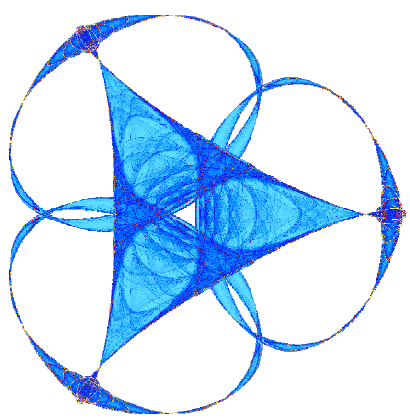
VISUALIZING THE PARETO SURFACE

By

Bamdad Hosseini, Guoqing Liu, Charles Puelz, Samantha Tracht, and Mikhail Smilovic

IMA Preprint Series #2401

(June 2012)



INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS
UNIVERSITY OF MINNESOTA
400 Lind Hall
207 Church Street S.E.
Minneapolis, Minnesota 55455-0436
Phone: 612-624-6066 Fax: 612-626-7370
URL: http://www.ima.umn.edu

Visualizing the Pareto Surface

Bamdad Hosseini[1], Guoqing Liu[2], Charles Puelz[3], Samantha Tracht[4], and Mikhail Smilovic[5]

[1]Simon Fraser University
[2]University of Pittsburgh
[3]Rice University
[4]University of Tennessee
[5]McGill University

June 29, 2012

Abstract

We introduce the existing Normal-Boundary Intersection method for approximating the Pareto surface of a multiobjective optimization problem, and suggest several extensions that resolve regions of interest. In particular, we focus on resolving regions of high curvature, and regions close to a particular area of interest in objective space. We also introduce the Sweeping method, another very natural approach to construct the Pareto surface.

Contents

1 Introduction

We are interested in studying multiobjective optimization problems of the form:

$$\begin{aligned} \min_{x} \quad & F(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_n(x) \end{bmatrix} \\ \text{s.t.} \quad & a(x) = 0 \\ & b(x) \leq 0 \\ & c \leq x \leq d. \end{aligned}$$

Minimizing a vector valued function F means in loose sense that we want to simultaneously get as close to the minima of each component function as possible. If we are a design engineer or a client interested in a multiobjective problem, how do we choose such an x, given that many configurations of objective function values exist? One such approach, often used in industry, is to visualize the set of Pareto optimal points, and apply some qualitative information in determining the best point for the problem at hand. We begin with a definition of a Pareto optimal point.

Definition 1. *A feasible point y is* ***Pareto optimal*** *provided there does not exist another feasible point x such that $f_i(x) \leq f_i(y)$ for all $i = 1, \ldots, n$, with a strict inequality for at least one i [1].*

Understanding, improving, and developing methods to visualize the set of Pareto optimal points, which we will informally call the Pareto surface or curve, is the goal of this exposition.

We will mainly focus on the case $n = 2$. In this case, to gain intuition, we use the toy problem with $f_1 =$ cost, and $f_2 = -$ payload of a rocket fuel tank to be designed [2]. Another example of a biobjective problem worth mentioning is portfolio optimization. In this example, we want to minimize volatility and simultaneously maximize return. Our understanding from a few discussions with people in industry is that the biobjective problem is transformed to a single objective in return, which is maximized for a fixed value of the volatility. This value of the volatility is precomputed based on the client's present portfolio, and perhaps is not always an accurate measure of the clients risk aversion or risk appetite. Instead, by visualizing the pareto surface, we can provide the client with multiple volatility, return values in the form of an approximate Pareto curve instead of a single point in objective space. We hope the reader keeps these two concrete examples in mind.

We will discuss two methods for visualization, the Normal-Boundary Intersection method, and the Sweeping method. In this exposition, we include several refinements to the Normal-Boundary Intersection method based on finding the regions of highest curvature of the Pareto surface, and determining Pareto optimal points which lie close to a point or region of interest in objective space (the utopia point, for example). The Sweeping method arises from the natural idea of minimizing one component objective function f_i, while keeping the others constant.

2 Normal-Boundary Intersection Method

2.1 Description of the method

We describe the Normal-Boundary Intersection method (NBI) for the case $n = 2$. This summary is drawn from [1].

The first step in the algorithm is to construct what we call the *shifted convex hull of individual minima*, or the SCHIM. We compute the vertices of the SCHIM as follows: first, we minimize f_1, and evaluate f_2 at that minimum, call it x_1^*. Likewise, we minimze f_2 and evaluate f_1 at the minimum x_2^*. Then the vertices are $v_1 := [f_1(x_1^*), f_2(x_1^*)]^T$, and $v_2 := [f_1(x_2^*), f_2(x_2^*)]^T$, and the SCHIM is defined as the line connecting these vertices. In the literature, we often reference the convex hull of individual minima, or the CHIM, and this is defined as the all convex linear combinations of the vectors $[f_2(x_1^*) - f_1(x_1^*), 0]^T$ and $[0, f_1(x_2^*) - f_2(x_2^*)]^T$ [1]. From the definition of the CHIM, we see that the SCHIM is equal to the CHIM shifted by the utopia point $u := [f_1(x_1^*), f_2(x_2^*)]^T$. In this exposition we will use CHIM and SCHIM interchangeably since one is simply a translation of the other; the qualitative understanding of the method is not affected by this difference.

Next, we take an equally spaced discretization of the SCHIM given by:

$$\{\alpha_1, \dots, \alpha_N\} \subset \{\gamma v_1 + (1-\gamma)v_2 \text{ s.t. } \gamma \in (0,1)\}$$

Note that the vertices of the SCHIM are not included in this discretization. For a fixed $i \in \{1, \dots, N\}$, we define the NBI subproblem corresponding to α_i as follows:

1. Construct $\hat{n}$ as the unit vector parallel to $u - \alpha_i$.

2. Solve the following optimization problem:

$$\begin{aligned} \max_{t,x} \quad & t \\ \text{s.t.} \quad & \alpha_i + t\hat{n} = F(x) \\ & a(x) = 0 \\ & b(x) \leq 0 \\ & c \leq x \leq d \end{aligned}$$

3. Let t_* be the solution to the NBI subproblem. Then we define $\beta_i = \alpha_i + t_*\hat{n}$.

A few words about the NBI subproblem. What we try to do is determine how far we can move away from the SCHIM, in the direction of $\hat{n}$ (towards the utopian point). The first constraint guarantees that we remain in the image of the feasible set under F, and the last set of constraints are from the original optimization problem.

