Understanding Knee Points in Bicriteria Problems and Their Implications as Preferred Solution Principles

Kalyanmoy Deb and Shivam Gupta
Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur
PIN 208016, Uttar Pradesh, India
{deb,sgupta}@iitk.ac.in

KanGAL Report Number 2010005

July 5, 2010

Abstract

A knee point is almost always a preferred trade-off solution, if it exists in a bicriteria optimization problem. In this paper, we make an attempt to better understand knee points and investigate the properties of bicriteria problems that may exhibit a knee on its Pareto-optimal front. Specifically, we review past studies and suggest a couple of new definitions of a knee point. One of the definitions allows us to define a knee region for problems in which, instead of one, a set of knee-like solutions exist. We also introduce edge-knee solutions which behave like a knee solution but lies near one of the extreme solutions. We argue here that knee solutions have far-reaching implications than they are currently known for. It is an interesting fact that in many problem-solving tasks, despite the existence of a number of solution methodologies, one or a few methodologies are commonly used. Here, we argue that often such common solution principles are knee solutions to a bicriteria problem formed from two conflicting goals of the underlying problem-solving task. We illustrate our argument by show-casing a number of popularly used problem-solving tasks, such as regression, sorting, clustering and a number of engineering design tasks. Each task, when viewed as a bicriteria problem, seems to exhibit a knee or a knee-region and the commonly used methodology seems to lie within the knee-region. Linking preferred solution methodologies as knee solutions to a related bicriteria problem is certainly an interesting idea and may have a long-term implication in the development of efficient solution methodologies for different scientific and other problem-solving tasks, such as game-playing and strategic decision-making activities.

1 Introduction

Multi-objective optimization involves consideration of number of conflicting objectives. Theoretically, such problems give rise to a set of optimal solutions, known as Pareto-optimal solutions, which together constitute a trade-off front. In most cases, the front obtained by the Pareto-optimal solutions are such that the choice of a single preferred solution is not straightforward and a multi-criteria decision-making methodology is needed to systematically choose a single preferred solution. However, there exists certain multi-objective optimization problems which exhibit a *knee* point on their Pareto-optimal fronts. A knee point is almost always the most preferred solution, since it requires an unfavorably large sacrifice in one objective to gain a small amount in the other objective. Knee points are well-recognized by multi-objective optimization researchers [3, 4, 5, 8, 13, 14, 15, 16, 18]. Due to their natural preference compared to other Pareto-optimal solutions, some evolutionary

optimization methodologies have been designed to find knee point(s) [3, 4, 8, 15, 16, 18]. However, a detailed study describing different plausible types of knee points and importantly addressing the issue on why some problems exhibit a knee solution is missing. In this paper, we review the existing definitions of a knee point and suggest a couple of new definitions for identifying a knee point in a bicriteria optimization problem. The definitions allow us to describe other knee-like points which have not been categorized well in the multi-objective literature.

It is well argued that a knee point is a natural preferred solution, if exists in a multi-objective optimization problem. In the subsequent part of this paper, we make an interesting and useful connection of a knee point with preferred solution methodologies often used in certain problem-solving tasks in practice. Scientists and engineers deal with different problem-solving tasks regularly, such as finding a mathematical regression equation fitting a set of two-dimensional data, clustering a set of two or three-dimensional data into a number of clusters, sorting a set of numbers in ascending order, finding root of an equation numerically, and others. In trying to solve such problems, there is usually one or two methods which are commonly preferred in practice. We tried to find a link between preferring specific methodologies in a problem-solving task with preferring knee points in a bicriteria optimization task. We argue that a plausible reason for why some methodologies are preferred over other methods may come from this linking of a problem-solving task with an equivalent bi-objective problem and realizing that the preferred method may lie on the knee-region of the corresponding trade-off front.

In the remainder of this paper, we review the existing definitions of a knee point. Thereafter, we suggest a couple of definitions based on a *bend-angle* based approach and another approach that is dependent on an user's preferred trade-off information. We describe systematic procedures for identifying a knee point and illustrate their working principle on a tunable test problem exhibiting a knee point. In some problems, instead of a sole knee point, there may exist a set of closely-packed trade-off points that together qualify as a *knee-region*. Moreover, in certain problems, the knee point may exist near one of the extreme points of the trade-off front. We distinguish such point(s) as *edge-knee* point(s). Based on gradient information, we then suggest some properties which may cause a trade-off front to exhibit a knee point. This allows us to suggest a procedure for designing bicriteria test problems having a knee point. In order to facilitate an user to estimate the trade-off parameters in tune with the given trade-off frontier, we have also suggested a upper bound estimation procedure. In the remaining part of the paper, we consider a number of generic and engineering problems solving tasks and demonstrate that commonly-used solution principles lie near the knee-region of the associated bi-objective Pareto-optimal set. The concept of linking preferred solution principles as knee solutions of a derived bicriteria problem opens up interesting and far-reaching implications for developing efficient solution methodologies. Conclusions of the study are made thereafter.

2 A Knee Solution in a Bi-Objective Optimization Problem

Here, we restrict ourselves for minimization of two conflicting objectives only ($f_i : \mathcal{S} \to \mathbf{R}$, $i = 1, 2$) as functions of decision variables $\mathbf{x}$:

$$\begin{array}{rl} \text{minimize} & \{f_1(\mathbf{x}), f_2(\mathbf{x})\}, \\ \text{subject to} & \mathbf{x} \in \mathcal{S}, \end{array} \tag{1}$$

where $\mathcal{S} \subset \mathbf{R}^n$ denotes the set of feasible solutions. A vector consisting of objective function values calculated at some point $\mathbf{x} \in \mathcal{S}$ is called an objective vector $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}))^T$.

2.1 Pareto-Optimal Points

Problem (1) gives rise to a set of *Pareto-optimal* solutions or a Pareto-optimal front (P^*), providing trade-off between two objectives. The *domination* between two solutions is defined as follows [7, 14]:

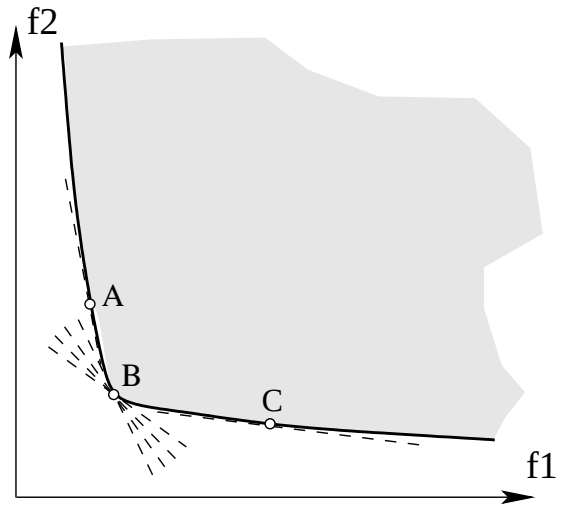


Figure 1: The utility based definition of a knee point is illustrated.

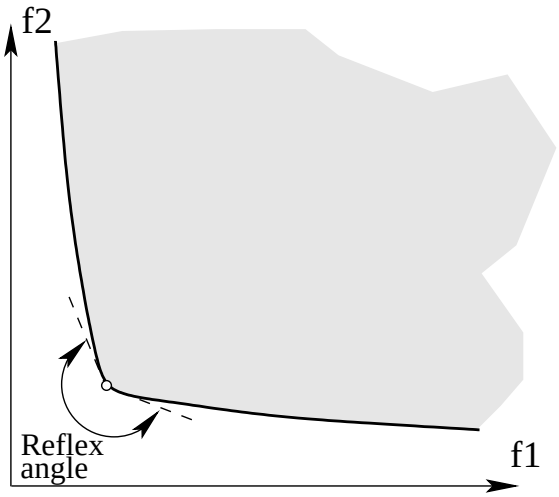


Figure 2: The reflex angle based definition of a knee point is illustrated.

Definition 1 *A solution* $\mathbf{x}^{(1)}$ *is said to dominate the other solution* $\mathbf{x}^{(2)}$*, if (i) the solution* $\mathbf{x}^{(1)}$ *is no worse than* $\mathbf{x}^{(2)}$ *in all objectives (that is, in the case of a minimization problem,* $f_i(\mathbf{x}^{(1)}) \leq f_i(\mathbf{x}^{(2)})$ *for all* $i = 1, 2$*) and (ii) the solution* $\mathbf{x}^{(1)}$ *is strictly better than* $\mathbf{x}^{(2)}$ *in at least one objective (that is, in the case of a minimization problem,* $f_i(\mathbf{x}^{(1)}) < f_i(\mathbf{x}^{(2)})$ *for at least one index* i*).*

Pareto-optimal solutions are then defined as follows [14]:

Definition 2 *A decision vector* $\mathbf{x}^* \in \mathcal{S}$ *is* Pareto-optimal *if there does not exist another decision vector* $\mathbf{x} \in S$ *that dominates* $\mathbf{x}^*$ *according to Definition 1.*

2.2 Previous Definitions of a Knee Point

A previous study by Branke et. al.[4] for identifying a knee point used a couple of ideas. The first definition was based on a linear marginal utility function: $U(\mathbf{f}, \lambda) = \lambda f_1 + (1 - \lambda) f_2$ with $\lambda \in [0,1]$ and $\mathbf{f} = (f_1, f_2)^T$. The idea works as follows. First, a uniform set of weight values (λ) are chosen in $[0, 1]$ and the best solution (having the minimum value of U) is identified for each λ. Following definition is then used to define a knee point:

Definition 3 *Let the minima-count of every Pareto-optimal solution is set to be zero initially. For every* λ_i *created uniformly in* $[0, 1]$*, a Pareto-optimal solution* j *having the minimum* $U(\mathbf{f}, \lambda_i)$ *is identified and its minima-count* c_j *is incremented by one. After all* $\lambda_i \in [0, 1]$ *are considered, the Pareto-optimal solution having the maximum value of minima-count is defined as the knee point.*

For a practical use of the above definition, a finite but a large set of λ_i values can be chosen. Figure 1 illustrates the scenario at which the knee[1] point (marked as B) is minimum for a larger number of utility functions than any other point on the Pareto-optimal front.

The second definition suggested in the study was based on finding two or more neighboring solutions and calculating the *reflex angle*, as shown in Figure 2:

[1]The word 'knee' is appropriate for a scenario in which both objectives are to be maximized, as then the front would look like a bended leg and the point of interest would lie on the knee part of the bended leg. In our opinion, the word 'elbow' is more appropriate for minimization problems, but due to its wide-spread use, we shall continue to refer to these solutions as 'knee' points, even for minimization problems.

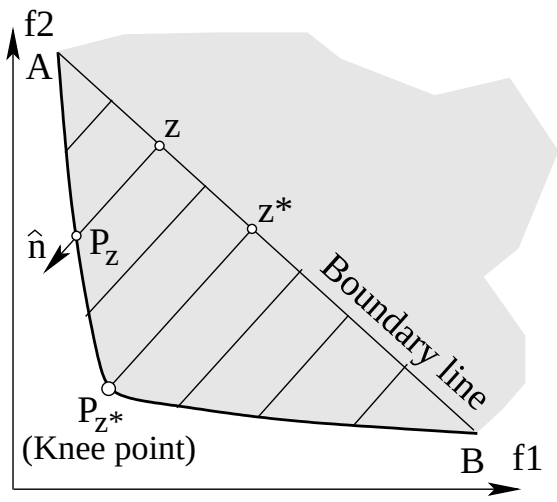


Figure 3: Knee point identification by the normal boundary intersection method.

Definition 4 *A knee point is the Pareto-optimal point having the maximum reflex angle computed from its neighbors.*

The reflex angle at a point on the Pareto-optimal front denotes the bend of the front from its left to its right and hence can be a measure of gain-sacrifice trade-off. However, since the reflex angle is measured using two neighboring points, it is a *local* property of the Pareto-optimal front and may not provide a global trade-off measure that truly defines a knee point. We shall modify this definition in one of our suggestions in the next section.

Das [5] defined a knee point from the concept of normal boundary intersection method illustrated in Figure 3.

Definition 5 *Given a Pareto-optimal front, where the objectives are normalized, two extreme points p_1 and p_2 are obtained to construct a boundary line $L(p_1, p_2)$. Thereafter, for any boundary point $\mathbf{z}$ (on the boundary line), the Pareto-optimal point ($P_{\mathbf{z}}$) along the normal ($\hat{n}$) of the boundary line is identified. The Pareto-optimal point ($P_{\mathbf{z}^*}$) with the maximum distance from its corresponding boundary point (z^*) along the normal direction is defined as the knee point.*

A couple of studies implemented the idea in an algorithm to find the knee point, instead of the usual procedure of finding the entire Pareto-optimal front. Bechikh et. al. [3] proposed an algorithm, *KR-NSGA-II* which is an extension of the reference point NSGA-II procedure [8]. It uses the normal boundary intersection method (Definition 5) to emphasize knee-like points. Schütze et. al. [18] suggested an archiving strategy which can be used in combination with any stochastic search algorithm to find the knee point or the knee-region present in a problem. By comparing trade-off solutions, the methodology attempts to find the two extreme points and an intermediate point that has a maximum distance from the line joining the extreme points.

Mattson et. al. [13] suggested a filtering approach which picks solutions with certain trade-off requirement supplied by the user. Although they did not suggest the idea for locating a knee point, the idea can be potentially used for this purpose.

As it is clear from the above description that most of the studies related to knee points are dedicated in utilizing certain properties of a knee point and efforts have been made in integrating the properties with an multi-objective optimization algorithm to find the knee point alone, instead of finding the entire trade-off frontier. We argue here that there are two limitations to these studies. First, most definitions for a knee point are defined in a local sense, that is, using neighboring

points, except Das's definition [5], which uses extreme point information. Second, all three definitions mentioned above suffer from another difficulty. The Pareto-optimal solution having the maximum minima-count or having the maximum reflex angle or having the maximum deviation from the boundary line may still not correspond to a knee point providing certain desired trade-off information. Since no specific mention of trade-off was used in those definitions, they will always refer to a specific point as a knee point in every convex Pareto-optimal front. The corresponding minima-count, reflex angle or deviation measure may be marginally better than their neighbors and hence the adequacy of the corresponding point as a knee point is questionable. In the following section, we present two new definitions that require that a knee point must have a substantially large trade-off value compared to other Pareto-optimal points, thereby having a global property. One of the definitions also utilizes user-supplied trade-off information directly. Later, we shall suggest some interesting extensions to define other knee related points.

3 Proposed Knee Point Definitions

First, we present a definition that is similar in principle to the reflex angle approach discussed above. Thereafter, we suggest a definition of a knee point that depends on user-supplied trade-off information directly.

3.1 Bend-Angle Approach

The computation of a reflex angle at a point $\mathbf{x}$ on the Pareto-optimal front requires two other points – a point ($\mathbf{x}^L$) left of $\mathbf{x}$ and another point ($\mathbf{x}^R$) right of $\mathbf{x}$. Instead of computing the reflex angle from two neighbors of $\mathbf{x}$, as suggested in the reflex angle approach [4], here, we define a *bend-angle*, that uses any two given Pareto-optimal points ($\mathbf{x}^L, \mathbf{x}^R$) left and right of $\mathbf{x}$, respectively. For the calculation of the bend-angle, we suggest here to use normalized values of the objectives, rather than the objectives themselves.

