum of all the given test functions, after 100 runs. It shows that, in general, the VVP has more probability for finding the global minimum than the two other restart procedures and the GA. Luersen and random restart are almost the same in this respect, and both are better than GA.

In terms of computational time, Luersen restart is about 5–10 times slower than the random and the VVP restarts, because it uses a multidimensional probability function that
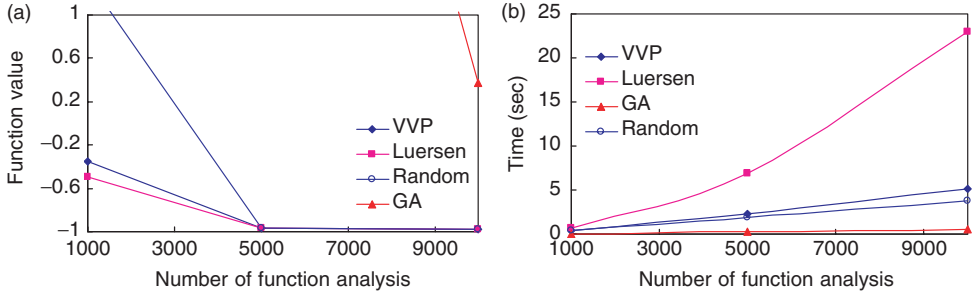
Figure 7. *Comparison of GBNM with Luersen, VVP, and random restarts and GA on Griewank's function (A₁) with 12 variables, 100 run; (a) Minimum function value; (b) Computational time.*
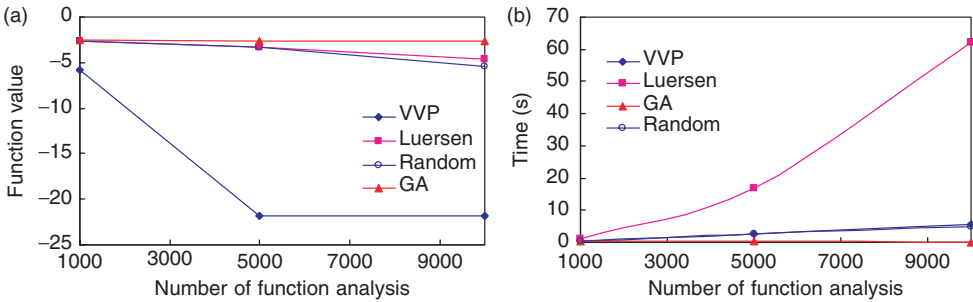

Figure 8. *Comparison of GBNM with Luersen, VVP, and random restarts and GA on Ackley's function (A₂) with 12 variables, 100 runs; (a) Minimum function value; (b) Computational time.*
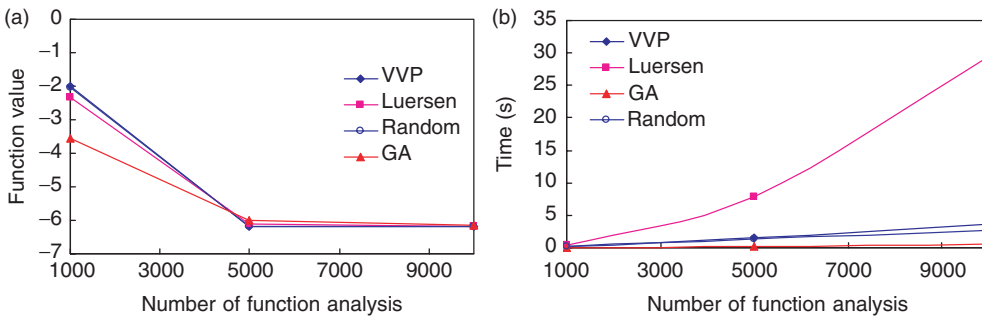

Figure 9. *Comparison of GBNM with Luersen, VVP, and random restarts and GA on B₁ test function with 12 variables, 100 runs; (a) Minimum function value; (b) Computational time.*

needs matrix calculation. Computational time of VVP and random restarts differs only by 30% and increases linearly with the number of iterations. Since the time for Luersen restart is increased exponentially with the number of restarts (and hence the number of iterations), the difference between Luersen and VVP will be more noticeable when the number of iterations is increased.