The NBI Method solves the NBI subproblems for many points on the SCHIM, and then uses the set $\{\beta_1, \dots, \beta_N\}$ to trace out an approximate shape of the Pareto surface. It can be shown that for each Pareto optimal point β, there exists α and t_β such that $\beta = \alpha + t_\beta \hat{n}$ with t_β solving the NBI subproblem corresponding to α. The converse is not true [1].

2.2 Modification 1: The θ method

Solving the NBI subproblems to develop an approximate Pareto curve has its drawbacks. Since we begin with a uniform discretization of the SCHIM, it can be challenging to effectively resolve complicated areas of the curve, such as regions of high curvature. In the context of the fuel tank problem, we would like to isolate regions where the cost changes just a small amount, but the payload rapidly increases, for example. Further, since we generally have no a priori information about the Pareto curve, we cannot select a nice, perhaps nonuniform, mesh of the SCHIM to start. Thus, we would like a way to better resolve interesting regions without having to take an ultra fine and highly expensive mesh (N is very big).

We begin with our first suggested modification of the NBI method. This modification, along with modification 2, resolve regions of the Pareto surface with high curvature.

Assume that the Pareto curve has been approximated using a piecewise linear interpolation of the points $\{\beta_1, \ldots, \beta_N\}$ acquired using NBI. At each point we can define an angle θ as depicted in Figure 1.

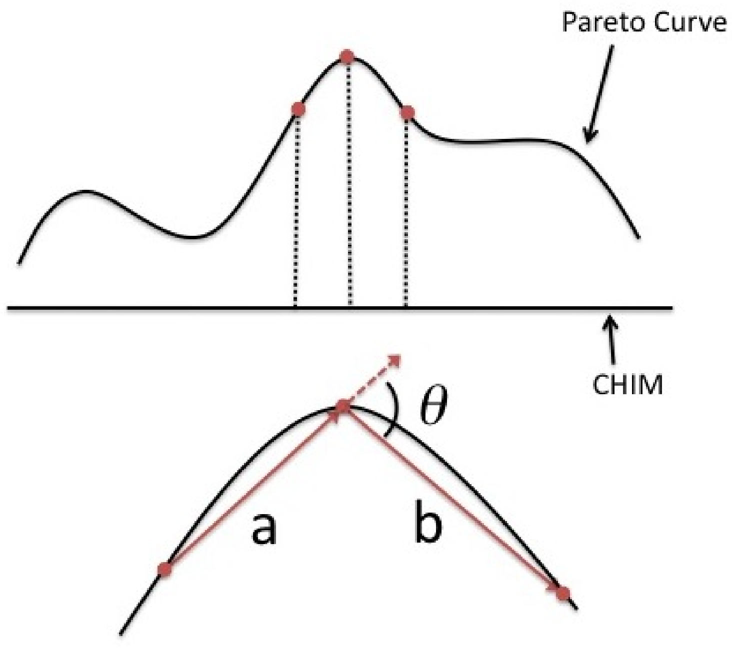


Figure 1: Schematics of the angle theta defined as an indicator of curvature.

This would be the angle between two successive piecewise linear approximations of the Pareto curve at each point. Since Pareto curves in general can be twisted and fold back on themselves, we consider the absolute value of the angle

θ. Therefore, a point with a large magnitude of θ is more likely to have fluctuations. To better resolve these regions of high curvature, we refine the mesh around the corresponding point on the CHIM, and then run the NBI method again with the finer mesh.

Various methods can be used to choose the points where refinements are required. One might wish to choose a portion of all points that have the highest curvature on each refinement. This approach is attractive if we want to have an interactive program that refines the points as long as it is required by the user. Another idea is to define a tolerance *tol* and refine points automatically until $|\theta| \leq tol$ at all points. Figure 2 shows results of our computations of the Pareto curve for the Fuel Tank problem starting with $N = 9$ points and then adding more points in two levels of refinement. In this case, we refined around points with the top 10% values of θ. We added two new generations of points and naturally each generation adds less points compared to the previous generation.

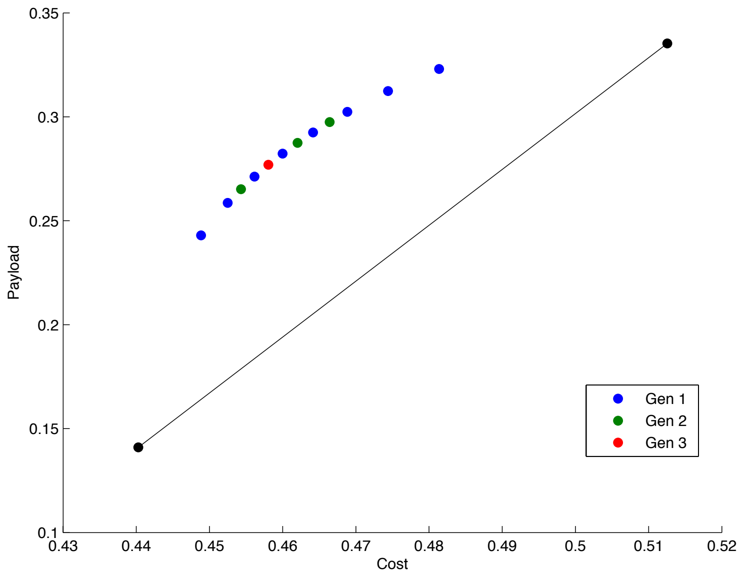


Figure 2: θ method used to determine the Pareto curve for the fuel tank problem formulated in [2].