Definition 6 *Let us say that for a given Pareto-optimal point* $\mathbf{x}$*, two other Pareto-optimal points* $\mathbf{x}^L$ *and* $\mathbf{x}^R$*, left and right of* $\mathbf{x}$ *are supplied. The bend-angle at* $\mathbf{x}$ *is defined as follows:*

$$\theta(\mathbf{x}, \mathbf{x}^L, \mathbf{x}^R) = \theta^L - \theta^R, \tag{2}$$

where

$$\theta^L = \arctan \frac{f_2(\mathbf{x}^L) - f_2(\mathbf{x})}{f_1(\mathbf{x}) - f_1(\mathbf{x}^L)}, \tag{3}$$

$$\theta^R = \arctan \frac{f_2(\mathbf{x}) - f_2(\mathbf{x}^R)}{f_1(\mathbf{x}^R) - f_1(\mathbf{x})}. \tag{4}$$

For any two supplied extreme trade-off points $\mathbf{x}^L$ and $\mathbf{x}^R$, we now define a bend-angle based knee-point ($\mathbf{x}^K_{BA}$) as follows:

Definition 7 *For a set of Pareto-optimal points, the point* ($\bar{\mathbf{x}}$) *having the maximum positive bend-angle* $\theta(\bar{\mathbf{x}}, \mathbf{x}^L, \mathbf{x}^R)$ *is identified. If the corresponding bend-angle is greater than a predefined threshold* S_{BA}*, the point* $\bar{\mathbf{x}}$ *is a bend-angle based knee point.*

It is clear from the above definition that there can be at most one knee point ($\mathbf{x}^K_{BA}$) within the range bounded by $\mathbf{x}^L$ and $\mathbf{x}^R$. There is a fundamental difference between our definition and the reflex angle based definition [4]. Certain trade-off fronts may not have a knee point according to our definition, but in reflex angle based definition there will always be at least one knee point for any trade-off front. Moreover, our definition requires specification of two extreme points within which a

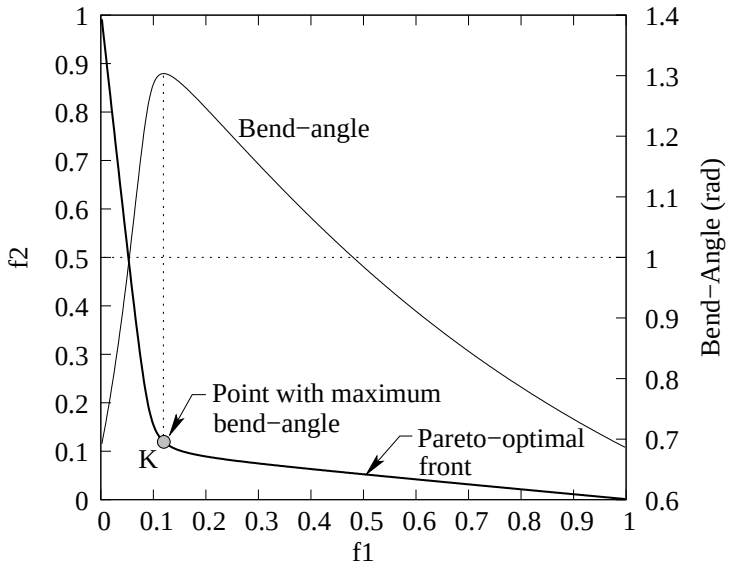


Figure 4: Pareto-optimal front for $a = 0.1$ and $c = 0.1$ and bend-angle values are shown.

knee point is sought. This definition has, therefore, a practical significance. Instead of seeking for a knee point on the entire Pareto-optimal front, the user can specify a region of interest and a knee point, if exists within the specified region would be identified. Furthermore, the specification of the threshold S_{BA} allows an user to achieve a desired *sharpness* of the obtained knee by calculating the deviation of the (max) bend angle with the threshold S_{BA}. Higher the deviation, more sharp and crisp the point is, as a knee. Based on our estimation in a number of practical problems (discussed later), we suggest $S_{BA} = 1$ rad, which is equivalent to about 57.3 degrees.

We illustrate the above knee point definition to a hypothetical Pareto-optimal front described mathematically as follows:

$$(af_2 + f_1 - a)(f_2 + af_1 - a) = ac^2, \tag{5}$$

where a and c are two parameters. The above function has two asymptotes: $af_2 + f_1 - a = 0$ and $f_2 + af_1 - a = 0$. The first asymptote has intercepts on f_1 and f_2 axes at a and 1, respectively, and the second asymptote has intercepts at 1 and a, respectively. The parameter a must be chosen less than one. The parameter c controls the degree of sharpness of the knee point. The smaller the value of c, more sharp is the knee point. Figure 4 shows the Pareto-optimal front for $a = 0.1$ and $c = 0.1$.

The corresponding bend-angle for every point on the front obtained using two extreme points $\mathbf{f}(\mathbf{x}^L) = (0, 1)^T$ and $\mathbf{f}(\mathbf{x}^R) = (1, 0)^T$ are shown in the figure. It is interesting to note how the bend-angle, starting with a small value close to left-extreme of the Pareto-optimal front, increases sharply and then falls quickly to small values near extreme right of the Pareto-optimal front. Since the knee point ($\mathbf{x}^K_{BA}$) corresponds to the point having the maximum bend-angle, the point K is a candidate point for a knee. Thereafter, we also observe from Figure 4 that the bend-angle at K is greater than chosen threshold $S_{BA} = 1$ rad. Thus, we declare the point K as the bend-angle based knee point.

The trade-off frontier given in equation 5 can also be used to study the effect of sharpness parameter (c) on the existence of a knee point. The above calculations were done for $c = 0.1$. Table 1 presents various knee point scenarios for different trade-off fronts obtained with different c values. As discussed above, as the value of c is increased the trade-off front becomes more and more flat, thereby reducing the sharpness of the knee point. The maximum bend-angle obtained for each c is also shown in the table. It is evident that with a threshold of 1 rad, all fronts below $c \leq 0.6$

Table 1: Effect of sharpness parameter c (with $a = 0.1$) on bend-angle based knee point.

c	Maximum Bend-angle	f_1	f_2
0.0	1.370	0.0920	0.0908
0.1	1.303	0.1200	0.1193
0.2	1.239	0.1480	0.1488
0.3	1.179	0.1740	0.1804
0.4	1.123	0.2000	0.2120
0.5	1.072	0.2240	0.2459
0.6	1.026	0.2460	0.2821
0.7	0.982	0.2680	0.3184
0.8	0.943	0.2880	0.3573
0.9	0.906	0.3080	0.3963
1.0	0.871	0.3280	0.4355

exhibit a bend-angle based knee, whereas for $c \geq 0.7$, the corresponding fronts do not exhibit any knee point. Figure 5 shows the trade-off fronts for four different c values: $c = \{0.0, 0.2, 0.4, 0.6\}$. The corresponding bend-angle based knee point is also marked on the respective front.

It is interesting to note that for a concave Pareto-optimal front, the bend-angle is negative for every point. Since our definition requires the knee point to have a positive bend-angle, no knee point ($\mathbf{x}^K_{BA}$) exists for a purely concave Pareto-optimal front.

3.2 Trade-off Approach

Our next definition is directly related to the desired trade-off information that will define a knee point. Recall that a point is said to be a knee point if there does not exist any other point surpassing a desired trade-off. Let us say that we are interested in a knee point $\mathbf{x}$ with the following trade-off: A unit gain in f_1 from $\mathbf{x}$ to any other trade-off point would require at least an amount of α sacrifice in f_2 and simultaneously a unit gain in f_2 from $\mathbf{x}$ to any other point would require at least an amount of β sacrifice in f_1. The parameters α and β are specified by the user. One advantage of this approach is that objectives need not be normalized; a proper choice (which can often be motivated by their practical significance) of α and β can take care of different scaling of objectives. Since this method is directly related to the user-defined critical trade-off information, it is a more pragmatic approach.

With this information, we can then identify the set of points which are (α, β)-*non-inferior (NI)* to $\mathbf{x}$. Figure 6 illustrates this concept. For a given α and β trade-off information, the point $\mathbf{x}$ (marked as A in the figure) is better than all Pareto-optimal points lying in the range BD and CE. Thus, the (α, β)-inferior points to $\mathbf{x}$ is the set BD and CE and the (α, β)-non-inferior points to $\mathbf{x}$ is the set BC. It is clear that if the range BC (or proportion of points in BC to all available Pareto-optimal points in DE) is adequately small, there is no motivation for the user to choose almost any other Pareto-optimal points. Hence, the point A can be declared as a desired knee point with respective to the specified (α, β) trade-off information. Motivated by this fact, we define a (α, β)-knee point.

Definition 8 *For a set of Pareto-optimal points, the point ($\bar{\mathbf{x}}$) having the maximum range of (α, β)-inferior points is identified. If the corresponding range is greater than a predefined threshold $S_{\alpha,\beta}$, the point $\bar{\mathbf{x}}$ is a called a (α, β)-knee point.*

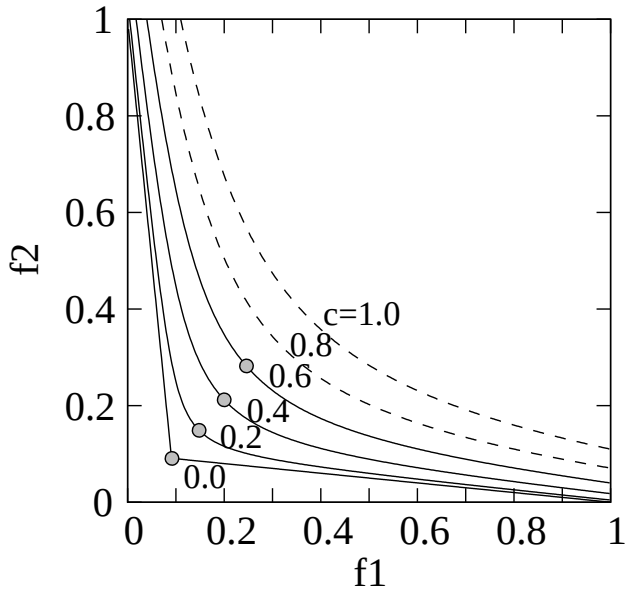


Figure 5: Pareto-optimal fronts for different values of c and $a = 0.1$ are shown. Fronts with $c \leq 0.6$ exhibit a bend-angle based knee.

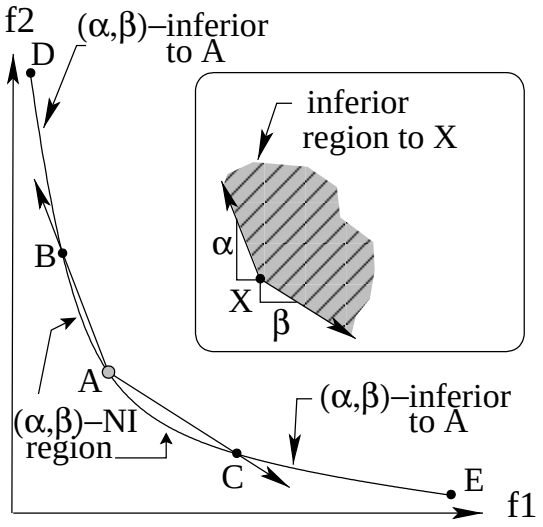


Figure 6: For a given trade-off of α and β, region BD and CE are inferior to A. The inset figure marks the inferior zone of X.

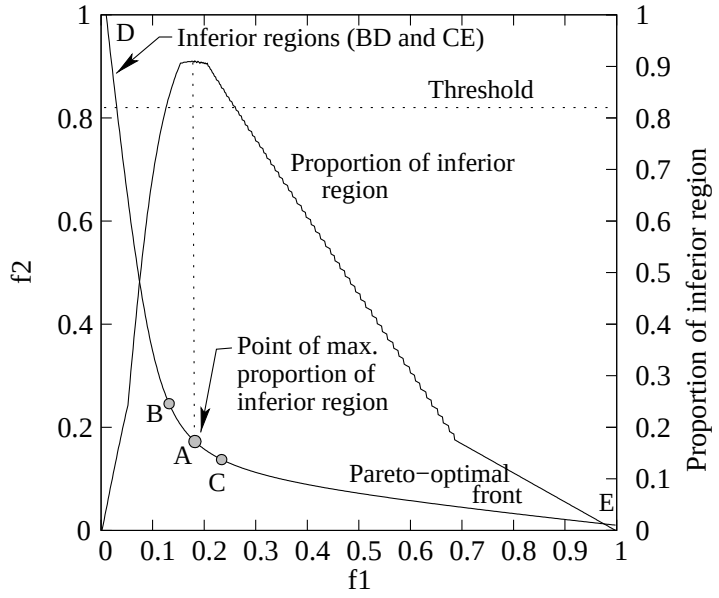


Figure 7: The (α, β)-knee point (A) for $a = 0.1$ and $c = 0.3$. The knee point maximizes the proportion of inferior region.

We illustrate this concept on the Pareto-optimal front given in equation 5. Figure 7 shows the front with $a = 0.1$ and $c = 0.3$ and the corresponding knee point and region of (α, β)-inferior points (BD and CE) with $\alpha = \beta = 1.5$. For different values of the sharpness factor c, the ratio of range of (α, β)-inferior points to the entire Pareto-optimal region is computed and tabulated in Table 2. It is clear from the second column that for $c = 0$, 100% points are (α, β)-inferior to the obtained point, thereby clearly making the obtained point as a knee point. The knee point is also shown in the table. As c is increased, the sharpness of the knee point reduces and the table indicates that the maximum proportion of (α, β)-inferior region to the entire front region decreases. To make an agreement with the bend-angle approach, we suggest and set $S_{\alpha,\beta} = 0.82$ here, so that all fronts with $c \leq 0.6$ qualify to have a (α, β)-knee point.

For concave Pareto-optimal fronts, extreme points (or their neighbors) are likely candidate of a (α, β)-knee point and it is impossible that an intermediate point will qualify as a (α, β)-knee point by the above definition.

Both bend angle and (α, β) approach tries to capture the trade-off information from the Pareto-optimal front and use a *metric* to quantify this trade-off information. If this metric for a particular point is greater than a chosen threshold, the point qualifies as a knee point, otherwise there is no knee point. This essential concept was missing in previous approaches, such as the reflex angle, utility and normal boundary intersection approaches. However, since each of these methods also utilize a metric namely, reflex angle, utility and distance in its core, the definitions can be modified by restricting their values to essentially lie beyond a certain threshold before a point can be declared as a knee point.

4 Knee-Region

The above two definitions allow us to determine the existence of a knee point (parameterized with respect to user specifications) and locate it from a given bi-objective Pareto-optimal front. However, if we focus on some fronts, such as the one with $c = 0.3$ in Figure 7, we find that there are other knee-like points around the obtained knee point. Although the point A is declared as

Table 2: Effect of sharpness parameter c (with $a = 0.1$) on (α, β)-knee point.

c	max. prop. of inferior region	(α, β)-Knee point		Left bound point		Right bound point	
		f_1	f_2	f_1	f_2	f_1	f_2
0.0	1.000	0.0920	0.0908	0.0920	0.0908	0.0920	0.0908
0.1	0.969	0.1240	0.1157	0.1040	0.1440	0.1340	0.1089
0.2	0.940	0.1500	0.1468	0.1200	0.1904	0.1880	0.1212
0.3	0.911	0.1820	0.1725	0.1320	0.2459	0.2340	0.1375
0.4	0.882	0.2020	0.2099	0.1520	0.2838	0.2980	0.1455
0.5	0.855	0.2340	0.2353	0.1640	0.3390	0.3440	0.1617
0.6	0.829	0.2520	0.2753	0.1860	0.3733	0.4120	0.1683
0.7	0.804	0.2860	0.2984	0.1960	0.4323	0.4540	0.1861
0.8	0.781	0.3180	0.3238	0.2080	0.4875	0.5000	0.2023
0.9	0.759	0.3440	0.3554	0.2240	0.5342	0.5540	0.2149
1.0	0.738	0.3700	0.3869	0.2400	0.5808	0.6100	0.2267

the (α, β)-knee point, there are points in its neighborhood which are (α, β)-non-inferior to A. The points in the region AB and AC have a better trade-off than the specified α and β trade-offs. In this sense, it may be unfair to declare only one point (A) as the (α, β)-knee point, as other points in its neighborhood may also be chosen for the same. In such situations, instead of a knee point alone, it may be better to identify a knee-region. First we define the concept of (α, β)-cone-domination and then we present the definition for a knee-region.

