Discrete search in design optimization

Martin Fuchs

CERFACS, Toulouse, France

in collaboration with Arnold Neumaier, University of Vienna, Austria

October 29, 2010

Design optimization

- black box cost function

  $\longrightarrow$ simulation based

  $\longrightarrow$ computationally expensive evaluation

  $\longrightarrow$ strong nonlinearities, discontinuities

  $\longrightarrow$ hidden constraints

- various design choices

  $\longrightarrow$ continuous, discrete, and categorical choices

  $\longrightarrow$ mixed variables

Problem formulation

$$
\begin{aligned}
\min_{\theta, z} \quad & F(z) \\
\text{s.t.} \quad & z = Z(\theta), \\
& \theta \in \mathbf{T}.
\end{aligned}
$$

- $F : \mathbb{R}^{n_z} \longrightarrow \mathbb{R}$ black box
- $\mathbf{T}$ is the domain of possible design choices θ
- Z is the design selection mapping

Problem generalization

$$\begin{aligned} \min_{\theta,x} \ & c^T x \\ \text{s.t.} \ & F(Z(\theta)) \le Ax, \\ & \theta \in \mathbf{T}. \end{aligned}$$

- special case $c = A = 1$ gives former formulation
- black box input z substituted by $z = Z(\theta)$

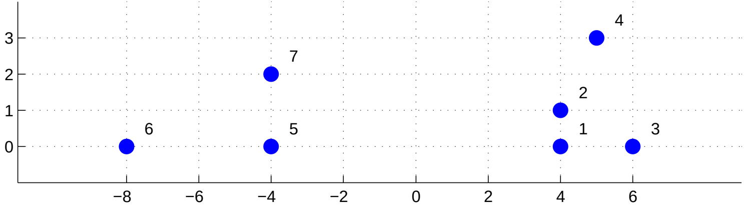
Design selection example: discrete case

θ	**Thruster**	F/N	I_{sp}/s	m_{thrust}/kg
1	Aerojet MR-111C	0.27	210.0	200
2	EADS CHT 0.5	0.50	227.3	200
3	MBB Erno CHT 0.5	0.75	227.0	190
4	TRW MRE 0.1	0.80	216.0	500
5	Kaiser-Marquardt KMHS Model 10	1.0	226.0	330

- contains specifications of design components and the associated choice variable θ
- the table mapping $Z : \theta \rightarrow (F, I_{sp}, m_{thrust})$ assigns an input parameter vector to a given design point θ
- in general F is a physical model based on $Z(\mathbf{T})$, rather than on $\mathbf{T}$

Search space $\mathbf{T}$

- $\mathbf{T}$ is the set of all possible designs
- $\mathbf{T} = T^1 \times T^2 \times \cdots \times T^{n_0}$
- $T^i = \begin{cases} \{1, 2, \ldots, N_i\} & \text{in the discrete case,} \\ [\underline{\theta^i}, \overline{\theta^i}] & \text{in the continuous case.} \end{cases}$
- for discrete T^i the design selection mapping provides a finite multidimensional set $Z^i(T^i)$, e.g.,

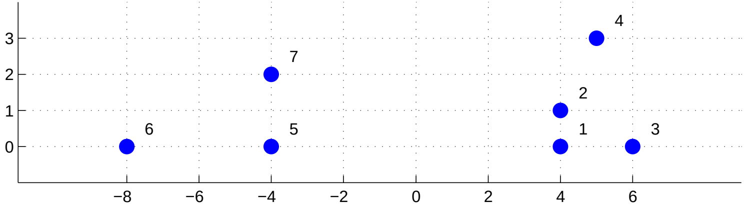


Heuristic approaches

- SNOBFIT (fits a quadratic model of the objective function and minimizes this model)
- Evolutionary algorithms
- Separable underestimation
- **Splitting based on convex relaxation**
- combination with methods for continuous variables

Convex relaxation based splitting: idea

- remember: F is based on $Z(\mathbf{T})$ rather than on $\mathbf{T}$



- for discrete T^i relaxation to the convex hull of $Z^i(T^i)$

Convex relaxation of $Z(\mathrm{T})$

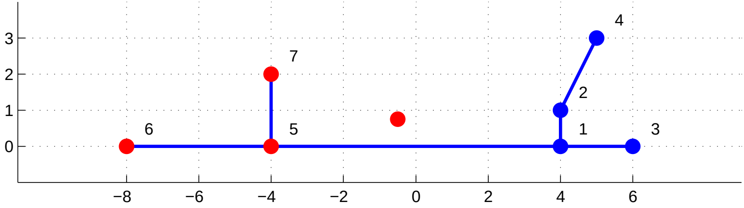
$$
\begin{aligned}
\min_{z,v,\lambda}\quad & c^T x \\
\text{s.t.}\quad & F(z) \le Ax, \\
& \left.
\begin{aligned}
& z = (v^1, \dots, v^{n_0}), \\
& v^i = \sum_{j=1}^{N_i} \lambda_j^i Z^i(j) \text{ for } i \in I_d, \\
& \sum_{j=1}^{N_i} \lambda_j^i = 1 \text{ for } i \in I_d, \\
& \lambda_j^i \ge 0 \text{ for } i \in I_d, 1 \le j \le N_i,
\end{aligned}
\right\} \text{convex combination} \\
& v^i \in [\underline{\theta^i}, \overline{\theta^i}] \text{ for } i \in I_c.
\end{aligned}
$$