A genetic algorithm is applied to all these functions with the same number of function analyses. In most of the cases, GA is not able to find a good minimum compare to the GBNM, with this number of iterations (see Figures 7, 8, and 10). The total computational time consists of time needed for functional analysis and time needed for optimization.
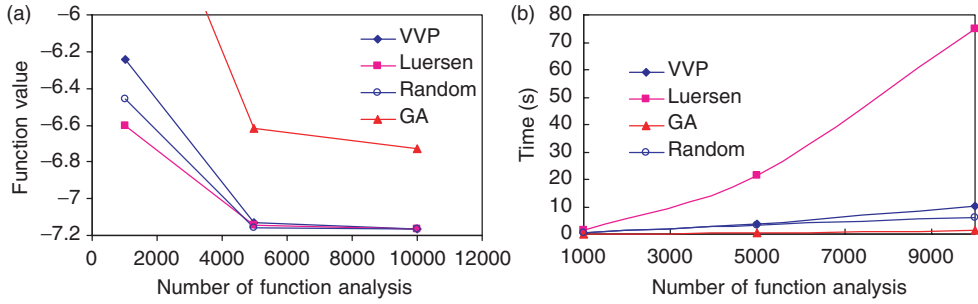
Figure 10. Comparison of GBNM with Luersen, VVP, and random restarts and GA on B_2 test function with 12 variables, 100 runs; (a) Minimum function value; (b) Computational time.
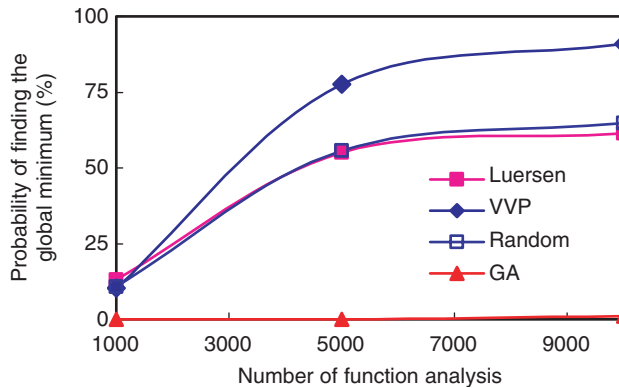

Figure 11. Average probability of finding the global minimum of the given test functions after 100 runs.

For mathematical functions, the computational time required for functional analysis is of the same order of the one required for optimization. Indeed, VVP requires longer optimization time than GA for a given mathematical test function. However, when functional evaluation becomes complex, such as in real composite structures where FEA calculations must be included, then the computational time for functional evaluation completely dominates the total computational time. The result is that VVP and GA end up with nearly the same total computational time (difference would be negligible). However, as shown in Figure 10(a), VVP provides a better accuracy in finding the optimum value for a give number of function analysis (i.e., 5000).

In summary, for a small number of function analyses, it is shown that several restarts of the N–M method (a local optimizer) can generally find a better solution than an evolutionary optimizer like a genetic algorithm. Three different restart procedures have been tested. The result shows that the VVP restart can find a better minimum for a general multimodal function than those obtained by Luersen and random restarts. The required computational time for VVP restart is considerably less than GBNM and is close to random restart.

Nonlinear Constraints

To test the back tracking procedure (Figure 3) we need to choose constraints described by nonlinear functions. Two hyper-spheres are added as constraints to the

third test function (B_1). Points within the two hyper-spheres are feasible. Figure 12 shows the feasible region and the function contour-plot in two dimensions. The GBNM with VVP restart is applied to this test function with two variables. All the local minima found within 1000 iterations are shown in this figure. The figure shows that the procedure can find the constrained local minima and the global ones (i.e., $f = -0.9085$ at $(-0.24, 0.65)$ and $(0.24, -0.65)$).

