Generating Biobjective Mixed Integer Programming Instances

We provide a method for generating BOMIPs for which the nondominated frontier is known. We present two classes of instances. The first class of instances are BOMIPs whose frontiers are characterized by a set of randomly generated, isolated NDPs. The second class of instances are characterized by a set of deterministically generated NDPs, each the apex of a cone of randomized width. Both classes have complex, intersecting frontiers.

Both classes of instances have the same fundamental structure. Each objective function is equal to one of the two continuous decision variables, i.e. $z_1(x) := x_1$ and $z_2(x) := x_2$, where $x \in \mathbb{R}^2$. Furthermore, a line segment L in the criterion space will have some nondominated portions. Specifically, $L = \{(x_1, x_2) \in \mathbb{R}^2 : x_1 + x_2 = 0, -k \leq x_i \leq k \text{ for } i = 1, 2\}$ where $k \in (0, \infty)$ is a parameter. Portions of the line segment will be dominated by a set of generated NDPs. If n NDPs are generated, then the BOMIP instance will have $n+1$ binary variables, i.e., $y \in \{0, 1\}^{n+1}$. Therefore, the feasible set of the BOMIP instance is $\mathbb{X} \subseteq \mathbb{R}^2 \times \{0, 1\}^{n+1}$.

Next, we discuss how to generate a set of n NDPs, $\{(a_i, b_i)\}_{i=1,2,...,n}$, where each dominates a portion of L. We note two properties that these NDPs must satisfy: (1) if (a_i, b_i) dominates any portion of L, then we must have $a_i + b_i < 0$ where $a_i, b_i \in [-k, k]$ for all i, and (2) if (a_i, b_i) and (a_j, b_j) are two distinct NDPs, then $a_i < a_j$ if and only if $b_i > b_j$. We use $U(x, y)$ to indicate a uniformly distributed random variable bounded within (x, y).

1 Fixed Cone-Width Instances

While the first class of instances does incorporate some randomness, their frontier is completely known. This is achieved by choosing the n NDPs in such a way that they are each equidistant from L and that two neighboring NDPs do not dominate the same portion of L. The former is easily accomplished by defining a line segment parallel to L but shifted vertically down by $d \in (0, \frac{2k}{n})$ units as $L_d = \{(x_1, x_2) : x_1 + x_2 = -d, -k \leq x_i \leq k - d \text{ for } i = 1, 2\}$. Then any point on L_d dominates a segment of L and has the same distance to L, so we can simply choose n points of L_d.

In order to ensure that any portion of L is not dominated simultaneously by two neighboring NDPs (see Figure 1), we must choose the NDPs such that $-b_i < a_{i+1}$ for $i = 1, 2, ..., n-1$. This result comes from considering the points (a_i, b_i) and (a_{i+1}, b_{i+1}) projected onto the line L horizontally and vertically, respectively. By avoiding such overlapping, choosing (a_1, b_1) such that it does not dominate the left endpoint of L, and similarly choosing (a_n, b_n) such that it does not dominate the right endpoint of L, this procedure will guarantee exactly $n + 1$ nondominated segments of L.

To prevent numerical issues, we use n equal-sized subintervals within $(-k, k)$ when choosing the NDPs to prevent dense clustering of the points in any one region. We also guarantee that the nondominated portions of L are no shorter in length than ϵ by requiring $a_{i+1} > -b_i + \epsilon$. The resulting procedure is given in Algorithm 1.

