Cycle decomposition on network flow problems

Jacques Desrosiers[1]

Jean Bertrand Gauthier[1]

([1]) HEC Montréal and GERAD

Column generation workshop

Búzios, Brazil
May 22–25, 2016

Outline

Programs

Network flow formulation

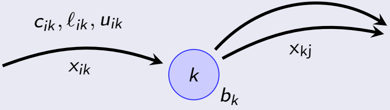
$$z^*_{CMCF} := \min \sum_{(i,j)\in A} c_{ij}x_{ij} \tag{1}$$

$$\text{s.t.} \quad \sum_{j:(k,j)\in A} x_{kj} - \sum_{i:(i,k)\in A} x_{ik} = b_k, \qquad [\pi_k], \qquad \forall k \in N, \tag{2}$$

$$0 \leq \ell_{ij} \leq x_{ij} \leq u_{ij}, \qquad \forall (i,j) \in A \tag{3}$$

- N: set of n nodes [b_k: demand π_k: node potential]
- A: set of m arcs [c_{ij}: cost ℓ_{ij}, u_{ij}: lower and upper bounds]

Network flow representation

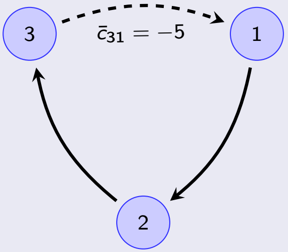


Edge direction

Reduced cost

$$\bar{c}_{ij} := c_{ij} - \pi_i + \pi_j, \ \forall (i,j) \in A$$

A cycle is a linear combination



To every entering arc-variable its cycle

$$\mathbf{a} = \sum_{(i,j)\in B} \mathbf{a}_{ij}\lambda_{ij}, \text{ where } \boldsymbol{\lambda} := \mathbf{A}_B^{-1}\mathbf{a}\,.$$

	x_{12}	x_{23}	x_{31}
1	1		-1
2	-1	1	
3		-1	1
$\boldsymbol{\lambda}$	-1	-1	1

Residual network $G(\mathrm{x}^0)$

Construction

- arc (i,j) with cost $d_{ij} := c_{ij}$ and residual capacity $r^0_{ij} := u_{ij} - x^0_{ij}$
- arc (j,i) with cost $d_{ji} := -c_{ij}$ and residual capacity $r^0_{ji} := x^0_{ij} - \ell_{ij}$

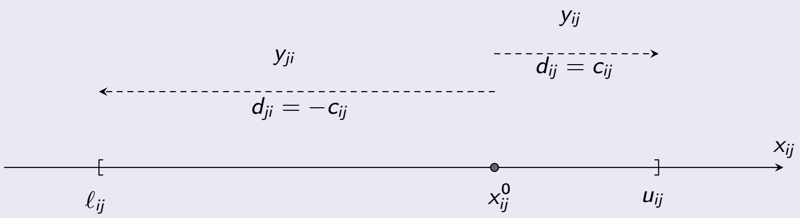
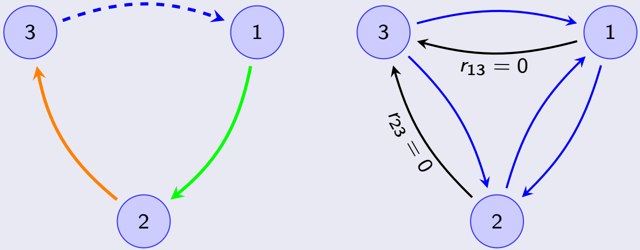


Figure: A change of variables

Residual network $G(x^0)$

A change of variables

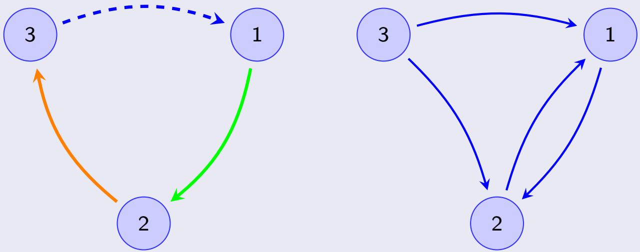


Notation

A': set of all y-variables

Residual network $G(\mathbf{x}^0)$

A change of variables



Notation

$A(\mathbf{x}^0)$: set of all residual y-variables ($r_{ij} > 0$)

Cycle definitions

Cycle

A cycle W is a sequence of arcs connecting once with the first and last arcs.

