

NUMERICAL METHOD FOR GENERATING THE ENTIRE PARETO FRONTIER IN MULTIOBJECTIVE OPTIMIZATION

Sergei V. Utyuzhnikov^{*†}, Paolo Fantini[†] and Marin D. Guenov[†]

^{*}University of Manchester, School of Mechanical, Aerospace & Civil Engineering,
Manchester M60 1QD
Email: s.utyuzhnikov@manchester.ac.uk

[†]Cranfield University, School of Engineering, Cranfield,
Bedfordshire MK43 0AL
Email: m.d.guenov@cranfield.ac.uk

Key words: Multiojective optimization, Pareto set, Physical programming.

Abstract. *In multidisciplinary optimization a designer solves a problem where there are different criteria usually contradicting each other. In general, the solution of such a problem is not unique. When seeking an optimal design, it is natural to exclude from the consideration any design solution which can be improved without deterioration of any discipline and violation of the constraints; in other words, a solution which can be improved without any trade-off. It leads to the Pareto optimal solutions. Mathematically, each Pareto point is a solution of the multidisciplinary optimization problem. Yet, it is important that the Pareto set is evenly distributed otherwise the representation of the Pareto surface may become inefficient.*

Although there are many numerical methods for vector optimization, only a few are able to generate the entire set of optimal solutions (Pareto frontier). It is very important for the decision maker, especially at the early design stages when most of the resources are committed, to be able to obtain maximum information on the Pareto surface at a minimal cost. The Physical Programming (PP) Method, suggested by Messac, is able to generate Pareto points on both convex and non-convex Pareto frontiers. The PP is substantially modified to make it simpler and more efficient for generating an even Pareto set. The algorithm does not provide non-Pareto solutions while local Pareto solutions may be easily recognized and removed in the framework of this method.

1 INTRODUCTION

Mathematically, the trade-off analysis can be formulated as a vector nonlinear optimization problem under constraints. Generally speaking, the solution of such a problem is not unique. It is natural to exclude from the consideration any design solution which can be improved without deterioration of any discipline and violation of the constraints; in other words, a solution which can be improved without any trade-off. This leads to the Pareto optimal solutions. Yet, it is important that the Pareto set is evenly distributed, otherwise the representation of the Pareto surface becomes inefficient.

Usually, vector optimization methods lead to a scalar optimization of an aggregated objective function (AOF) which includes a combination of objective (cost) functions. The most conventional case of the AOF is represented by a linear (weight) combination of the objective functions. Although this method is relatively simple, it has many drawbacks related mainly with the uncertainty about weights. It may require many iterations to find the combination of the weights leading to a solution which corresponds to the decision maker's expectations. Furthermore, it is well known that this method can generate only the convex part of a Pareto surface while structural problems often result in non-convex Pareto surfaces.

The Normally Boundary Intersection (NBI) Method [1, 2] was developed for generating an even distribution of Pareto points by Das and Dennis. The method appeared to generate non-Pareto and locally Pareto points that require a filtering procedure [4]. The new Normal Constraint (NC) Method [4] developed recently looks very promising. Both methods have clear geometrical interpretation. Both methods may fail to generate Pareto solutions over the entire Pareto frontier [5] in multidimensional case. The recent modification of the NC [5] eliminates this drawback and guarantees the complete representation of a Pareto frontier. Meanwhile, both methods may generate non-Pareto and local Pareto solutions [4] though the NC does it less likely [5].

The Physical Programming (PP) Method was suggested by Messac in [3]. This method also generates Pareto points on both convex and non-convex Pareto frontiers as it was shown in [6]. The method does not use any weight coefficients and allows one to take into account the DM experience immediately. In this sense, it looks to be the most interesting method for practical applications under the above stated conditions. In the PP, the designer assigns each objective to one of the four categories (class-functions). The optimization is based on minimization of an aggregate preference function determined by the preference functions (class-functions) with preferences set *a priori*. A new (more compact) approximation of the class-functions is given in [6].

The PP is modified below to make it simpler and more efficient for practical applications. The class-functions are generalized to shrink the search domain and make its location in space more optimal. This is critical when generating an even set of the Pareto frontier. The proposed modification allows us to combine the advantages of the PP, NBI and NC methods. The algorithm to obtain an even distribution of the Pareto set is described in this paper. One of its main advantages is that it does not provide non-Pareto solutions while local Pareto solutions may be easily recognized and removed. A comparison against the standard approach shows that the modified approach is able to generate a much more even Pareto set.