Then the constrained optimization procedure is applied to B_1 with 12 design variables. Figure 13 shows the average minimum of the function value found with and without nonlinear constraints. The global minimum of B_1 with nonlinear constraints is -5.45
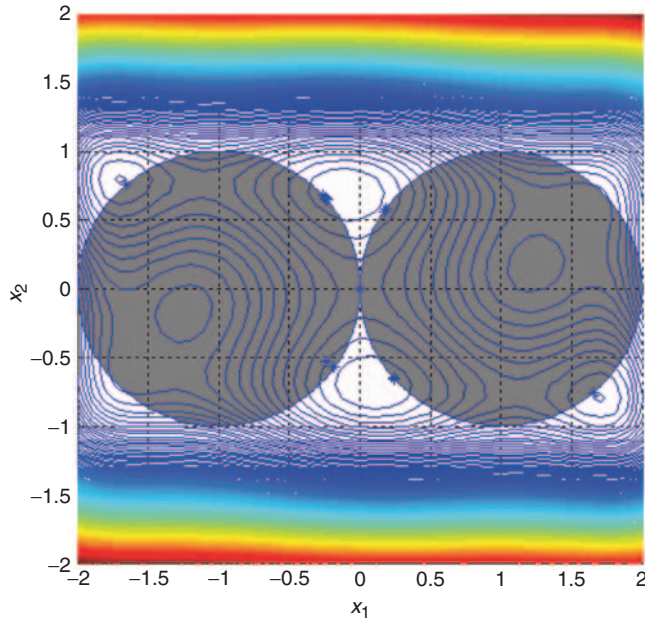

Figure 12. *The feasible region (two gray circles) and the contour plot of the six-hump camelback test function (B_1) in two dimensions.*
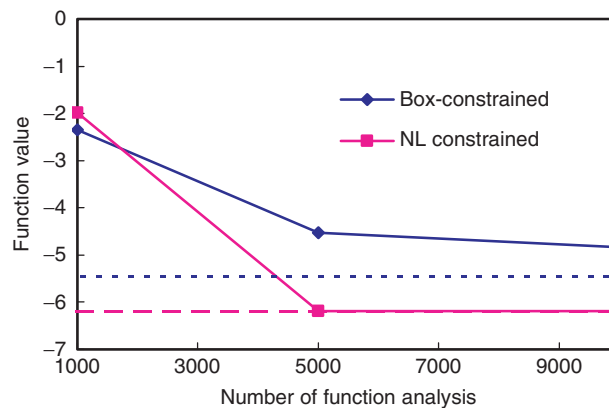

Figure 13. *Comparison of the minimum function value found by GBNM with VVP restart on a box/nonlinear constrained B_2 test function with 12 variables, 100 runs.*

(dashed line in Figure 13) and the average minimum found by the optimization procedure is −4.83. Considering more challenges in nonlinear constrained problem, the average minimum is close to the global solution. The results are not compared with GA or GBNM with Luersen restart because neither of these algorithms is able to directly work with nonlinear constraints.

COMPOSITE DESIGN PROBLEM

Finding the optimum structural design of composite materials is a difficult task due to the high degree of freedom in tailoring material properties and shape design. A variety of optimization methods, from simple mathematical methods, such as a linear programming [11], to combinations of computationally expensive methods, such as genetic algorithms and topology optimization [12], has been used for this purpose. The genetic algorithm is among the most popular ones in this field [13] because of capability of global optimization and independency to gradient information.

The optimum design of a composite part is a trade-off between structural and manufacturing aspects. Composite designers usually try to simplify the problem by separating the two parts [14] and performing the process tuning after the structural design. But, it is acknowledged that, once the preliminary design has been selected, up to 80% of the manufacturing cost cannot be changed [15]. Also a significant weight reduction in final part is reported by incorporating the manufacturing constraints into the structural design problem [16,17]. There are several researches [14–18] confirming the approach to the design and optimization of composite materials must be multidisciplinary. Such an approach, called *simultaneous* optimization, is studied in this article.

Simultaneous optimization is more complex than the separate design, because it requires taking into account a large number of variables and local minima. Figure 14 shows the objective functions of a two-layer rectangular plate made of laminated composite material under specific in-plane loads. Maximum structural objective (part (a)) corresponds to the maximum strength. Maximizing the manufacturing objective (part (b)) is also desired, which corresponds to the minimum warpage and maximum permeability. The simultaneous objective function is illustrated in Figure 14(c). The simultaneous objective is neither the manufacturing nor the structural one, but it is a trade-off that lies between them.

