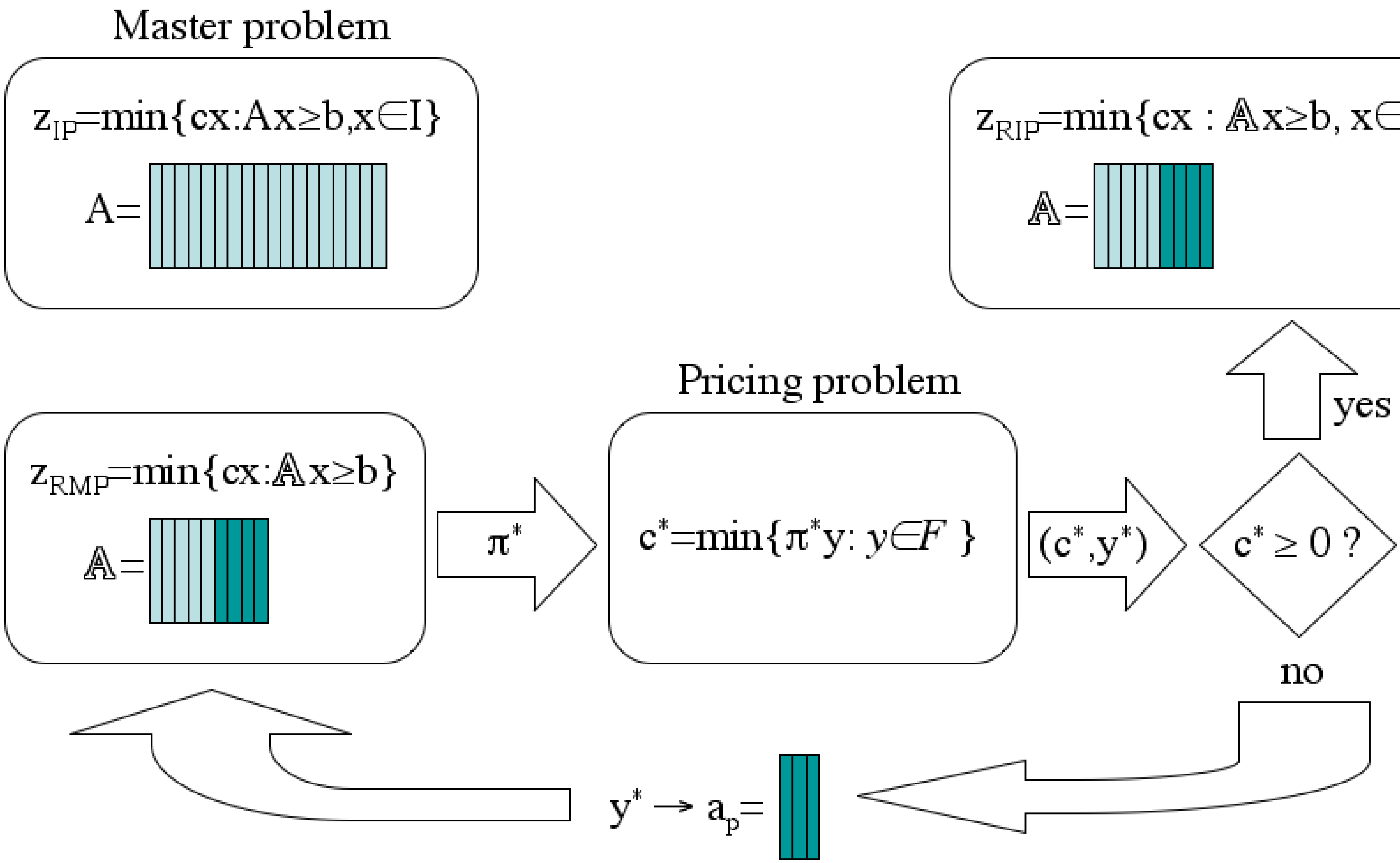
Resource Constrained Shortest Paths with Side Constraints and Non Linear Costs

Stefano Gualandi
stefano.gualandi@unipv.it
http://www-dimat.unipv.it/~gualandi
twitter: @famo2spaghi

Outline

1. Introduction
2. Non Linear Costs
3. Iterated Preprocessing
4. Lower bounding via Lagrangian Relaxation
5. Computational Results

Column Generation Overview



What is *F* in Crew Scheduling problems?

Resource Constrained Shortest Path

The problem of putting together a set of pieces of work into a single duty, that is a column or variable of problem (LP-MP), is formalized as a