2 MULTIOBJECTIVE OPTIMIZATION PROBLEM. PARETO OPTIMIZATION

It is assumed that an optimization problem is described in terms of a design variable vector $\mathbf{x} = (x_1, x_2, \dots, x_N)^T$ in the design space $\mathbf{X} \in R^N$. A function $\mathbf{f} \in R^M$ evaluates the quality of a solution by assigning it to an objective vector $\mathbf{y} = (y_1, y_2, \dots, y_M)^T$ ($y_i = f_i(\mathbf{x}), f_i: R^N \rightarrow R^1, i = 1, 2, \dots, M$) in the objective space $\mathbf{Y} \in R^M$. Thus, \mathbf{X} is mapped by \mathbf{f} onto $\mathbf{Y}: \mathbf{X} \mapsto \mathbf{Y}$. A multiobjective optimization problem may be formulated in the following form:

$$\text{Minimize } [\mathbf{y}(\mathbf{x})] \quad (1)$$

subject to K inequality constraints

$$g_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \dots, K \quad (2)$$

and P equality constraints

$$h_j(\mathbf{x}) = 0, \quad j = 1, 2, \dots, P \quad (3)$$

The feasible design space \mathbf{X}^* is defined as the set $\{\mathbf{x} | g_j(\mathbf{x}) \leq 0, j = 1, 2, \dots, K; \text{ and } h_i(\mathbf{x}) = 0, i = 1, 2, \dots, P\}$. The feasible criterion (objective) space \mathbf{Y}^* is defined as the set $\{\mathbf{Y}(\mathbf{x}) | \mathbf{x} \in \mathbf{X}^*\}$. The feasibility means no constraint is violated.

A design vector \mathbf{a} ($\mathbf{a} \in \mathbf{X}^*$) is called a Pareto optimum iff it does not exist any $\mathbf{b} \in \mathbf{X}^*$ such that: $\mathbf{y}(\mathbf{b}) \leq \mathbf{y}(\mathbf{a})$ and exist $l \leq M: y_l(\mathbf{b}) < y_l(\mathbf{a})$.

A design vector is called a local Pareto optimum if it is a Pareto optimum within its some neighbourhood.

3 GENERATION OF EVEN DISTRIBUTION OF A PARETO SET

The PP allows one to generate an even distribution of the Pareto frontier. The appropriate algorithm is given in [6]. The original approach is briefly described below, followed by the description of our proposed modification which aims to make the algorithm more efficient, especially in the case of a concave Pareto frontier. Also, the algorithm is generalized on the $2S - 4S$ class-functions.

Let us define the trade-off matrix T as follows:

$$T = \begin{bmatrix} F_{1,\min} & F_{12} & \dots & F_{1n_{sc}} \\ F_{21} & F_{2,\min} & \dots & F_{2n_{sc}} \\ \dots & \dots & \dots & \dots \\ F_{n_{sc}1} & F_{n_{sc}2} & \dots & F_{n_{sc},\min} \end{bmatrix}, \quad (4)$$

In the trade-off matrix T , an i -th row represents the coordinates of an anchor point $\boldsymbol{\mu}_i^*$ corresponding to the solution of single-optimization problem $\min F_i$ in the feasible criterion space \mathbf{Y}^* .

In the feasible space \mathbf{Y}^* a hypercube H limiting the search domain is defined in the following manner. We set the pseudo nadir point [5] $F_{i,\max} = \max_j F_{ij}$ that corresponds to the maximum i -th component among all anchor points. Then, the hypercube H is represented as follows: $H = [F_{1,\min} F_{1,\max}] \times [F_{2,\min} F_{2,\max}] \times \dots \times [F_{n_{sc},\min} F_{n_{sc},\max}]$.

For the sake of simplicity, consider the 2D case where there are only two design metrics. We assume that each of the design metrics belongs to class *IS*. Following [6], for each of the design metrics let us introduce the vector of pseudo-preferences P_i :

$$P_i \equiv (F_{i1}, F_{i2}, F_{i3}, F_{i4}, F_{i5})^T = F_i^{(0)}(1,1,1,1,1)^T + a_i(0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1)^T, \quad (5)$$

where $a_i = (F_{i,\max} - F_{i,\min})/n_d$, $F_i^{(0)}$ is a free parameter. The parameter n_d defines the box size.

In such a case, either region $F_1 > F_{15}$ or region $F_2 > F_{25}$ becomes unacceptable. Thus, the box $D = [F_{11}F_{15}] \times [F_{21}F_{25}]$ defined by the pseudo-preferences limits the search domain from the right and upper sides leaving it in the other directions unlimited for a formal search. Changing the free vector $\mathbf{F}^{(0)} = (F_1^{(0)}, F_2^{(0)}, \dots, F_{n_{sc}}^{(0)})^T$, it is possible to shift the box D in the hypercube H to seek Pareto solutions. The current location of the box D determines a possible location of a Pareto point since the Pareto points outside D (more precisely – higher or on the right of D) are excluded from the current consideration. In some sense, it operates similarly to the ε -constraint method, but in contrast, space reduction is simultaneously performed for all objectives. For example, moving the box to the lower-right vertex of the hypercube H (in 2D case), we give preferences to low values of the second objective at the expense of high values of the first objective. In [6], the algorithm is given for shifting the box D over the space \mathbf{Y}^* to seek the Pareto frontier. To provide this, the free vector $\mathbf{F}_i^{(0)}$ is specially chosen to move the box along lines parallel to a diagonal of the hypercube H passing through the lower-right and the upper-left vertices. A few free parameters are introduced to control the displacement of the box. No algorithm is given to determine these parameters. The approach described is only applicable to the minimization problem when only the class function *IS* is involved.