2.3 Modification 2: Approximation of Curvature

In this section we would like to propose another modification to the NBI method that helps in refining regions of the Pareto surface with high curvature. The algorithm is as follows:

1. Begin with a uniform discretization of the SCHIM, $\{\alpha_1, \ldots, \alpha_N\}$, and solve the corresponding NBI subproblems to get $\{\beta_1, \ldots, \beta_N\}$.

2. For each β_i, compute an approximation to the curvature, c_i, of the Pareto curve at this point. We will discuss two ways to do this computation below.

3. Fix a tolerance *tol* between 0 and 1, let c_{diff} =difference between the maximum and minimum values of the curvature, and let $c_{\text{absmax}} = \max\{|c_1|, \ldots, |c_n|\}$. Then for each i such that

$$\frac{c_{\text{absmax}} - |c_i|}{c_{\text{diff}}} < tol,$$

we refine the original mesh of the SCHIM around α_i, and compute solutions to the NBI subproblems corresponding to the refined mesh. By *refine the mesh*, we take a finer discretization of the line segment:

$$\left\{ \frac{1-t}{2}(\alpha_{i-1} + \alpha_i) + \frac{t}{2}(\alpha_i + \alpha_{i+1}) \text{ s.t. } t \in [0,1] \right\}.$$

We have developed two ways to compute an approximation to the curvature c_i at the point $\beta_i := [\beta_1^{(i)}, \beta_2^{(i)}]^T$ in step 2. Our first way to approximate curvature is as follows. Let

$$m_{+,i} = \frac{\beta_2^{(i+1)} - \beta_2^{(i)}}{\beta_1^{(i+1)} - \beta_1^{(i)}}.$$

and

$$m_{-,i} = \frac{\beta_2^{(i)} - \beta_2^{(i-1)}}{\beta_1^{(i)} - \beta_1^{(i-1)}}.$$

Then we take,

$$c_i = \frac{m_{+,i} - m_{-,i}}{2}. \tag{1}$$

This intuition in this approximation is that a large difference in slopes of the piecewise linear interpolant through the points β_{i-1}, β_i, and β_{i+1} indicate concavity of the curve at the point β_i, with a negative c_i implying concave down and positive c_i implying concave up.

Below is a result of applying this approximation to an example. The following figures were created by making a synthetic Pareto curve, SCHIM, and utopia, and mimicking the NBI method by drawing lines from the SCHIM to the utopia point. Then the β_i's, the blue asterisks on the Pareto curve, are determined by finding the intersection of the line and the synthetic Pareto curve. We then depict regions of refinement as red asterisks.

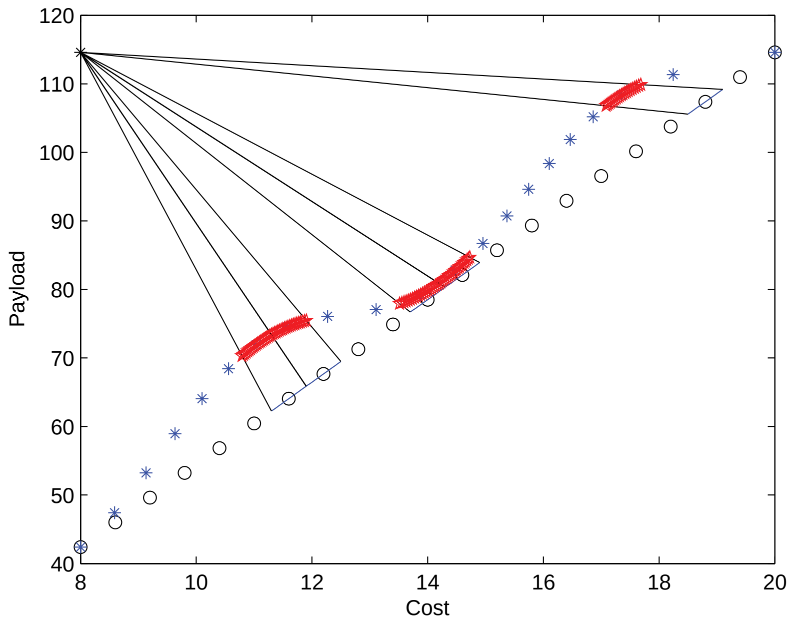


Figure 3: Approximation of curvature (1) with tolerance 0.15.

The second approximation to the curvature we employ is the following. Let l_i be the line orthogonal to the SCHIM comtaining the point β_i, and let b_i be the point of intersection of l_i and the SCHIM. Then we define:

$$d_i = \left\{ \begin{array}{ll} ||b_i - \beta_i||_2 & \text{if } i \in \{1, \dots, N\} \\ 0 & \text{if } i = N+1 \text{ or } 0 \end{array} \right]$$

$$\delta_i = \left\{ \begin{array}{ll} ||b_i - b_{i-1}||_2 & \text{if } i \in \{2, \dots, N\} \\ ||b_i - v_1||_2 & \text{if } i = 1 \\ ||v_2 - b_{i-1}||_2 & \text{if } i = N+1. \end{array} \right]$$

We use these two values to compute the approximate second derivative at each point β_i as follows. Let

$$\sigma_i^+ = \left\{ \begin{array}{ll} (d_{i+1} - d_i)/\delta_{i+1} & \text{if } i \in \{0, \dots, N\} \\ 0 & \text{if } i = N+1 \end{array} \right]$$

$$\sigma_i^- = \left\{ \begin{array}{ll} (d_i - d_{i-1})/\delta_i & \text{if } i \in \{1, \dots, N+1\} \\ 0 & \text{if } i = 0 \end{array} \right]$$

With these two values, σ_i^+ and σ_i^-, for the forward and backward difference respectively, we compute an approximation to the slope at each point in $\{\beta_1, \dots, \beta_N\}$ by taking the average:

$$\bar{\sigma}_i = (\sigma_i^+ + \sigma_i^-)/2, \ i \in \{0, \dots, N+1\}$$

Here, the slope at β_i is approximated by $\bar{\sigma}_i$ for $i \in \{1, \dots, N\}$, and the values $\bar{\sigma}_{N+1}$ and $\bar{\sigma}_0$ are used in the next computation.