Resource Constrained Shortest Path Problem

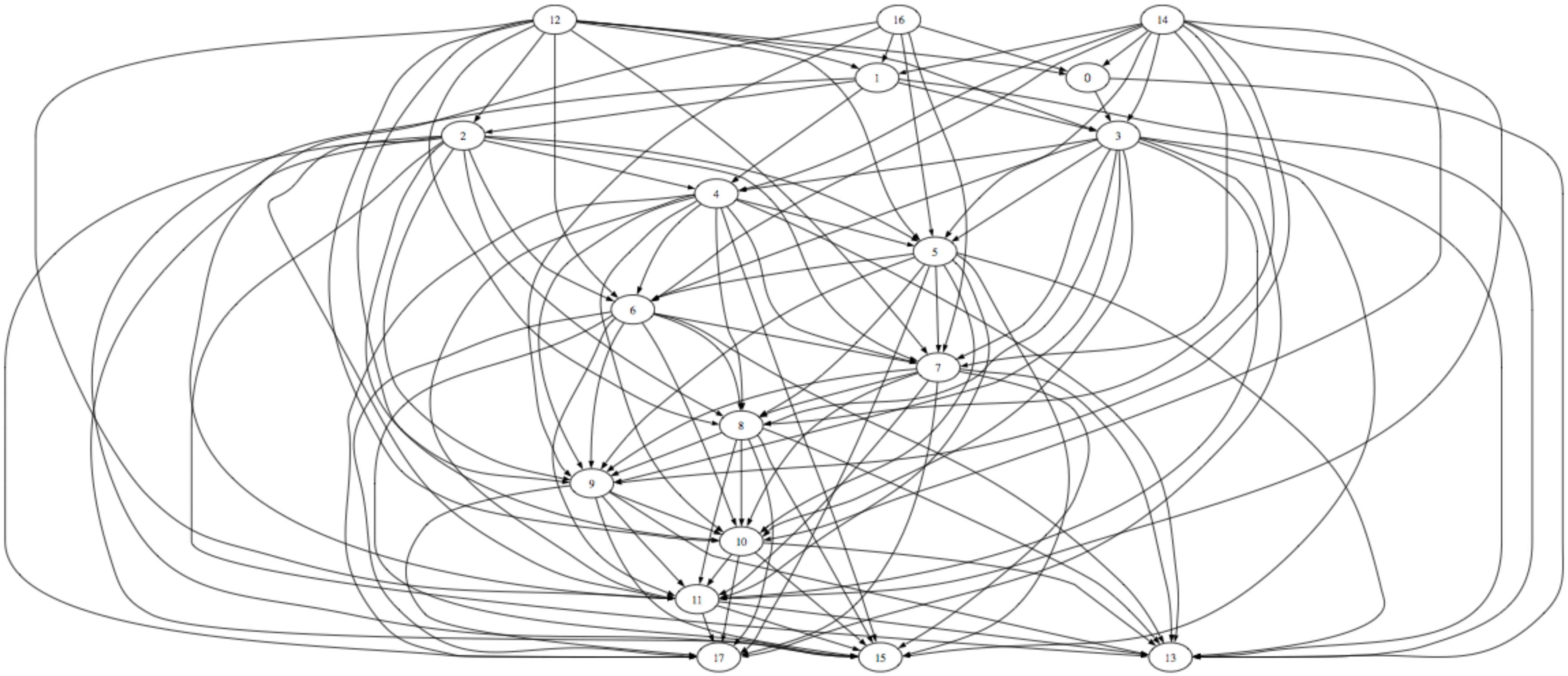
Example 12 pieces of works, 3 depots

ID	Da	A	Inizio	Fine
0	NETTPO	RMANAG	04:30	06:20
1	NETTPO	RMLAUREN	04:40	06:20
2	RMLAUREN	NETTPO	06:20	08:15
3	APRILI	LATINA	07:25	08:05
4	ANZICO	NETTPO	13:00	13:40
5	NETTPO	ANZIO	14:00	14:25
6	ANZIO	NETTPO	14:30	14:50
7	NETTPO	ANZIO	14:50	15:20
8	ANZIO	NETTPO	15:30	16:00
9	NETTPO	ANZIO	16:00	16:20
10	ANZIO	NETTPO	16:30	16:55
11	NETTPO	ANZIO	17:30	18:00

Resource Constrained Shortest Path

Let $G = (N, A)$ be the compatibility graph, weighted, directed, and acyclic:

- $N = P \cup \{\{s^h, t^h\} | h \in D\}$ a node for each PoW, and a pair of nodes for each depot
- A has an arc for each pair (i, j) of compatible PoW, and (s^h, i) (pull-out) and (i, t^h) (pull-in) $\forall h \in D$ and $i \in P$

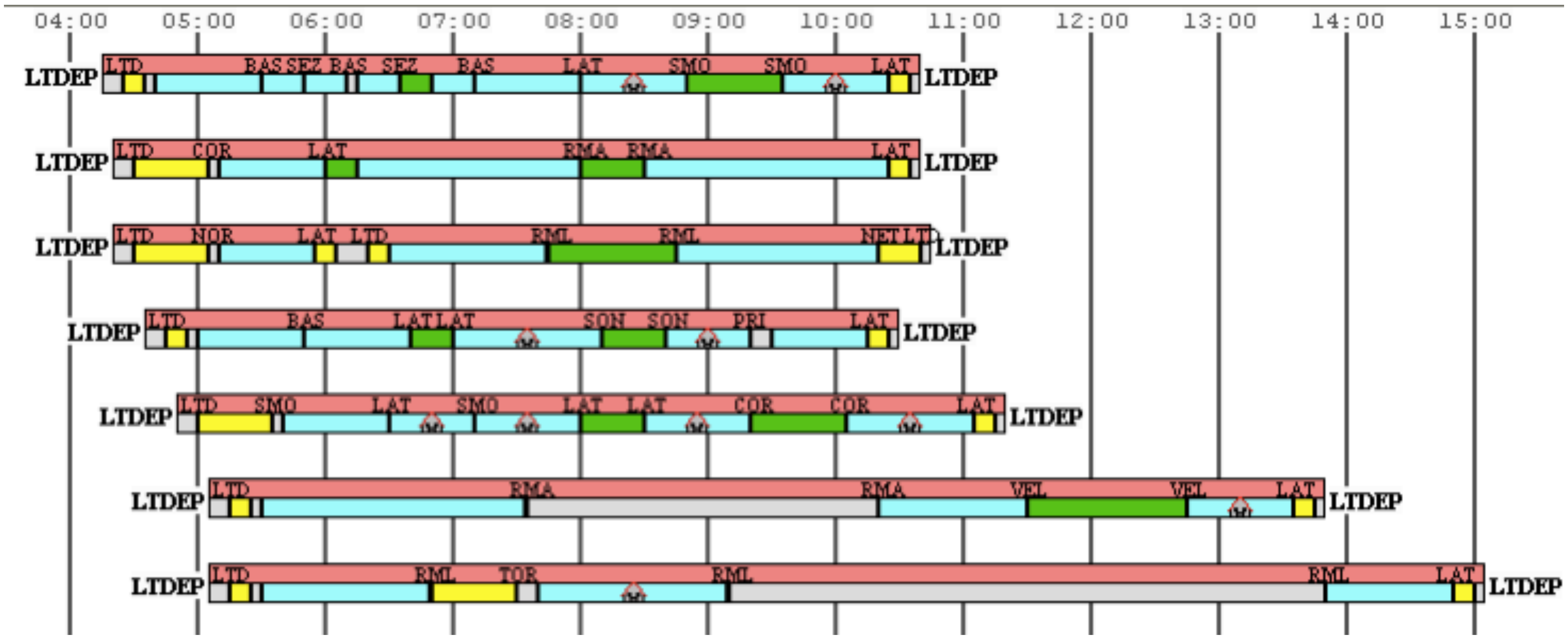


Resource Constrained Shortest Path

- each arc (i, j) has associated a set of resources r_{ij}^k, for each $k \in K$, e.g. **working time**, driving time, and break time (other resources may be used to model working regulation)