We suggest another strategy to seek the Pareto frontier. For the sake of simplicity, we assume initially that only a minimization problem is considered (all class functions are *IS*) and the problem is solved in the objective space \mathbf{Y} . The generalization on the arbitrary case is given subsequently.

Similar to the NC method, let us consider the utopia plane created by anchor points $\boldsymbol{\mu}_i^*$. It is well known that any point \mathbf{p} belonging to the interiority of a convex polygon spanned by n_{sc} vertexes $\boldsymbol{\mu}_i^*$ can be represented as follows:

$$\mathbf{p} = \sum_{i=1}^{n_{sc}} \alpha_i \boldsymbol{\mu}_i^* \quad (6)$$

where the parameters α_i must satisfy the following conditions:

$$0 \leq \alpha_i \leq 1, \quad \sum_{j=1}^{n_{sc}} \alpha_j = 1 \quad (7)$$

In this approach the notion of the anchor point is used. Although not crucial, an important general remark on the definition of the anchor point should be made. As it was mentioned above, the standard definition assumes an anchor point $\boldsymbol{\mu}_i^*$ corresponds to the solution of the single-optimization problem $\min F_i$ in feasible criterion space \mathbf{Y}^* (see, e.g.,

[5]). This definition allows the anchor point corresponding to some objectives to be non-unique. Furthermore, it may not even belong to the boundary of \mathbf{Y}^* . Such an example is given below. We suggest the following specification which guarantees the uniqueness of the anchor point for each objective. If the solution of the problem $\boldsymbol{\mu}_i^* = \mathbf{F}(\mathbf{X}^{i*}) \{ \mathbf{X}^{i*} : \mathbf{X}^{i*} = \arg \min_{\mathbf{Y}^*} F_i \}$ is

not unique, then the point corresponding to the minimal values of the other design metrics is to be chosen. It may lead to the problem of trade-off minimization for the remaining objectives. To avoid this, priority in the minimization is introduced. First, F_i is to be minimized, then F_{i+1} and so on up to F_{i-1} . The prioritization is introduced in a circular order: $i+1, i+2, \dots, n_{sc}, 1, 2, \dots, i-1$. A k -th prioritization assumes that the k -th minimization must not violate all the previous $k-1$ ones. At this definition, it is easy to prove that all anchor points belong to the Pareto frontier because they are on the boundary of the feasible space \mathbf{Y}^* and no objectives can be improved without deterioration of any other objective.

The free vector $\mathbf{F}^{(0)} = (F_1^{(0)}, F_2^{(0)}, \dots, F_{n_{sc}}^{(0)})^T$ is determined in the following way. Let us consider the box D . The box D is shifted in such a way that its vertex M corresponding to the maximal values of the design metrics $(M = (F_{1,\max}, F_{2,\max}, \dots, F_{n_{sc},\max})^T)$ lies in the utopia plane (see Figure 1). This means

$$\mathbf{F}^{(0)} = \sum_{i=1}^{n_{sc}} \alpha_i \boldsymbol{\mu}_i^* - \mathbf{a}, \quad (8)$$

where

$$\begin{aligned} \mathbf{F}^{(0)} &= (F_1^{(0)}, F_2^{(0)}, \dots, F_{n_{sc}}^{(0)})^T, \\ \mathbf{a} &= (a_1, a_2, \dots, a_{n_{sc}})^T. \end{aligned}$$

An even distribution of the coefficients α_i gives us an even distribution of the Pareto set. In comparison to the NC and NBI methods, the approach described below allows us to generate the complete Pareto frontier considering only non-negative coefficients α_i from (7).

One of the possible algorithms for calculation of the coefficients α_i is given in [5] where the following induction procedure is used. First, a uniform distribution of coefficient α_1 is considered. The sum of the rest coefficients $\sum_{j=2}^{n_{sc}} \alpha_j$ equals to $1 - \alpha_1$ for each selected value of α_1 . Then, a uniform distribution of the coefficient α_2 is considered for each of these variants and so on until either the last coefficient $\alpha_{n_{sc}}$ is reached or the sum of the coefficients already determined equals to 1. In the latter variant the remaining coefficients equal zero.

It will be shown that the distribution of the Pareto set may be sensitive to the displacement of the box D along the utopia plane especially if the Pareto frontier is concave. To avoid this, generalization of the class functions is performed. It allows us to shrink the search domain substantially.

In order to shrink the search domain defined by the hypercube H , it is suggested to introduce generalized class-functions as follows:

$$\bar{F}(\tilde{F}_i) \quad (i = 1, \dots, n_{sc}), \quad (9)$$

where \tilde{F}_i is defined by an affine transform

$$\tilde{F}_i = F_j B_{ji}, \quad (i, j = 1, \dots, n_{sc}). \quad (10)$$

In the objective space \mathbf{Y} , it is equivalent to the introduction of a new coordinate system with the basic vectors

$$\mathbf{a}_i = A_{ij} \mathbf{e}_j, \quad (i, j = 1, \dots, n_{sc}), \quad (11)$$

$$A^{-1} = B$$

where \mathbf{e}_j ($j = 1, \dots, n_{sc}$) are the basic vectors of the original coordinate system.