We compute an approximation to the curvature, c_i $(i = 1, \ldots, N)$ by computing

$$\kappa_i^+ = (\bar{\sigma}_{i+1} - \bar{\sigma}_i)/\delta_{i+1}$$

$$\kappa_i^- = (\bar{\sigma}_i - \bar{\sigma}_{i-1})/\delta_i$$

And taking the average:

$$c_i = \frac{\kappa_i^+ + \kappa_i^-}{2}. \tag{2}$$

Here is the same example with this approximation of curvature,

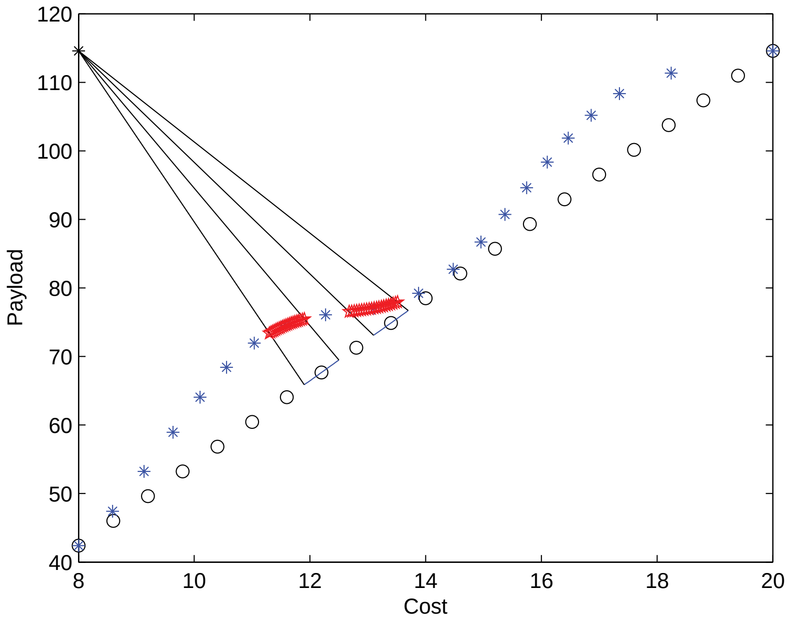


Figure 4: Approximation of curvature (2) with tolerance 0.15.

We compare and contrast these two approximations to the curvature with more figures below.

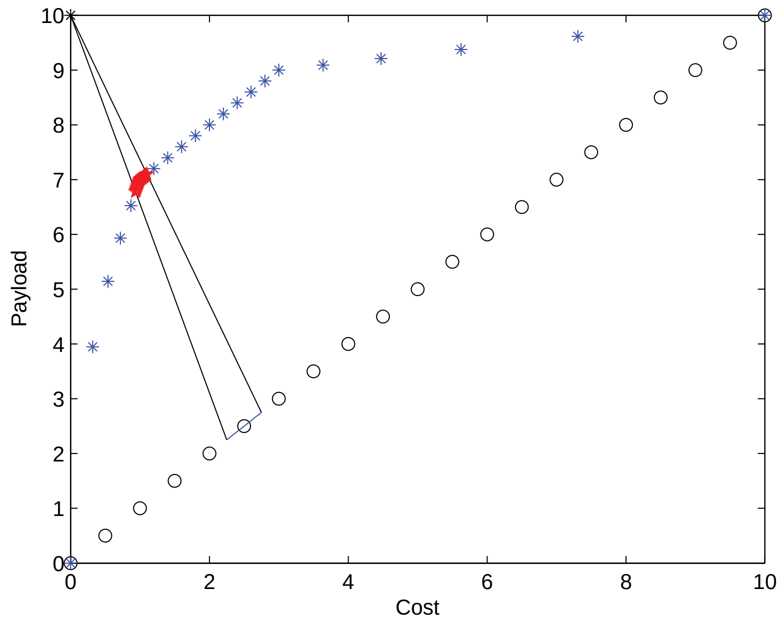


Figure 5: Approximation of curvature (1) with tolerance 0.9.

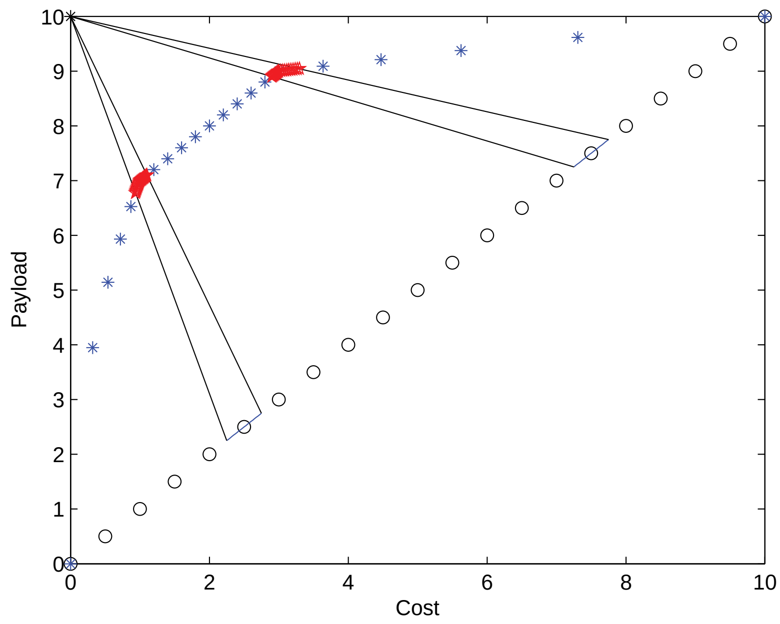


Figure 6: Approximation of curvature (2) with tolerance 0.9.