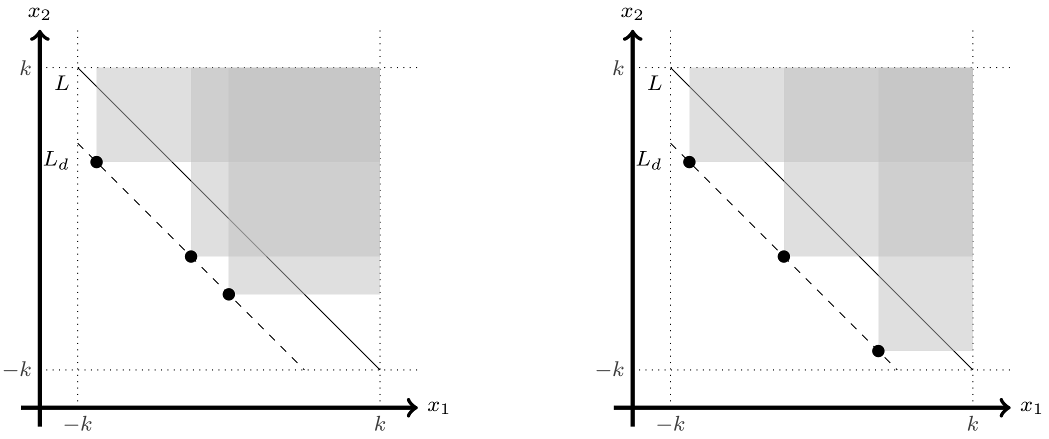

Figure 1: NDPs and associated cones generated by the fixed cone-width class of instances. (Left) If the pointed cones from NDPs overlap on L, then there are fewer than $n+1$ nondominated segments of L. (Right) Choosing the NDPs such that $-b_i < a_{i+1}$ prevents such overlapping, thus guaranteeing exactly $n+1$ line segments of L are nondominated.

Algorithm 1 Fixed Cone-Width NDP Generation

```
$d = k/4$ (distance by which $L_d$ is shifted down)
$w = (2k - d)/n$ (the width of subintervals)
$a_1 = U(-k, -k + w)$
$b_1 = -a_1 - d$
for $i = 2, 3, ..., n$ do
    $a_i = U(\max\{-k + (i-1)w, -b_{i-1} + \epsilon\}, -k + iw)$
    $b_i = -a_i - d$
end for
```

For constructing the associated BOMIP, assume that the line L, NDPs $\{(a_i, b_i)\}_{i=1..n}$, and parameters k, n are known. With $\vec{z}((x, y)) = (x_1, x_2)$ as our objective vector, we set the uniform bound $-k \leq x_i \leq k$ for $i = 1, 2$. Then we have the following polytopes of interest in the criterion space:

- The line segment L is the nondominated frontier of the polytope

$$P_0 = \{(x_1, x_2) : x_1 + x_2 \geq 0, -k \leq x_i \leq k, i = 1, 2\}.$$

- For $i = 1, 2, ..., n$, the *orthogonal* pointed cone[1] dominated by its vertex NDP (a_i, b_i) is defined by

$$P_i = \{(x_1, x_2) : \ x_1 \geq a_i, \ x_2 \geq b_i\}.$$

To have the exact nondominated frontier we desire, we require any NDP (x_1^*, x_2^*) to belong to at least one of these polytopes (since the polytopes intersect, it may belong to more than one). Let

[1] The fact that we use the orthogonal pointed cone (with radius of 90 degrees) for *all* NDPs led to the name of "fixed cone-width."

$y \in \{0,1\}^{n+1}$ be a binary decision vector, where $y_0 = 1$ implies $(x_1^*, x_2^*) \in L$ and for $i = 1, 2, ..., n$, $y_i = 1$ indicates that $(x_1^*, x_2^*) \in P_i$ for $i = 0, 1, ..., n$. We also require $\sum_{i=0}^{n} y_i = 1$. We therefore define an MILP with constraints that enforce such implications (note we do not enforce that $y_i = 0$ implies $(x_1^*, x_2^*) \notin P_i$). The associated MILP for the first class of instances is thus defined as:

$$\begin{aligned}
\text{minimize} \quad & (x_1, x_2) \\
\text{s.t.} \quad & x_1 + x_2 \geq -2k(1 - y_0) && (1) \\
& x_1 \geq a_i - 2k(1 - y_i) \quad \forall i = 1, 2, ..., n && (2) \\
& x_2 \geq b_i - 2k(1 - y_i) \quad \forall i = 1, 2, ..., n && (3) \\
& \sum_{i=0}^{n} y_i = 1 && (4) \\
& -k \leq x_i \leq k \qquad \forall i = 1, 2 && (5) \\
& y \in \{0,1\}^{n+1}.
\end{aligned}$$

Observe that when $y_0 = 1$, we have that (1) and (5) together characterize the polytope P_0 containing the line segment L. Similarly, when $y_i = 1$ for $i = 1, ..., n$, (2) and (3) together characterize the pointed cone polytope P_i containing NDP (a_i, b_i). Otherwise, when $y_i = 0$ for $i = 0, 1, ..., n$, the associated constraints from (1), (2), and (3) including y_i become redundant.