	NEDEP	ANZICO	12:35	12:55	VAV
4	ANZICO	NETTPO	13:00	13:40	PG
5	NETTPO	ANZIO	14:00	14:25	PG
6	ANZIO	NETTPO	14:30	14:50	PG
7	NETTPO	ANZIO	14:50	15:20	PG
8	ANZIO	NETTPO	15:30	16:00	PG
9	NETTPO	ANZIO	16:00	16:20	PG
10	ANZIO	NETTPO	16:30	16:55	PG
11	NETTPO	ANZIO	17:30	18:00	PG
	ANZIO	NEDEP	18:00	18:10	VAV
			durata:	5:35	

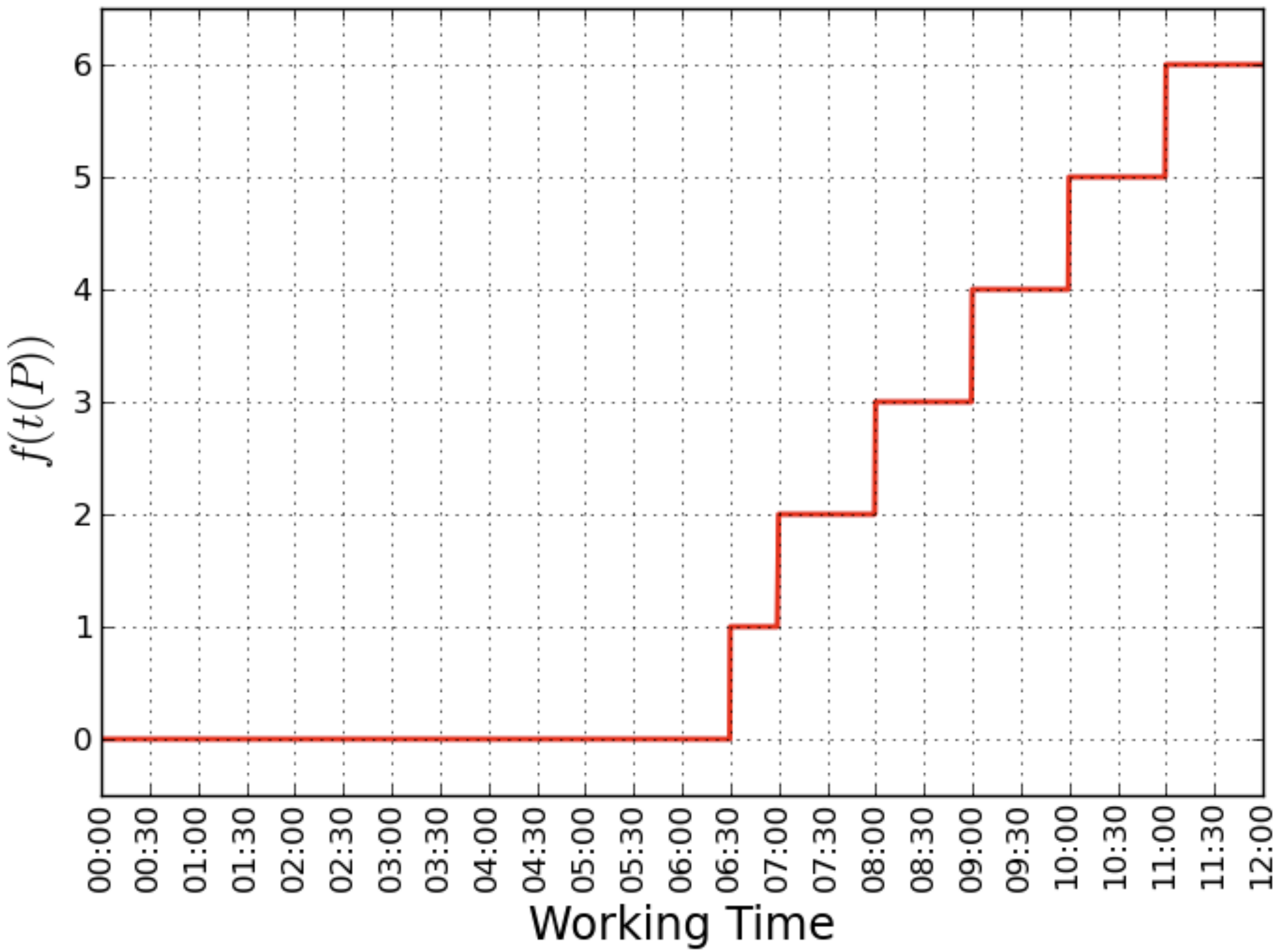
Example of Crew Schedules (with reosurces)



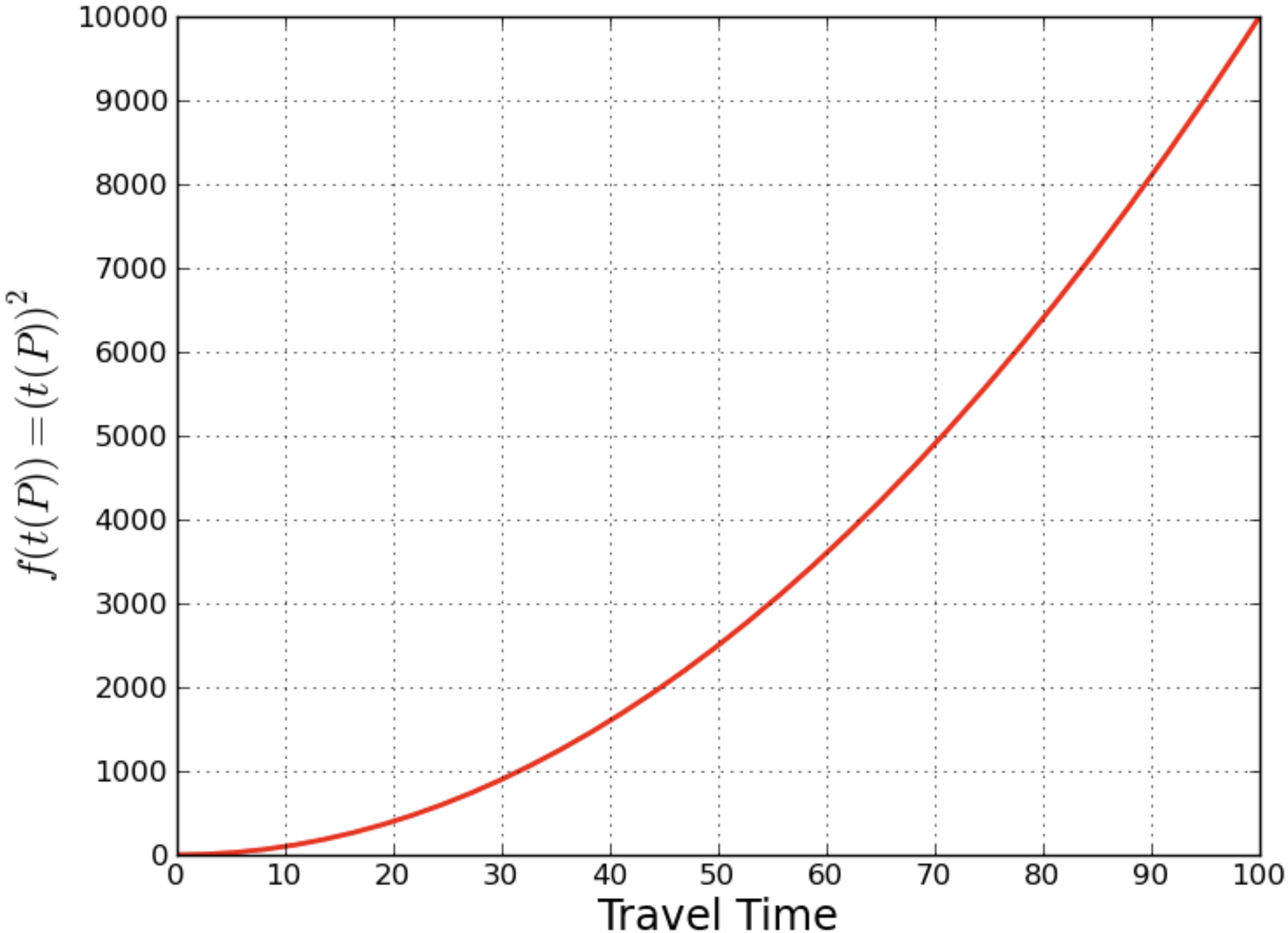
Resources:

1. spread time (red)
2. driving time (light blue), corresponds to PoW
3. *out-of-service* time (yellow)
4. long break (grey)
5. breaks (green), very important how they are located

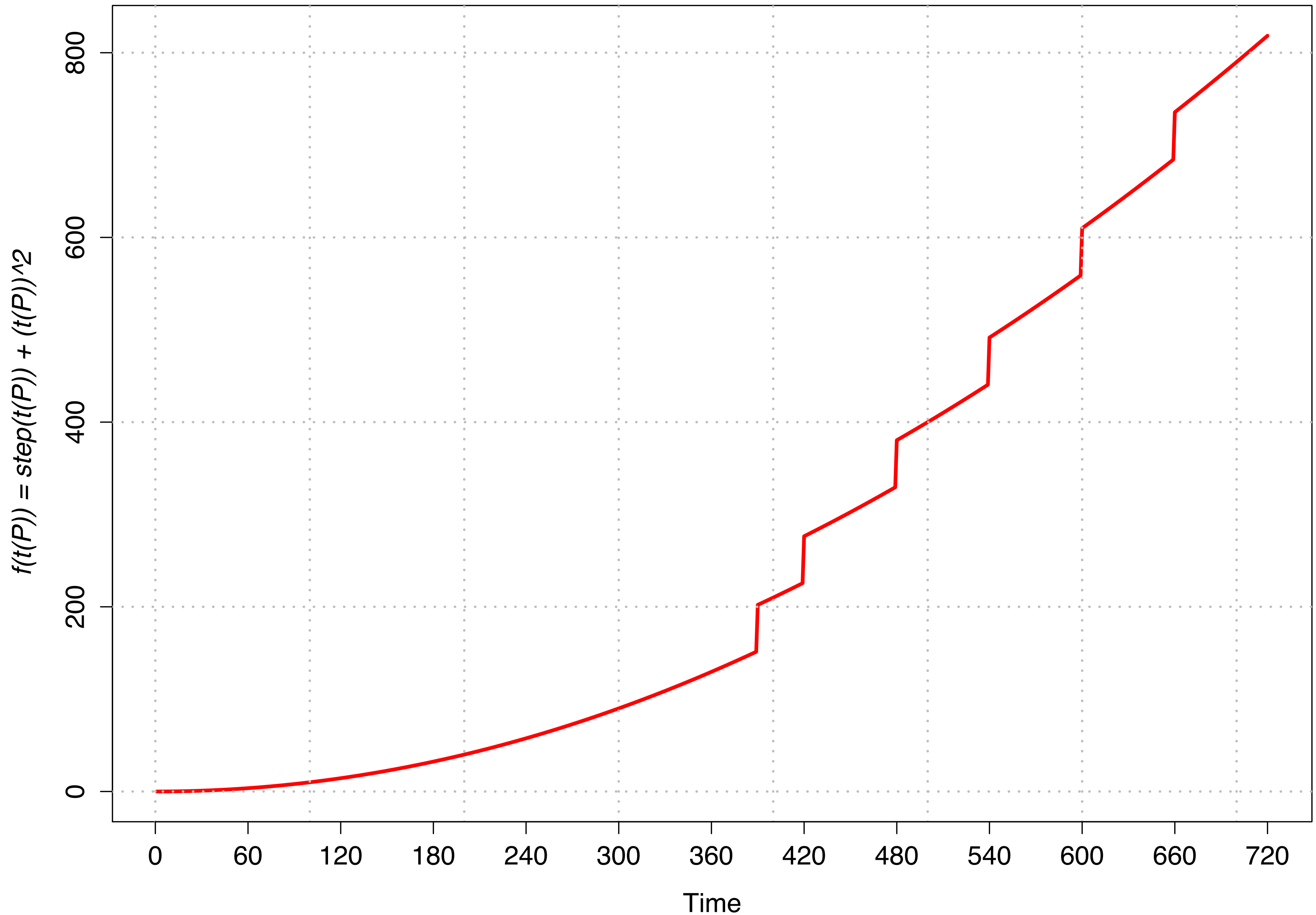
Non Linear Costs



Non Linear Costs



Non Linear Costs

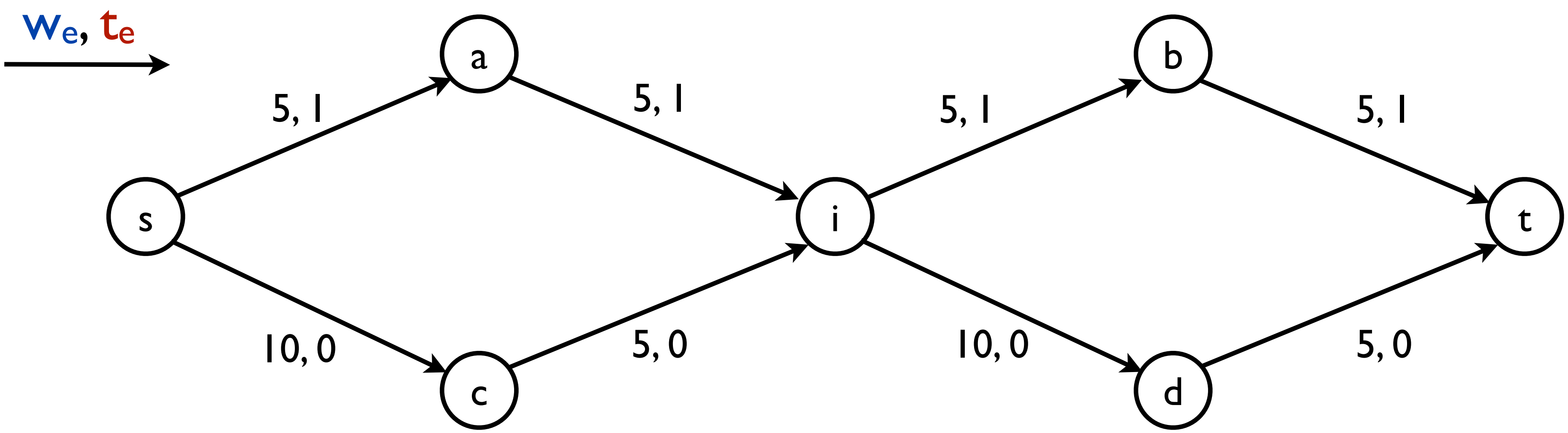


Resource Constrained Shortest Paths (RCSP)

Arc-flow IP formulation with non linear costs $f_h(\cdot)$:

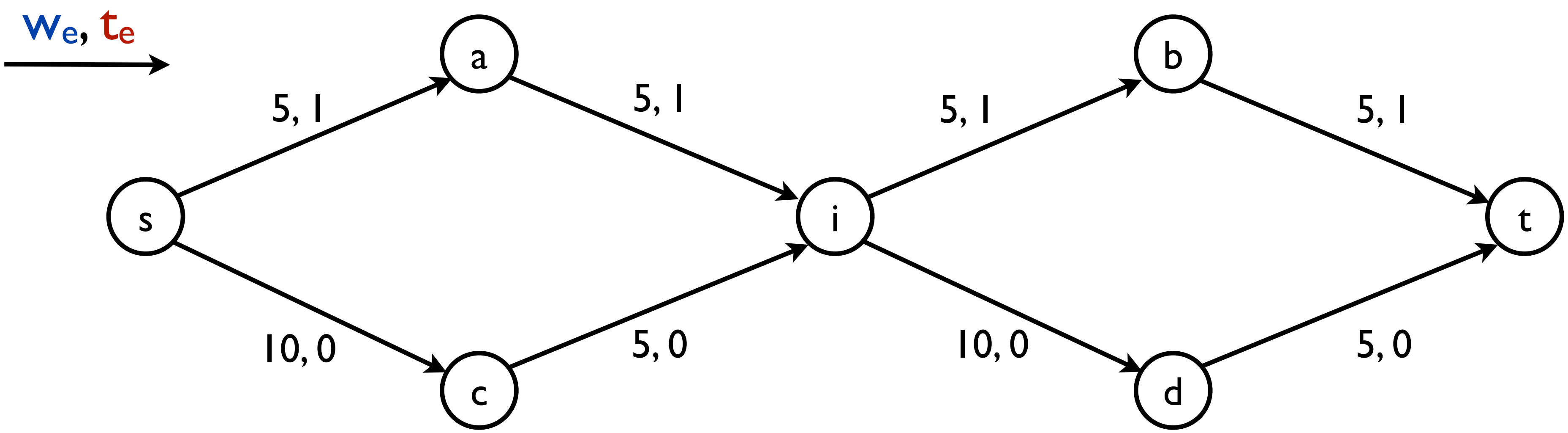
$$
\begin{aligned}
\min \quad & \sum_{e\in A} w_e x_e + \sum_{k\in K}\sum_{h\in H} f_h\Big(\sum_{e\in A} r_e^k x_e\Big) & \\