Then, the search domain can be changed as shown in Figure 2. In particular, it is possible to choose basic vectors \mathbf{a}_i ($i = 1, \dots, n_{sc}$) which form an angle γ_c to a selected direction \mathbf{l} . The 2D case is shown in Figure 2. Matrixes \mathbf{A} and \mathbf{B} can be easily determined as follows:

$$A = \begin{pmatrix} \cos \gamma_- & \sin \gamma_- \\ \cos \gamma_+ & \sin \gamma_+ \end{pmatrix}, \quad B = \frac{1}{\sin 2\gamma_c} \begin{pmatrix} \sin \gamma_+ & -\sin \gamma_- \\ -\cos \gamma_+ & \cos \gamma_- \end{pmatrix}, \quad (12)$$

where $\gamma_+ = \gamma_n + \gamma_c$, $\gamma_- = \gamma_n - \gamma_c$, $\mathbf{l} = (\cos \gamma_n, \sin \gamma_n)^T$.

In the general case of $R^{n_{sc}}$, it is possible to suggest the following algorithm. There are the following conditions on \mathbf{a}_i :

$$(\mathbf{a}_i, \mathbf{l}) = \cos \gamma_c \quad (i = 1, \dots, n_{sc}) \quad (13)$$

All the vectors \mathbf{a}_i are parallel to the lateral area of the hypercone having the angle γ_c and the axis along vector \mathbf{l} . It is important to guarantee a spread distribution of these vectors. At least, the basis created by these vectors must not vanish. It appears possible to obtain even a fully uniform distribution of the basis vectors of such a polyhedral cone.

First, suppose that \mathbf{l} is directed as follows:

$$\mathbf{l} = \mathbf{l}_0, \quad (14)$$

$$\mathbf{l}_0 = (l_0, l_0, l_0, \dots, l_0)^T$$

Assuming that \mathbf{l} is a unit vector, we obtain its coordinates:

$$l_0 \equiv \cos \gamma_0 = \frac{1}{\sqrt{n_{sc}}} \quad (15)$$

The basis vector \mathbf{a}_i can be determined in the plane created by the vectors \mathbf{e}_i and \mathbf{l}_0 . It is possible to show that

$$\mathbf{a}_i = \frac{\sin \gamma_c}{\sin \gamma_0} \mathbf{e}_i + \frac{\sin(\gamma_0 - \gamma_c)}{\sin \gamma_0} \mathbf{l}_0 \quad (16)$$

From (11), (15) and (16) we obtain

$$A = \frac{\sin \gamma_c}{\sin \gamma_0} I + \frac{\sin(\gamma_0 - \gamma_c) \cos \gamma_0}{\sin \gamma_0} E, \quad (17)$$

where all elements of the matrix E are unities: $\|E_{ij}\| = 1$.

In many cases the shrinking around the lines parallel to the vector \mathbf{l}_0 is already sufficient. Nevertheless, it is important to obtain matrix A in the general case of an arbitrary

unit vector \mathbf{l} . For this purpose, it is enough to perform a linear transform mapping the previous pattern in such a way that the vector \mathbf{l}_0 is mapped onto the \mathbf{l} . This purpose is reached by multiplying both parts of equation (16) by an orthogonal matrix R :

$$R\mathbf{l}_0 = \mathbf{l} \quad (18)$$

Then, we obtain the basis of vectors $\{\mathbf{a}'_i\}$ ($i = 1, \dots, n_{sc}$) uniformly distributed in the lateral area of the hypercone having the axis parallel to the vector \mathbf{l} :

$$\mathbf{a}'_i = \frac{\sin \gamma_c}{\sin \gamma_0} \mathbf{e}'_i + \frac{\sin(\gamma_0 - \gamma_c)}{\sin \gamma_0} \mathbf{l}, \quad (19)$$

where $\mathbf{e}'_i = R\mathbf{e}_i$ are the components of the Cartesian coordinate system in which the vector \mathbf{l} has equalled components. It is easy to see that the columns of transition matrix R are the coordinates of the vectors \mathbf{e}'_i in the basis $\{\mathbf{e}_j\}$. Since the transform is orthogonal, all angles are preserved. In particular, $(\mathbf{a}'_i, \mathbf{l}) = \cos \gamma_0$. Now we can write the matrix A in the general form:

$$A = \frac{\sin \gamma_c}{\sin \gamma_0} R^T + \frac{\sin(\gamma_0 - \gamma_c)}{\sin \gamma_0} E, \quad (20)$$

where $\|E_{ij}\| = \|l_j\|$.

