

Capacitated Multi-Commodity-Flow Cuts

Part II: Separation

Tobias Achterberg

Christian Raack



Abstract: Given an MCF network, we derive cutting planes based on network cuts. We use the complemented mixed integer rounding (c-MIR) framework but replace the standard aggregation algorithm by a procedure exploiting the network structure.

General c-MIR Procedure:

Marchand & Wolsey [01], Louveaux & Wolsey [03]

Aggregation: Combine (weighted) rows to obtain a single mixed integer constraint (standard: less than 10 rows)

Bound substitution: Substitute some variables by their (variable) upper bounds

Scaling and MIR: Scale row and apply MIR

Properties:

- Many rows in aggregation but sparse inequalities due to cancellation \Rightarrow **Not possible with standard aggregation**
- Inequalities with problem-specific nature
- Cuts with connected shores \Rightarrow **Strong inequalities**

Extensions:

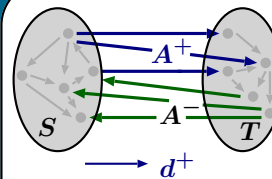
- Consider all single-node cuts
- Consider subsets of the commodities and subsets of A^+ (flow-cutset inequalities)
- Apply knapsack cover and flow cover separation to base inequalities

Network cut selection by graph contraction:

- Shrink arcs with large slack in the capacity row, use duals for tight rows \Rightarrow **Shores stay connected**
- Enumerate all cuts of the resulting partition

Aggregation based on network cut:

- Aggregate all flow rows for S and commodities k with positive cut demand $d^k \Rightarrow$ **Flow variables in S cancel out**
- Add all capacity rows for A^+



$$\sum_{a \in A^+} f_a^k - \sum_{a \in A^-} f_a^k \geq d^k, k \in K^+$$

$$C x_a - \sum_{k \in K} f_a^k \geq 0, a \in A^+$$

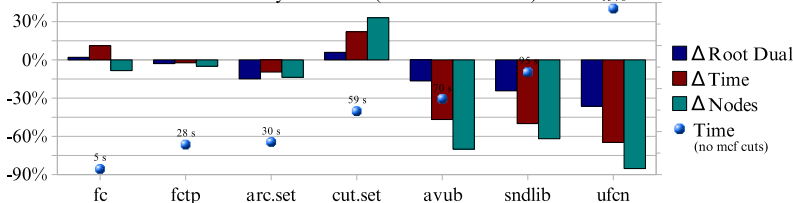
Base inequality:

$$\sum_{a \in A^+} C x_a - \sum_{\substack{k \in K \setminus K^+ \\ a \in A^+}} f_a^k - \sum_{\substack{k \in K^+ \\ a \in A^-}} f_a^k \geq \sum_{k \in K^+} d^k =: d^+$$

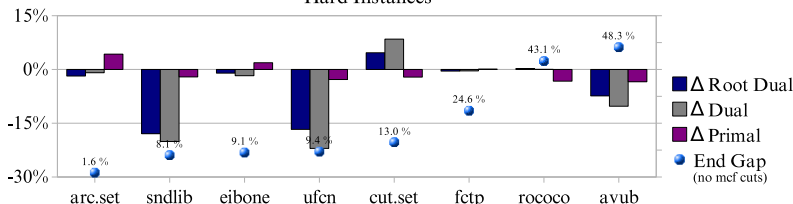
MIR inequality: $\sum_{a \in A^+} x_a \geq \left\lceil \frac{d^+}{C} \right\rceil$

Preliminary results – geometric means

Easy Instances (solvable within 1h)



Hard Instances



Relative improvements w.r.t. default SCIP 1.0.7 (including all default cuts)

- \rightarrow Negative values imply better results (% average improvement)
- \rightarrow Dual (Primal) is considered as gap to best known Primal (Dual)

Test sets:

set	#	origin	description
arc.set	35	A. Atamtuerk	MCF, unsplitable and splittable, binary capacity
avub	60	A. Atamtuerk	randomly generated, SCF, binary capacities + GUB
cut.set	15	A. Atamtuerk	MCF, integer capacities
fc	20	A. Atamtuerk	SCF, fixed charge, binary capacity
eibone	20	Eibone proj.	MCF on complete graph, 2-layers, integer capacities
fctcp	28	J. Gottlieb	SCF, complete bipartite, binary capacity
sndlib	52	sndlib.zib.de	MCF, integer capacities or binary capacities +GUB
rococo	20	Rococo proj.	MCF, unsplitable, binary capacity
ufcn	84	L.A. Wolsey	SCF, uncapacitated, fixed charge, binary cap, big M

Discussion:

- For most of the instances we improve dual as well as primal bounds
- Success largely depends on the quality of the network detection, for **avub**, **sndlib**, **ufcn**, **arc.set** we do very well, for **cut.set** we fail
- No change for **fctcp**, **rococo**, **eibone**, overhead for very easy **fc** test set outweighs node decrease
- 11 instances can be solved within 1h only using our separator (5 **avub**, 4 **ufcn**, 2 **sndlib**)