Approximate linear solution

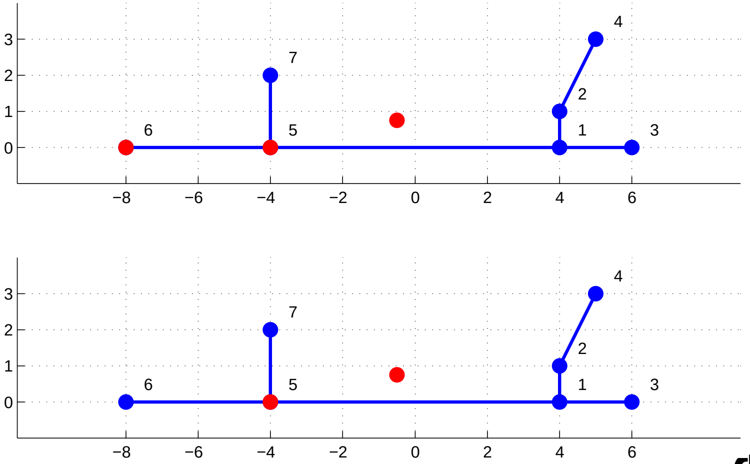
$$
\begin{aligned}
\min_{z,x,\mu,v,\lambda} \quad & c^T x + \varepsilon \|\mu\|_p \\
\text{s.t.} \quad & \sum_{j=1}^{N_0} \mu_j F_j \leq Ax, \\
& z = \sum_{j=1}^{N_0} \mu_j z_j, \\
& \sum_{j=1}^{N_0} \mu_j = 1, \\
& z = (v^1, \ldots, v^{n_0}), \\
& v^i = \sum_{j=1}^{N_i} \lambda_j^i Z^i(j) \text{ for } i \in I_d, \\
& \sum_{j=1}^{N_i} \lambda_j^i = 1 \text{ for } i \in I_d, \\
& \lambda_j^i \geq 0 \text{ for } i \in I_d, 1 \leq j \leq N_i, \\
& v^i \in [\underline{\theta^i}, \overline{\theta^i}] \text{ for } i \in I_c.
\end{aligned}
$$

Splitting

- use the coefficients of the convex relaxation as weights on the minimum spanning tree (MST) of $Z(\mathbf{T})$
- split the MST in two of parts of similar total weight
- Example:



Splitting ctd.

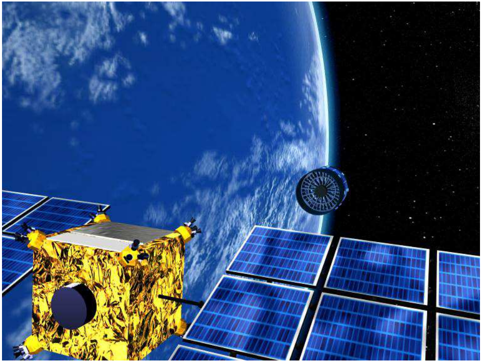


Solver strategy

1. Find a relaxed approximate solution on the current branch.
2. Round the relaxed solution $\widehat{z}$ to the next feasible point, i.e., $\widehat{z}_{\mathrm{round}} := \arg\min_{\{z \in Z(\mathbf{T})\}} \|z - \widehat{z}\|_2$.
3. Start neighborhood search from $\widehat{z}_{\mathrm{round}}$.
4. Split on the variable with maximal deviation during Step 3.
5. Select the branch with the best function value in Step 3 for the next iteration.

1. Introduction
2. Discrete search space
3. Design optimization
4. Real-life application
5. Summary

XEUS mission – permanent space-borne X-ray observatory



- complex design problem, 10 dimensions
- $4 \times 14 \times 6 \times 8 \times 5 \times 20 \times 9 \times 44 \times 30 \geq 3 \cdot 10^9$ discrete choices
- 1 continuous choice variable

Optimization results

- total mass $m = 1566$ kg
- found in 4 out of 5 runs of 2500 function evaluations each
- 1 run failed because we found no feasible starting point
- previous study used ≥ 50000 function calls

Summary

- Exploit structural knowledge about the discrete search space.
- Speed up the splitting procedure in branching algorithms.
- Solve successfully higher dimensional real-life design optimization problems.

Visit my website: `http://www.martin-fuchs.net`