If $\gamma_c = \gamma_0$, obviously $\mathbf{a}'_i = \mathbf{e}'_i$ that means the transform becomes orthogonal and is only reduced to a turn of the original Cartesian coordinate system. As a consequence in this case, the matrix A is orthogonal and $B = A^T$.

The general presentation requires the calculation of the orthogonal matrix R , the components of which must satisfy the following additional requirements:

$$\cos \gamma_0 \sum_{j=1}^{n_{sc}} R_{ij} = l_i \quad (21)$$

Matrix R is not unique. The simplest way to obtain it is to consider the rotation from the vector \mathbf{l}_0 to the vector \mathbf{l} in a Cartesian coordinate system related with these vectors so that

$$R = DT_R D^{-1}, \quad (22)$$

where T is an elementary rotation matrix describing the rotation in the plane created by the first two basis vectors

$$T_R = \begin{bmatrix} (\mathbf{l}_0, \mathbf{l}) & -\sqrt{1 - (\mathbf{l}_0, \mathbf{l})} & 0 & \dots & 0 \\ \sqrt{1 - (\mathbf{l}_0, \mathbf{l})} & (\mathbf{l}_0, \mathbf{l}) & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}, \quad (23)$$

and the matrix D is the transition matrix from the original basis to some orthogonal basis $\{\mathbf{b}_i\}$: $\mathbf{b}_1 = \mathbf{l}_0$, $\mathbf{b}_2 = \frac{\mathbf{l} - (\mathbf{l}, \mathbf{l}_0)\mathbf{l}_0}{\sqrt{1 - (\mathbf{l}, \mathbf{l}_0)^2}}$, The remaining basis vectors $\mathbf{b}_3, \mathbf{b}_4, \dots, \mathbf{b}_{n_{sc}}$ can be easily

obtained by the Gram-Schmidt orthogonalization procedure.

Obviously the vectors \mathbf{a}_i ($i = 1, \dots, n_{sc}$) create a basis in \mathbf{Y} which does not vanish. The basis vectors form a search cone similar to the 2D case shown in Figure 2. In the 2D case, $n_{sc} = 2$, $\gamma_0 = \pi/4$ and we obtain formulas (12).

The boundaries of the preference ranges are mapped in according to (10):

$$\tilde{F}_{ik} = F_{jk} B_{ji}, \quad (i, j = 1, \dots, n_{sc}; k = 1, \dots, 5) \quad (24)$$

Transform (9), (10) allows us to shrink the search domain and focus on a much smaller area on the Pareto surface. It makes the algorithm more flexible and much less sensitive to the displacement of box D . The transform shrinks the search domain, much like a “light beam” which emits from point M and highlights a spot on the boundary of feasible space \mathbf{Y}^* . The direction of the search is easy to handle. It is natural to choose this direction (vector $-\mathbf{l}$) in alignment with the normal to the utopia hyperplane towards the decrement of the objective functions. If no solution is found, the direction is switched on the opposite one. The box \tilde{D} , which is the image of the box D after transform (9), (10) is translated along the vector \mathbf{l} until the whole box \tilde{D} intersects the utopia plane. In this case, the new pseudo-preference \tilde{F}_{i1} corresponds to the former pseudo-preference \tilde{F}_{i5} and the search domain is limited by the box \tilde{D} only.

It should be emphasised that the general representation of matrix A can play a substantial part in seeking the Pareto set nearby its boundary. As it was mentioned above, if we consider orthogonal projection of the Pareto set onto the utopia hyperplane, the images of some Pareto points may not belong to the interiority of a convex polygon spanned by the n_{sc} vertexes $\boldsymbol{\mu}_i^*$ (6), (7). This fact was first noted in [5]. One of the possibilities to resolve this problem, suggested in [5] for the NC method, is based on the use of negative coefficients α_i . However, this will cause another problem, this time with the lower evaluation of the coefficients. Inevitably some points in the utopia plane corresponding to negative coefficients may not be orthogonal images of any Pareto point. Another opportunity, offered specifically by our modified PP is described below.

Let us consider the edge vectors of polygon (6), (7): $\mathbf{v}_i = \boldsymbol{\mu}_{i+1} - \boldsymbol{\mu}_i$ ($i = 1, \dots, n_{sc}-1$). The point \mathbf{p}_i belongs to a k -th edge of the polygon iff $\alpha_m = 0$ ($m \neq k, k+1$). Assuming that vector \mathbf{l} is related with the normal of the utopia hyperplane. Then, when point M lies on an edge of the polygon, vector \mathbf{l} is rotated in the direction outwards from the polygon. In other words, \mathbf{l} is changed in such a way that the orthogonal projection of the end of the vector, drawn from an edge, onto the utopia hyperplane must not fall in the interiority of the polygon. For this purpose, in the utopia plane we introduce a unit vector which is the outer normal to the edge considered. The vector can be defined as:

$$\mathbf{s}_i = \frac{\mathbf{v}_{i-1} + \beta_i \mathbf{v}_i}{|\mathbf{v}_{i-1} + \beta_i \mathbf{v}_i|}, \quad \beta_i = -\frac{(\mathbf{v}_{i-1}, \mathbf{v}_i)}{(\mathbf{v}_i, \mathbf{v}_i)}. \quad (25)$$