2.4 Modification 3: Computing the Shortest Distance

We could also focus our attention on a subset of the Pareto curve which is closest to a certain special point or region. One natural selection for this point would be the utopia point, and we run our calculations with this particular point. The third modification to the NBI method arises from this idea. The algorithm is as follows:

1. As before, begin with a uniform discretization of the SCHIM, $\{\alpha_1, \ldots, \alpha_N\}$, and solve the corresponding NBI subproblems to get $\{\beta_1, \ldots, \beta_N\}$.

2. Fix a tolerance *tol* between 0 and 1. Express the vertices of the SCHIM as $v_i = [v_1^{(i)}, v_2^{(i)}]^T$ $(i = 1, 2)$. Let $c_{\mathrm{diff},i} = |v_i^{(1)} - v_i^{(2)}|$. Then for each i such that

$$\sqrt{\left(\frac{\beta_i^{(1)} - v_1^{(1)}}{c_{\mathrm{diff},1}}\right)^2 + \left(\frac{\beta_i^{(2)} - v_2^{(2)})}{c_{\mathrm{diff},2}}\right)^2} < tol$$

   We refine around the point α_i as described in the curvature method, and rerun the NBI method.

As seen above, before we compute the minimum distance to a point, say the utopia point $u = [v_1^{(1)}, v_2^{(2)}]$, we shift all the points $\{\beta_1, \ldots, \beta_N\}$ by the utopia point, and then scale each axis to ensure the maximum value attainable is 1. The SCHIM under this transformation becomes the line segment $\{te_1 + (1-t)e_2\}$, where e_1 and e_2 are unit vectors corresponding to the x and y axes. This scaling in each axis ensures that a computed distanced equally weighs the contribution of each objective function, and converts the values on each axis to dimensionless values. For example, in the fuel tank problem, in order to avoid comparing a payload value with a cost value, we need to make the axes dimensionless.

Below we include some figures in which we minimized the transformed distance to the utopia point.

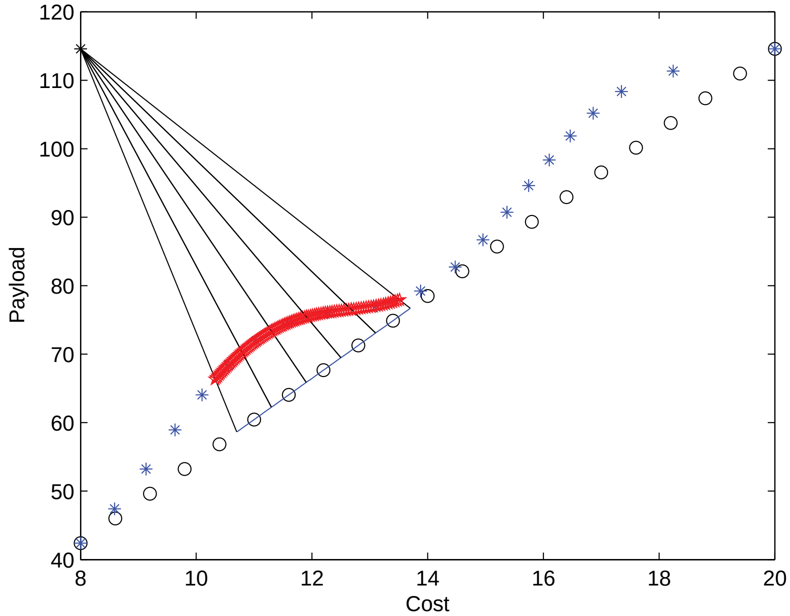


Figure 7: Approximation of distance with tolerance 0.15.

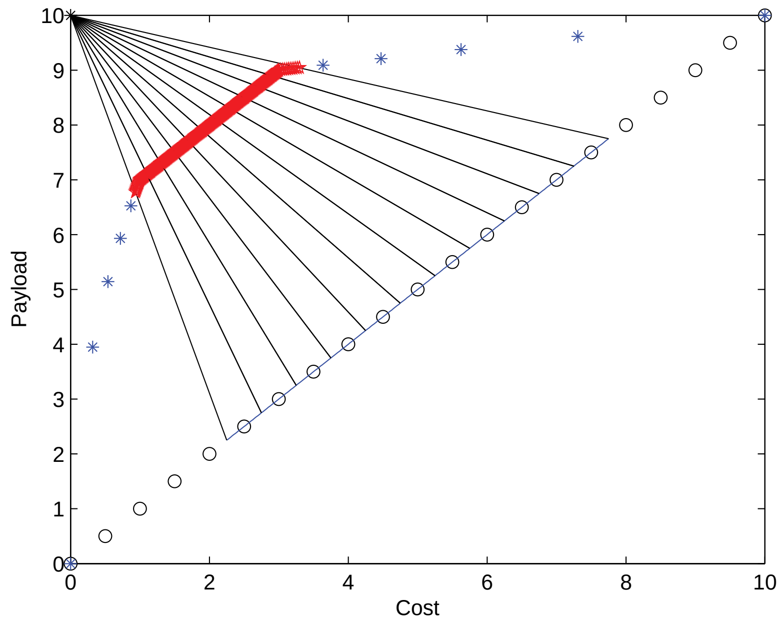


Figure 8: Approximation of distance with tolerance 0.1.

2.5 Modification Details: The Tolerance

In this section, we will discuss the choice of the tolerance parameter *tol*. Our goal is to determine interesting points of high curvature or optimal location on the Pareto curve with respect to a region of interest. Now, perhaps we have other requirements, such as a priori knowledge of a minimum payload we must achieve. In other words, one may prefer points with high curvature *and* values of f_1 or f_2 less than some known value. In order to satisfy this kind of requirement, we can increase the tolerance to obtain more points around which we wish to refine. Taking the curvature as an example, different points actually have different priority. While increasing the tolerance from 0 to 1, the points are being captured according to their magnitude of approximated curvature. The following figures show the change of points depending on the value of the tolerance.