$$W := \{(i,j) \in A' \mid y_{ij} > 0\}$$

Directed cycle

A cycle W is *directed* if all its composing arcs have a positive residual capacity

$$r_{ij} > 0, \ \forall (i,j) \in W \qquad \equiv \qquad W \subseteq A(\mathbf{x}^0)$$

Negative cycle

A *negative cycle* is a directed cycle with a negative cost

$$W \subseteq A(\mathbf{x}^0) \qquad \text{and} \qquad c(W) < 0$$

Reduced costs: arcs vs. cycles

Arc

$$\bar{d}_{ij} := d_{ij} - \pi_i + \pi_j$$

Cycle

$$c(W) = \sum_{(i,j)\in W} d_{ij} = \sum_{(i,j)\in W} d_{ij} - \pi_i + \pi_j = \bar{c}(W)$$

Type 1 vs. Type 2 cycles

- Type 1: $\bar{d}_{ij} < 0, \ \forall (i,j) \in W$
- Type 2: $\exists (i,j) \in W \mid \bar{d}_{ij} \geq 0$

Type 1 focus

Admissible network $G(\mathbf{x}^0, \boldsymbol{\pi})$

$$A(\mathbf{x}^0, \boldsymbol{\pi}) \subseteq A(\mathbf{x}^0) \mid \bar{d}_{ij} < 0$$

Type 1 cycle cancellation

$$A(\mathbf{x}^1, \boldsymbol{\pi}) \subset A(\mathbf{x}^0, \boldsymbol{\pi})$$

Algorithm

Phase h

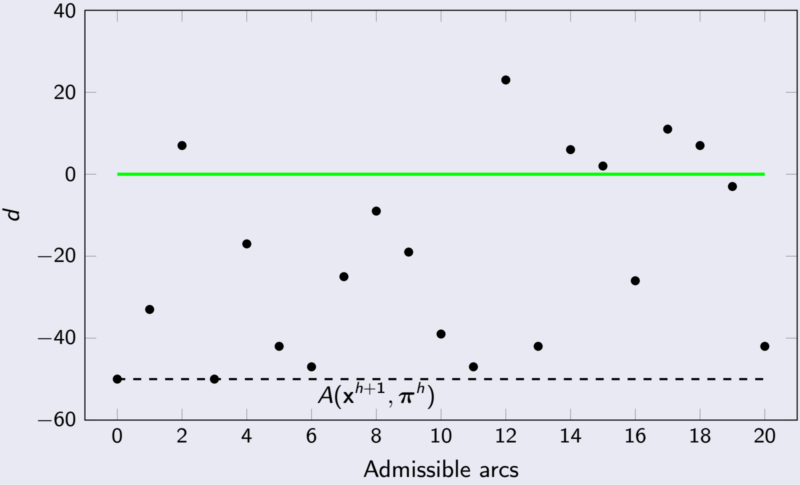
A sequence of Type 1 cycle (with respect to $\boldsymbol{\pi}^h$) cancellations terminated when none remains, i.e.,