Then, the current vector \mathbf{l}_r is determined via \mathbf{s}_i and normal \mathbf{n} to the utopia hyperplane towards the utopia point $(F_1(\mu_1), F_2(\mu_2), \dots, F_{n_{sc}}(\mu_{n_{sc}}))$ as follows:

$$\mathbf{l}_r = -\cos \theta_r \mathbf{n} + \sin \theta_r \mathbf{s}_i, \quad (0 < \theta_r < \pi/2) \quad (26)$$

The angle θ_r is a parameter. Changing θ_r from $\pi/2$ to 0, the vector \mathbf{l}_r is turned from the normal vector $-\mathbf{n}$ to the vector \mathbf{s}_i (see Figures 3a and 3b). Thus, the algorithm is formulated in the following way. If point M in (8) belongs strictly to the interiority of the polygon, then the vector \mathbf{l}_r coincides with the normal $-\mathbf{n}$. If point M lies on an edge of the polygon, then an additional rotation of the vector may be required. To obtain an even distribution of the Pareto set, the number of additional points N_r related with the rotation of a vector \mathbf{l}_r depends on the distance to the vertexes of the edge. For example, the rotation is not required at the anchor points. Generally speaking, it is reasonable to choose the maximal value of N_r at the centre of an edge. The following evaluation of N_r is suggested for a k -th edge:

$$N_r = \text{int}(4m\alpha_k\alpha_{k+1}) \quad (m \geq 1) \quad (27)$$

Finally, it is worth noting that this number can be substantially optimized if the information on the current local distribution of the Pareto set is taken into account. For example, if a Pareto solution appears to be at an edge of the polygon, no additional rotation is needed and $N_r = 0$.

The approach described above guarantees catching the entire Pareto surface. In some very special cases, like the one shown in Figure 4, local Pareto points (e.g., a point \mathbf{P}) may be obtained which are not global Pareto solutions. The algorithm excludes such points easily. Let us locate the box D in such a way that the point M is at some point \mathbf{P} investigated as a candidate Pareto solution and set $A = I$ in (11), (17). If the point is a global Pareto solution (e.g., a point \mathbf{P}'), no any other solution can be obtained. It immediately follows from the contact theorem. Thus, we have a criterion for verification if the solution is a global Pareto solution.

To avoid undesirable severe skewing of the search domain in the algorithm one may recommend preliminary scaling of the objective functions:

$$F_i^{sc} = \frac{F_i - F_{i,\min}}{F_{i,\max} - F_{i,\min}} \quad (28)$$

4 EXAMPLE

The method described above is validated using a few test cases. The test cases include examples with both Pareto convex and concave frontiers. It is shown that the standard realization of the PP may lead to a high sensitivity of the location of the Pareto points to the displacement of box D (or D^*) in the case of a concave Pareto frontier.

Example 1:

First, we consider the following simple algebraic test cases:

$$\min (x, y)^T \quad (29)$$

case one, constraints:

$$x^2 + y^2 \leq 1, x > -1, y > -1 \quad (30)$$

case two, constraints:

$$x^2 + y^2 \geq 1, x \geq 0, y \geq 0 \quad (31)$$

In the first case, (29) and (30), the Pareto surface is convex while in the second case, (29) and (31), it is concave. The latter case also represents an example where the standard definition of an anchor point leads to non-uniqueness. In this case any point with coordinates $x = 0, y \geq 1$ is a solution of the single-objective minimization problem: $\min_{x,y} x$. In our formulation we have only two anchor points in total: (0,1) and (1,0).

The solution of problem (29), (30) is shown in Figure 5. An even displacement of point M (or box D) along the utopia line does not lead to a completely even distribution of the Pareto points since there are gaps nearby the anchor points. In the case of shrinking the search domain via transform (9) we obtain an even representation of the Pareto frontier (Figure 6).

In problem (29), (31) the Pareto frontier is concave. The displacement along the utopia line does not provide any solution, except the utopia points because box D is in the unfeasible space $\mathbf{Y} \setminus \mathbf{Y}^*$. If the displacement of point M is performed along the line parallel to the utopia line as shown in Figure 7, a complete Pareto frontier can be obtained. Yet, this Pareto set is not evenly distributed. The performance of the algorithm is demonstrated in Figure 8. In this figure, the contour plots of the AOF are located inside the search domain. The surface shows the values of the AOF inside box \tilde{D} . The complete solution is represented in Figure 8 along with the contour lines located in the corresponding boxes \tilde{D} for each of the utopia plane points. If a solution is not found (as shown in Figure 7), the direction of the search is switched to the opposite one.

The developed method has been verified on a number of test cases in [7].