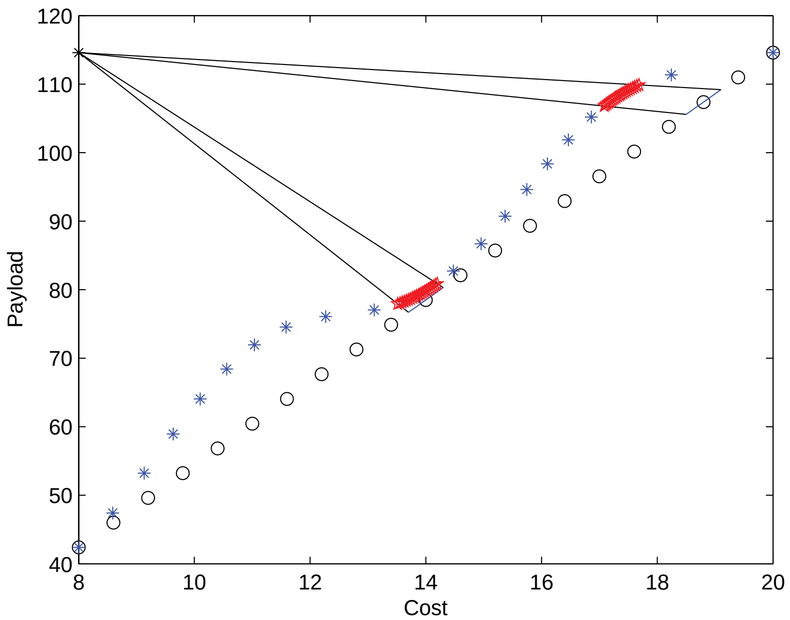


Figure 9: Approximation of curvature (1) with tolerance 0.05.

From the figures, as we increase the tolerance from 0.05 to 0.25, more turning points are chosen. This approach give us more potentially interesting points.

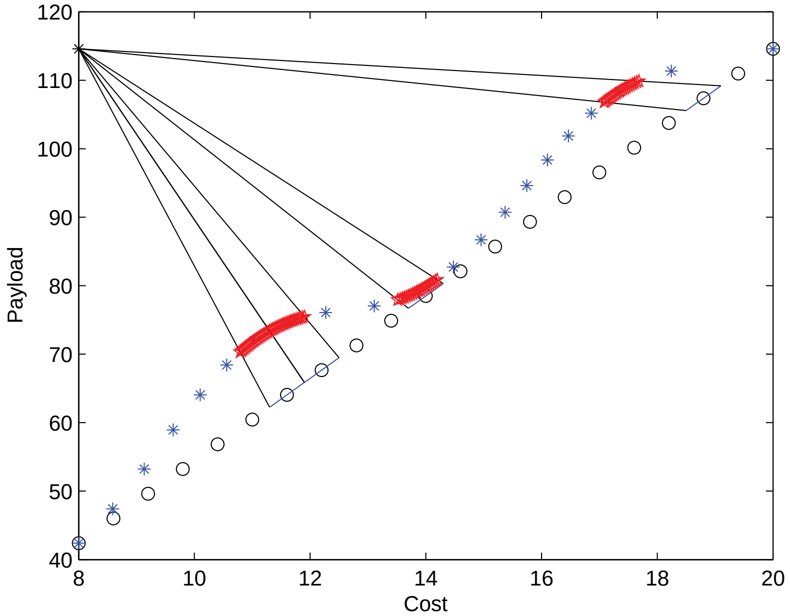


Figure 10: Approximation of curvature (1) with tolerance 0.10.

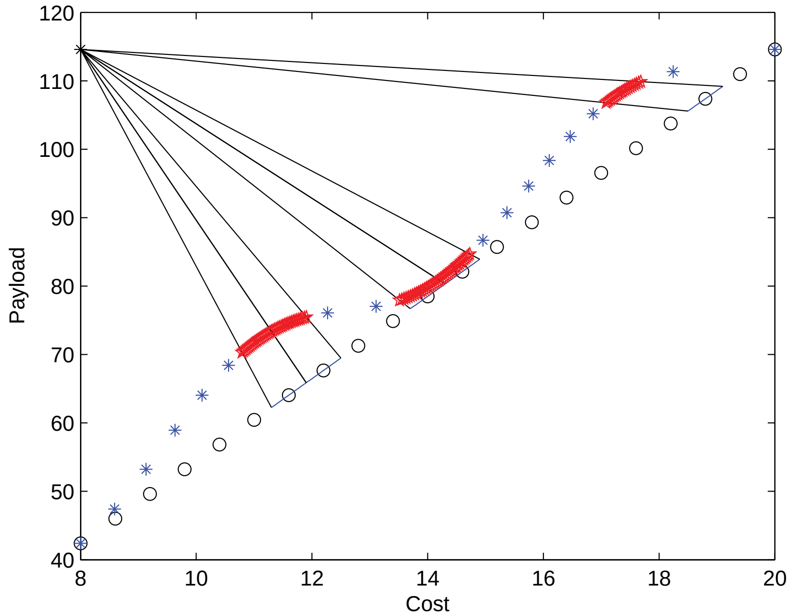


Figure 11: Approximation of curvature (1) with tolerance 0.15.