There are only a few papers working on simultaneous optimization of composite materials, and most of them use evolutionary methods (i.e., a genetic algorithm) [18–22], despite their low convergence rate and inherently high computational cost. Previous
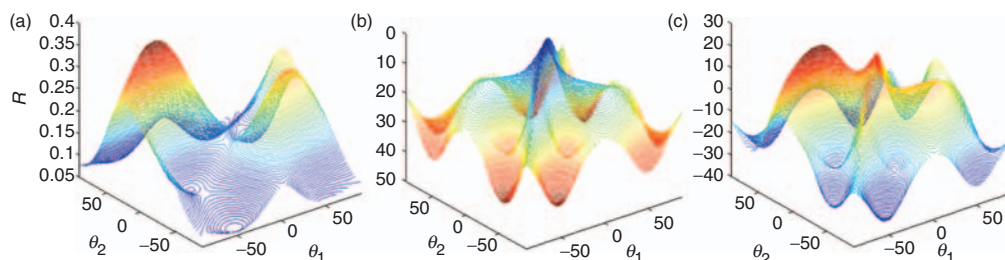

Figure 14. *Comparing a typical structural objective (a), manufacturing objective (b), and simultaneous objective (c) in a tow-layer rectangular plate $[\theta_1/\theta_2]_t$ under in-plane loads.*

section has shown that the improved local-global search is generally faster than an evolutionary algorithm for small number of function analyses. In this section the improved GBNM, described in preceding sections, is used for simultaneous optimization of a composite bracket.

Composite Bracket Design Problem

The Z-shaped composite bracket shown in Figure 15 is made of 16 plies balanced symmetric laminate of graphite/epoxy (AS4/8552) with fiber orientation of $[\pm\theta_1/\pm\theta_2/\pm\theta_3/\pm\theta_4]_s$. The optimization problem is to find the structural and geometrical variables to achieve the optimum value of the objective functions as shown in Table 1. The objectives are shown in the first row of the table. This table also shows the strong/weak effect of the design variables on the objectives by a solid/non-solid circle in the corresponding cell. The part should not fail or delaminate anywhere within a safety factor of 1.5 and 2, respectively. Delamination is calculated in the curved regions where the angle shape causes high interlaminar normal stresses. The vertical deflection of less than 1 mm and the spring-in of less than $0.5°$ are strictly required for an acceptable design.

Problem Simulation

To evaluate the objective function, an appropriate processing and structural simulation is required. A semi-analytical model is developed in MATLAB$^©$ for quick evaluation of the objectives and constraints.
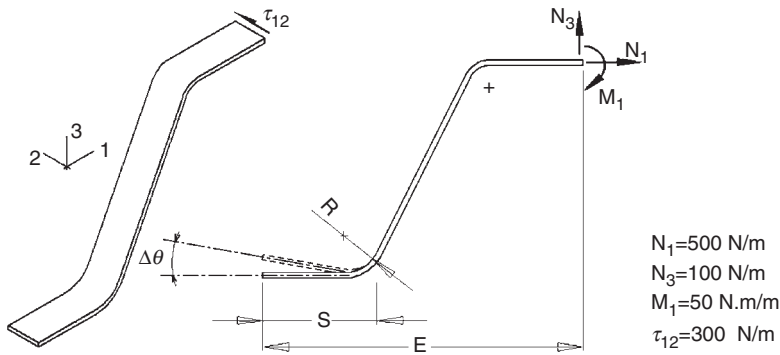

$N_1=500$ N/m
$N_3=100$ N/m
$M_1=50$ N.m/m
$\tau_{12}=300$ N/m

Figure 15. Geometrical variables and applied loads on the composite bracket.

Table 1. Structural, geometrical, and manufacturing design variables and objectives.

Design variable	Structural objectives/constraints				Manufacturing objectives/constraints	
	Min: W	Max: R	δ_y, 1mm	D $\geq$ 2	$\|\Delta\theta\|0.5°$	Min: $\|\Delta\theta\|$
$\theta_i i = 1,\ldots,4$	∘[a]	•[b]	•	•	•	•
E	•	•	•	•	•	•
S	•	•	•	∘	•	•
R	•	•	•	•		

∘, Indirect or weak interconnection; •, direct and strong coupling.

