A Theoretical and Algorithmic Characterization of Bulge Knees

Pradyumn Kumar Shukla, Marlon Alexander Braun, and Hartmut Schmeck

Institute AIFB, Karlsruhe Institute of Technology
Karlsruhe, D-76128, Germany
pradyumn.shukla@kit.edu

Abstract. This paper deals with the problem of finding convex bulges on the Pareto-front of a multi-objective optimization problem. The point of maximum bulge is of particular interest as this point shows good trade-off properties and it is also close to the non-attainable utopia point. Our approach is to use a population based algorithm to simultaneously promote convex bulges and improve the current approximation of the individual minimum of the objectives. This is done by changing the ranking of the solutions, and by proposing a new domination scheme that is used to sort the solutions. Theoretical results characterize the interrelationships between the bulge knee, the weighted sum method, and the guided domination approach.

Keywords: bulge knee; domination; convexity ; evolutionary algorithms

1 Introduction

This work focusses on searching for interesting points on the Pareto front of a multi-objective optimization problem where m objective functions $f_1 : S \to \mathbb{R}$, $f_2 : S \to \mathbb{R}, \ldots, f_m : S \to \mathbb{R}$, need to be minimized. The Pareto front of an m-objective problem is usually an $m-1$ dimensional manifold and an analytical or a detailed approximation of it is either intractable or computationally prohibitive. Due to these reasons and in the absence of any preference information, the knee point is almost always an interesting solution, and is the most sought after.

There exist different notions of knees in multi-objective optimization ([1–5]); trade-offs, (bend) angles, and curvature are some of the properties that are used to quantify knees. Convex bulges on the Pareto front give rise to solutions that have good trade-off properties and these are known as bulge knees. Additionally, the point of maximal bulge is also a point that is close to the non-attainable utopia point (simultaneous minimizer of all the objectives). The main difficulty in finding bulge knees is that one needs to know the extremes of the Pareto front to quantify a bulge knee and this prohibits the use of a trade-off, cone, or a utility based approach for finding these points. Trade-offs, cones, and utility functions are all prone to relative scaling of the objective function values (on the Pareto front) and bulge knee, to the best of our knowledge, is the only knee notion that is independent of relative scaling of the objectives.

There are some exact [6, 4] and population based approaches [7] for finding the bulge knee. The exact approaches use additional hard nonlinear constraints to limit the feasible region and require the information about exact individual minima of the objectives and thus, limiting their applicability. Nevertheless, these exact formulations are helpful to obtain a deeper understanding of the bulge knee, especially its relation to other scalarization techniques. The only population based approach designed to obtain bulge knee is, to the best of our knowledge, to use an archive maintenance scheme in an evolutionary algorithm. The aim there is to investigate bi-objective problems where bulge knees can be obtained easily.

The approach in this paper is to theoretically characterize the bulge knee, and use this characterization to investigate new ranking and domination schemes to approximate the bulge knee. The approach can be used for any number of objectives and uses an adaptive approximation of the extremes of the Pareto front to quantify the bulge.

The main contributions of this paper are as follows:

- The relationship of the bulge knee with the weighted sum method is described, the invariance of the bulge knee with respect to positive scaling of the objectives is shown, and the effect of guided domination on the bulge knee is investigated. (Section 2)
- Two algorithms for finding solutions in the bulge knee region are proposed. (Section 3)
- Theoretical and algorithmic properties are empirically analyzed. (Section 4)

2 Theoretical Characterization

The *bulge knee* was proposed by Das in [6, 4] and uses the normal-boundary intersection method [8]. Normal-boundary intersection is a deterministic search method that constructs an approximation of the Pareto front by solving a series of non-linear optimization problems. It uses a surface normal to the convex hull of individual minima of the objective functions (CHIM) as search direction.

Let $f^* = (f_1^*, f_2^*, \dots, f_m^*)^\top$ be the ideal point of the multi-objective optimization problem. Henceforth, we shift the origin (in objective space) to f^* so that all the objective functions are non-negative. Let the individual minimum of the objective functions be attained at $\mathbf{x}_i^*$ for $i = 1, 2, \dots, m$. The CHIM hyperplane can now be expressed as $\Phi\beta$, where $\Phi = (f(\mathbf{x}_1^*), f(\mathbf{x}_2^*), \dots f(\mathbf{x}_m^*))$ is a $m \times m$ matrix and $\beta = \{(\beta_1, \beta_2, \dots, \beta_m)^\top \mid \sum_{i=1}^m b_i = 1\}$.