$$G(\mathbf{x}^{h+1}, \boldsymbol{\pi}^h) \text{ is acyclic}$$

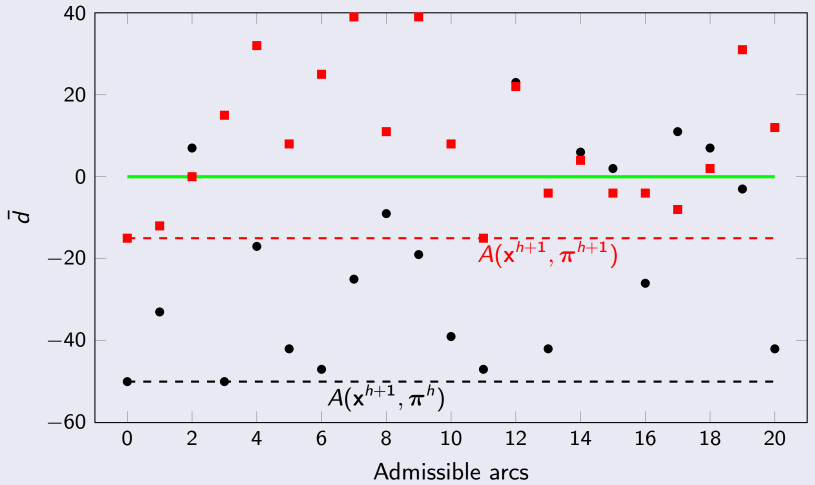
Persistent dual multipliers

Dual multipliers $\boldsymbol{\pi}^{h+1}$ are *updated* at the end of a phase

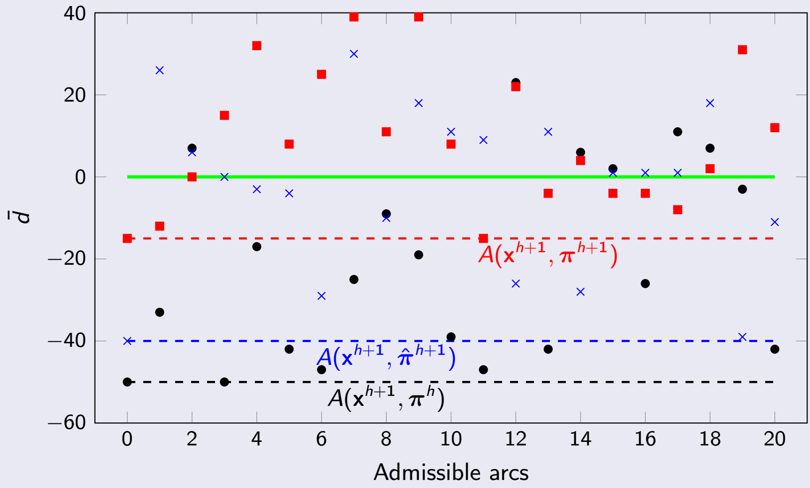
Reduced cost of admissible arcs with respect to dual variables



Reduced cost of admissible arcs with respect to dual variables



Reduced cost of admissible arcs with respect to dual variables



Updating dual multipliers

Optimization design

$$\mu^{h+1} := \max \mu \mid \mu \leq d_{ij} - \pi_i + \pi_j, \ \forall (i,j) \in A(\mathbf{x}^{h+1})$$

Heuristic design

$$\hat{\mu}^{h+1} := \min_{(i,j) \in A(\mathbf{x}^{h+1})} d_{ij} - \hat{\boldsymbol{\pi}}_i^{h+1} + \hat{\boldsymbol{\pi}}_j^{h+1}$$

Tighten step

- Establish $\hat{\boldsymbol{\pi}}$ in order to *ensure* a lift of $\hat{\mu}^{h+1}$
- $\pi_i^h \leq \pi_i^{h+1}, \ \forall i \in N, h \geq 0$

Minimum mean cycle-canceling algorithm

Seminal papers

- Goldberg and Tarjan (1989)
- Radzik and Goldberg (1994)

Strongly polynomial

Cancellation order *matters*

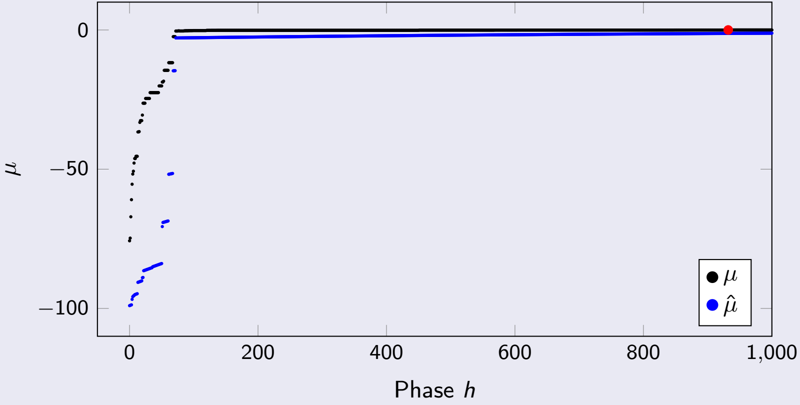
Minimum mean cycle-canceling algorithm

Complexity analysis summary (Gauthier et al., 2015)

	Outer loop		Global runtime complexity Cancel-and-Tighten strategy	
Point of view	# iterations	# phases	Without	With
Theorem 1	$O(mn\log(nC))$	$O(n\log(nC))$	$O(m^2n^2\log(nC))$	$O(mn\log n\log(nC))$
Theorem 2	$O(m^2n\log n)$	$O(mn\log n)$	$O(m^3n^2\log n)$	$O(m^2n(\log n)^2)$
Theorem 3	$O(m^2n)$	$O(mn)$	$O(m^3n^2)$	$O(m^2n\log n)$