Objectives of the coupled problem include: failure index, vertical deflection, and spring-in. Failure index (R) is calculated by first-ply-failure in classical lamination theory and Hashin stress criterion [23]. Vertical deflection is calculated by a numerical integration and energy method [24]. Finally, spring-in, which is the angular deformation of a part after demolding, is a function of cure shrinkage and thermal expansion, and is given by:

$$\Delta\theta = \theta\left[\left(\frac{(\alpha_l - \alpha_t)\Delta T}{1 + \alpha_t\Delta T}\right) + \left(\frac{\phi_l - \phi_t}{1 + \phi_t}\right)\right] \tag{12}$$

where θ is the angle of the bracket, $\Delta\theta$ is spring-in, ΔT is temperature change which is the difference between the cure temperature and room temperature. ϕ and α are coefficients of shrinkage and thermal expansion, subscripts l and t respectively stand for longitudinal and through thickness direction.

Delamination is a critical mode of failure in composite materials, and it is due to the interlaminar stresses between subsequent laminates. In a flat plate, interlaminar stresses are created only by the free-edge effect [25], but in a curved part, the 3D stress field also creates significant interlaminar stresses that may cause delamination at the curved region. A convenient, albeit crude, model of free edge interlaminar stresses by Pagano [25] is used. In this model the interlaminar normal stress is estimated as shown in Figure 16.

Interlaminar normal stresses created by the angle-shape effect are shown in Figure 17. Sequentially solving the equilibrium equations for all layers, starting form the innermost layer results in interlaminar normal stress between layers n and $n+1$:

$$\sigma_{z,n-(n+1)} = \frac{\sum_{k=1}^{n} \sigma_{x,k} t_k}{R + \sum_{k=1}^{n} t_k} \tag{13}$$

where t_i shows the thickness of ith layer, and R is the inner radius of the curved part. If off-axis stresses change along the curve, Equation (13) would be valid only for a differential angle $d\theta$.

Interlaminar shear stresses are of minor importance with respect to interlaminar normal stresses [23], thus an approximation of shear stress in a prismatic member under a transverse load is used.

The semi-analytical models of first-ply-failure, delamination, deflection, and spring-in will be used during the optimization process in the next section.
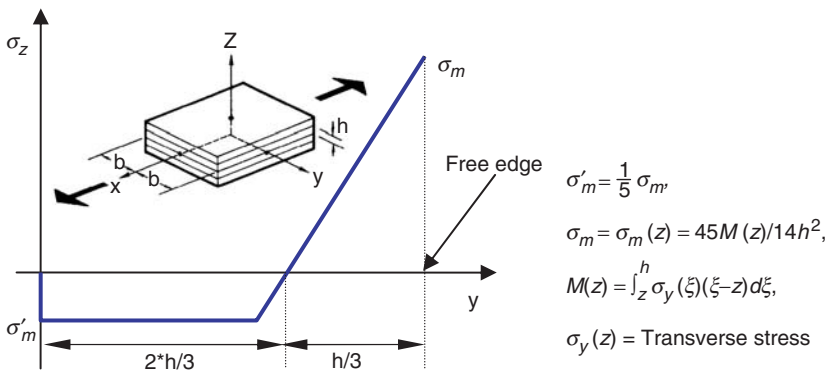

Figure 16. *Approximate distribution of interlaminar normal stress at a free edge.*
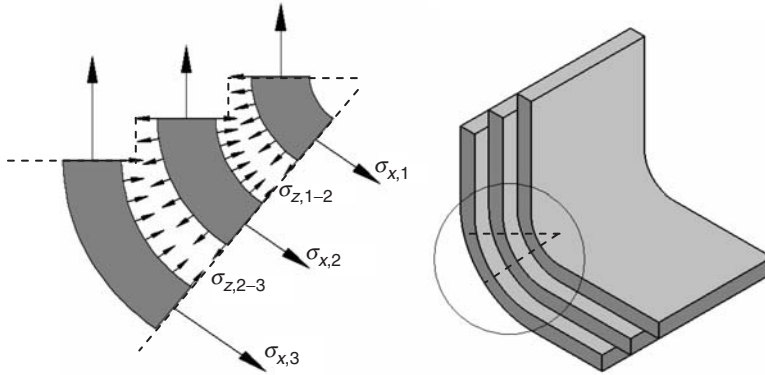
Figure 17. *Interlaminar normal stresses in the curved region of an angle-shaped flange.*