The bulge knee is characterized as the point maximizing the distance between the Pareto front and CHIM. Using points on CHIM, the bulge knee point can be computed in the following way (see Figure 1). An arbitrary point $\Phi\beta \in$ CHIM serves as position vector. Starting at $\mathbf{u}$ the normal to CHIM $\mathbf{n}$ is used as directional vector pointing towards the Pareto front. Extending the inverted Pareto cone along the normal vector until it reaches the boundary of the objective space yields a (weakly) Pareto optimal point. Mathematically, the following subproblem (βNBI) for a given vector β can be solved to obtain the bulge knee.

$$\begin{aligned} &\max_{(\mathbf{x},t,\beta)} t, \\ &\text{subject to } \Phi\beta + t\hat{n} \geq f(\mathbf{x}), \\ &\qquad\qquad \mathbf{x} \in S, \\ &\qquad\qquad \beta^\top e = 1, \\ &\qquad\qquad \beta_i \geq 0 \quad \forall i = 1, 2, \ldots, m, \end{aligned} \tag{1}$$

where $S = \{\mathbf{x} \mid h(\mathbf{x}) = 0; g(\mathbf{x}) \leq 0, a \leq \mathbf{x} \leq b\}$ is the set of feasible vectors. This above subproblem is from [4] and it has computational and theoretical advantages over the original one from [6].

We next investigate the relationship between the bulge knee obtained by solving the single objective optimization problem (1) and the weighted sum method. For this, we will use the theory of Lagrange multipliers.

Theorem 1 (Relationship with weighted sum) *Let* $(\mathbf{x}^*, t^*, \beta^*)$ *be the solution of the (1). Then,* $\mathbf{x}^*$ *is the solution to*

$$\begin{aligned} &\min_{\mathbf{x}} \qquad w^\top f(\mathbf{x}), \\ &\textit{subject to } \mathbf{x} \in S, \end{aligned} \tag{2}$$

where $w = \Phi^{-\top} e$ *and* $e = (1, 1, \ldots, 1)^\top$ *is the* m*-dimensional unit vector.*

Proof. By adding positive slack variable s_i^2, we can reformulate the inequalities $g(\mathbf{x})_i \leq 0$ to $g_i(\mathbf{x}) + s_i^2 = 0$. Using this reformulation, let $\bar{h}(\mathbf{x}) = 0$ denote the set of all constraints in S. Although by adding s_i's we increase the number of variables, for simplicity, we denote the combined the combined set of variable $(\mathbf{x}, s)^\top$ by $\mathbf{x}$.

Let us assume that $\beta^* > 0$ (this means that there is a bulge in the Pareto front and it only removes trivial linear scenarios and scenarios where each of the objective functions are concave). Note that the normal to CHIM is $\hat{n} = -\Phi e$.

Considering the modification above, the Lagrangian of (1) can be written as

$$\begin{aligned} \Lambda(x, t, \beta, \lambda_\beta, \lambda_N, \lambda) &= -t + (e^\top \beta - 1)\lambda_\beta \\ +(f(\mathbf{x}) - f^* - \Phi\beta + t\Phi e)^\top \lambda_N &+ \bar{h}(x)^\top \lambda, \end{aligned}$$

where λ_β is the vector of Lagrange multipliers of the constraint $\beta^\top e = 1$, λ_N is the vector of Lagrange multipliers of the constraint $\Phi\beta + t\hat{n} \geq F(\mathbf{x})$, and λ is the vector of Lagrange multipliers of the constraint $\bar{h}(\mathbf{x}) = 0$.