Definition 9 *Given a set of Pareto-optimal points and trade-offs (α, β), a set of (α, β)-cone-dominated points are those that are cone-dominated with respect to (α, β)-cone.*

For a two-objective problem, this refers to identifying the points that are dominated (by definition 1) with respect to the following transformed objectives:

$$F_1 = f_1 + \frac{1}{\alpha} f_2, \tag{6}$$

$$F_2 = \frac{1}{\beta} f_1 + f_2. \tag{7}$$

We now define a (α, β)-knee-region.

Definition 10 *The set of Pareto-optimal points which are not (α, β)-cone-dominated is defined as the (α, β)-knee-region.*

Since a (α, β)-knee point and a (α, β)-knee-region are defined from two different concepts, we require the following theorem to establish a relationship between them:

Theorem 1 *For a bicriteria problem having a convex Pareto-optimal front, the (α, β)-knee point always lies within the (α, β)-knee-region.*

We outline a proof of this theorem by the help of Figure 8. The figure shows the Pareto-optimal front on the transformed objective space (F_1-F_2 space) obtained from the original objective space (f_1-f_2 space) using a cone-domination concept described by the parameters α and β, given in equations 6 and 7. The original Pareto-optimal front is marked as the entire front AB, while the (α, β)-knee-region obtained using the cone-domination concept is marked as CD. Let us now consider a point E in the region AC (outside CD), as shown in the figure. The point E has AE and

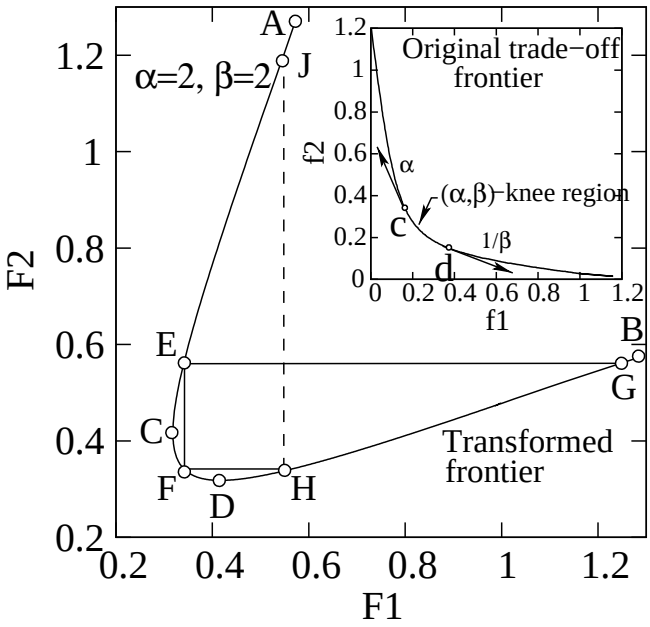


Figure 8: The inset figure shows the Pareto-optimal front. The transformed frontier with a number of points are shown to illustrate a proof Theorem 1

BG as (α, β)-inferior regions. We shall show here that there exists at least one point in the region CD, which has a larger (α, β)-inferior region than E. There can be two scenarios while we attempt to locate another point: vertical projection having an identical F_1 value to that of E or horizontal projection having an identical F_2 to that of E. First, we consider the scenario in which the vertically projected point (the point F) lies in the region CD, as shown in the figure. The region AE and BH are (α, β)-inferior to F. Since BH is larger than BG, the (α, β)-inferior region of F is more than E. This is true with any point E in the region AC. Take another point J in the region AC, which has only AJ as it's (α, β)-inferior region. Here the second scenario happens when the vertically projected point H (having an identical F_1 value) does not lie in the region CD. For such points, we make a horizontally projection from H and locate the point F which has an identical F_2 value to H and lies on CD. Then, we compare the point J with the point F in terms of their (α, β)-inferior regions. The point F has regions AE and BH. Since AE is greater than AJ, we conclude that the point F has a larger (α, β)-inferior region than the point J. A similar argument can also be made for any point in the region BD. Thus, we conclude that the point having the maximum (α, β)-inferior region cannot lie outside CD. This proves the stated theorem.

The theorem suggests that the (α, β)-knee point defined in the previous section will always lie in the (α, β)-knee region. However, for some choice of α (small) and β (large), the (α, β)-knee-region may collapse to a single (α, β)-knee point.

Figure 9 shows the Pareto-optimal front for different c values, their (α, β)-knee point, and their (α, β)-knee-region for $\alpha = \beta = 1.5$. It is important to note that the existence of a (α, β)-knee point is a necessary condition to the existence of a (α, β)-knee-region. For $c = 0.8$ and 1.0, the knee point does not exist with $\alpha = \beta = 1.5$; hence there is no knee-region for these two Pareto-optimal fronts.

The choice of α and β affect the knee point as well as the knee-region. Figure 10 shows the knee point and knee-region for $\alpha = \beta = 1.1$ and 2.0. With a small value of α and β, trade-off requirement is small and hence the knee-region is small. Due to a smaller trade-off requirement, the knee point exist even for the Pareto-optimal front with $c = 1.0$. However, when large values of α and β are chosen, the knee-region is wider and the Pareto-optimal front needs to be sharply changing near the knee point for it to exist.

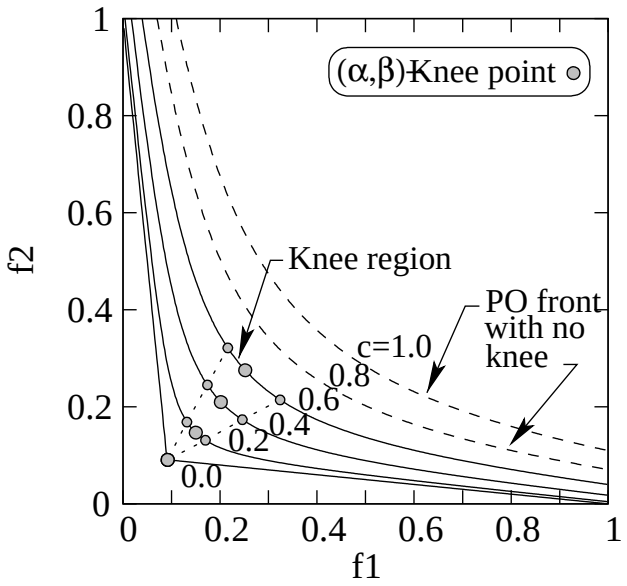


Figure 9: The (α, β)-knee-region are shown for different values of c (with $a = 0.1$). Here, $\alpha = \beta = 1.5$ is used.

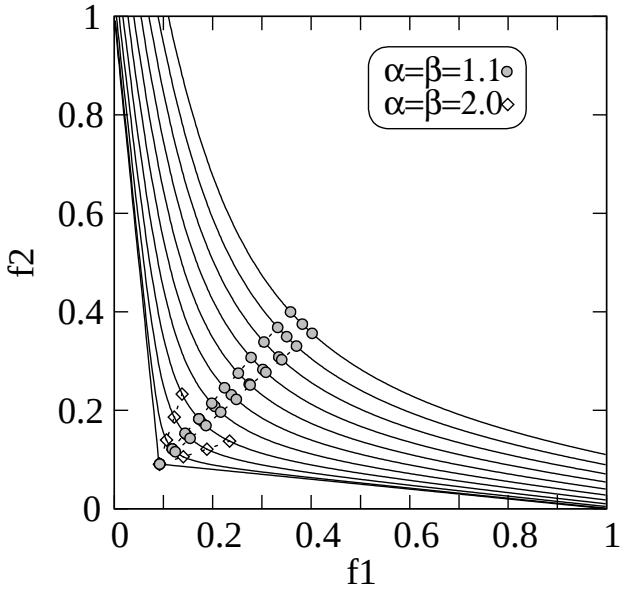


Figure 10: The effect of α and β values on the obtained knee-region is shown.

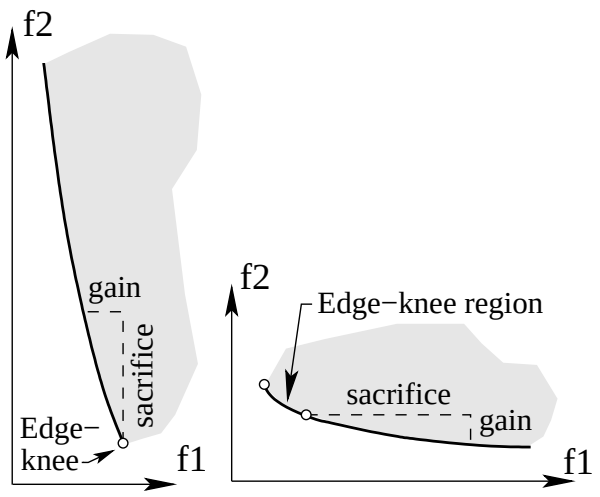


Figure 11: The edge-knee points are illustrated. A large sacrifice is needed to make a small gain to move to any other Pareto-optimal point.

We now state a relationship between (α, β)-knee-region and the (α, β)-non-inferior region, which may be useful in certain circumstances.

Theorem 2 *Given a set of Pareto-optimal points, (α, β)-knee-region will always be a subset to (α, β)-non-inferior region corresponding to the (α, β)-knee point.*

This theorem can be easily verified by using mean value theorem related to first derivative of continuous functions. Let us revisit Figure 6. Given the (α, β)-knee point, A and its (α, β)-non-inferior region defined by region BA and AC, one can obtain the point $\hat{B}$ within the region AB, having the first derivative matching the slope $(-\alpha)$ and $\hat{C}$ having the first derivative matching the slope $(-1/\beta)$. Then, the range bounded by $\hat{B}$ and $\hat{C}$ is our (α, β)-knee region and can be given by:

$$f_1 \in \left[f_1|_{\left(\frac{df_2}{df_1}=-\alpha\right)}, f_1|_{\left(\frac{df_2}{df_1}=-\frac{1}{\beta}\right)} \right]. \tag{8}$$

Since the points of tangencies $\hat{B}$ and $\hat{C}$ are within BA and AC, it proves the theorem.

5 Edge Knees

The above definitions of a knee point takes into account the trade-off information on both sides of the knee point. However, even in the case of a convex Pareto-optimal front, the knee point can have an unfavorable trade-off on only one side of the point, so that there is no motivation for an user to choose any other point, other than an extreme point. In this case, we define the point as an *edge-knee point*:

Definition 11 *A Pareto-optimal point is defined as a γ-edge-knee point, if it lies near an extreme point and a unit gain in one of the objective function would require an at least γ amount of sacrifice in the other objective.*

Pareto-optimal fronts and corresponding edge-knee points are illustrated in Figure 11. Like a knee-region, there can also be an *edge-knee-region*, as shown in the adjoining figure.

The (α, β)-knee point identification procedure can detect a γ-edge knee point. The (α, β)-knee point procedure can be applied with $\alpha = \beta = \gamma$. If the procedure finds an extreme point as the knee

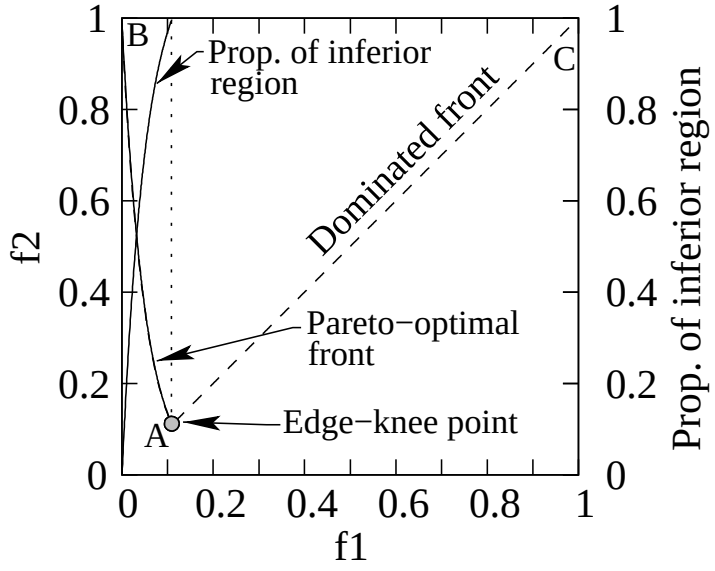


Figure 12: The edge-knee point is found for the problem stated in Equation 9

point, the corresponding point is a γ-edge knee point. The identification of the edge-knee-region, if any, can be similarly used.

We illustrate the procedure on a specific Pareto-optimal front described as a mathematical relationship between f_1 and f_2:

$$f_2 = \max\left(\exp(-50f_1), f_1\right). \tag{9}$$

Figure 12 clearly shows that the trade-off between the objectives exists in region BA. The region AC gets dominated by the point A and hence is not a part of the Pareto-optimal front. Also, the point A is a potential edge-knee point, as, to move out from point A, there is a large sacrifice in f_2 needed to have a small gain in f_1. When we apply our procedure with $\alpha = \beta = \gamma = 1.5$, the point A is obtained as the γ-edge-knee point.

6 Some Front Properties to Exhibit a Knee Point

Identification of a knee point or a knee-like point is one aspect. Another fundamental task would be to investigate any mathematical property that a given Pareto-optimal front should have in order for it to exhibit a knee point. We address this issue in this section.

Let us consider that the underlying Pareto-optimal front is a continuous and twice-differentiable function: $f_2 = f_2(f_1)$. Since both objectives are to be minimized, the trade-off information between the objectives states that an increase in f_1 must come from a decrease in f_2. This indicates that the function f_2 must be a *strictly decreasing* function. For a strictly decreasing function, we have the following condition:

$$\frac{df_2}{df_1} < 0. \tag{10}$$

The front obtained from Pareto-optimal solutions can be of convex or non-convex shape. Since a knee point exists for convex Pareto-optimal front, we realize that a necessary condition on the second derivative of the $f_2(f_1)$ front to exhibit a knee point would be non-negativity of its second derivative function, or

$$\frac{d^2 f_2}{df_1^2} \geq 0. \tag{11}$$

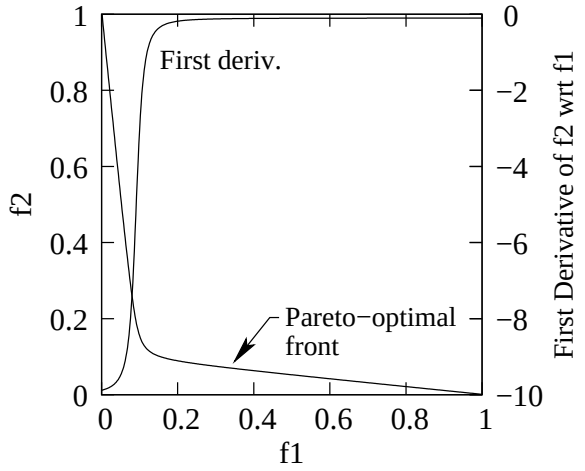


Figure 13: First derivative function for the front relationship given in equation 5.