By construction, we are guaranteed to have $n + 1$ distinct binary vectors $(e_1, e_2, ..., e_{n+1})$ that each belong to at least one efficient solution in the efficient frontier. The nondominated frontier associated with this BOMIP has exactly n isolated NDPs and $n + 1$ line segments (2 half-open at the ends of L and $n - 1$ open segments in between isolated NDPs).

Note: Even large instances of this class of BOMIPs are solved quite quickly by most methods, so they are not ideal for testing computational efficiency. However, since the nondominated frontiers are known exactly, they can be used for validating the correctness of an algorithm.

2 Randomized Cone-Width Instances

The second class of instances randomizes the pointed cones associated with each NDP rather than only using orthogonal pointed cones. Similar to the first class of instances, we restrict the cones of the NDPs from overlapping the same portion of L, as in Figure 2, which requires a careful choice of the NDPs. We postpone a discussion of how to choose the NDPs until after introducing the randomizing of cone widths.

Let (a, b) be a point that dominates line L. Note that the simple orthogonal pointed cone with (a, b) as its vertex is given by $\{(x_1, x_2) \in \mathbb{R}^2 : x_1 \geq a, x_2 \geq b\}$, as before. We will generalize these inequalities, as shown in Figure 3. To define any pointed cone with the vertex (a, b), requires two inequalities of the form

$$\theta_1 x_1 + (1 - \theta_1) x_2 \geq \theta_1 a + (1 - \theta_1) b \qquad (6)$$

$$\theta_2 x_1 + (1 - \theta_2) x_2 \geq \theta_2 a + (1 - \theta_2) b. \qquad (7)$$

The cones must be derived from $\theta_1 \in [0, \frac{1}{2})$ and $\theta_2 \in (\frac{1}{2}, 1]$ based on the following observations. First, a line defined by the inequalities held at strict equality – that is neither horizontal ($\theta_i = 0$) nor vertical ($\theta_i = 1$) – must both have negative slope, i.e. $-\frac{\theta_i}{1-\theta_i} < 0$, so θ_i must be within $(0, 1)$.

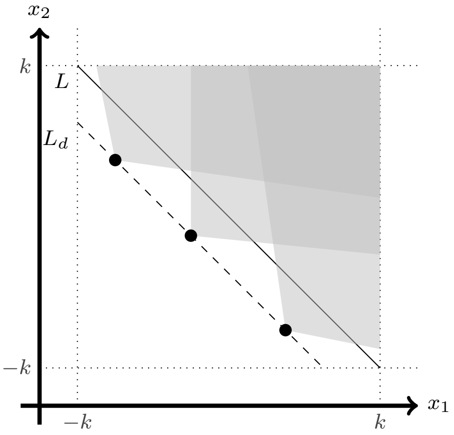

Figure 2: NDPs and associated cones generated by the randomized cone-width class of instances.

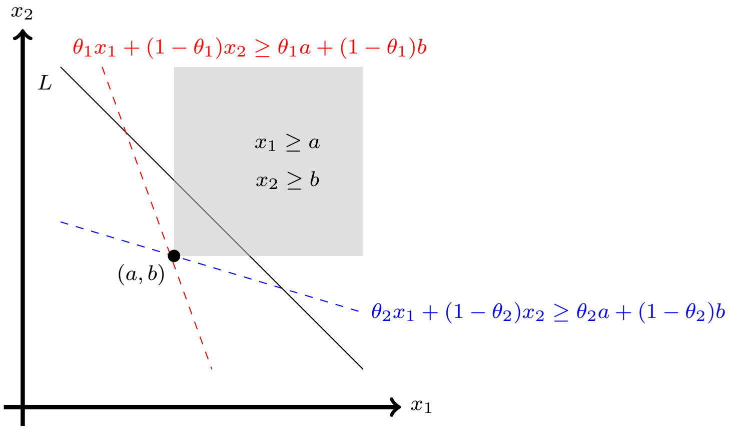
x_2
$\theta_1 x_1 + (1-\theta_1)x_2 \geq \theta_1 a + (1-\theta_1)b$
L
$x_1 \geq a$
$x_2 \geq b$
(a, b)
$\theta_2 x_1 + (1-\theta_2)x_2 \geq \theta_2 a + (1-\theta_2)b$
x_1

Figure 3: The orthogonal pointed cone with vertex (a, b) is shaded. The generalized pointed cone is defined between the red and blue inequalities. We randomize the cone widths by choosing θ_1 and θ_2.