A part (with respect to variable t) of the necessary KKT optimality condition for the optimization problem (1) is

$$\nabla_t \left((f(\mathbf{x}^*) - \Phi\beta^* + t\Phi e)^\top \lambda_N \right) - 1 = 0. \tag{3}$$

This means that $\lambda_N = \Phi^{-\top} e$. Another part (with respect to. variable $\mathbf{x}$) of the necessary KKT optimality condition for (1) is

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) \lambda_N + \nabla_x \bar{h}(\mathbf{x}^*) \lambda = 0. \tag{4}$$

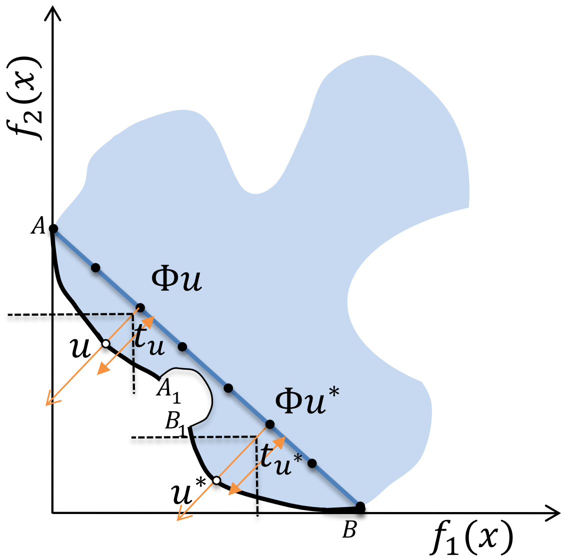

Fig. 1. Bulge knee for a nonconvex bijective optimization problem. The Pareto front is the union of curves AA_1 and B_1B; the line segment AB is the CHIM. The knee subproblem corresponding to Φp_{u^*} yields the bulge knee u^* as the maximal distance along the normal to the plane towards the Pareto front is t_{u^*}.

Dividing both sides of Equation 4 by the (non-zero) positive scalar $\sum_{i=1}^{m} \lambda_N$ we obtain

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*)^\top w + \nabla_x \bar{h}(\mathbf{x}) \frac{\lambda}{\sum_{i=1}^{m} \lambda_N} = 0, \tag{5}$$

where $w = \lambda_N = \Phi^{-\top} e$. This is exactly the necessary KKT optimality condition for the weighted sum problem (2) and hence the equivalence follows. □

The next results shows the invariance of the bulge knee with respect to any scaling of the objective functions. The proof technique is from [6] and is adapted to the inequality constraint $\Phi\beta + t\hat{n} \geq F(\mathbf{x})$ (instead of the original equality constraint used in[6]).

Theorem 2 (Invariance property) *The bulge knee is invariant to any positive scaling of the objective functions.*

Proof. Let the objective function be scaled by positive scalars s_i. The scaling changes $f_i(\mathbf{x})$ to $s_i f(\mathbf{x})$ for all indices i. The scaling changes the Φ matrix to $S\Phi$ and $\hat{n} = -\Phi e$ to $-S\Phi e$, where S is the diagonal matrix with entries s_i in the i-th diagonal place. The (only) constraint that is affected by scaling is $\Phi\beta + t\hat{n} \geq F(\mathbf{x})$ and this constraint changes to $s_i(\Phi\beta)_i + ts_i(\hat{n})_i \geq s_i f_i(\mathbf{x})$. Now as $s_i > 0$ the scaled constraint is the same as the original $(\Phi\beta)_i + t(\hat{n})_i \geq f_i(\mathbf{x})$, and thus the solution set also coincides. □

Remark 1. In many real world applications, the objectives usually have different magnitudes (stress versus displacement trade-off that is inherent in engineering design problems, for example); the invariance of the bulge knee as shown in Theorem 2 makes it suitable for solving such problems. Various other knee concepts, like the trade-off knee, bend knee, proper knee, are prone to different scaling of the objectives. The above result is algorithmically relevant, as normalization is quite often used in population based algorithms to calculate the distance (or volume) that are then used in the selection criteria.

The subproblem 1 in its original form uses the equality constraints $\Phi\beta + t\hat{n} = f(\mathbf{x})$. If this formulation is used, then an equivalence to guided domination could be shown. The guided domination approach works with a linearly transformed set of objectives that come from using a polyhedral domination cone. This domination cone could be represented as $D = \{d \in \mathbb{R}^m : Ad \geq 0\}$, where A is an $m \times m$ matrix.