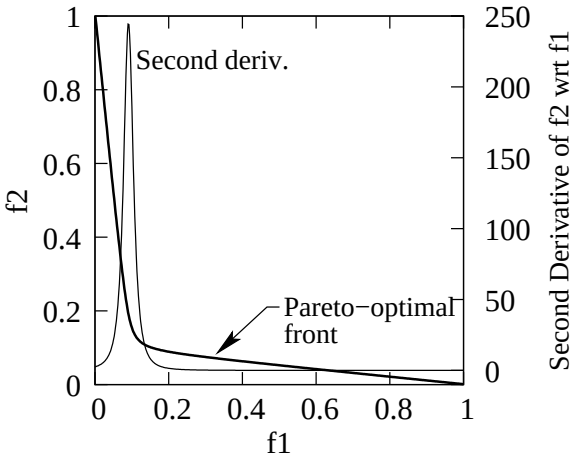


Figure 14: Second derivative function for the front relationship given in equation 5.

Based on the above two conditions, we conclude that for any two Pareto-optimal points (with f_1^L and f_2^R having $f_1^L < f_1^R$), the slope (negative of $\frac{df_2}{df_1}$) at f_1^L is more than the slope at f_1^R:

$$-\frac{df_2}{df_1}\Big|_{f_1^L} > -\frac{df_2}{df_1}\Big|_{f_1^R}. \tag{12}$$

Although this condition is true for any Pareto-optimal front, but for a Pareto-optimal front exhibiting a knee point, the difference between the slopes at a point slightly left and right of the knee point is expected to be large. Thus, we state the following:

Definition 12 *In the context of a bi-objective optimization problem, a Pareto-optimal front would exhibit a knee point, if the second derivative of the front at the knee point has a sudden large positive value.*

$$\frac{d^2 f_2}{df_1^2} = \frac{2ac^2/(1-a^2)}{[(f_1 - a/(1+a))^2 + (2ca/(1-a^2))^2]^{3/2}}. \tag{13}$$

Equation 13 gives second derivative of the front given in equation 5. It shows a sharp increase in it's value at $f_1 = a/(1+a)$. Figures 13 and 14 show the variation of the first and second derivatives of the front given in equation 5 for $a = 0.1$ and $c = 0.1$. It can be seen that the first derivative is negative throughout the front, while the second derivative has a sudden increase in its value near the knee point. Given this, the (α, β)-knee-region can be calculated using equation 8 as:

$$f_1 \in \frac{a}{1+a}\left[1 - \frac{c(2a\alpha - (1+a^2))}{(1-a)\sqrt{a(1-a\alpha)(\alpha - a)}}, 1 - \frac{c(2a/\beta - (1+a^2))}{(1-a)\sqrt{a(1-a/\beta)(1/\beta - a)}}\right]. \tag{14}$$

6.1 Development of Test Problems Exhibiting a Knee

The above properties of a knee-based Pareto-optimal front may be useful in designing test problems exhibiting a knee point. We illustrate the procedure with a simple example problem. We observe that the first derivative function $\frac{df_2}{df_1}$ requires to be a S-type function. We approximate it by using an arctan function as follows:

$$\frac{df_2}{df_1} = \arctan\left(a(f_1 - b)\right) - c. \tag{15}$$

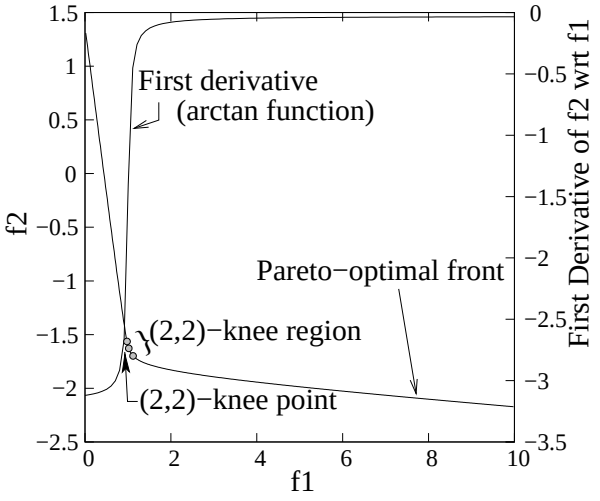


Figure 15: Obtained front relationship for the arctan first derivative function. (2,2)-knee point and region are marked.

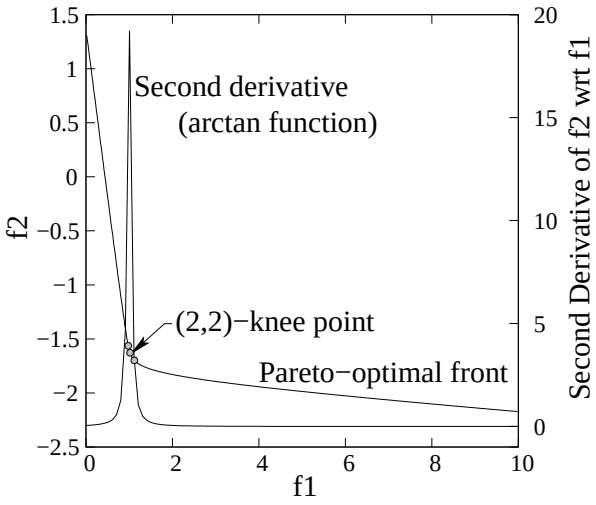


Figure 16: Obtained second derivative function for the arctan first derivative function.

The parameter c is used to make sure that the first derivative function is always negative. By integrating and differentiating the above function, we obtain the front (f_2-f_1) relationship and the second derivative function as follows:

$$f_2(f_1) = (f_1 - b)\arctan\left(a(f_1 - b)\right) - \frac{1}{2a}\ln(1 + a^2(f_1 - b)^2) - cf_1, \quad (16)$$

$$\frac{d^2 f_2}{df_1^2} = \frac{a}{1 + a^2(f_1 - b)^2}. \quad (17)$$

Figure 15 shows the first derivative and its front for $a = 20$, $b = 1$ and $c = 1.6$. Figure 16 shows the second derivative function. The front relationship shows clearly that there is a knee point. Other functional forms of the first derivative function, such as the logistic function, are possible and can be used to design test problems exhibiting a knee as well.

Although in the above construction process, we have assumed a functional form for the first derivative function and then computed the front relationship, a suitable functional form for the second derivative can also be assumed first and then the first derivative and the front relationship can be obtained by integrating the assumed second derivative function.

When we apply our proposed (α, β)-approach (with $\alpha = \beta = 2$) to the above front (arctan function as the first derivative), a knee point and a knee-region are found to exist. They are marked in Figure 15. The S-nature of the first derivative is clear from the figure. Figure 16 clearly shows that the second derivative shoots up to a very high value near the knee-region. Again using equation 8 we can write the knee-region directly as:

$$f_1 \in \left[\frac{1}{a}\tan(c - \alpha) + b, \frac{1}{a}\tan(c - 1/\beta) + b\right]. \quad (18)$$

The above function is simplistic and provides us with conditions on the objectives that may cause a knee or an edge-knee on its Pareto-optimal front. It is clear that different conditions may exist, but problems must have some restrictive conditions for demonstrating a knee point. Particularly the conditions on first and second derivatives which we discussed above are quite restrictive. Thus, real-world problems exhibiting a knee or an edge-knee point may follow some special characteristics in their functional relationships before a knee is exhibited. But there is no

denying that if there exists a knee to a two-objective optimization problem, the knee point is the most preferred. We shall argue a connection between the above property of a knee point with popular solution methodologies used in certain problem-solving tasks, but before that we suggest a procedure which may help provide some a priori information about adequate values of trade-off parameters (α and β) in a trade-off frontier exhibiting a knee point.

7 Upper Bound Estimation for Trade-off Parameters

In the trade-off approach mentioned above, it is expected that the user will supply two trade-off parameters: left-side trade-off, α and right-side trade-off, β. This allows the user to provide problem-specific information in determining a knee point or a knee-region. But, given a trade-off frontier, we can also analyze and find the upper bounds on these parameters which will cause the frontier to exhibit a knee like property.

Recall that in addition to a knee point, a trade-off frontier is expected to have a knee-region of certain width, depending on the chosen α and β parameters. Since these two parameters will be supplied by the user before a knee-point or a knee-region is found, the user may be interested in knowing the upper limit of these two parameters for a particular width of the resulting knee-region. If the chosen width is small, it is expected that a sharp knee point is desired, and vice versa. To determine such limits, we suggest an inverse analysis process to that portrayed in the previous section here.

For this purpose, we fix a width δ of the knee-region and determine which part of the trade-off frontier (with the given width δ) will correspond to the knee-region. Here, we use the following strategy. We locate the strip of width δ from the left extreme point by a parameter s, thereby initializing s to zero. We then compute the slope of tangency at the left extreme point of the strip and inverse of the slope at the right extreme of the strip. Say, the corresponding quantities are $\alpha(s)$ and $\beta(s)$. We now define a measure: $\mathcal{M}(s) = \arctan(-\alpha) - \arctan(-1/\beta)$. Thereafter, we slide the strip by a small distance ($s = s_t$) at a time. Say, the corresponding slope quantities are $\alpha(s_t)$ and $\beta(s_t)$ and the measure is $\mathcal{M}(s_t)$. Following the procedure till the other extreme of the frontier, we then compute the location s^δ for the measure $\mathcal{M}$ to have its highest value, thereby locating the knee region of a specified width δ. This will result in the upper bounds of the trade-off parameters $\alpha(s^\delta)$ and $\beta(s^\delta)$ for the given strip width of δ. The above procedure can then be extended for various values of δ and an idea of the variation of upper bounds of the trade-off parameters can be observed. Figure 17 plots the obtained $\alpha(s^\delta)$ and $\beta(s^\delta)$ for different values of δ in terms of the proportion of knee-region to the complete length of the frontier for the trade-off frontier given in equation 5. Recall that the abscissa needs to be plotted till $(1 - S_{\alpha,\beta})$, as beyond that a frontier cannot exhibit a knee point according to our definition 8. Two c values are used here. For $c = 0.02$, the knee is sharper and the figure shows that the upper bounds on the trade-off values increase sharply. It is clear that a knee-region exists with both upper bounds varying in the range $[1, 10]$. For example, if an user is interested in locating the knee-region with a width of 4% of the entire length of the trade-off frontier, the plot with $c = 0.02$ indicates that the maximum value of α and β can be around 8.0. On the other hand, if the user wants a sharper knee-region (say 1%, instead of 4%), the maximum value of α and β can be around 3.0. Thus, Figure 17 allows the user to understand this important relationship of trade-off parameters with the desired sharpness of the obtained knee-region. For $c = 0.02$ frontier, it does not seem to be wise to choose α and β values beyond 8.0, as an increased value of α and β beyond 8.0 comes with a large sacrifice in the sharpness of the resulting knee-region. On the other hand, for $c = 0.2$ frontier, the relationship between upper bounds of α (and β) with δ is almost linear. This means that the upper bound of α and β is directly proportional to the choice of δ. Another interesting observation is that to have a knee point, $c = 0.2$ frontier needs a smaller range of α and β values than that for $c = 0.02$ frontier.

In addition to the information about any preferred choice for α and β values, another advantage

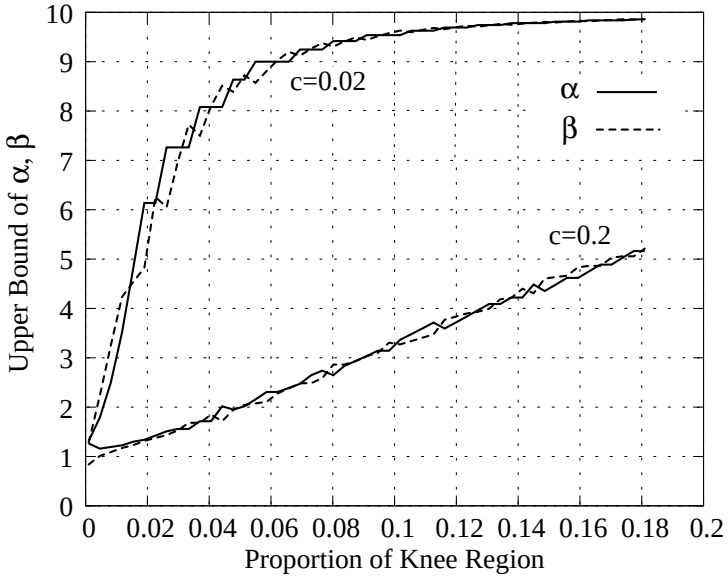


Figure 17: Obtained α and β values as a function of obtained proportion of knee-region.

of this analysis is that the user can estimate the width of the knee region a priori for a chosen upper bound of α and β values, thereby allowing the user to make a proper choice of trade-off parameters before looking for a knee or knee-line point in a trade-off frontier. We shall illustrate this analysis task for an engineering design problem later.

8 Problem-Solving Tasks and Knee Solutions

To discuss the connection between a knee point with preferred solution methodologies in certain problem-solving tasks, let us consider the task of sorting a set of integers. The first methodology comes to any mind is the *quicksort* procedure. There are two other methods – the bubble sort method which is the simplest and most straightforward sorting method and the binary-tree sort which is computationally the most efficient method. A plethora of other sorting algorithms exist, but an important question to ponder is when a sorting task is required why most researchers resort to the quicksort procedure. Next, let us consider the task of fitting a curve through a set of two-dimensional data points. Despite the existence of a dozen of different methods, why do researchers prefer the polynomial regression method? Besides these generic problem-solving tasks, many engineering problems are found to have preferred solutions. For example, it is recommended that driving around 55 miles/hr is the most fuel-efficient approach, and so on.

Could it be that the corresponding popular solution methodology or principle is an optimal solution of a single goal of solving the problem. Let us consider the sorting example again. An obvious goal any user is concerned about is the computational time. A method which takes the smallest time to sort a set of numbers accurately is of interest to the user. But in such problem-solving tasks, a single objective is often not everything an user is interested. A method may be fastest in computational time, but if it is very difficult to implement on a computer or if it takes a large amount of computer memory or if it has restricted application domain, the user may not be interested in that method. Usually, in most problem-solving tasks, we can think of at least two conflicting goals, both of which would of interest to an user. In the case of the sorting task, two objectives of minimizing computational time and computer memory would be conflicting to each other. Table 3 presents the average-case computational time complexity and required computer

Table 3: Computational time complexity and computer memory requirement of three sorting methods.

Method	Computational time	Computer Memory
Bubble sort	$O(n^2)$	$O(1)$
Quicksort	$O(n \log n)$	$O(\log n)$
Binary-tree sort	$O(n \log n)$	$O(n)$

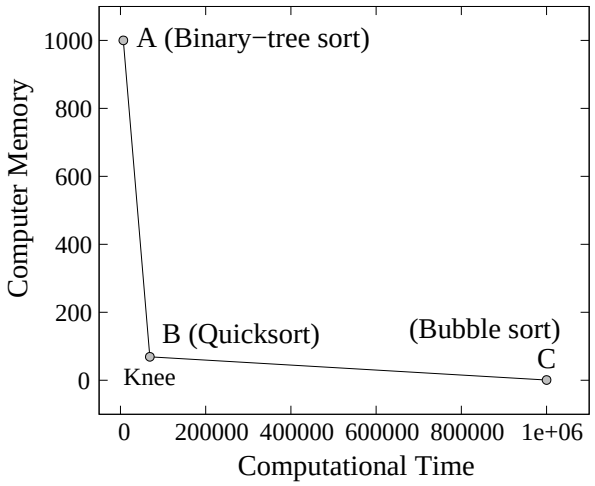


Figure 18: Three sorting algorithms are compared for two conflicting objectives and the quicksort seems to be a knee solution.

memory of three sorting methods. The trade-off between the two objectives is clear from the table. The 'big Oh' notation ($O(x)$) indicates that the value is bounded by kx, where 'k' is a constant. For large values of n, $O(n \log n)$ is much smaller than $O(n^2)$ and $O(\log n)$ is much smaller than $O(n)$. Since the objectives are conflicting in nature, a method which is good from computational point of view is expected to demand more computer memory and vice versa. In such problems, there is a set of Pareto-optimal solutions, which become optimal choices. In the above case, all three sorting methods are different Pareto-optimal choices from the bicriteria point of view.

