Mixed-Integer Nonlinear Optimisation: Branching

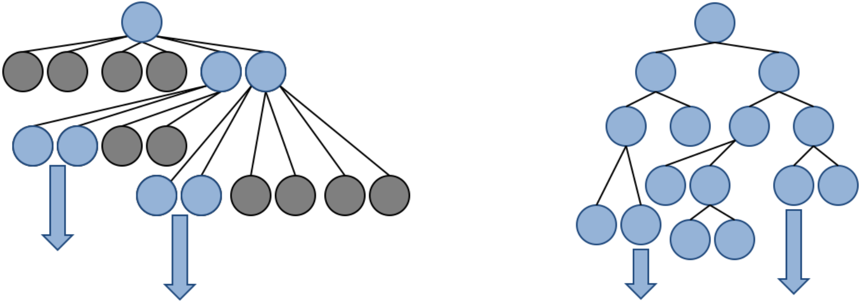
Ruth Misener
r.misener@imperial.ac.uk

Computational Optimisation Group
Centre for Process Systems Engineering
Department of Computing

2017/05/16

Strong Branching

Two complementary schemes for partitioning the search space are strong branching and maximum error branching.



Strong Branching - Key Idea

Tests several possible branching candidates before selecting the locally optimal variable for partitioning. Advantage: May explore relatively few nodes in the branch-and-bound tree. Disadvantage: Selecting variables for branching is computationally intensive.

Maximum error branching

Maximum error branching

Automatically selects the variable contributing to the greatest discrepancy between the MILP relaxation and the nonlinear representation at the relaxation solution $\hat{w}^{\mathrm{aux}}$; $\hat{x}$:

$$\arg\max_i \sum_j \left| \hat{w}_{i,j}^{\mathrm{aux}} - f(\hat{x}) \right|$$

Where to branch?

Once we've chosen a variable for branching, where should we divide the search space?

- At the mid point?

$$x_i^b = x_i^L + (x_i^U - x_i^L)/2$$

- At the solution to the relaxation?

$$x_i^b = \hat{x}_i$$

- At a point minimising the error in the next iteration?
- At a convex combination of the previous items?

$$x_i^b = \lambda \cdot \left(x_i^L + (x_i^U - x_i^L)/2 \right) + (1 - \lambda)\hat{x}_i$$