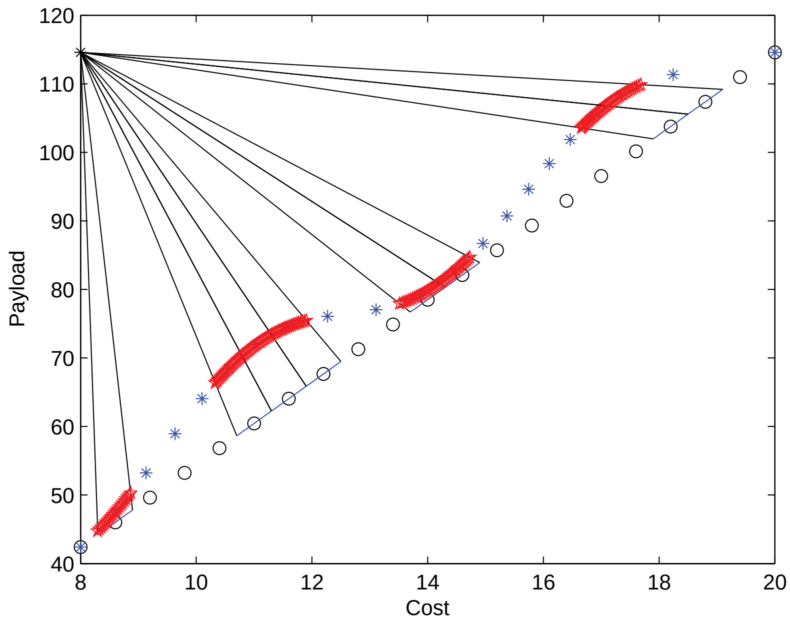


Figure 12: Approximation of curvature (1) with tolerance 0.20.

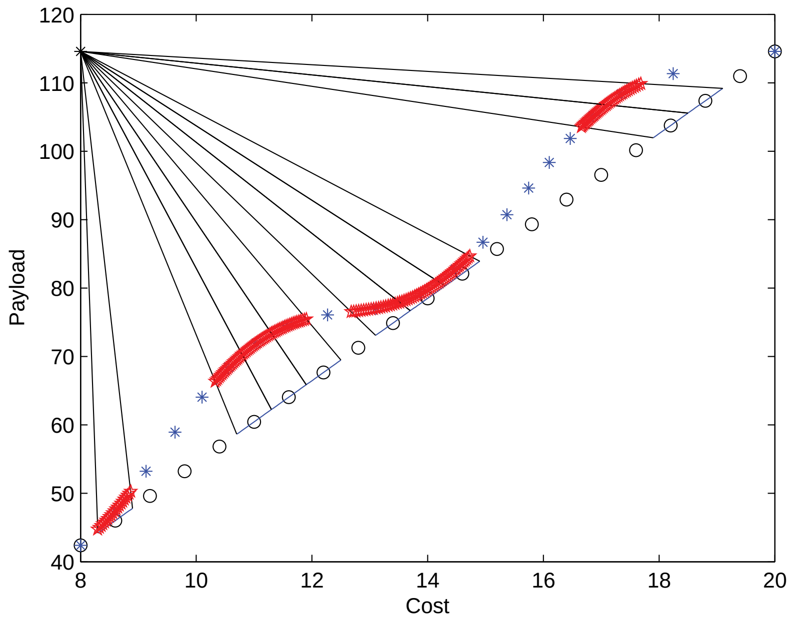


Figure 13: Approximation of curvature (1) with tolerance 0.25.

3 Sweeping method: an Alternative to NBI

3.1 Biobjective optimization

In the setup of NBI in two dimensions, we compute the values $f_1(x_1^*)$, $f_1(x_2^*)$, $f_2(x_1^*)$, and $f_2(x_2^*)$. We are then interested in the part of the Pareto curve in the set

$$B = [f_1(x_1^*), f_1(x_2^*)] \times [f_2(x_2^*), f_2(x_1^*)].$$

Without loss of generality, we can assume that B sits at the origin. Explicitly,

$$f_1(x_1^*) = f_2(x_2^*) = 0,$$

$$f_1(x_2^*) \leftarrow f_1(x_2^*) - f_1(x_1^*), \text{ and}$$
$$f_2(x_1^*) \leftarrow f_2(x_1^*) - f_2(x_2^*).$$

The fact that we are interested in the information inside B leads to a natural algoritm we call the Sweeping method. Beginning at the origin, we walk along either axis, fixing points and then calling the optimizer with the other axis as our objective function. What results is a sweeping action parallel to the other axis. If necessary, the method can be repeated with the axes in opposite roles. The union of the two sets of optimizer-given points is our desired Pareto curve.

The idea is formulated as follows:

Following in the footsteps of NBI, we choose a discretization size N. We then move along the f_j axis (with the other axis denoted f_k):

$$f_j^{fixed} = \frac{i}{N+1}(f_j(x_k^*) - f_j(x_j^*)), \quad 0 \leq i \leq N+1$$

and call the optimizer with $f_1 - f_1^{fixed}$ as an added equality constraint.

3.2 Higher Dimensions

Given n objective functions, we expand on the previous section by considering the information inside the n-dimensional cube (once again assuming a shift to the origin) given by

$$[f_1(x_1^*), \max_{1 \leq m_1 \leq n} f_1(x_{m_1})] \times \ldots \times [f_n(x_n^*), \max_{1 \leq m_n \leq n} f_n(x_{m_n})].$$

The sweeps are illustrated as follows. Let us consider a triobjective optimization problem and decide to first walk along the f_1 axis. Then with an f_1 value fixed, we must choose to walk along one of the other axes. Let us assume we have f_3 as our objective function, so we walk along the f_2 axis. Then for each step along the f_1 axis, we walk along the entire f_2 axis, and send to the optimizer the equality constraints $f_1 - f_1^{fixed} = f_2 - f_2^{fixed} = 0$. For each sweep there are therefore N^2 calls to the optimizer giving an f_3 value in each case (N is the chosen discretization size).