Consider Figure 18 which is plotted by using the objective values taken from Table 3 with $n = 1,000$. For the sake of plotting the figure, we have assumed the constant in $O()$ as 10. The figure clearly shows the similarity of the shape of the trade-off front with the hypothetical front exhibiting a knee and that the quicksort procedure (solution B) is a knee solution, which is what is most commonly used by practitioners. It is clear from the figure that a comparison between solutions A and B would prefer solution B, as there is a need for a large sacrifice in the memory to make a small gain in the time. On the other hand, there is no motivation for one to choose solution C in comparison with solution B as well, as it requires a large sacrifice in the time to make a small gain in the memory. Thus, solution B is usually preferred in such a problem and this may remain as an explanation for the popular choice of the quicksort procedure in a sorting task.

We argue, although not viewed in this bi-objective manner while choosing a procedure for a task, if different procedures of a task inherently possess a knee solution in an associated bicriteria problem, the preferred procedure is likely to be the knee solution. We now investigate this argument on a number of problem solving tasks.

9 Some Practical Problem-Solving Tasks

In this section, we take three different generic problem-solving tasks and four engineering design problems from transportation to mechanical engineering design tasks to understand and analyze the reasons for popularity of common solution methodologies from a bi-objective point of view.

9.1 Bucket Sorting Procedure

We have considered the sorting problem above, purely from the reported computational time and memory requirement of three sorting methodologies. There exist a number of sorting algorithms with differences in performance in terms of computational complexity, memory usage, and stability. For some algorithms, average and worst-case quantities of the performance indicators are worked out, but their actual values depend on the data set being considered.

In this section, we consider the bucket sorting algorithm the performance of which can be controlled by using a parameter, called the bucket size d. In this algorithm, all numbers are divided into d buckets, each bucket with an equal range of values. Each bucket is then sorted using the *Insertion Sort* procedure. Let us assume that there are 'n' numbers that needs to be sorted. Then, the space complexity is given by $O(2n + d)$, of which the output takes $O(n)$ space, the buckets in which individual numbers are stored takes up a total of $O(\sum d_i)$ $(=O(n))$ space and a vector that keeps the count of the number of elements present in the buckets will have a size of $O(d)$. The time complexity is calculated as follows: O(n), time is required to scan the data set and place each number into it's corresponding bucket, $O(1)$ time is required for each call made to insertion sort which sorts the numbers in a bucket, and $O(1)$ time is required for each swap made in the insertion sort. Although these numbers give an indication of their orders of magnitude as a function of size of data set(n) and the number of buckets(d), the actual implementation may produce different results (depending upon the data set).

For our experiment, we consider a data set consisting of 3,334 randomly generated numbers ranging from 3 to 9999. The data set is sorted repeatedly using the bucket sort procedure, each time with a different number of buckets. In some sense, each bucket size constitutes a different sorting strategy and we are interested in knowing if there exists a strategy that could give a good trade-off between time and memory. For this purpose, the number of buckets, d is varied from 1 to 100. Figure 19 shows the variation of time and space complexities observed while sorting the data set for different number of buckets. It is observed that when only one bucket is chosen, the bucket sort algorithm become equivalent to an insertion sort algorithm and leads to computationally expensive sorting. When more buckets are chosen, the computational effort to sort each bucket reduces at the cost of extra memory space. On closely examining the trade-off curve, it can be seen that a knee-region exists near d equal to 10. Considering $\alpha = 75,000$ and $\beta = 1/25,000$, we obtain $d = 7$ solution as the (α, β)-knee point. These parameters mean that for a unit *gain* in space complexity we are not interested in solutions having more than 75,000 units of *sacrifice* in time complexity. Similarly, for a unit gain in time complexity, we are not interested in solutions having a sacrifice of more than 1/25,000 units of space complexity. In a sense, the time complexity is considered three times more important than space complexity. Interestingly, there are four consecutive bucket sizes ($d = 7$ to 10) qualified to be in the knee region, according to our chosen α and β values.

Table 4 shows the proportion of inferior region for each of these points. When we use the bend-angle approach, $d = 8$ turns out to be the knee point, with a bend angle of about 1.37 radians.

A similar study is performed on different data sets and they also exhibit a knee-region near d equal to 10. This motivates us to investigate why d equal to 10 (in the entire range from 1 to 100) is a good trade-off strategy for computational time and computer memory usage! This can be explained by bringing a logic from another sorting procedure called the *radix sort*, where the data set is sorted starting from the least significant digit (rightmost), followed by the second-least

Table 4: Knee points (shown in bold) and knee-region found by the proposed knee-finding methodologies for the bucket sorting algorithm.

d	(α, β) approach			Bend-angle approach		
	f_1	f_2	prop. of inferior region	f_1	f_2	Bend-angle
7	**6675**	**398499**	0.925			
8	6676	358652	0.865	**6676**	**358652**	1.369
9	6677	313787	0.878			
10	6678	285438	0.845			

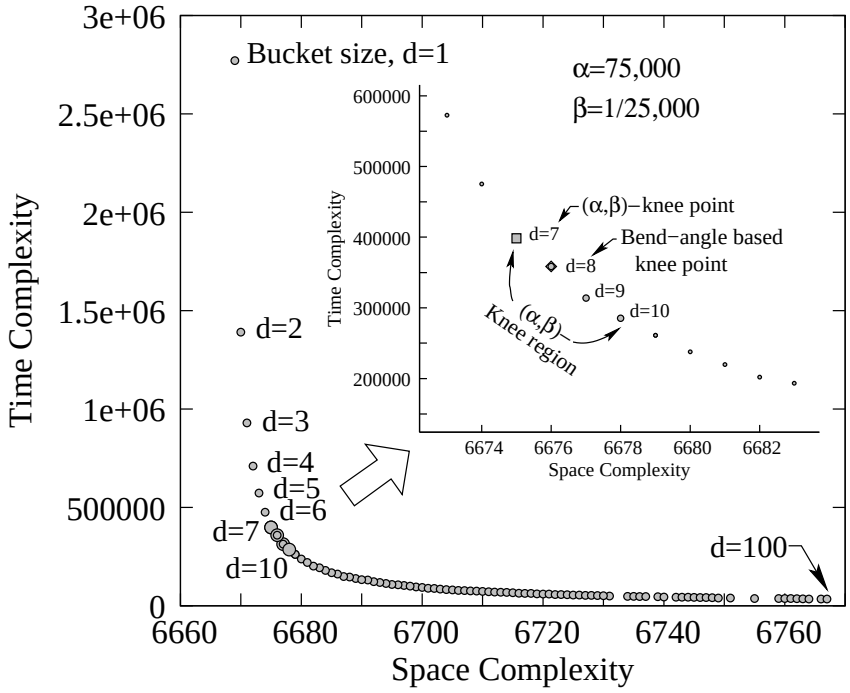


Figure 19: The bucket sort algorithm exhibits a knee-region with bucket size varying between 7 to 10 for $\alpha = 75,000$ and $\beta = 1/25,000$.

significant digit and so on till the most significant digit (leftmost). Since we are dealing with data set with all elements belonging to the decimal number system, the number of buckets required for one pass of radix sort will be 10 (corresponding to digits 0 to 9). Any bucket size smaller than 10 will not consider some digits properly and any size larger than 10 is redundant. In a sense, the bucket sort ($d = 10$) and radix sort are similar in principle when dealing with natural numbers and a consideration of 10 buckets seem to have a connotation with the radix sort algorithm. It is worth mentioning that $d = 2$ is considered equivalent to the quicksort procedure, provided each bucket is also sorted using the same quicksort procedure. Thus, the performance of the bucket sort depends on the sorting procedure used to sort individual buckets and with our use of insertion sort, around 10 buckets seem to provide a good trade-off.

9.2 Curve Fitting

Curve fitting involves fitting a curve to a given data set. There are a wide number of curve fitting techniques available in literature. These models have different ideologies which mark them apart

from one another and produce different kinds of fit. Some fits are closer to the data points though they do not quite match the data trend very well. Some are smoother and takes care of few or more random points in the data set. It is observed that in practice most people prefer to use polynomial regression as their first choice of model for curve fitting. Why is the polynomial regression a preferred choice? Does anything else governs this decision other than personal choice?

We investigate this aspect by considering two conflicting goals of a curve fitting task – the *degree-of-freedom* (DOF) and the *residual sum of squares* (RSS). The DOF provides a measure of the complexity of the fitted curve or rather a measure of number of parameters that are required to mathematically represent the fit, thereby relating to the complexity involved in the curve fitting task. However, the RSS gives the measure of the error in approximating data with the curve. When the data set is fairly complicated, one may fit it with a simpler polynomial regression model having lesser DOF but the accuracy or RSS obtained need not be within satisfactory bounds. If one tries to improve the RSS, more parameters may have to be introduced that would refine the curve closer to the data set increasing the complexity or DOF. Hence the objectives that we try to optimize in this problem are conflicting to one another. Following subsections provide a brief description of curve fitting models considered in this study (for details reader is suggested to refer [17]).

9.2.1 Polynomial Regression

A general linear model can be written as:

$$\begin{aligned} y_i &= \beta_0 + \beta_1.x_{i1} + \beta_2.x_{i2} + \cdots + \beta_k.x_{ik} + \varepsilon_i, \\ \boldsymbol{y} &= \boldsymbol{X\beta} + \boldsymbol{\varepsilon}. \end{aligned} \tag{19}$$

For polynomial regression, the basis vectors x_{ik} becomes equal to x_i^k. Ordinary least squares estimator gives $\hat{\boldsymbol{\beta}}$ and matrix $\boldsymbol{H}$ called *hat matrix*:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} \tag{20}$$

$$\begin{aligned} \hat{\boldsymbol{y}} &= \boldsymbol{X}\hat{\boldsymbol{\beta}}, \\ &= \boldsymbol{X}(\boldsymbol{X^T X})^{-1}\boldsymbol{X^T y}, \\ &= \boldsymbol{H y}. \end{aligned} \tag{21}$$

Using these approximation parameters the objective functions RSS and DOF are calculated as follows:

$$\begin{aligned} DOF &= trace(\boldsymbol{H}) \\ RSS &= \sum_{i=1}^{n} \varepsilon_i^2. \end{aligned} \tag{22}$$

Even for other fitting models namely splines, penalized splines and quadratic local regression, DOF and RSS can be calculated using equation 22. For a better curve fitting model, one would be interested in choosing a model having smaller values of DOF and RSS.

9.2.2 Splines

The linear and quadratic spline models can be given by

$$f(x) = \beta_0 + \beta_1 x + \sum_{k=1}^{K} b_k (x - \kappa_k)_+, \tag{23}$$

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \sum_{k=1}^{K} b_k (x - \kappa_k)_+^2. \tag{24}$$

where κ represents the *knots* (knots define the curve segment for which splines are defined). The total number of knots are taken as K that can be chosen in x.

$$(x-\kappa_k)_+ = \begin{cases} (x-\kappa_k), & \forall\ x \geq \kappa_k, \\ 0, & \text{otherwise.} \end{cases} \tag{25}$$

9.2.3 Penalized Linear Spline Regression

When a large number of knots are introduced, the fit becomes wiggly in nature. To make it smooth, constrained ordinary least squares fit is performed which is written as follows:

$$\min \|\boldsymbol{y} - \boldsymbol{X\beta}\|^2 + \lambda^2 \boldsymbol{\beta^T D \beta} \tag{26}$$

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{0}_{2\times 2} & \boldsymbol{0}_{2\times K} \\ \boldsymbol{0}_{K\times 2} & \boldsymbol{I}_{K\times K} \end{bmatrix} \tag{27}$$

for some number $\lambda \geq 0$. The amount of smoothing is controlled by λ, which is therefore referred to as *smoothing parameter*. It is varied in the range [0.01-200] depending upon the nature of data set. This has solution, given as follows:

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{\lambda}} = (\boldsymbol{X^T X} + \lambda^2 \boldsymbol{D})^{-1} \boldsymbol{X^T y}. \tag{28}$$

The values of DOF and RSS can be calculated from the estimated parameters β_λ and matrix $\boldsymbol{H}$ using equation 22.

9.2.4 Quadratic Local Regression

Considering a second degree polynomial, a smooth curve is obtained by fitting the quadratic model using the weighted least squares with *kernel weights* $K\left(h^{-1}(x_i - x)\right)$. The kernel function $K(x)$ is taken as e^{-x^2}. The parameter $h > 0$ is the smoothing parameter (varied in range [0.01-1]). The solution is given as follows:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X_x^T W_x X_x})^{-1} \boldsymbol{X_x^T W_x y} \tag{29}$$

$$\boldsymbol{X_x} = \begin{bmatrix} 1 & (x_1 - x) & (x_1 - x)^2 \\ \vdots & \vdots & \vdots \\ 1 & (x_n - x) & (x_n - x)^2 \end{bmatrix} \tag{30}$$

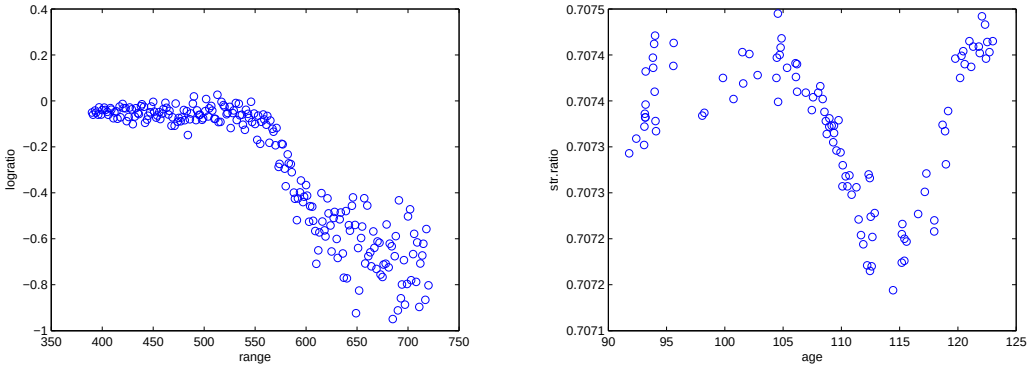
$$\boldsymbol{W_x} = diag\left\{ K\left(\frac{x_1 - x}{h}\right), \cdots, K\left(\frac{x_n - x}{h}\right) \right\}. \tag{31}$$

9.2.5 Bi-objective Analysis

Four data sets are taken from `http://www.stat.tamu.edu/~carroll/semiregbook` and are known as *lidar*, *fossil*, *electrical usage*, *simulated*. These data sets are fitted with different approximation models with variation in their parameters. Figure 20 gives the plot of different data sets. They clearly indicate that a different DOF would be required to fit each dataset well.

Figure 21 gives the variation of RSS with DOF for these four data sets using different modeling techniques discussed above. We have used both the bend-angle approach and the (α, β)-knee point approach (with $\alpha = 0.5$ and $\beta = 5$). In all cases except lidar, the (α, β) approach also finds a single knee point, identical to that found by the bend-angle approach. The knee points are marked in each figure.