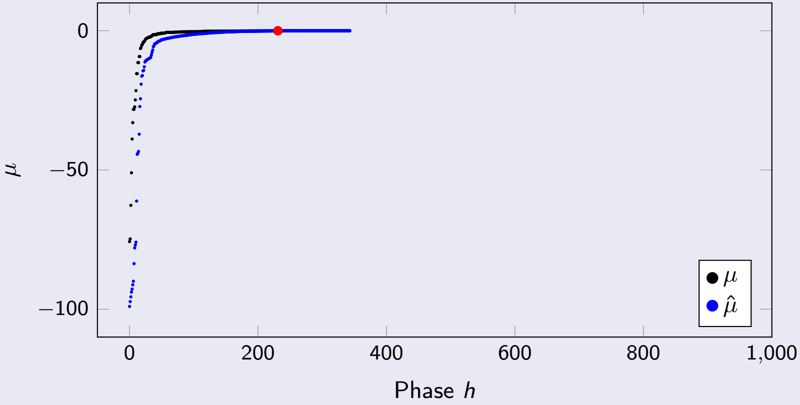
Cancel-and-Tighten

Optimality parameter μ and estimate $\hat{\mu}$ for Instance 1 [phase base - Update $L^h_{(a)}$]



Cancel-and-Tighten

Optimality parameter μ and estimate $\hat{\mu}$ for Instance 1 [phase base - Update $L^h_{(b)}$]



Cancel-and-Tighten

Computational results

Resolution	CPU (sec)
Instance 1	
CT	0.82
CPLEX	1.60
gridgen_sr_13a	
CT	36.17
CPLEX	15.46
goto_8_15a	
CT	281.44
CPLEX	282.61
goto_8_16a	
CT	10184.10
CPLEX	2587.94
grid_long_13a	
CT	4.92
CPLEX	0.25
netgen_lo_sr_15a	
CT	890.86
CPLEX	182.75
gridgen_8_17a	
CT	506.11
CPLEX	719.28

Miscellaneous observations

Culprits

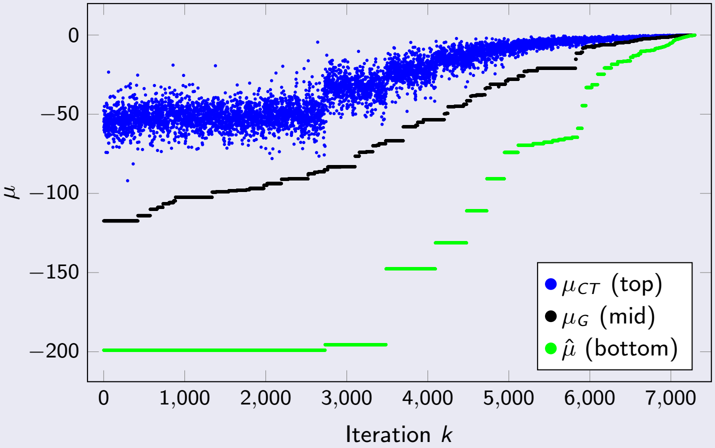
- Initial multipliers
- Tail

Exploitable ideas?

- Acyclic admissible network
- Push intermediate dual variables to integer values

Cancel-and-Tighten flashback

Comparison of μ_{CT}, μ_G and $\hat{\mu}$



Cycle decomposition

Cycle subset

$$\overline{\Omega} \subset \Omega$$

Dantzig-Wolfe decomposition

$$
\begin{aligned}
& \min z^0 + \sum_{w \in \overline{\Omega}} c(w) \cdot \phi_w \\
\text{s.t.} \quad & -r_{ji} \leq \sum_{w \in \overline{\Omega}} a_{ij}(w) \cdot \phi_w \leq r_{ij}, && \forall (i,j) \in A \\
& \phi_w \geq 0, && \forall w \in \overline{\Omega}
\end{aligned}
$$

Cycle decomposition impact

Phase	CT	DW	Number of cycles	CPU time (sec)
1	5415562800	74063734	1629	0.070
2	123041650	689478	76	0.000
3	690058	690058	4	0.000
4	690058			
5	690058			
6	690058			
7	690058			
8	690058			
9	690058			
10	689389			

Conclusion

Key ideas

- Exact algorithm which inherently exploits heuristic ideas
- Nondecreasing dual multipliers
- Update mechanism matters significantly
- No basis maintenance

Research paths

- Improve Tighten heuristic
- Adapt dual variables from the cycle master problem to the Tighten step
- Transfer Cancel-and-Tighten to linear programs