Theorem 3 (Relationship with guided domination) *Let the matrix A be invertible and let the constraints $\Phi\beta + t\hat{n} = f(\mathbf{x})$ in subproblem 1 be changed to equality constraints. Then, the bulge knee is invariant to the polyhedral domination cone induced by the matrix A.*

Proof. The use of the polyhedral domination cone D can be incorporated by transforming the objectives from f to Af [9, lemma 2.3.4]. This changes (only) the constraint $\Phi\beta + t\hat{n} = f(\mathbf{x})$ to

$$A\Phi\beta + tA\hat{n} = Af(\mathbf{x}).$$

As the matrix A is invertible, multiplying both sides by of the above by A^{-1} yields the original constraint $\Phi\beta + t\hat{n} = f(\mathbf{x})$ and the result follows. □

3 Algorithms

In the following we will outline two algorithm for finding bulge knee points. The first algorithm can be used to approximate a solutions near the bulge knee, and additionally uses a crowding measure to generate a well-diverse set of solutions in the other part of the front. The second algorithm uses a complete ordering of solutions by computing their distance to the bulge knee to rank the solutions. The distance to the bulge knee is computed by first scaling the objectives and then computing the distance to the hyperplane. According to Theorem 2, the scaling of the objective could be done without changing the knee location.

Initially, a random population P_0 of size N is generated and a non-dominated sorting to P_0 is applied to identify different fronts: $\mathcal{F}_i \subseteq P_0$, $i = 1, 2, \ldots$, etc. A solution in $\mathcal{F}_i$ is given a rank i and using binary tournament selection, recombination and mutation operators an offspring population Q_0 size N is generated. At any generation t, let P_t and Q_t denote the parent and the offspring populations, respectively. Let R_t be the combined population, i.e., $R_t = P_t \cup Q_t$.

3.1 Bulge Approximation Evolutionary Algorithm (Baea)

Baea ranks the solutions in R_t as follows:

Step 1: Perform a non-dominated sorting to R_t to identify different fronts: $\mathcal{F}_i \subseteq R_t$, $i = 1, 2, \ldots,$ etc.

Step 2: Calculate the crowding distance (or some other diversity measure) of each solution in R_t.

Step 3: Using the points from $\mathcal{F}_1$ (the non-dominated set of R_t), compute the individual minimum of each objective f_i. Let $f^i \in \mathbb{R}^m$ be the point (in $\mathbb{R}^m$) at which f_i attains its minimum[1]. Based on this, let $\underline{f}, \overline{f} \in \mathbb{R}^m$ be such that

$$\underline{f}_i = \min\{f_i^1, f_i^2, \ldots, f_i^m\},\ \overline{f}_i = \max\{f_i^1, f_i^2, \ldots, f_i^m\} \quad \forall i = 1, 2, \ldots, m.$$

Step 4: For an arbitrary solution $u \in R_t$, let $d(u)$ be defined by

$$d(u) = \frac{-1 + \sum_{i=1}^{m} \left(\left(\frac{u_i - \underline{f}_i}{\overline{f}_i - \underline{f}_i} \right) \cdot \mathbf{1}_{(\overline{f}_i - \underline{f}_i \neq 0)} \right)}{\sqrt{m}}, \tag{6}$$

where $\mathbf{1}_{(\overline{f}_i - \underline{f}_i \neq 0)}$ equals one if $\overline{f}_i - \underline{f}_i \neq 0$ and zero otherwise.

Let $d_{\min}, d_{\max}$ denote the minimum and maximum values of all the $d(u)$'s in R_t. The normalized function $\bar{d}(u) \in (0, 1)$ is defined by

$$\bar{d}(u) := \frac{d(u) - d_{\min}}{d_{\max} - d_{\min} + 1}. \tag{7}$$

Step 5: Every solution $u \in \mathcal{F}_k$ has two attributes, its rank $\mathrm{r}(u) = k + \bar{d}(u)$ and its crowding distance $\mathrm{c}(u)$.

The distance to the hyperplane in Step 4 is directed and is measured away from the origin. Hence, smaller (e.g. negative) values of d are better. If the set of individual minima lie on plane parallel to the axes, then, for distance computation, we ignore those components where the minima and the maxima of the set of individual minima coincide.

Now, we have to create the new parent population P_{t+1} and the new offspring population Q_{t+1}. This is done as follows:

Step a: Set new population $P_{t+1} = \emptyset$. Set a counter i=1. Until $|P_{t+1}| + \mathcal{F}_i < N$, perform $P_{t+1} = P_{t+1} \cup \mathcal{F}_i$ and $i = i + 1$. Now, we need to include $N - |P_{t+1}|$ solution more into P_{t+1} from $\mathcal{F}_l$ (hence, the l-th front cannot be fully accommodated).