Numerical Results

The improved GBNM is applied to the simultaneous structural and manufacturing design of the bracket. The following weighted summation of the objectives is used as a cost function to be minimized:

$$\min F(x) = \alpha\left(\frac{\text{Weight}}{0.005} - \frac{R}{1.5}\right) + \beta\left(\frac{|\Delta\theta|}{0.25}\right); \quad \alpha, \ \beta \in \{0, 1\}$$

$$\text{S.T.} \quad \{R \geq 1.5, \ D \geq 2.0, \ S_r \geq 0.01, \quad\quad\quad\quad (14)$$

$$\alpha(|\delta| - 0.001) \leq 0, \quad \beta(|\Delta\theta| - 0.5) \leq 0\}$$

where R and D are delamination and load factors. S_r is the shoulder length after applying the fillet, measured in meters. Δ is the vertical deflection in meters. $\Delta\theta$ is the spring-in expressed in degrees. α and β are two dimensionless factors defining the relative importance of structural and processing objectives. The optimum found by $\alpha = 1$ and $\beta = 0$ only considers the structural objectives and constraints, thus called the *structure-only* design. In contrast the case with $\alpha = 0$ and $\beta = 1$ is the *manufacturing-only* design. Optimization with $\alpha = \beta = 1$ is called a *simultaneous* design. Here, α and β are restricted to 0 and 1, but in general, a designer can set them to any real value by considering the relative importance of structural and manufacturing design.

Since the problem includes variables of different kinds and scales, the variables are normalized to their design domain. The stopping criteria are set to be one percent of the smallest discrete portion of the design domain for the small simplex and 10^{-4} for a flat simplex. The optimization procedure is performed up to 5000 iterations. The optimum point obtained in continuous optimization process is rounded off toward the near discrete value.

We now consider three design scenarios described by different objective functions. To do this, we control the values of α and β in Equation (14). In the first scenario, only the structural performance is to be optimized. In the second one, both structural and manufacturing objectives are considered. In the third one, the objective function is described only by manufacturing parameters. Table 2 and Figure 18 show the best solutions to the structural-only, manufacturing-only, and simultaneous design problem.

Table 2. Optimum design obtained by improved GBNM after 5000 iterations.

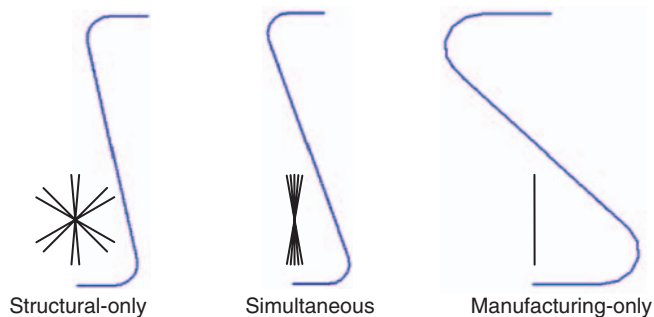|		Structure-only	Simultaneous	Mfg-Only
Fiber orientation		$[\pm5/\pm45/\pm60/\pm60]_s$	$[10/0_4/5_8]_s$	$[0_8]_s$
Geometry	E (mm)	21	10	6
	S (mm)	20	20	50
	R (mm)	8	7	12
Structural objectives/constraints	Weight (gr)	6.83	6.91	9.34
	Deflection (mm)	−0.98	−0.24	0.141
	Load factor (R)	2.17	1.74	1.55
	Delamination factor	12	23	40
Mfg. obj./cons.	Spring-in	−0.27	−0.014	−0.005

Figure 18. *Optimum geometry of the bracket and fiber orientation of the laminate.*

Structural-only   Simultaneous   Manufacturing-only

Structural-only Simultaneous Manufacturing-only

Table 2 shows the geometry, the fiber orientation, and the performance criteria of solutions obtained by three design scenarios. Figure 19 shows the performance indices of the three solutions. These have been obtained by normalizing the objective functions so that the solution obtained by the structure-only approach has 100% structural performance and 0% manufacturing performance. In contrast, the solution obtained by the manufacturing-only approach has 100% manufacturing performance and 0% structural performance. As shown in this figure, the structural performance of the simultaneous design is 67% of that obtained by the structural-only design. Its manufacturing performance is 97% of that provided by the manufacturing-only design. It shows that the simultaneous design is a trade-off between the structure-only and the manufacturing-only solutions.