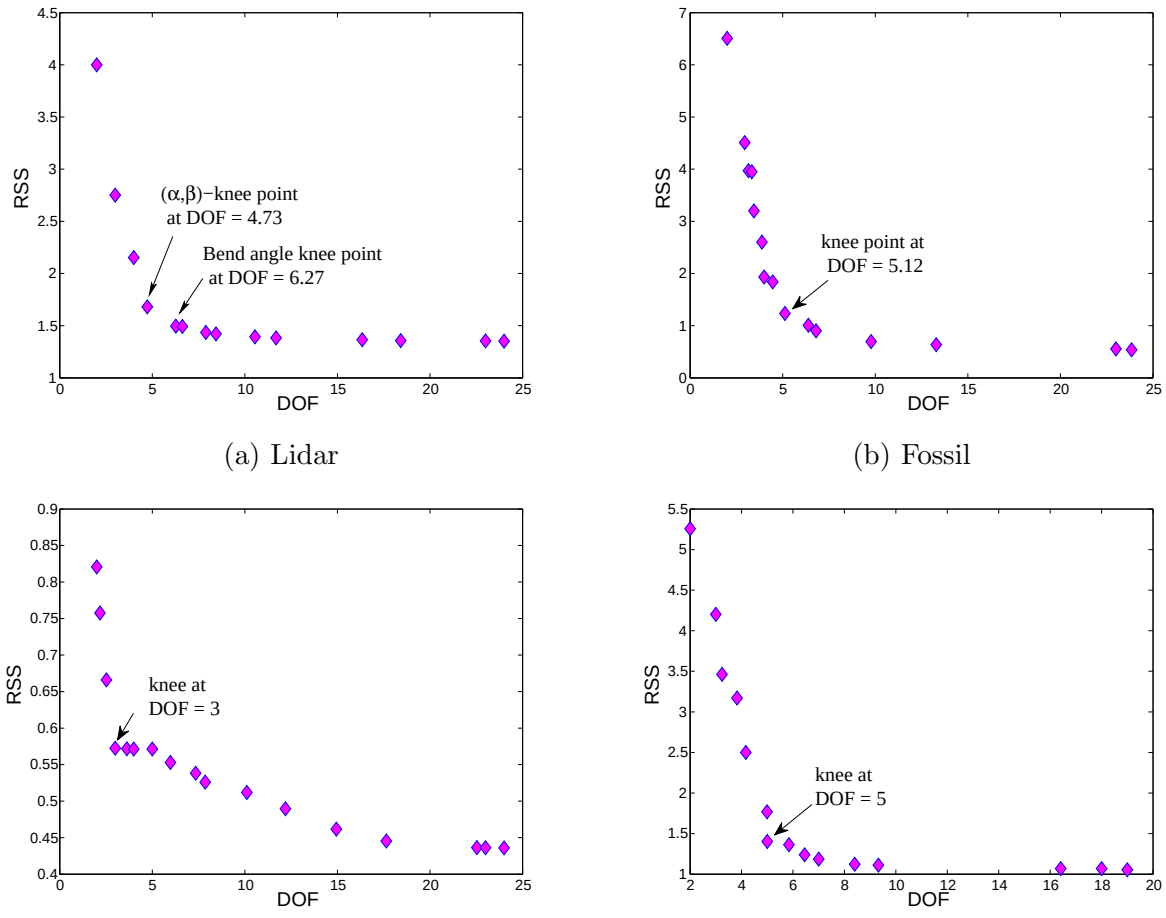
Table 5 tabulates the DOF and RSS values of 'Lidar' dataset and corresponding knee points detected by both approaches. The (α, β)-knee point is found to be the point with $DOF = 4.73$,



(a) Lidar (b) Fossil

(c) Electrical Usage (d) Simulated

Figure 20: Four different data sets used for Regression Analysis.



(a) Lidar

(b) Fossil

(c) Electrical Usage

(d) Simulated

Figure 21: Plot of RSS-DOF for different curve fitting models applied on the given data sets

Table 5: Knee points (shown in bold) found by the proposed methodologies on 'Lidar' dataset. The fourth point is declared a knee point by the (α, β) approach and the fifth point is by bend-angle approach.

Curve Fitting Model	DOF	RSS	Prop. of Inferior Region	Bend-angle
Linear Reg.	2.00	4.00		
Poly. Reg. (2^{nd} order)	3.00	2.75	0.749	0.969
Poly. Reg. (3^{rd} order)	4.00	2.15	0.879	1.121
Pen. Linear Spline ($\lambda = 200$)	**4.73**	**1.68**	**1.000**	1.289
Pen. Linear Spline ($\lambda = 100$)	**6.27**	**1.50**	0.844	**1.298**
Quadratic Local Reg. ($h = 0.1$)	6.63	1.49	0.828	1.285
Quadratic Local Reg. ($h = 0.08$)	7.89	1.44	0.703	1.254
Pen. Linear Spline ($\lambda = 50$)	8.44	1.42	0.679	1.241
Pen. Linear Spline ($\lambda = 30$)	10.55	1.39	0.587	1.170
Quadratic Local Reg. ($h = 0.05$)	11.68	1.38	0.538	1.132
Pen. Linear Spline ($\lambda = 10$)	16.34	1.37	0.334	0.968
Quadratic Local Reg. ($h = 0.03$)	18.42	1.36	0.243	0.913
Linear Spline	23.00	1.35	0.044	0.809
Quadratic Spline	24.00	1.35		

close to polynomial regression with a fourth-order polynomial, whereas the bend-angle approach finds the regression with a $DOF = 6.27$ – close to fifth-order polynomial. Figure 21(a) shows that these two points on the objective space are close to each other and are visually likely to be a knee point.

For the 'Fossil' dataset, the knee point is obtained at $DOF = 5.12$ using both (α, β) and bend angle approach. This is close to a fourth-order polynomial regression (Table 6). For the 'Electric' data-set (Table 7), there are three regression approaches (penalized linear spline with $\lambda = 100$ and 200, and the polynomial regression of order two) that have maximum proportion of inferior region with the chosen α and β values. In principle, any of these three approaches can be considered as a knee solution, however, if a smaller α value is chosen, the polynomial regression of order two becomes the knee solution. For the the 'Simulated' data-set (Table 8), the knee points occur *exactly* at $DOF = 5$ by both approaches. It is interesting that both approaches of defining a knee point provide similar outcomes to all the four datasets,

It is also clear from the plots that polynomial regression points are clustered around the knee point, indicating that polynomial regression with $DOF \in [3, 6]$, in general, provides a good trade-off between computational complexity and obtained error of regression. Based on this finding, we argue that this can be one of the reasons why researchers prefer polynomial regression with a small DOF.

9.3 Data Clustering

Clustering is a widely-used computational task in almost any field of science and engineering. Researchers have preferences of one clustering algorithm over another. In many clustering problems, the number of clusters is usually not known beforehand and the part of the clustering task is also to determine how many clusters is optimum for the given data set. Here, we consider a simple data set, which can be visually clustered into a set of five clusters. Figure 22 shows the data set. Instead of trying different clustering algorithms, here, we use the well-known *k-mean* clustering algorithm and keep the number of cluster as a parameter. Thus, when p clusters are fixed in the clustering algorithm, the algorithm is called p-clustering algorithm, denoting that this specific algorithm can only cluster points data sets into p different clusters.

Table 6: Knee point (shown in bold) found by the proposed methodologies on 'Fossil' dataset.

Curve Fitting Model	DOF	RSS	Prop. of Inferior Region	Bend-angle
Linear Reg.	2.00	6.51		
Pen. Linear Spline ($\lambda = 50$)	2.95	4.51	0.500	0.834
Pen. Linear Spline ($\lambda = 40$)	3.15	3.97	0.530	0.903
Quadratic Local Reg. ($h = 0.15$)	3.33	3.95	0.523	0.883
Pen. Linear Spline ($\lambda = 30$)	3.45	3.20	0.560	1.006
Quadratic Local Reg. ($h = 0.1$)	3.88	2.60	0.588	1.079
Poly. Reg. (3^{rd} order)	4.00	1.93	0.792	1.201
Quadratic Local Reg. ($h = 0.08$)	4.46	1.84	0.774	1.187
Poly. Reg. (4^{th} order)	5.00	1.83	0.753	1.152
Pen. Linear Spline ($\lambda = 10$)	**5.12**	**1.23**	**1.000**	**1.277**
Quadratic Local Reg. ($h = 0.05$)	6.39	1.01	0.879	1.258
Pen. Linear Spline ($\lambda = 5$)	6.80	0.90	0.879	1.264
Quadratic Local Reg. ($h = 0.03$)	9.78	0.70	0.690	1.178
Pen. Linear Spline ($\lambda = 1$)	13.29	0.64	0.500	1.052
Linear Spline	23.00	0.55	0.033	0.761
Quadratic Local Reg. ($h = 0.01$)	23.85	0.54		

Table 7: Knee point (shown in bold) found by the proposed methodologies on 'Electric' dataset.

Curve Fitting Model	DOF	RSS	Prop. of Inferior Region	Bend-angle
Linear Reg.	2.00	0.82		
Pen. Linear Spline ($\lambda = 200$)	**2.17**	**0.76**	**0.953**	0.818
Pen. Linear Spline ($\lambda = 100$)	**2.52**	**0.67**	**0.953**	0.956
Poly. Reg. (2^{nd} order)	**3.00**	**0.57**	**0.953**	**1.158**
Quadratic Local Reg. ($h = 1.0$)	3.62	0.57	0.925	1.106
Poly. Reg. (3^{rd} order)	4.00	0.57	0.908	1.074
Poly. Reg. (4^{th} order)	5.00	0.57	0.862	0.989
Quadratic Local Reg. ($h = 0.2$)	5.98	0.55	0.818	0.982
Quadratic Local Reg. ($h = 0.15$)	7.34	0.54	0.756	0.918
Pen. Linear Spline ($\lambda = 5$)	7.86	0.53	0.733	0.923
Quadratic Local Reg. ($h = 0.1$)	10.11	0.51	0.630	0.862
Quadratic Local Reg. ($h = 0.08$)	12.19	0.49	0.536	0.840
Pen. Linear Spline ($\lambda = 1$)	14.94	0.46	0.411	0.888
Quadratic Local Reg. ($h = 0.05$)	17.64	0.45	0.289	0.849
Pen. Linear Spline ($\lambda = 0.1$)	22.53	0.44	0.067	0.820
Linear Spline	23.00	0.44	0.045	0.809
Quadratic Spline	24.00	0.44		

Table 8: Knee point (shown in bold) found by the proposed methodologies on 'Simulated' dataset.

Curve Fitting Model	DOF	RSS	Prop. of Inferior Region	Bend-angle
Linear Reg.	2.00	5.26		
Poly. Reg. (2^{nd} order)	3.00	4.20	0.128	0.670
Pen. Linear Spline ($\lambda = 1.0$)	3.24	3.46	0.623	0.849
Quadratic Local Reg. ($h = 0.8$)	3.83	3.17	0.623	0.844
Pen. Linear Spline ($\lambda = 0.5$)	4.17	2.50	0.697	1.002
Quadratic Local Reg. ($h = 0.5$)	4.99	1.77	0.875	1.157
Poly. Reg. (4^{th} order)	**5.00**	**1.40**	**1.000**	**1.280**
Quadratic Local Reg. ($h = 0.4$)	5.84	1.36	0.937	1.237
Quadratic Local Reg. ($h = 0.35$)	6.45	1.24	0.904	1.242
Poly. Reg. (6^{th} order)	7.00	1.19	0.818	1.228
Quadratic Local Reg. ($h = 0.25$)	8.40	1.12	0.671	1.179
Pen. Linear Spline ($\lambda = 0.07$)	9.32	1.11	0.582	1.134
Pen. Linear Spline ($\lambda = 0.01$)	16.41	1.07	0.135	0.834
Linear Spline	18.00	1.07	0.052	0.733
Quadratic Spline	19.00	1.05		

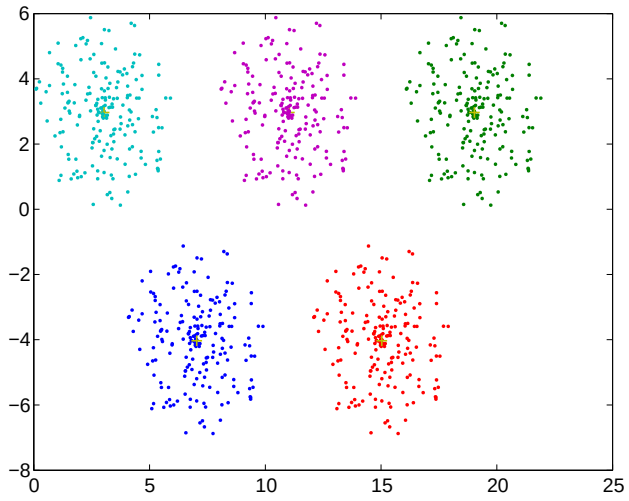


Figure 22: Data set for the clustering problem.

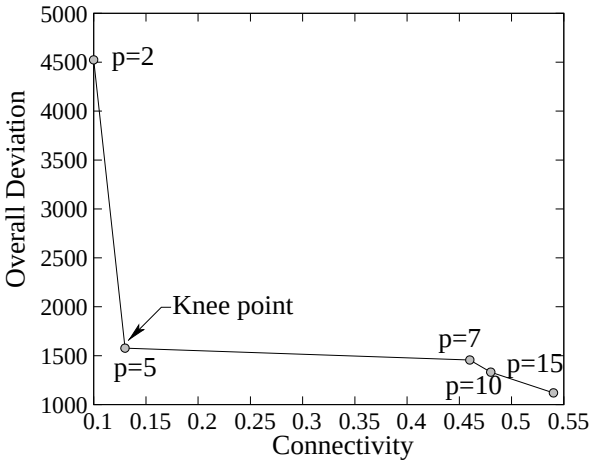


Figure 23: Trade-off frontier and the identified knee point for the clustering problem.

To evaluate how well a p-clustering algorithm behaved for a data set will then require one or more criteria. The following two criteria are important for a clustering task: *Overall Deviation* (OD) and *Connectivity* (CON). In the k-mean clustering method, cluster centroids are calculated as mean of data points belonging to the cluster. Assuming that $\mathcal{N}$ be the total number of points in the data set, p be the number of clusters that p-clustering algorithm desires, N_i be the number of data points that belong to cluster $\mathcal{C}_i$ with centroid c_i, where i goes from one to p, x_j be the j^{th} data point, where j goes from one to $\mathcal{N}$ then:

$$c_i = \frac{1}{N_i} \sum_{\forall j \ \in \ \mathcal{C}_i} x_j, \tag{32}$$

$$OD = \sum_{i=1}^{p} \sum_{\forall j \ \in \ \mathcal{C}_i} \|x_j - c_i\|_2, \tag{33}$$

$$CON = \frac{1}{\min \|c_i - c_k\|_2}, \quad \forall \ i, k \in [1, p], k \neq i. \tag{34}$$

The objective OD is calculated as the sum of distances (Euclidean distance) of all the data points with their respective cluster centroids. The objective CON is calculated as the reciprocal of the minimum of centroid to centroid distances between different clusters. As more clusters are introduced, the data points get closer to the cluster centroids reducing the OD and the cluster space become more compact, thereby reducing the cluster to cluster distances. This increases the connectivity, CON. An adequate number of clusters would be one that tries to reduce OD and CON simultaneously, but since the two objectives are conflicting in nature a preferred solution will be the one that would make a good compromise to both objectives. In the clustering literature, often intra-cluster distance and inter-cluster distances are minimized. The objectives OD and CON have similar connotations.

The k-mean clustering with different clusters ($p = 2$ to 15) is applied to the data set shown in Figure 22. Figure 23 shows the bi-objective trade-off points in the space of OD and CON values. There is a clear knee which can be observed at $p = 5$. It is clear that for smaller p values, the CON value is better, but the OD value is worse and for larger p values, the opposite is true. But it is not obvious that the trade-offs in objectives on the left and right of $p = 5$ point are so large. The trade-offs do not seem to be large for any other p value. The clustering algorithm working with the five-cluster assumption ($p = 5$) is ideal for this data set, since intuitively the data set has clearly five clusters of points.