[1] It could happen that $f^i = f^j$ for some $i \neq j$, e.g., if there is just one point in $\mathcal{F}_1$. Additionally, if in $\mathcal{F}_1$ there are more than one minima of some f_i, then f^i is any one of them.

Step b: Let $\delta \in [0,1]$ be a user defined threshold parameter. Sort the solutions of $\mathcal{F}_l$ into increasing order of their ranks (or equivalently, increasing order of their $\bar{d}$ values). Take the first $\lceil \delta(N - |P_{t+1}|) \rceil$ solutions into P_{t+1} from the sorted list. Take the remaining solutions by considering the crowding distance values (for solutions in $\mathcal{F}_l$ but not in P_{t+1}).

Step c: Create offspring population Q_{t+1} using lexicographic tournament selection (winner based on lexicographic comparison of $(\mathrm{r}(u), -\mathrm{c}(u))$ tuple), recombination and mutation operators.

3.2 Bulge Domination Evolutionary Algorithm (Bdea)

The domination based algorithm to find bulge knee points uses a different domination to rank the solutions.

Definition 1 (Bulge-domination) *Let us assume that the individual minima $f^i \in \mathbb{R}^m$ for all i be supplied. Then, a point $\mathbf{v} \in \mathbb{R}^m$ is said to* bulge-dominate *another point $\mathbf{u} \in \mathbb{R}^m$ if $d(\mathbf{v}) < d(\mathbf{u})$ holds.*

The bulge domination is a total ordering. It can be shown that bulge dominance is Pareto dominance compliant, i.e., it does not contradict the order induced by the Pareto dominance. However, it requires the knowledge of the individual minima. One could first approximate the individual minima and then use this to rank the solutions by computing the distance to the hyperplane in which the individual minima lie, however the bulge knee found will be dependent on the goodness of the approximation of the individual minima. Bdea tries to approximate the individual minima and the bulge knee *simultaneously.*

Minimizing the distance to the hyperplane (recall that negative values mean that the point is away from the hyperplane towards the origin) would not emphasize the corner points (individual minima) that are needed to estimate the distance itself. Bdea tries to emphasize the corner points by carrying them to the next population directly, in addition to choosing points in the bulge knee region based on their d values.

In Bdea, the ranking, creation of the new parent population P_{t+1} and of the new offspring population Q_{t+1} is done as follows.

Step i: Find the non-dominated set of R_t. Let it be denoted by $\mathcal{F}_1$.

Step ii: Calculate $d(u)$ for every $u \in R_t$ using Step 3 and Step 4 of the ranking procedure of Baea.

Step iii: Create m-sorted lists, where the solutions are sorted with respect to each of the objectives in increasing order. Let $\mathcal{L}_i$ be the list corresponding to f_i.

Step iv: Let a user defined parameter $p \in (0,1)$ be given. Now, copy the first $\lceil \frac{Np}{m} \rceil$ solutions from each of the lists $\mathcal{L}_i$, to P_{t+1}.

Steps i-iv above cluster a fraction of solutions near the individual minima. The remaining solutions are filled in based on their d-values. This is detailed next.

Step v: Perform a bulge-non-dominated sorting (i.e., non-dominated sorting using bulge-domination instead of Pareto-domination) to R_t to identify different fronts: $\mathcal{F}_i^{\mathcal{B}} \subseteq R_t$, $i = 1, 2, \ldots$, etc.

Step vi: Calculate the crowding distance (or some other diversity measure) of each solution in R_t (using the fronts $\mathcal{F}_i^{\mathcal{B}}$).

Step vii: Set a counter $j = 1$. Until $|P_{t+1}| + \mathcal{F}_j^{\mathcal{B}} < N - m\lceil \frac{Np}{m} \rceil$, perform $P_{t+1} = P_{t+1} \cup \mathcal{F}_j^{\mathcal{B}}$ and $j = j + 1$.

Step viii: Include the most widely spread solutions to fill the remaining $N - |P_{t+1}|$ slots.