\text{s.t.} \quad & \sum_{e\in\delta_i^+} x_e - \sum_{e\in\delta_i^-} x_e = b_i = \begin{cases} +1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases} & \forall i \in N \\
& \sum_{e\in A} r_e^k x_e \le U^k & \forall k \in K \\
& + \text{ side non linear constraints} & \\
& x_e \in \{0,1\} & \forall e \in A.
\end{aligned}
$$

Resource Constrained Shortest Paths (RCSP)



Resource Constrained Shortest Paths (RCSP)

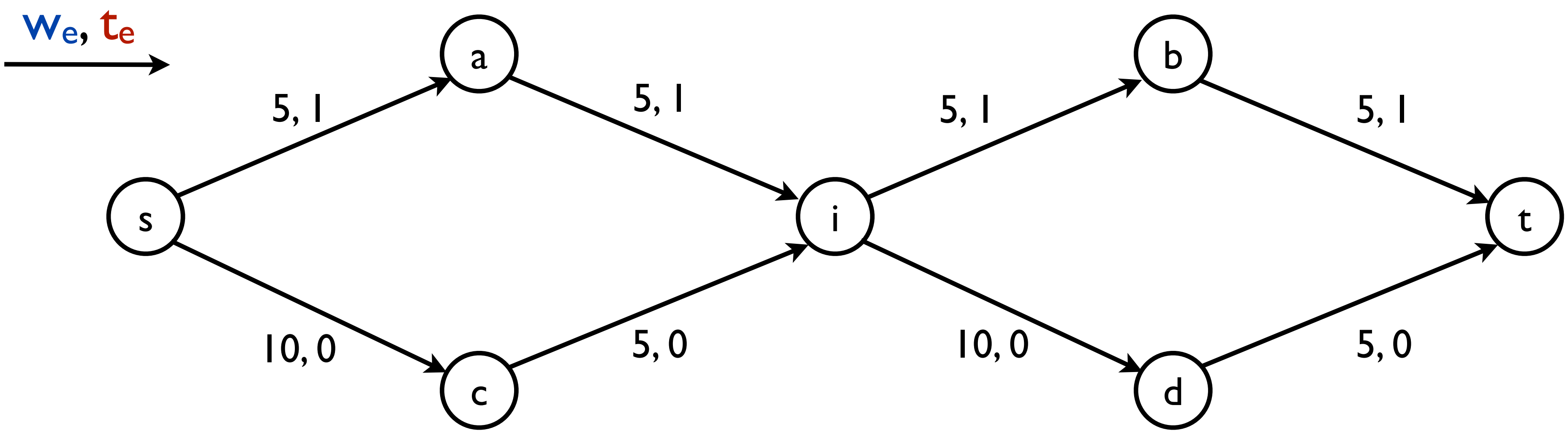
We restrict to super additive functions: $c(P_1 \cup P_2) \geq c(P_1) + c(P_2)$