Second, neither line may be parallel to L, so $-\frac{\theta_i}{1-\theta_i} \neq -1$ and so $\theta_i \neq \frac{1}{2}$. Lastly, in order for both edges of the pointed cone to contribute to the nondominated frontier, we require $\theta_1 < \frac{1}{2}$ and $\theta_2 > \frac{1}{2}$.

The orthogonal cone given by $x_1 \geq a$ and $x_2 \geq b$ results from substituting $\theta_1 = 1$ and $\theta_2 = 0$ into (6) and(7), respectively. Having some orthogonal cones is desired in order to induce some isolated points (and open endpoints) into the nondominated frontier. Therefore, we define a parameter $\pi \in [0, 1]$ representing the probability that a cone is orthogonal. In practice, we use $\pi = 0.05$. Algorithm 2 shows how we generate a list of ordered pairs of thetas.

Algorithm 2 Theta Generation

```
thetalist= $\emptyset$
for $i = 1, 2, ..., n$ do
    if $U(0, 1) \leq \pi$ then
        $\theta_1 = 0$
        $\theta_2 = 1$
    else
        $\theta_1 = U(0, \frac{1}{4})$
        $\theta_2 = U(\frac{3}{4}, 1)$
    end if
    thetalist.append($(\theta_1, \theta_2)$)
end for
```

We tighten the bounds on θ_1, θ_2 to yield smaller-width pointed cones, namely $\theta_1 \in [0, \frac{1}{4})$ and $\theta_2 \in (\frac{3}{4}, 1]$. Note that these bounds on the slopes of the linear inequalities allow us to compute the minimum distance between two NDPs required to ensure that their cones do not overlap on the line segment L. By a geometric argument, we have that if all NDPs are chosen from a line segment shifted down d from L, i.e. L_d, then any two neighboring NDPs should be at least distance $2d$ from each other in both coordinates. To ensure that the nondominated portions of L have length of at least ϵ, we force NDPs to be at least $2d + 1$ apart. If we have $n + 1$ of such intervals (one for each NDP and one additional to allow for the endpoints of L to be nondominated), then we can compute the appropriate d in order to fit n NDPs within our bounds, $x_i \in [-k, k]$ for $i = 1, 2$:

$$(n + 1)(2d + 1) = 2k \quad \Leftrightarrow \quad d = \tfrac{k}{n+1} - \tfrac{1}{2}.$$

This results in Algorithm 3 for generating the NDPs for this class of instances.

Algorithm 3 Randomized Cone-Width NDP Generation

```
$d = k/(n + 1) - 0.5$
$a_1 = -k + 0.5d$
$b_1 = -a_1 - d$
for $i = 2, ..., n$ do
    $a_i = a_{i-1} + 2d + 1$
    $b_i = -a_i - d$
end for
```

The BOMILP for the second class of instances is defined as:

$$\text{minimize} \qquad (x_1, x_2)$$

$$\text{s.t.} \qquad x_1 + x_2 \geq -2k(1 - y_0) \tag{8}$$

$$\theta_1^i x_1 + (1 - \theta_1^i)x_2 \geq \theta_1^i a_i + (1 - \theta_1^i)b_i - 2k(1 - y_i) \quad \forall i = 1, 2, ..., n \tag{9}$$

$$\theta_2^i x_1 + (1 - \theta_2^i)x_2 \geq \theta_2^i a_i + (1 - \theta_2^i)b_i - 2k(1 - y_i) \quad \forall i = 1, 2, ..., n \tag{10}$$