Step ix: Create offspring population Q_{t+1} using lexicographic tournament selection (winner based on lexicographic comparison of $(i, -\mathrm{c}(u))$ tuple, where $u \in \mathcal{F}_i^{\mathcal{B}}$), recombination and mutation operators.

The above algorithm has some similarities with the Pareto corner search algorithm [10]. The Pareto corner search algorithm tries to distribute solutions on the corner of the Pareto front; BDEA tries to approximate the corners of individual minima and the bulge knee point. The entire corner 'curves' are not needed for determining the hyperplane. They are more useful for determining the nadir point instead.

3.3 Complexity of One Generation of BAEA and BDEA

Step 1 of BAEA can be done in $\mathcal{O}(N \log^{m-1} N)$ operations using a divide-and-conquer based fast non-dominated sorting from [11] while Step 2 needs $\mathcal{O}(mN \log N)$ operations. Steps 3-5 and Steps a-c require $\mathcal{O}(Nm)$ and $\mathcal{O}(N \log N)$ operations, respectively. Hence, the overall complexity of BAEA is the maximum of $\mathcal{O}(N \log^{m-1} N)$ and $\mathcal{O}(mN \log N)$.

In the case of BDEA, we require to find the non-dominated front in Step i, this costs $\mathcal{O}(N \log^{m-2} N)$ operations [12]. It can also be that Steps ii-iv require $\mathcal{O}(mN \log N)$ operations. As the d values form a total ordering, Step v-vi can be done in $\mathcal{O}(mN \log N)$ time. Hence, the overall complexity of BAEA is either $\mathcal{O}(N \log^{m-2} N)$ or $\mathcal{O}(mN \log N)$ whichever number is larger.

For a fixed m, and an increasing number of points, it can be seen that BDEA is slightly faster than BAEA.

4 Simulation Results

The algorithms from Section 3 haven been empirically evaluated on a large number of problems. In this section we will present and discuss result on some of these. The results are after 15000 evaluations using SBX recombination and polynomial mutation operators with $\eta_c = \eta_m = 10$, and $N = 100$. Both BAEA and BDEA are implemented in the jMetal framework.

Figures 2 and 3 shows the obtained solutions by BAEA with different δ values on ZDT1 problem. For ZDT1, The distance of the solutions from the CHIM-hyperplane can be analytically obtained as $d(f_1, f_2) = \frac{f_1 - \sqrt{f_1}}{\sqrt{2}}$. Hence, the bulge

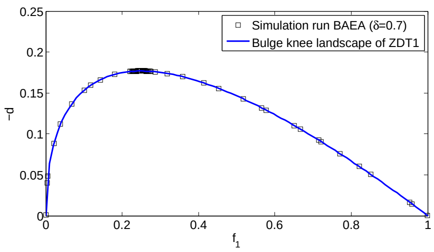

Fig. 2. Simulation results of BAEA with $\delta = 0.7$ on ZDT1. The bulge knee landscape is also shown.

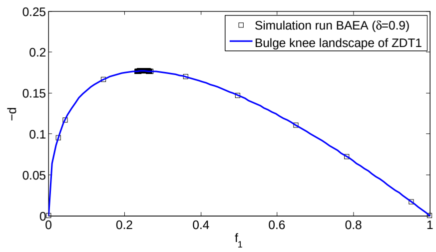

Fig. 3. Simulation results of BAEA with $\delta = 0.9$ on ZDT1. More solutions are concentrated near the global bulge.

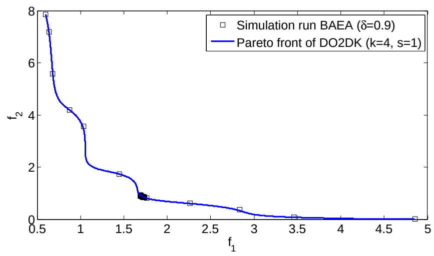

Fig. 4. Simulation results of BAEA with $\delta = 0.9$ on DO2DK ($k = 4, s = 1$).