Table 9 shows the proportion of inferior region and bend-angle for the points. We have used

Table 9: Knee point (shown in bold) found by the proposed methodologies on the clustering dataset.

p	Connectivity	Deviation	prop. of inferior region	Bend-angle
2	0.10	4523.90		
5	**0.13**	**1576.70**	**1.000**	**1.349**
7	0.46	1456.50	0.099	0.336
10	0.48	1331.50	0.062	0.399
15	0.54	1120.20		

$\alpha = 10,000$ and $\beta = 1/10,000$. No knee-region is found by the (α, β) approach. It is clear that both methods identified $p = 5$ solution as the knee point. Although this result is obtained for a specific clustering problem, it clearly shows that if there is clear solution to the problem, it usually lies on a knee.

9.4 Vehicle Fuel Consumption

A lot of research has been done on driving conditions to improve the fuel economy of a vehicle. Efforts are also being made to improve the driving conditions on road, but besides technical improvisations, often it is observed that people know few tricks, like maintaining a constant speed as much as possible or following the specified speed limits thinking that there must have been some basis for fixing them, etc. which leads to savings in fuel. It is observed that 55 mph is a speed that most people prefer to drive [12]. The question we would like to ask is why 55 mph is preferred and suggested?

To find an answer to the above question, let us first investigate what does a person intend to achieve while covering a distance. Of course, an increased speed will take the person quicker to the destination. If minimization of driving time is the only consideration, everyone would have driven at the maximum speed that a car can allow. Of course, safety and maneuvering ability of a person at a high speed are conflicting concerns. But since these are difficult to account for, we shall consider another conflicting concern that directly affect the pocket of the person – the fuel economy. Does more speed means more fuel consumption?

The fuel consumption economy can be given by the following empirical relationship [1]:

$$FCE = a + \frac{b}{s} + cs + ds^2, \tag{35}$$

where FCE is the *fuel consumption economy* in (lt/100 km), s is the vehicle speed in (km/h), and parameters a, b, c and d are the empirical parameters that define the model of the car [1]. The time taken to drive 100 km at a constant speed of s km/h is $T = 100/s$ h, and the total fuel consumption to drive 100 km is FCE. Figure 24 shows that the FCE and TT values are in conflict with each other only when the speed is higher than 80 kmph. For less than 80 kmph speed, the time taken to cover 100 km and required fuel consumption both increase. Thus, all solutions with $TT > 1.25$ h are dominated.

Table 10 presents the proportion of inferior region by the (α, β) approach (with $\alpha = 5$ and $\beta = 1/5$) for each non-dominated solution. It can be seen that the solution with $TT = 1$ h is considered to be the (α, β)-knee point. This corresponds to a speed of 100 km/h. However, the bend-angle approach finds a knee at $TT = 0.71$ h, with a speed of $s = 140.85$ kmph. Since the bend-angle approach normalizes the dataset for computing the bend-angle, the solution with the maximum bend-angle does not provide an adequate knee point, near an edge-knee scenario.

Despite the trade-off between fuel consumption and travel time in driving at a speed higher than 80 kmph, it seems that driving at 100 kmph becomes a knee point. We argue that this solution

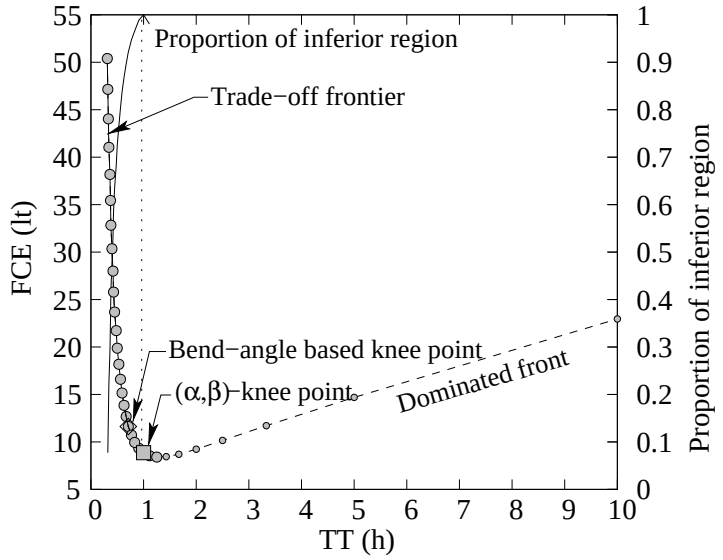


Figure 24: Plot of FCE with TT shows a near edge-knee at $TT = 1$ h.

Table 10: Knee points (shown in bold) found by the proposed methodologies for the fuel consumption problem. $TT = 1.00$ solution is (α, β)-knee point and $TT = 0.71$ solution is bend-angle based knee point.

TT (h)	FCE (lt)	prop. of inferior region	Bend-angle
0.31	50.39		
0.32	47.14	0.077	0.684
0.33	44.03	0.151	0.717
0.34	41.04	0.222	0.752
0.36	38.18	0.290	0.747
0.37	35.44	0.355	0.791
0.38	32.83	0.417	0.833
0.40	30.35	0.476	0.849
0.42	28.00	0.532	0.868
0.43	25.78	0.585	0.913
0.45	23.69	0.634	0.936
0.48	21.72	0.681	0.942
0.50	19.88	0.725	0.969
0.53	18.18	0.765	0.979
0.56	16.60	0.803	0.992
0.59	15.16	0.837	1.004
0.63	13.85	0.868	1.004
0.67	12.67	0.896	1.004
0.71	**11.62**	0.921	**1.006**
0.77	10.72	0.947	0.985
0.83	9.95	0.966	0.967
0.91	9.33	0.989	0.931
1.00	**8.86**	**1.000**	0.891
1.11	8.54	0.981	0.842
1.25	8.40		

is not chosen due to the fact that it causes smallest fuel consumption, but because the saving in driving time with a larger speed is not commensurate with the sacrifice in fuel economy. Since 100 kmph is roughly about 62 mph, we argue that driving around 62 mph (which is close to 55 mph) makes a good trade-off between time and fuel cost and is probably a popular suggestion while driving on straight roads.

9.5 Signalized Intersection Design

Most of the intersections in the city we observe are signalized intersections. A signalized intersection employs a phasing scheme which divides the common space on an intersection such that only a limited number of pre-defined approaches could use the intersection in a particular phase. What objectives should a traffic engineer consider while designing a signalized intersection? Minimizing the time spent by a vehicle on the intersection (*delay*) should have the first priority. Also for a smooth transit system, the number of vehicles that could use the intersection should be maximized. But can this be done simultaneously? The answer is no. This is because given an intersection and limiting our focus to only pre-timed signals, as soon as one tries to increase the number of vehicles passing through an intersection, the time taken by the intersection to service that demand increases and hence increasing the delay faced by the vehicles. In such a scenario, what would be the best solution for a traffic engineer? If the design parameters could be changed such that delay is lesser than a maximum limit and the intersection could service vehicles near its capacity, then that would be 'the best'. The delay at a signalized intersection can be given by [10, 11]:

$$\begin{aligned} D &= D_1 + \sigma_{D_1} + D_2 + \sigma_{D_2}, \\ D_1 &= 0.5C\frac{(1-\lambda)^2}{1-\lambda\cdot x_1}, \\ D_2 &= 900T\left((X-1)+\sqrt{(X-1)^2+\frac{8kX}{cT}}\right), \\ \sigma_{D_1} &= \frac{C^2(1-\lambda)^3(1+3\lambda-4\lambda x_1)}{12(1-\lambda x_1)^2}, \\ \sigma_{D_2} &= \frac{TX}{2c}+\frac{T^2(1-X)^2}{12}, \end{aligned} \tag{36}$$

where D is the *total delay*, C is cycle length (60 sec), λ is greentime/cycletime (0.5), T is time of evaluation (15 min), X is *volume to capacity ratio* (varied from 0.1 to 1.6), x_1 is min(X,1), c is capacity flow rate (900 veh/hr), and k is 0.5.

Figure 25 shows the variation between *delay* and *capacity-volume ratio* for a single approach single lane signalized intersection. Here, we have used $\alpha = 100$ and $\beta = 1/100$ for the (α, β) approach. The knee point obtained by both approaches are identical to each other: (Cap/vol=1.25 and Delay=60.04).

At this solution the intersection is operating near capacity though under-saturated, as on over-saturation would lead to very high delays. Such a design is preferred by a traffic engineer and our analysis shows that this specific desired solution corresponds to a knee solution to the underlying bi-objective optimization problem. This reiterates the fact that a knee solution, if exists, is not an arbitrary solution to a problem, but usually corresponds to a preferred solution.

9.6 Leg Mechanism Design

A robotic leg-mechanism design resorts to a complex optimization task which involves optimization of a number of objectives and satisfaction of a number of trigonometric constraints [9]. Although the original study considered a number of objectives, here we discuss the two-objective scenario

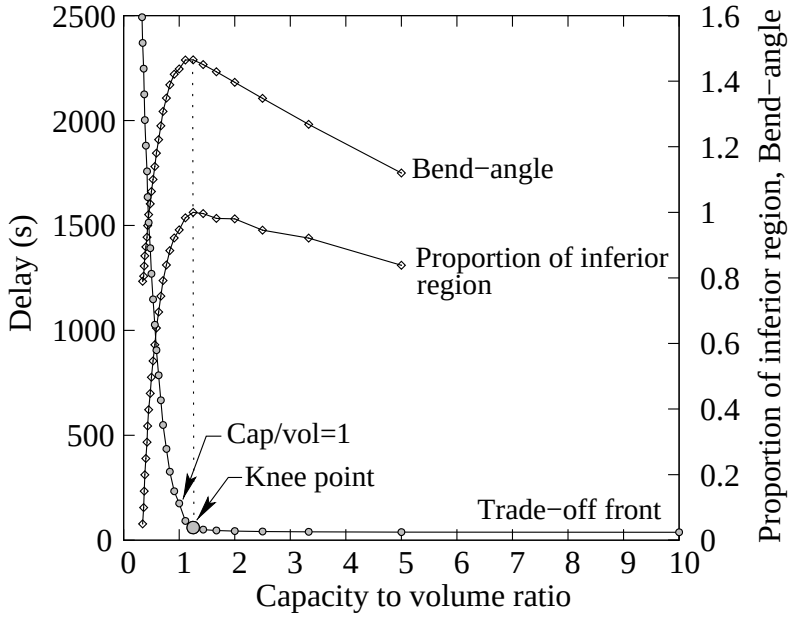


Figure 25: Trade-off frontier and the identified knee point for the signalized intersection delay problem.

in which minimization of force applied to the ground and energy required to execute the walking motion are considered. The force allows the leg to be moved forward and the energy is needed to lift the leg vertically. Of many constraints, there were two constraints which restrict a lower bound on horizontal and vertical stride. For more details, readers are referred to the original study [9]. Figure 26 shows the trade-off front and the presence of a knee point.

Both the (α, β)-knee approach (with $\alpha = 10$ and $\beta = 0.1$) and the bend-angle approach find the same point as the knee point: (Force, Energy)=(1.948, 3.535) (kN, J). The point is marked on the figure. The corresponding proportion of inferior region is 1.0 and the bend-angle is 1.333 rad. Figure 27 shows the variation of the proportion of inferior region and the bend-angle. It can be clearly seen that the knee point is obtained for the maximum value of both these quantities. Interestingly, solutions applying a force smaller than that needed for the knee point requires a larger vertical leg movement beyond the specified lower bound, thereby requiring more energy for leg operation. Similarly, the solutions applying a larger force than that at the knee solution requires a larger horizontal leg movement beyond the specified lower bound, thereby requiring a larger actuating force. It is the knee solution which adheres to both specified horizontal and vertical bounds. Interestingly, the additional energy needed to lift the leg to gain an advantage on the actuating force is quite high and on the other hand, the gain in energy needed for making larger horizontal stride requires a much larger actuating force. Due to these requirements, a designer would not be motivated to adopt any design other than that at the knee solution.

This problem illustrates a generic phenomenon. If constraints are absent, the resulting solution becomes a trivial one – no walking at all, involving zero actuating force and zero required energy. Since a minimum horizontal stride is recommended as a hard constraint, some amount of force is needed to be applied. On the other hand, since a lower bound on the vertical leg movement is specified, some energy is required. When both objectives are of importance, the Pareto-optimal front is likely to divided around a specific point. On either side, one constraint is expected to be active, while the other constraint is inactive and the amount of constraint value provides the trade-off between the objectives. The specific point is a unique solution at which both constraints are active. Additionally, if the trade-off on either side is unfavorable in terms the gain-to-loss ratio (as in this case), the specific point becomes a knee point and stands as an obvious preferred solution.

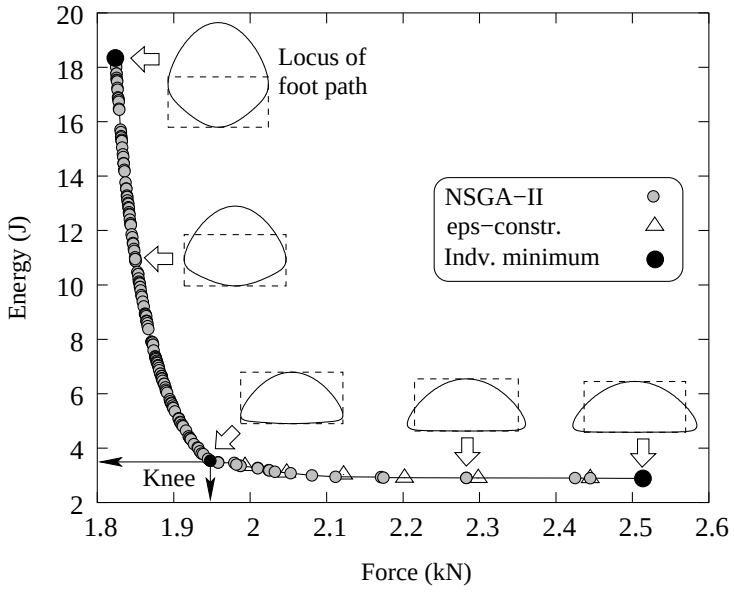


Figure 26: Energy versus actuating force for the leg mechanism design.

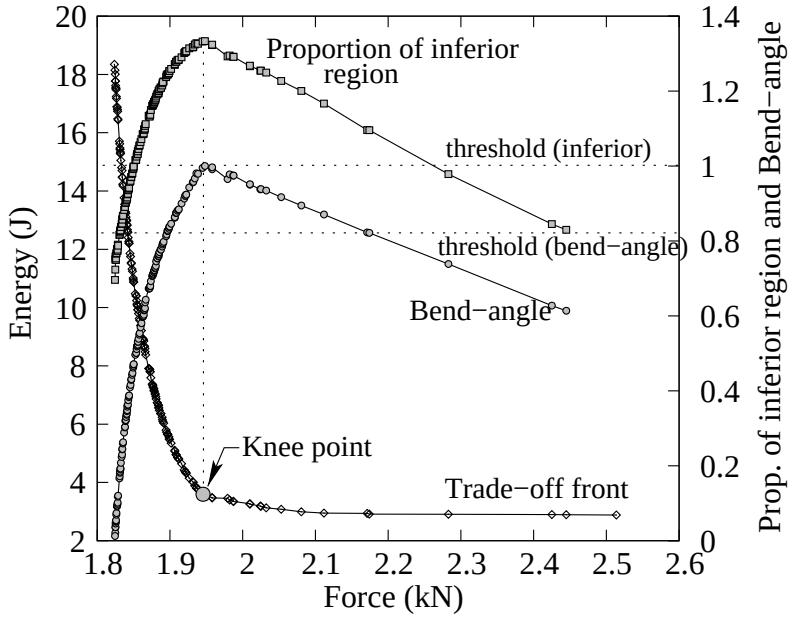


Figure 27: The trade-off frontier and proportion of inferior region and bend-angle are shown for the leg mechanism design problem.