The genetic algorithm needs a penalty function and its performance depends on the appropriate selection of penalty factor. Here, a GA with different penalty functions is used, and the best result obtained is compared with the improved GBNM. After 2000 function analyses, solution by the improved GBNM is 17% better than GA. The difference is reduced to 10% after 5000 function analyses, but still GBNM finds a better design.

Table 2 shows only the best solution, although there are around 50 other local solutions found during the optimization process that a designer can select from. In this respect, this
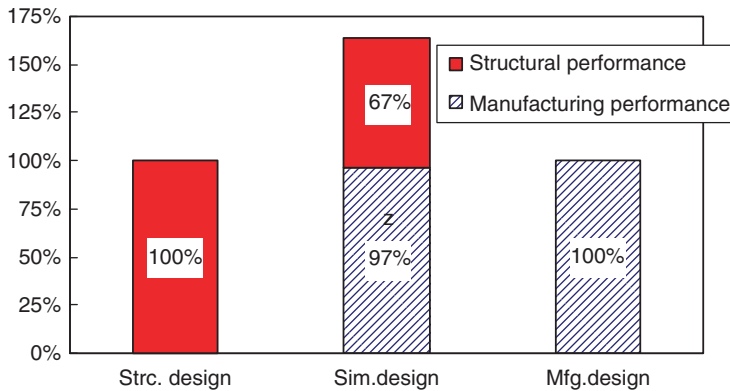
Figure 19. *The simultaneous approach provides a big improvement in manufacturing aspect with a slight loss in the structural performance.*

optimization procedure is comparable to an evolutionary procedure that provides a family of optimal solutions instead of just one specific solution. This feature is important, especially for multi-objective optimization.

The composite bracket is optimized for structural, manufacturing, and simultaneous objectives. The result attests to the necessity of the simultaneous approach for composite material design. It also confirms that for a small number of function analyses the improved GBNM is more efficient than a common evolutionary algorithm, both in terms of time and the optimal solution found.

CONCLUSIONS

A local-global search based on several restarts of the N–M local optimizer is introduced. The VVP restart is presented to improve the performance of this local-global search. Different mathematical functions are used to show that the VVP restart generally performs better than Luersen and random restarts and even an evolutionary algorithm like GA. The computational cost of VVP restart is much less than that of Luersen restart. A back-tracking procedure is also presented and tested to incorporate nonlinear constraints into the design problem. The resulting algorithm can work on a problem with nonlinear constraints. It is simple to use and is capable of being terminated at any time. The numerical results show that it is generally faster than an evolutionary algorithm for a small number of function analyses. Finally, it provides a family of solutions (local optima) instead of just one specific solution.

The developed optimization procedure is applied to simultaneous structural and manufacturing optimization of a Z-shaped composite bracket and is compared with GA. The observed trade-off between structural and manufacturing optimum designs confirms the need of using a simultaneous approach in this field. The proposed procedure performed better than an evolutionary algorithm on this type of problem by providing a better solution with the same number of function analyses. Future work will give more emphasis on the manufacturing objectives (e.g., processing time, cost, and quality parameters) and will include manufacturing design variables (e.g., injection pressure and temperature, gate and vent location, mold design, variables, etc).

REFERENCES

1. Luersen, M.A. and Riche, R.L. (2004). Globalized Nelder-Mead Method for Engineering Optimization, *Computers and Structures*, **82**: 2251–2260.

2. Nelder, J.A. and Mead, R. (1965). Simplex Method for Function Minimization, *Computer Journal*, **7**(4): 308–313.

3. Lagarias, J.C., Reeds, J.A., Wright, M.H. and Wright, P.E. (1998). Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions, *SIAM Journal of Optimization*, **9**(1): 112–147.

4. Kolundzija, B.M. and Olcan, D.I. (2003). Antenna Optimization using Combination of Random and Nelder-Mead Simplex Algorithms, *IEEE Antennas and Propagation Society*, *AP-S International Symposium (Digest)*, **1**: 185–188.

5. Chelouah, R. and Siarry, P. (2003). Genetic and Nelder-Mead Algorithms Hybridized for a More Accurate Global Optimization of Continuous Multiminima Functions, *European Journal of Operational Research*, **148**(2): 335–348.