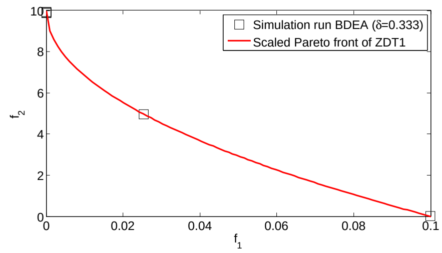

Fig. 5. Simulation results of BDEA with $\delta = \frac{1}{3}$ on a scaled version ZDT1.

knee corresponds to $x_1 = 0.25$. It is seen that the higher value of δ gives a more dense distribution of solutions near the bulge knee. As the value δ is not exactly one, we see that the extremes are also included (as these solutions are best with respect to diversity).

Similar results can also be seen on the knee test problem DO2DK (Figure 4). The bulge knee is found and a also a diverse set of other solutions. These other solutions need not be the local bulge knees and the focus of BAEA is only to find the global bulge knee. However, niching techniques can be used to find local bulge knees we well, if desired.

The invariance property shown in Theorem 2 is tested on a scaled ZDT1 problem where f_1 is scaled by 0.1 and f_2 scaled by 10. BAEA gives the exact location of the knee on this problem (corresponding to $x_1 = 0.25$). The value $\delta = \frac{1}{3}$ is used so that equal number solutions near the extreme of both objectives and solutions in the bulge knee are equally emphasized.

From simulations we obtained that just including the extremes is not enough, it is essential to have a certain fraction of solutions close to the extremes. This helps the adaptive CHIM to converge to the true CHIM, as the bulge in the Pareto-front depends on the current approximation of CHIM, and this in turn depends on how good the individual minima are approximated.

5 Conclusions and Future Works

The paper provided a theoretical and algorithmic characterization of the bulge knee on the Pareto front of a multi-objective problem. Theoretical results showed the relationship of the bulge knee to the weighted sum method, and showed the invariance of the bulge knee with respect to positive scaling of the objectives. Using these results, we proposed two new algorithms for approximating the bulge knee. The first algorithm aimed at approximating the bulge knee whereas the second algorithm used an adaptive approximation of the corner points to induce a total order, and then search for the bulge knee. Empirical results verified the theory and the two algorithms. We are currently exploring the use of niching techniques to extend the algorithm so as the local bulges as found as well.

References

1. Deb, K., Gupta, S.: Understanding knee points in bicriteria problems and their implications as preferred solution principles. Engineering Optimization **43**(11) (2011) 1175–1204
2. Rachmawati, L., Srinivasan, D.: Multiobjective evolutionary algorithm with controllable focus on the knees of the pareto front. Evolutionary Computation, IEEE Transactions on **13**(4) (aug. 2009) 810 –824
3. Branke, J., Deb, K., Dierolf, H., Osswald, M.: Finding knees in multi-objective optimization. In: In PPSN VIII. LNCS, Springer (2004) 722–731
4. Shukla, P.K., Braun, M.A., Schmeck, H.: Theory and algorithms for finding knees. In: Evolutionary Multi-Criterion Optimization, Springer (2013) 156–170
5. Bechikh, S., Ben Said, L., Ghédira, K.: Searching for knee regions in multi-objective optimization using mobile reference points. In: Proceedings of the 2010 ACM Symposium on Applied Computing. SAC '10, New York, NY, USA, ACM (2010) 1118–1125
6. Das, I.: On characterizing the "knee" of the pareto curve based on normal-boundary intersection. Structural Optimization **18**(2-3) (1999) 107–115
7. Schtze, O., Laumanns, M., Coello Coello, C.: Approximating the knee of an mop with stochastic search algorithms. In Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N., eds.: PPSN X. Volume 5199 of LNCS. Springer (2008) 795–804
8. Das, I., Dennis, J.E.: Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. SIAM Journal on Optimization **8**(3) (1998) 631–657
9. Sawaragi, Y., Nakayama, H., Tanino, T.: Theory of multiobjective optimization. Volume 176 of Mathematics in Science and Engineering. Academic Press Inc., Orlando, FL (1985)
10. Singh, H., Isaacs, A., Ray, T.: A pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems. Evolutionary Computation, IEEE Transactions on **15**(4) (Aug 2011) 539–556
11. Jensen, M.: Reducing the run-time complexity of multiobjective eas: The nsga-ii and other algorithms. Evolutionary Computation, IEEE Transactions on **7**(5) (oct. 2003) 503 – 515
12. Kung, H.T., Luccio, F., Preparata, F.P.: On finding the maxima of a set of vectors. J. ACM **22**(4) (1975) 469–476