Resource Constrained Shortest Paths (RCSP)

We restrict to super additive functions: $c(P_1 \cup P_2) \geq c(P_1) + c(P_2)$

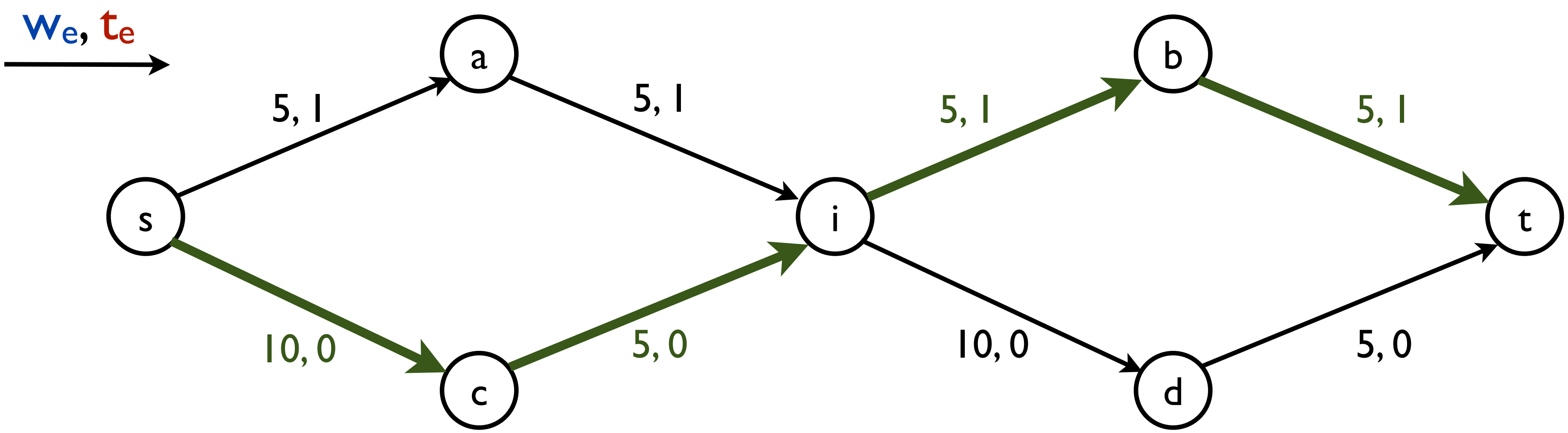
Example: $c(P) = w(P) + f(P) = \sum_{e \in P} w_e + \left(\sum_{e \in P} t_e\right)^2$



Resource Constrained Shortest Paths (RCSP)

We restrict to super additive functions: $c(P_1 \cup P_2) \geq c(P_1) + c(P_2)$

Example: $c(P) = w(P) + f(P) = \sum_{e \in P} w_e + \left(\sum_{e \in P} t_e\right)^2$

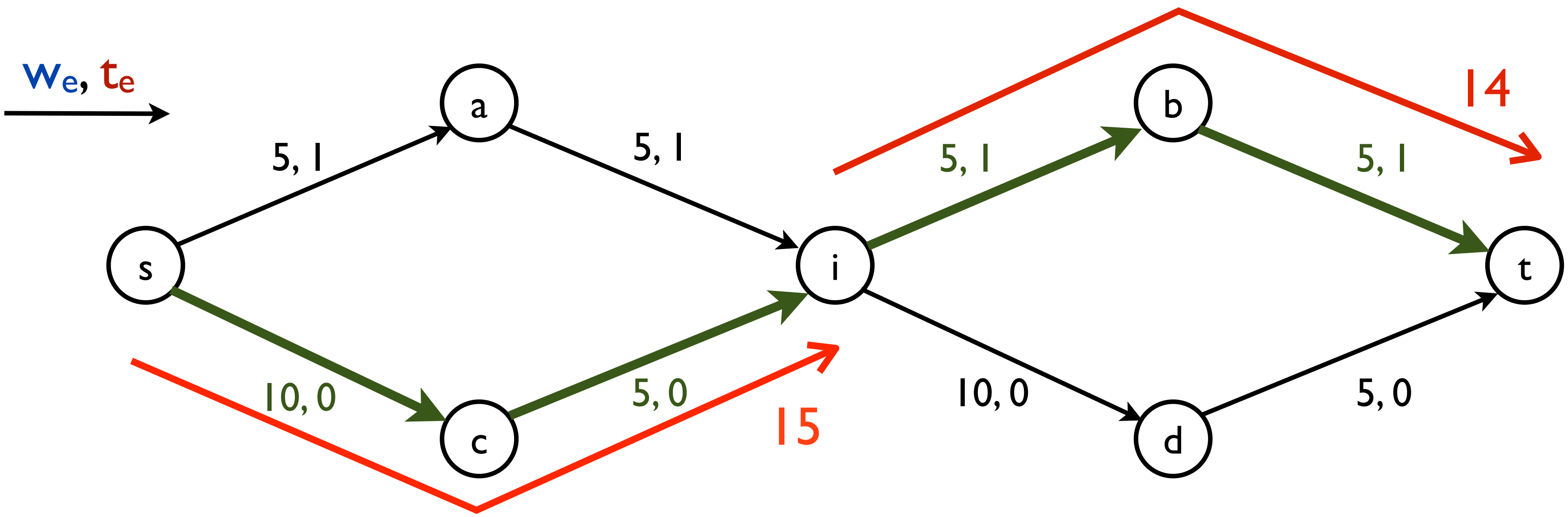


Optimal Path: $P_2 = \{s, c, i, b, t\}$, $c(P_2)$ = **25** + **4** = **29**

Resource Constrained Shortest Paths (RCSP)