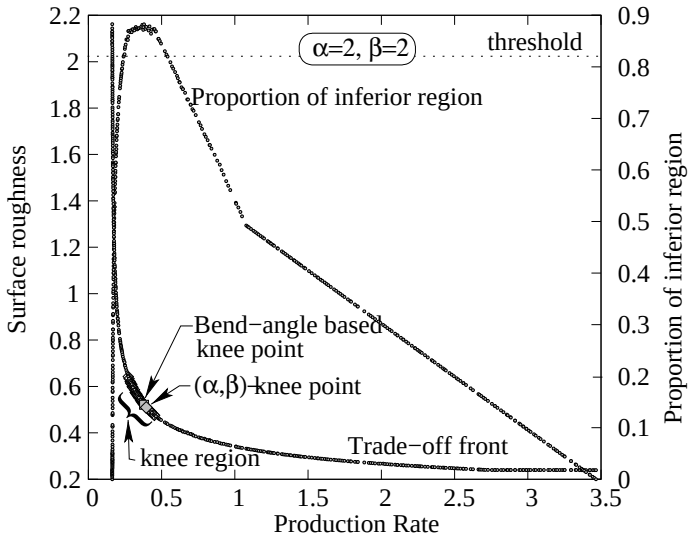


Figure 28: Surface roughness versus production rate for the metal cutting process design problem.

When individual optimal solutions to each objective is trivial, the presence of constraints causes a parsimonious solution to be preferred. In such situations, a knee solution is often the parsimonious solution.

9.7 Metal Cutting Process Design

Finding the optimum machining parameters is an important task in every machining process. This helps in saving time, energy and also helps in producing a better product. This is another engineering problem which usually has a number of conflicting goals. Let us consider a turning process in which minimization of production rate (f_1) and minimization of surface roughness (f_2) are two important goals. The parameters which control these two goals are rotational speed (v), feed of cutter per revolution (f) and depth of cut (a). The optimization problem is given as follows with $\mathbf{x} = (f, v, a)^T$:

$$\begin{array}{rl} \text{minimize} & f_1(\mathbf{x}) = 0.16 + 2313.76(1 + 0.26/T(v,f,a))/MRR(v,f,a), \\ \text{minimize} & f_2(\mathbf{x}) = 1.001\left(v^{0.0088} f^{0.3232} a^{0.3144}\right), \\ \text{subject to} & g_1(\mathbf{x}) \equiv 5 - P(v,f,a) \geq 0, \\ & g_2(\mathbf{x}) \equiv 230 - F(v,f,a) \geq 0, \\ & \mathbf{x}^{(L)} \leq \mathbf{x} \leq \mathbf{x}^{(U)}, \end{array} \tag{37}$$

where

$$\begin{array}{ll} T(v,f,a) = 1575134.21(v^{-1.70} f^{-1.55} a^{-1.22}), & MRR(v,f,a) = 1000vfa, \\ F(v,f,a) = 1.38 f^{1.18} a^{1.26}, & P(v,f,a) = 0.000626 v f^{1.18} a^{1.26} \\ \mathbf{x}^{(L)} = (0.1, 70, 0.1)^T, & \mathbf{x}^{(U)} = (2, 90, 5)^T. \end{array}$$

A recent study [6] showed that production rate and surface roughness are conflicting to one another (Figure 28). A knee-region is evident in the figure. This shows that there is indeed a combination of speed, feed and depth of cut that can be preferred without compromising much on any of the two objectives.

In this example, we demonstrate the use of the upper bound estimation procedure for α and β parameters on the width of the obtained knee-region, as described in Section 7. Figure 29 shows the variation of the maximum α and β values on the proportion of knee-region to the entire length of the trade-off frontier. Several interesting observations can be made from the figure. First, the

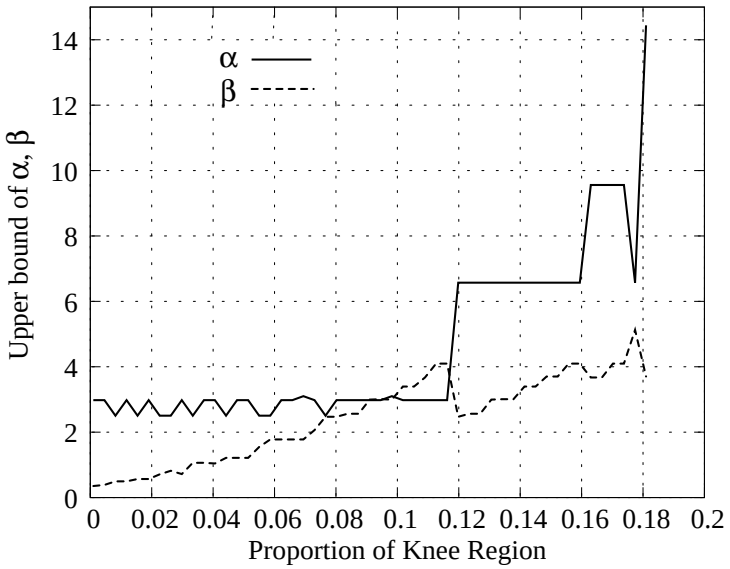


Figure 29: Variation of upper bound of α and β versus resulting proportion of knee-region for the metal cutting process design problem.

Table 11: Knee points found by the proposed methodologies for the metal cutting problem.

Approach	Time	SR	prop. of inferior region	Bend-angle
Bend-angle	0.3946	0.5094	0.87741	**1.33926**
(α, β)	0.3796	0.5209	**0.88230**	1.33844

figure suggests that the left-side trade-off (α) value can be chosen to be higher than the right-side trade-off (β) value. Second, the upper bound of the variation of α can be restricted within $[2.5, 14.5]$, whereas the same for β may be restricted within $[0.5, 5.0]$. Any trade-off value higher than these upper bounds will not result in any knee point for the given frontier. Third, since α values are more or less same for a δ value up to 12% of the entire length of the frontier, it is wise to use α smaller than approximately 2.5 to obtain a sharp knee.

Based on the above analysis, we choose $\alpha = \beta = 2.0$ and use the (α, β)-approach to locate the knee point. Table 11 shows the (α, β)-knee point and also the bend-angle based knee point for the given Pareto-optimal points. For both approaches, the obtained knee points are close to each other. These points are marked in Figure 28. The proportion of inferior region is also shown in the figure. It is clear that the proportion increases sharply near the knee point and its maximum is greater than the chosen threshold of 0.82. A similar plot is also observed for the bend-angle variation.

With $\alpha = \beta = 2$, there is also an (α, β)-knee-region, varying in the range (PR $\in [0.2720, 0.4507]$), as shown in the figure. To show the effect of the α and β parameters, we use $\alpha = 1$ and $\beta = 1$ and recompute the knee point again. Now, we obtain a sole (α, β)-knee point: (PR, SR) = (0.3237, 0.5702). With such a small trade-off, there is no knee-region for the same front. This example clearly shows that the definition of the (α, β)-knee point is dependent on the chosen α and β values.

9.7.1 Innovization within the Knee-Region

Among all the knee-region points obtained with $\alpha = \beta = 2$, we observe that $v = 90$ m/min is common. None of these solutions make constraints g_1 and g_2 active. By investigating the two

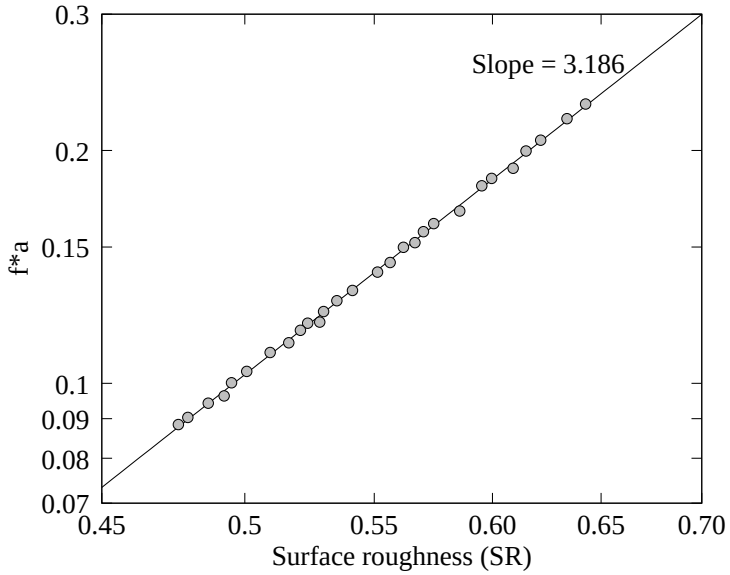


Figure 30: The product of depth of cut and feed increases polynomially with the required surface finish (SR or f_2), obtained as a common principle among the knee solutions.

objectives, we notice the following:

$$f_1(\mathbf{x}) \approx 0.16 + \frac{0.257}{vfa}, \tag{38}$$

$$f_2(\mathbf{x}) = 1.001 v^{0.0088} f^{0.3232} a^{0.3144}. \tag{39}$$

Substituting $v = 90$ m/min for the points in the knee-region, we obtain another empirical relationship (shown also in Figure 30):

$$fa = 0.9336(\text{SR})^{3.186}, \quad 0.4761 \leq \text{SR} \leq 0.6426. \tag{40}$$

Since these solutions are close to the knee point, investigating common properties, vis-a-vis the entire trade-off frontier, should provide an user valuable information about important trade-off solutions. By analyzing the knee solutions of the metal cutting problem, we found the following common properties:

1. The cutting speed is fixed to all knee solutions having $v = 90$ m/min (upper limit of v set by the user).

2. In order to achieve a particular surface finish, depth of cut is inversely proportional to feed. For a particular feed f, the optimal depth of cut (a) can be calculated using equation 40.

Such information is certainly not intuitive and requires first finding a set of preferred trade-off solutions using the knee concept and then analyzing the common principles hidden to the optimized data. Although we have used a manual analysis of the solutions in the knee-region here, recent automated procedures [2] can also be used.

10 Conclusions

In this paper, we have suggested a couple of approaches for identifying a knee point in a bicriteria optimization problem. The first approach uses the concept of a bend-angle of a point computed from

two extreme points and identifies the point with the maximum bend-angle as the knee point. On the other hand, the second approach uses the supplied critical trade-off information to identify the point which maximizes the region that is inferior to the point based on the trade-off information. Based on the cone-dominated approach, we have also proposed the idea of a knee-region for problems in which instead of a single solution, a set of consecutive solutions may qualify as knee-like solutions. Similarly, instead of the knee point lying in the intermediate region of the trade-off frontier, a knee point may also exist near one of the extreme solutions. We have proposed the concept of an edge-knee point for this purpose. The ideas proposed for identifying a knee point, a knee region and an edge-knee point have been illustrated well on a tunable test problem exhibiting a knee with varying sharpness.

We have also reviewed the various approaches that have been employed by researchers to define or find knee related points. Most of these techniques fall short in completely elaborating the concept of knee but with the ideas propagated through this paper, efforts can be made to fill the gaps in these definitions as well. This paper has made a comprehensive discussion of knee and related preferred points and suggested procedures for identifying them.

Furthermore, to facilitate an user to choose appropriate trade-off parameter values, we have suggested an upper bound estimation procedure from the given trade-off frontier. This analysis provides a sensitivity of the upper bounds of trade-off parameters on the width of the resulting knee-region. Such an analysis should help researchers and applicationists to get a priori information about the range of trade-off parameters which may provide a knee or a knee-like point on a given trade-off frontier.

There is no denying the fact that knee solutions, if exist, are the most important solutions to a decision maker but the fact that not all bicriteria problems have a knee or a knee-like solution suggests that there could be some special relationship between problem parameters and objectives that may result in a trade-off front exhibiting a knee. Using the first and second-order gradients of the trade-off frontier, we have revealed certain properties which will cause the frontier to exhibit a knee. Such properties are important and can be used to design test problems having knees and should help advance the understanding and discovery of knee points better.

Since the knee points are usually preferred solutions, we have argued that the reason for the existence of a popular preferred solution methodology in certain problem-solving tasks is related to it being a knee point in a underlying bicriteria problem. We have chosen a number of generic problem-solving tasks, such as sorting, regression, clustering and a few engineering design problems to illustrate this aspect. The relation between knee points in a bicriteria problem and popularity of certain solution methodologies is interesting and should help develop efficient solution methodologies in other problem-solving tasks.

Acknowledgments

The study is funded by Department of Science and Technology, Government of India under SERC-Engineering Sciences scheme (No. SR/S3/MERC/091/2009).

References

[1] H. K. Al-Omishy. Prediction of the fuel consumption of petrol and diesel vehicles from a computer simulation model. *Energy*, 14(9):563–570, 1989.

[2] S. Bandaru and K. Deb. Automated discovery of vital knowledge from pareto-optimal solutions: First results from engineering design. In *World Congress on Computational Intelligence (WCCI-2010)*. IEEE Press, 2010.

[3] S. Bechikh, L. Ben Said, and K. Ghédira. Searching for knee regions in multi-objective optimization using mobile reference points. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1118–1125. ACM, 2010.

[4] J. Branke, K. Deb, H. Dierolf, and M. Osswald. Finding knees in multi-objective optimization. In *Parallel Problem Solving from Nature (PPSN-VIII)*, pages 722–731. Heidelberg, Germany: Springer, 2004. LNCS 3242.

[5] I. Das. On characterizing the "knee" of the Pareto curve based on normal-boundary intersection. *Structural Optimization*, 18:107–115, 1999.

[6] R. Datta and K. Deb. A classical-cum-evolutionary multi-objective optimization for optimal machining parameters. In *World Congress on Nature and Biologically Inspired Computing (NaBIC-2009)*, pages 607–612, 2009.

[7] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, Chichester, UK, 2001.

[8] K. Deb, J. Sundar, N. Uday, and S. Chaudhuri. Reference point based multi-objective optimization using evolutionary algorithms. *International Journal of Computational Intelligence Research (IJCIR)*, 2(6):273–286, 2006.

[9] K. Deb and S. Tiwari. Multi-objective optimization of a leg mechanism using genetic algorithms. *Engineering Optimization*, 37(4):325–350, 2005.

[10] F. Dion, H. Rakha, and Y. S. Kang. Comparison of delay estimates at under-saturated and over-saturated pre-timed signalized intersections. *Transportation Research Part B*, 38:99–122, 2004.

[11] L. Fu and B. Hellinga. Delay variability at signalized intersections. *Transportation Research Record: Journal of the Transportation Research Board*, (1710):215–221, 2000.

[12] D. B. Kamerud. The 55 mph speed limit: Costs, benefits, and implied trade-offs. Technical report, Societal Analysis Department, General Motors Research Laboratories,, Warren, U.S.A., 1981.

[13] C. A. Mattson, A. A. Muller, and A. Messac. Smart pareto filter: Obtaining a minimal representation of multiobjective design space. *Engineering Optimization*, 36(6):721–740, 2004.

[14] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer, Boston, 1999.

[15] L. Rachmawati and D. Srinivasan. A multiobjective evolutionary algorithm with weighted-sum niching for convergence on knee regions. In *Proceedings of IEEE Congress on Evolutionary Computation (CEC-2006)*, pages 1916–1923, 2006.

[16] L. Rachmawati and D. Srinivasan. Multiobjective evolutionary algorithm with controllable focus on the knees of the pareto front. *IEEE Transactions on Evolutionary Computation*, 13(4):810–824, 2009.

[17] D. Ruppert, M. P. Wand, and R. J. Carroll. *Semiparametric regression*. Cambridge University Press, 2003.

[18] O. Schütze, M. Laumanns, and C. A. Coello Coello. Approximating the knee of an mop with stochastic search algorithm. In *Parallel Problem Solving from Nature (PPSN X), (LNCS 5199)*, pages 795–804. Heidelberg, Berlin: Springer, 2008.