$$\sum_{i=0}^{n} y_i = 1 \tag{11}$$

$$-k \leq x_i \leq k \qquad \forall i = 1, 2 \tag{12}$$

$$z \in \{0, 1\}^{n+1}$$

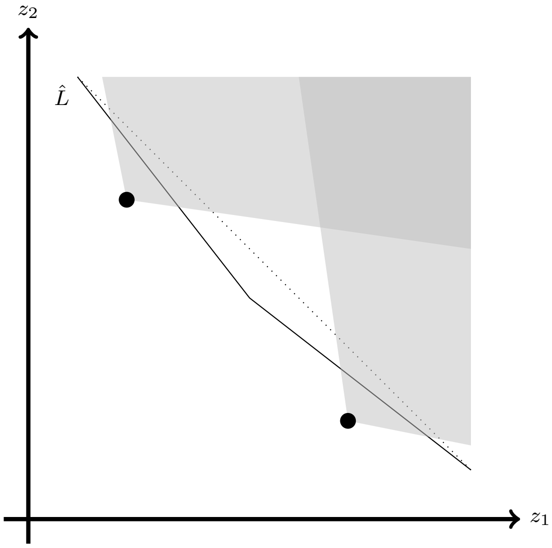

Figure 4: A Bent Randomized Cone-Width instance for $n = 2$. The bent line segment $\hat{L}$ comprises two line segments of a very wide cone. Solving a scalarized IP with respect to the gradient of either piece has at most one optimal solution but (for large n) multiple near-optimal solutions.

By construction, we will have no more than $n + 1$ distinct binary vectors that each belong to at least one efficient solution in the efficient frontier. The nondominated frontiers associated with this BOMIP will have no more than $3n+1$ line segments ($n+1$ from L and at most 2 per NDP), including open endpoints induced by any orthogonal cones and closed endpoints induced by intersecting slices.

Note that the BOMIP for both classes of instances has $n + 3$ variables and $4 + 2n$ constraints, so the size of the problem is linear with respect to n.

Note: The second class of instances, compared to the first class, is more difficult to solve because of the high frequency of intersecting integer slices in the frontier.

3 Bent Randomized Cone-Width Instances

As the name suggests, the bent randomized cone-width instances represent a variation of the randomized cone-width instance. It employs the same method for generating the n NDPs (at uniform distance from L) and their cone widths. However, the straight line segment L is replaced by a slightly bent line segment, $\hat{L}$, i.e. the boundary of a cone that is very wide (see Figure 4).

Now, when solving a scalarized IP with respect to the gradient of either part of the bent line segment, there are no longer multiple optimal solutions, but there are many *near*-optimal solutions.

The "bent" line segment has corner point $(a_0, b_0) = (-d/4, -d/4)$, which may be dominated or nondominated, $\theta_1^0 = (k + d/4)/(2k)$, and $\theta_2^0 = (k - d/4)/(2k)$. Then the bent line segment $\hat{L}$ fits the previous definition for a cone while substituting the corner point (a_0, b_0), θ_1^0, and θ_2^0. Therefore, in the BOMIP for the randomized cone-width instances, we replace constraint (8) with two constraints of the form (9) and (10). Let the generated NDPs be $\{(a_i, b_i)\}_{i=0,1,\ldots,n}$ and their cones be defined by $\{(\theta_1^i, \theta_2^i)\}_{i=0,1,\ldots,n}$. We then have the following BOMIP for the bent randomized cone-width instance:

$$
\begin{aligned}
\text{minimize} \quad & (x_1, x_2) \\
\text{s.t.} \quad & \theta_1^i x_1 + (1-\theta_1^i) x_2 \geq \theta_1^i a_i + (1-\theta_1^i) b_i - 2k(1-y_i) \quad \forall i = 0,1,2,...,n && (13) \\
& \theta_2^i x_1 + (1-\theta_2^i) x_2 \geq \theta_2^i a_i + (1-\theta_2^i) b_i - 2k(1-y_i) \quad \forall i = 0,1,2,...,n && (14) \\
& \sum_{i=0}^{n} y_i = 1 && (15) \\
& -k \leq x_i \leq k \qquad \forall i = 1,2 && (16) \\
& y \in \{0,1\}^{n+1}.
\end{aligned}
$$