A1.1 - Computational Design with Combinatorial Optimization

Jaan Tollander de Balsch, 452056

November 5, 2018

The Problem

- E denotes the set of elements $\{e_1, ..., e_n\}$ in the menu.
- r is a positive real number denoting the time to read an element between top of menu and target.
- c is an integer denoting the desired number of columns in the grid.
- $d : E \times E \to \mathbb{R}^+$ is a distance function that measures the distance between two elements in the grid layout.
- $w : E \to \mathbb{R}^+$ denotes the weight function which
- $a : E \times E \to \mathbb{R}^+$ association function
- x denotes a design candidate in the design space Ω, i.e. a sequence that is a permutation of the elements of set E.

Linear selection time (`Linear_ST`)

$$f_1(x) = \sum_{e \in x} w(e) \cdot d(e_1, e) \cdot r \qquad \text{(selection-time-objective)}$$

Associations between elements (`myObjective`)

$$f_2(x) = \sum_{(e_i, e_j) \in x \times x} d(e_i, e_j) \cdot a(e_i, e_j) \qquad \text{(association-objective)}$$

The combinatorial design problem aims to find the design that minimized the objective function.

A1.1 Multi-objective Optimization

1. Weighted sum multi-objective criterion

If the objective function as

$$f_1(x) + w_A f_2(x), \qquad \text{(objective)}$$

where w_A is positive real number denoting the weight of the association objective, then increasing the weight w_A increases the importance of the associations between the elements, i.e. solutions in which the distance between the associated element is smaller will be favored more. Vice versa, by decreasing the value of w_A, the importance of the associations between elements will decrease.

2. Normalization

The problem with defining the objective function like (objective), is that if the weights w are scaled by multiplying by a constant (not zero), the ratios between the weight do not change and therefore the optimal solution for only optimizing for linear selection time would not change. However, it would diminish the effect of the weight w_A and therefore the objective on the associations between the elements. Also, the weight for function f_1 is set 1 and cannot be tuned by the designer. In order to combat this the objective function can be normalized.

Normalization can be implemented as shown by (Mausser, 2006)

$$\sum_{i=1}^{k} u_i \theta_i f_i(x), \qquad \text{(normalized-objective)}$$

where u_i are the weights for which $\sum_{i=1}^{k} u_i = 1$ and $u_i \geq 0$. The normalization factors are defined

$$\theta_i = \frac{1}{z_i^N - z_i^U}, \qquad (1)$$

where the coefficients are defined

$$x^{[i]} = \operatorname{argmin}\{f_i(x) : x \in \Omega\} z_i^U = f_i(x^{[i]}) z_i^N = \max_{1 \leq j \leq k}(f_i(x^{[j]})).$$

In our case $k = 2$.

3. Pareto Frontier

Using (normalized-objective) the Pareto optimal designs can be obtained by solving the optimization problem for different values of the weights u_i. (Mausser, 2006)

TODO: Plot the values of f_1 and f_2 as a function of the weights u_i.

- TODO: track independent values of f_1 and f_2
- List 4 pareto optimal designs

References

Mausser, H., 2006. Normalization and Other Topics in Multi-Objective Optimization. *Proceedings of the Fields–MITACS Industrial Problems Workshop*, 2, pp.89–101.