6. Chelouah, R. and Siarry, P. (2005). A Hybrid Method Combining Continuous Tabu Search and Nelder-Mead Simplex Algorithms for the Global Optimization of Multiminima Functions, *European Journal of Operational Research*, **161**(3): 636–654.

7. Humphrey, D.G. and Wilson, J.R. (2000). A Revised Simplex Search Procedure for Stochastic Simulation Response Surface Optimization, *INFORMS Journal on Computing*, **12**(4): 272–283.

8. Regis, R.G. and Shoemaker, C.A. (2004). Local Function Approximation in Evolutionary Algorithms for the Optimization of Costly Functions, *IEEE Transactions on Evolutionary Computation*, **8**(5): 490–505.

9. Floudas, C.A. and Pardalos, P.M. (1990). A Collection of test Problems for Constrained Global Optimization Algorithms, *Lecture Notes in Computer Science*, **455**. Springer Verlag, Berlin.

10. Tomick, J.J., Arnold, S.F. and Barton, R.R. (1995). Sample Size Selection for Improved Nelder–Mead Performance. In: *Proceeding of the 1995 Winter Simulation Conference*, December 3–6, pp. 341–345.

11. Schmit, L.A. and Farshi, B. (1973). Optimum Laminate Design for Strength and Stiffness, *International Journal for Numerical Methods in Engineering*, **7**(4): 519–536.

12. Hansel, W., Treptow, A., Becker, W. and Freisleben, B. (2002). A Heuristic and a Genetic Topology Optimization Algorithm for Weight-minimal Laminate Structures, *Composite Structures*, **58**(2): 287–294.

13. Venkaraman, S. and Hafteka, R.T. (1999). Optimization of Composite Panels–A review. In: *14th Annual Technical Conference of the American Society of Composites*, September 27–29, pp. 479–488.

14. Le Riche, R., Saouab, A. and Breard, J. (2003). Coupled Compression RTM and Composite Layup Optimization, *Composite Science and Technology*, **63**(15): 2277–2287.

15. Wang, K., Kelly, D. and Dutton, S. (2002). Multi-objective Optimisation of Composite Aerospace Structures, *Composite Structures*, **57**(1): 141–148.

16. Wang, B.P. and Costin, D.P. (1992). Optimum Design of a Composite Structure with Three Types of Manufacturing Constraints, *AIAA Journal*, **30**(6): 1667–1669.

17. Costin, D.P. and Wang, B.P. (1993). Optimum Design of a Composite Structure with Manufacturing Constrains, *Thin-walled Structures*, **17**: 185–202.

18. Henderson, J.L., Gurdal, Z. and Loos, A.C. (1999). Combined Structural and Manufacturing Optimization of Stiffened Composite Panels, *Journal of Aircraft*, **36**(1): 246–254.

19. Park, C.H., Lee, W.I., Kim, S.J., Han, W.S. and Vautrin, A. (2001). Simultaneous Optimization of Design and Manufacture of Composite Structures, *Collection of Technical Papers AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, **1**: 186–196.

20. Park, C.H., Lee, W., Han, W.S. and Vautrin, A. (2003). Weight Minimization of Composite Laminated Plates with Multiple Constraints, *Composites Science and Technology*, **63**(7): 1015–1026.

21. Park, C.H., Lee, W., Han, W.S. and Vautrin, A. (2004). Simultaneous Optimization of Composite Structures Considering Mechanical Performance and Manufacturing Cost, *Composite Structures*, **65**(1): 117–127.

22. Park, C.H., Lee, W., Han, W.S. and Vautrin, A. (2005). Multiconstraint Optimization of Composite Structures Manufactured by Resin Transfer Molding Process, *Journal of Composite Materials*, **39**(4): 347–374.

23. Tsai, S.W. (1992). *Theory of Composites Design*, Think Composites, a Division of ILT Corporation, Dayton, Paris, Tokyo.

24. Megson, T.H.G. (1999). *Aircraft Structures for Engineering Students*, **3rd edn**, J.W. Arrowsmith Ltd., Bristol, England.

25. Pagano, N.J. and Pipes, R.B. (1989). *Interlaminar Response of Composite Materials*. Composite Materials Series, Volume 5, Elsevier Science Publishing Company Inc., New York.