A multi-objective strategy for Pareto set refinement

Why refining the Pareto set?
Why refining the Pareto set?

- custom sampling of Pareto set
  - uniform
  - by curvature
  - refine where you desire to refine

- good sampling of Pareto set
  - avoid repetition of samples
  - promote enough distance between samples

\[ f_1(x) \quad f_2(x) \]

\[ f_1(x) \quad f_2(x) \]

\[ f_1(x) \quad f_2(x) \]
A supporting theorem for refinement
Quasi-convex function

- sublevel sets are convex
Quasi-convex function

\[ f(c) < \max\{f(a), f(b)\}, \forall c \in (a, b) \]
Theorem: refinement for quasi-convex functions

**Theorem (quasi-convex Pareto)**

For any two strictly quasiconvex functions $f_1$ and $f_2$ and for any two Pareto optimal points $a$ and $b$ where $f_1(a) < f_1(b)$, any $c = a + \lambda(b - a)$, $\lambda \in (0, 1)$, is dominated only by Pareto optimal points $d$ where $f_1(d) \in (f_1(a), f_1(b))$ and $f_2(d) \in (f_2(b), f_2(a))$. 
Theorem: refinement for quasi-convex functions

Proof.

Because $f_1$ and $f_2$ are strictly quasiconvex functions, any point $c = a + \lambda(b - a)$, $\lambda \in (0, 1)$, must lie in the cone $\{x \mid f_1(x) < f_1(b)\} \cap \{x \mid f_2(x) < f_2(a)\}$ since $f_1(a) < f_1(b)$ and $f_2(b) < f_2(a)$. Because $a$ and $b$ are Pareto optimal points, no feasible point can lie in $\{x \mid f_1(x) \leq f_1(a), f_2(x) \leq f_2(a), f(x) \neq f(a)\} \cup \{x \mid f_1(x) \leq f_1(b), f_2(x) \leq f_2(b), f(x) \neq f(b)\}$. Hence, the remaining space for points dominating $c$ to lie is $f_1(x) \in (f_1(a), f_1(b))$ and $f_2(x) \in (f_2(b), f_2(a))$, including Pareto optimal points $x = d$ because $f(d) \leq f(c)$, $f(d) \neq f(c)$. \qed
Geometrical interpretation of theorem
Refinement algorithm
Refinement algorithm

1. find optimal Pareto vertices
2. loop
   1. normalize objective functions
   2. elect a segment to refine
   3. solve with a dominating optimization algorithm

- $f_1(x)$
- $f_2(x)$
Performance in practice
Performance in practice

- two quadratic functions
- one stretched, displaced and rotated
Why refining the Pareto set? A supporting theorem for refinement Refinement algorithm Performance in practice Concluding remarks

Performance in practice

- epsilon constrained (left)
- Pareto refinement (right)
Performance in practice

- two quasiconvex functions
- one stretched, displaced and rotated

![Graph showing two quasiconvex functions with one stretched, displaced and rotated](chart.png)
Performance in practice

- epsilon constrained (left)
- Pareto refinement (right)
Performance in practice

- refinement is about 10% faster epsilon constrained

![Graph showing comparison between epsilon-constrained refinement and refinement. The x-axis represents the dimension, n, and the y-axis represents the number of problem evaluations, k. The graph plot shows a positive trend with both methods increasing as the dimension increases, with refinement being faster.]
Performance in practice

- test on seasonalizing of energy problem
- income versus risk
- formulation

\[
\begin{align*}
\text{minimize} \quad & f_1(x, x_0) = - \text{mean}_j \sum_{i=1}^{n} x_i (p_{i,j} \eta_{i,j} + s_i) + x_0 \delta_i r_{i,j} [b + p_{i,j} (\eta_{i,j} - 1)] \\
& f_2(x, x_0) = - \text{mean}_{j \in \mathcal{W}} \sum_{i=1}^{n} x_i (p_{i,j} \eta_{i,j} + s_i) + x_0 \delta_i r_{i,j} [b + p_{i,j} (\eta_{i,j} - 1)] \\
\text{subject to} \quad & \sum_{i=1}^{n} x_i \leq G \\
& x \geq 0
\end{align*}
\]

where

\[
\eta_{i,j} = \min \left\{ 1, \frac{h_{i,j}}{x_i + x_0 \delta_i + g_i} \right\}
\]
Performance in practice
Performance in practice

![Graph showing income mean, E - ePaR (millions R$) vs income mean, E (millions R$)]
Concluding remarks
Concluding remarks

- the refinement works for many variables
- the refinement typically works faster than scalarization methods
- the refinement generalizes for more than 2 objective functions
Concluding remarks

- presentation available at www.enacom.com.br

Thanks