We restrict to super additive functions: $c(P_1 \cup P_2) \geq c(P_1) + c(P_2)$

Example: $c(P) = w(P) + f(P) = \sum_{e \in P} w_e + \left(\sum_{e \in P} t_e\right)^2$

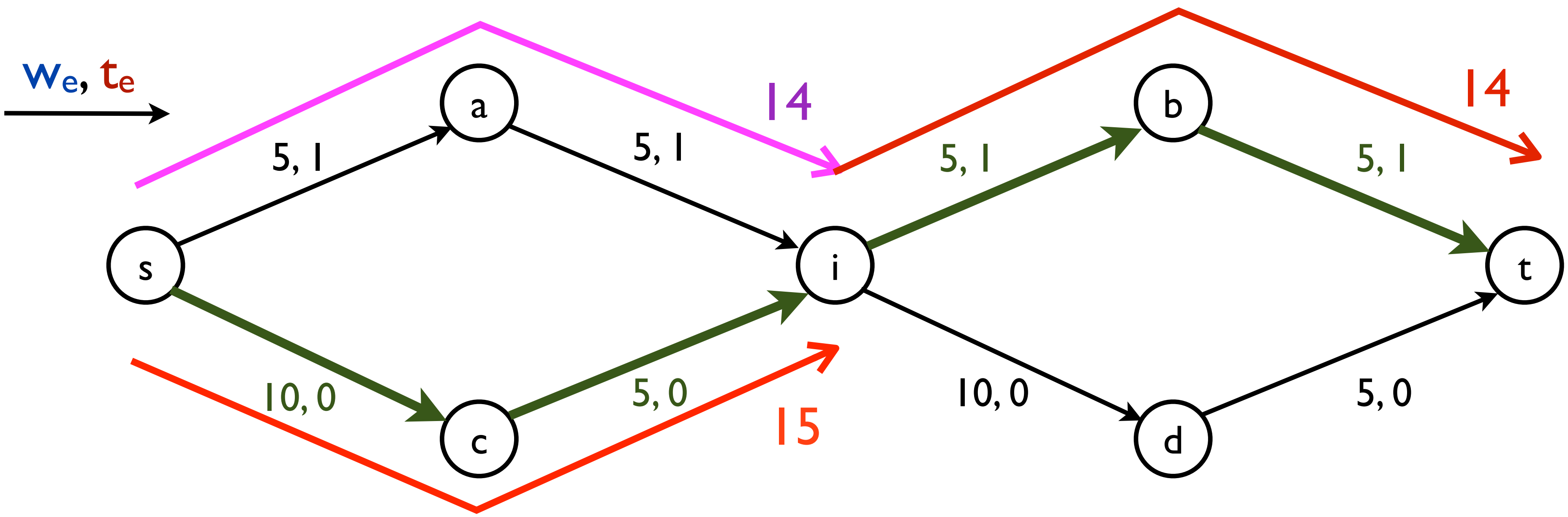


Optimal Path: P_2 = {s, c, i, b, t}, $c(P_2)$ = **25** + **4** = **29**

Resource Constrained Shortest Paths (RCSP)

We restrict to super additive functions: $c(P_1 \cup P_2) \geq c(P_1) + c(P_2)$

Example: $c(P) = w(P) + f(P) = \sum_{e \in P} w_e + \left(\sum_{e \in P} t_e\right)^2$

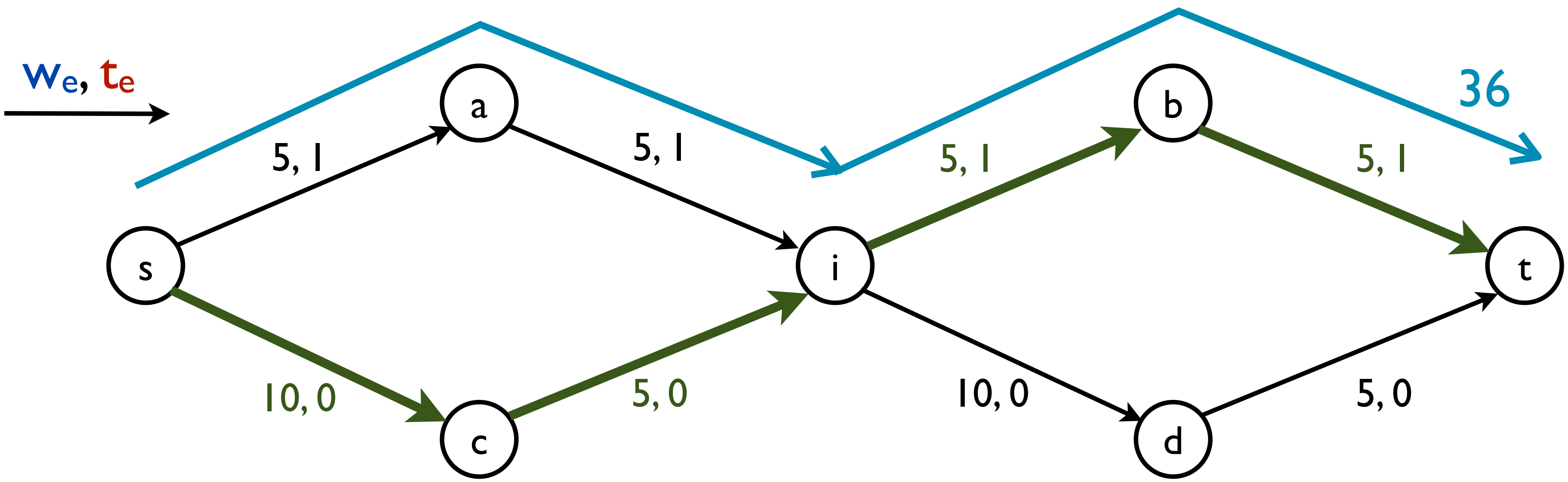


Optimal Path: $P_2 = \{s, c, i, b, t\}$, $c(P_2) = \mathbf{25} + \mathbf{4} = \mathbf{29}$

Resource Constrained Shortest Paths (RCSP)

We restrict to super additive functions: $c(P_1 \cup P_2) \geq c(P_1) + c(P_2)$

Example: $c(P) = w(P) + f(P) = \sum_{e \in P} w_e + \left(\sum_{e \in P} t_e\right)^2$