5 CONCLUSIONS

The Physical Programming method has been modified to generate an even distribution of the entire Pareto set. The modification is based on the generalization of the class-functions which leads to shrinking of the search domain. The orientation of the search domain in space can be easily conducted. It allows the method to provide an even distribution of the entire Pareto surface. The generation is performed for both convex and non-convex Pareto frontiers. The method does not generate non-Pareto solutions. A simple algorithm has been proposed to remove local Pareto solutions which are not the global Pareto solutions.

7 ACKNOWLEDGEMENTS

This research was conducted as part of VIVACE (Value Improvement through a Virtual Aeronautical Collaborative Enterprise)- integrated project AIP3 CT-2003-502917, partly sponsored by the Sixth Framework Programme of the European Community under priority 4, "Aeronautics and Space".

REFERENCES

- [1] Das, I.: WCSMO-3 Proceedings, Buffalo, NY, March, 1999.
- [2] Das, I, Dennis, J.E.: Proceeding of ASME Design Automation Conference, Montreal, Quebec, Canada, Sept. 17-20, 1989.

- [3] Messac, A.: AIAA Journal, 34, N1, 1996.
- [4] Messac, A., Ismail-Yahaya, A., Mattson, C.A.: Structural and Multidisciplinary Optimization, 25, 2, 2003.
- [5] Messac, A., Mattson, C.: AIAA J., 2004, 42, 10.
- [6] Messac, A., Mattson, C.: Optimization and Engineering, 3, December 2002.
- [7] Guenov, M.D., Utyuzhnikov, S.V., Fantini, P.: EUROGEN 2003, Proceedings of the Sixth Conference on Evolutionary Methods for Design, Optimization and Control with Application to Industrial and Societal Problems, Barcelona, September 12-14, 2005, CIMNE, Munich, Germany

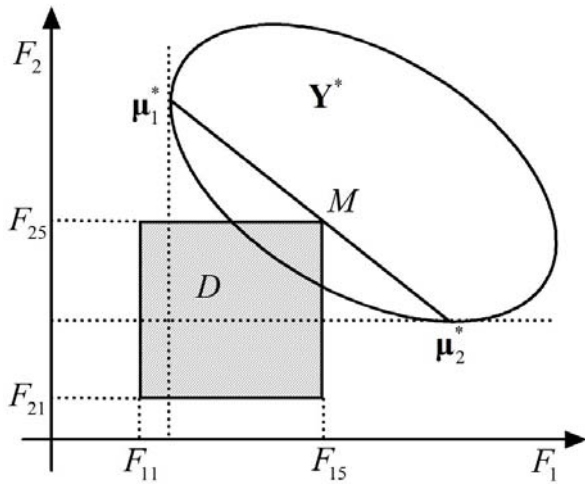


Fig. 1: Original search domain in the objective space.

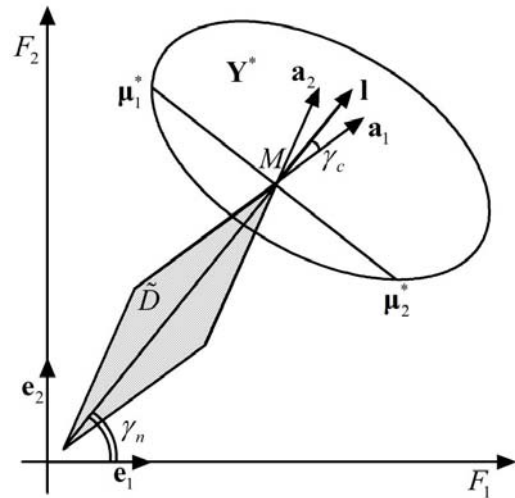


Fig. 2: New search domain after transform

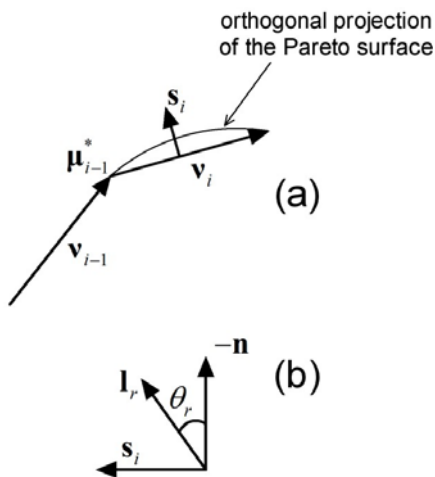


Fig. 3: Rotation of the search domain.

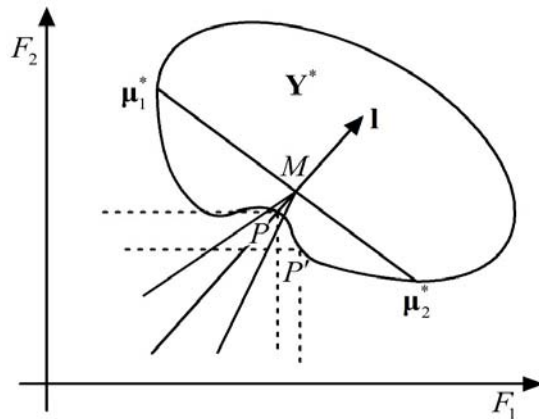


Fig. 4: Verification of global Pareto.