3.3 Sphere example

Let us define a triobjective optimization problem as follows:

$$
\begin{aligned}
\min_{x} \quad & F(x, y, z) = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\
\text{s.t.} \quad & x^2 + y^2 + z^2 \geq 1 \\
& x + y \geq \frac{1}{\sqrt{3}} \\
& x + z \geq \frac{1}{\sqrt{5}} \\
& x, y, z \geq 0
\end{aligned}
$$

The utopia point is the origin and the resulting Pareto curve is the unit sphere restricted to the first octant, with two modifications. This Pareto curve was constructed with the Sweeping method. The challenge in solving this problem with NBI is determining a *good* choice for the CHIM. Instead of dealing with three minima, one for each objective, a triobjective problem can result in up to six minima. One must take into consideration the order in which the objectives are minimized. For example, if we wish to minimize x, then y and lastly z, the resulting minima is $\left(0, \frac{1}{\sqrt{3}}, \sqrt{\frac{2}{3}}\right)$. If we reverse the order of x and y we get $\left(\frac{1}{\sqrt{3}}, 0, \sqrt{\frac{2}{3}}\right)$.

The set of minima for the sphere example:

$$
\left\{ \left(0, \frac{1}{\sqrt{3}}, \sqrt{\frac{2}{3}}\right), \left(\frac{1}{\sqrt{3}}, 0, \sqrt{\frac{2}{3}}\right), \left(0, \frac{2}{\sqrt{5}}, \frac{1}{\sqrt{5}}\right), \left(\frac{1}{\sqrt{5}}, \frac{2}{\sqrt{5}}, 0\right), \left(1, 0, 0\right) \right\}.
$$

One solution to this challenge in choosing a CHIM is to consider all possibilities: $\binom{5}{3}$ CHIMS. For each three points, we construct the CHIM as the convex set formed by the points, and run NBI. This is highly computationally infeasible, and unnecessary as there are overlaps in the CHIMS. Further work must be done to understand which CHIMS should be chosen to minimize the number used. Another method for solving this with NBI would to simply produce the following three pseudo-minima,

$$
(\max_{i\in\{2,3\}} f_1(x_i^*), f_2(x_2^*), f_3(x_3^*)),\ (f_1(x_1^*), \max_{i\in\{1,3\}} f_2(x_i^*), f_3(x_3^*)),
$$
$$
(f_1(x_1^*), f_2(x_2^*), \max_{i\in\{1,2\}} f_3(x_i^*)),
$$

not all necessarily in the space of feasibility, and construct the CHIM as the convex set (triangle) formed by these three points. A drawback of this approach is that the holes in the feasibility space may leave a given optimizer running to its maximum number of iterations. Further work must be done on this notion of tailoring the CHIM around the holes; perhaps using the inequality constraints to cut off certain areas of the plane could be useful.

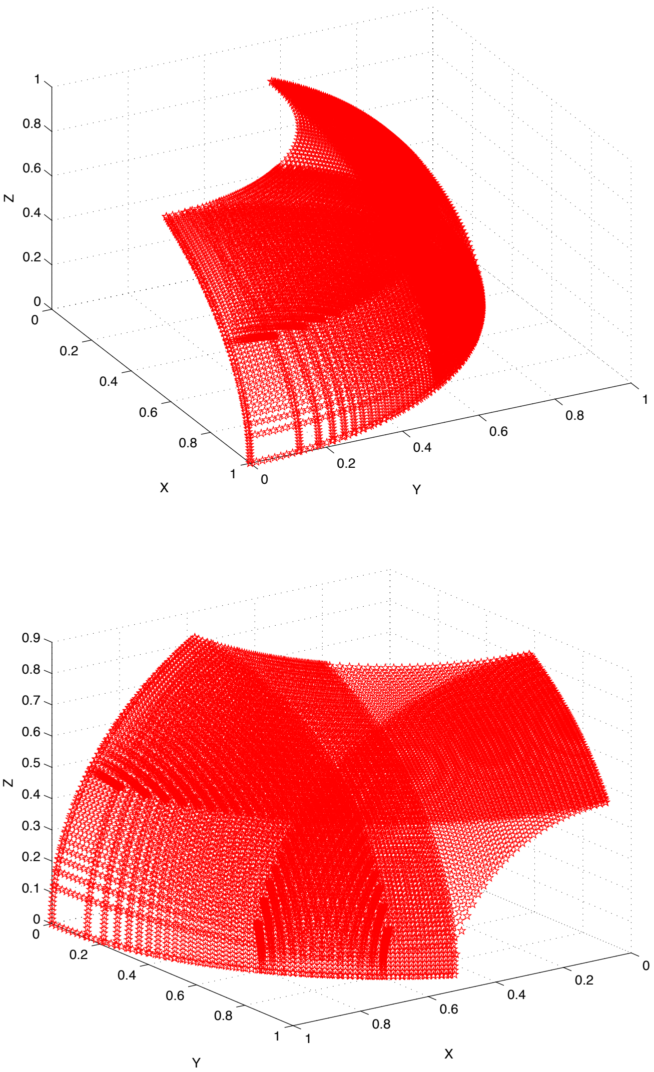


Figure 14: Two views of the Pareto curve for the Sphere example.

4 Conclusion

Understanding the shape of the Pareto curve is a crucial part of industrial projects seeking to minimize several objectives. We have discussed two methods for visualization, the NBI method, with several improvements, and the Sweeping method. We hope to further understand the efficiency of each method in terms of function evaluations, and continue tweaking and building adaptive modifications to the NBI algorithm.

5 Acknowledgments

The authors are extremely grateful for guidance from their mentor, Dr. Laura Lurati, and to IMA and PIMS for providing a wonderful research experience.

References

[1] Das I., Dennis J. Normal-boundary intersection: an alternate method for generating Pareto optimal points in multicriteria optimization problems. ICASE 1996.

[2] Schuman T., Weck OL. Integrated system-level optimization for concurrent engineering with parametric subsystem modeling. 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference. AIAA 2005-2199.

[3] Cramer E., Dennis J.E., Frank P., Lewis R.M., Subin G. Problem Formulation for Multidisciplinary Optimization. Rice University Center for Research on Parallel Computation Technical Report 93334. August 1993.