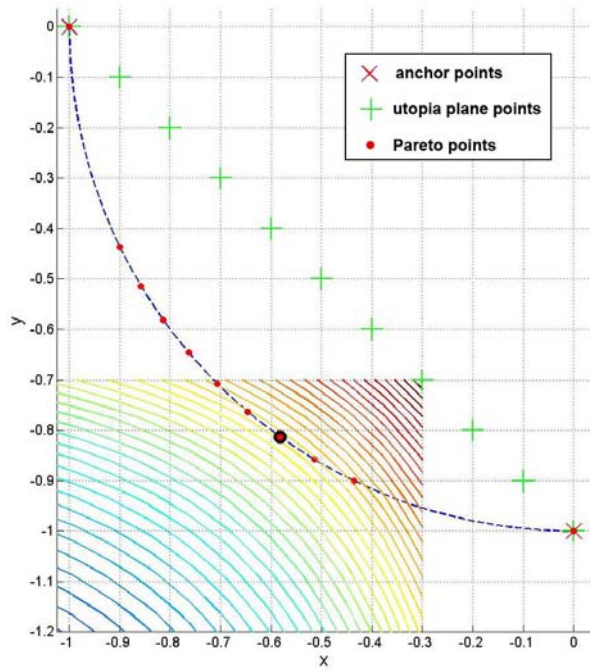


Fig. 5: Standard search domain.
Convex Pareto frontier.

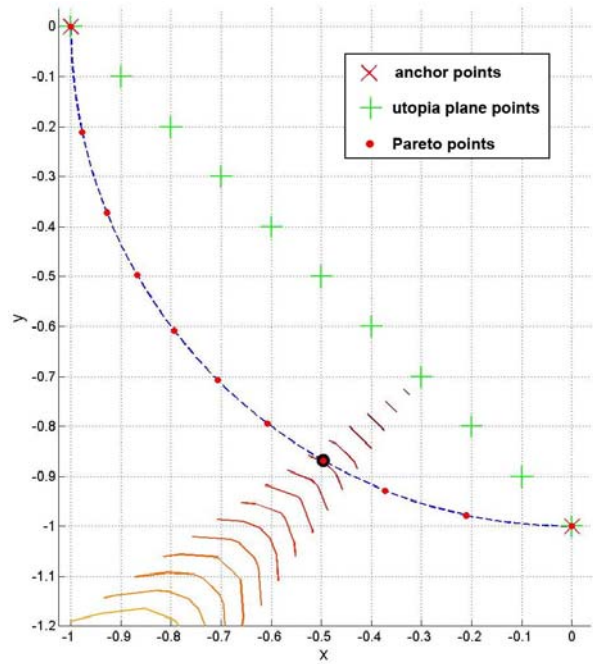


Fig. 6: Transformed search domain. Convex Pareto frontier.

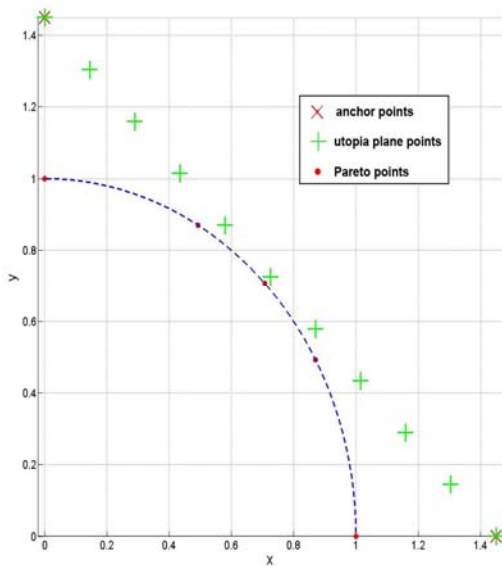


Fig. 7: Standard search domain.
Concave Pareto frontier.

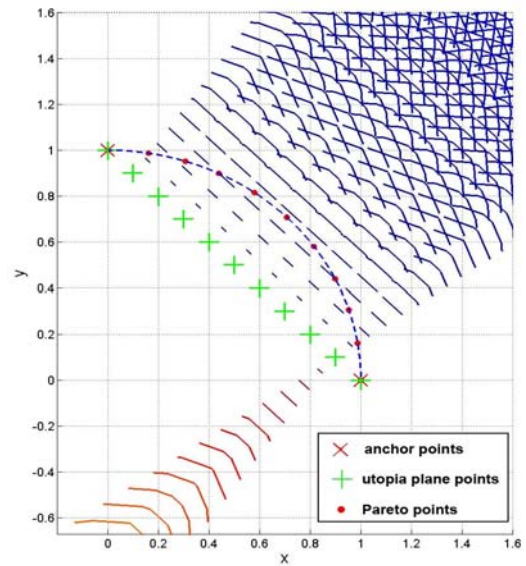


Fig. 8: Transformed search domain. Concave Pareto frontier.