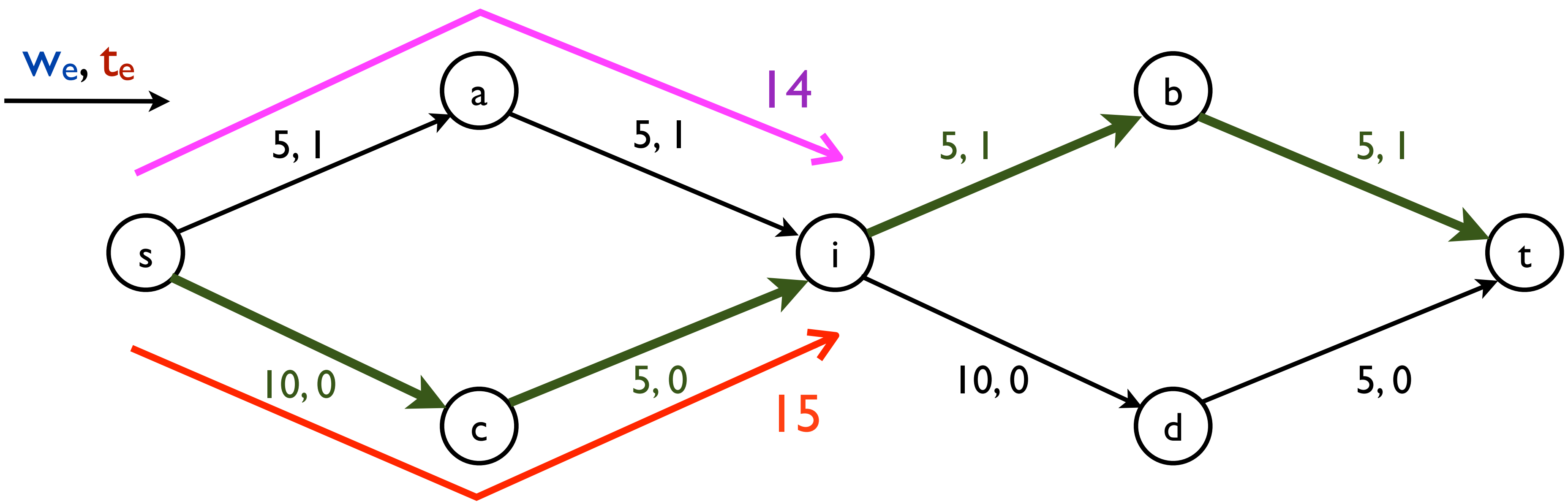


Path: P_3 = {s, a, i, b, t}, $c(P_2)$ = **20** + **16** = **36**

Resource Constrained Shortest Paths (RCSP)

We restrict to super additive functions: $c(P_1 \cup P_2) \geq c(P_1) + c(P_2)$

Example: $c(P) = w(P) + f(P) = \sum_{e \in P} w_e + \left(\sum_{e \in P} t_e\right)^2$



Bellmann's optimality conditions do not hold!

Outline

1. Introduction
2. Non Linear Costs
3. Iterated Preprocessing
4. Lower bounding via Lagrangian Relaxation
5. Computational Results

Resource-based Preprocessing

(Beasley and Christofides, 1989; Dumitrescu and Boland, 2003; Sellmann et al., 2007)

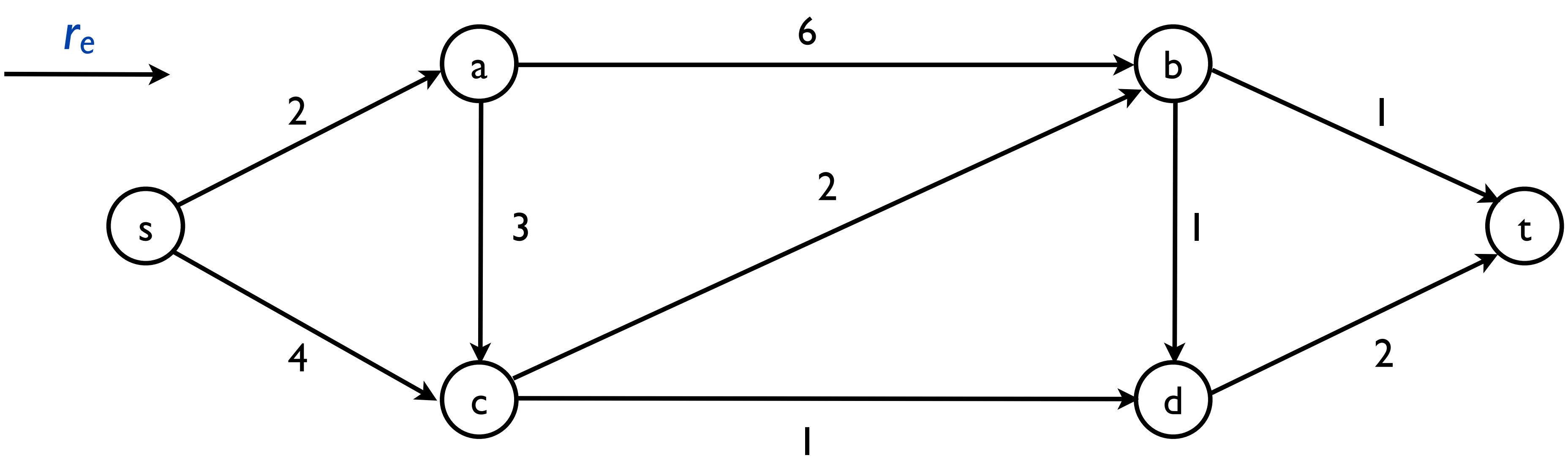
if $r^k(P^*_{si}) + r^k_e + r^k(P^*_{jt}) > U^k$ **then** remove arc $e = (i, j)$
where P^*_{si} and P^*_{jt} are shortest (k-th resource) paths.

Resource-based Preprocessing

(Beasley and Christofides, 1989; Dumitrescu and Boland, 2003; Sellmann et al., 2007)

if $r^k(P^*_{si}) + r^k_e + r^k(P^*_{jt}) > U^k$ **then** remove arc $e = (i, j)$
where P^*_{si} and P^*_{jt} are shortest (k-th resource) paths.

Resource consumption of each arc. Upper resource bound $U = 7$.

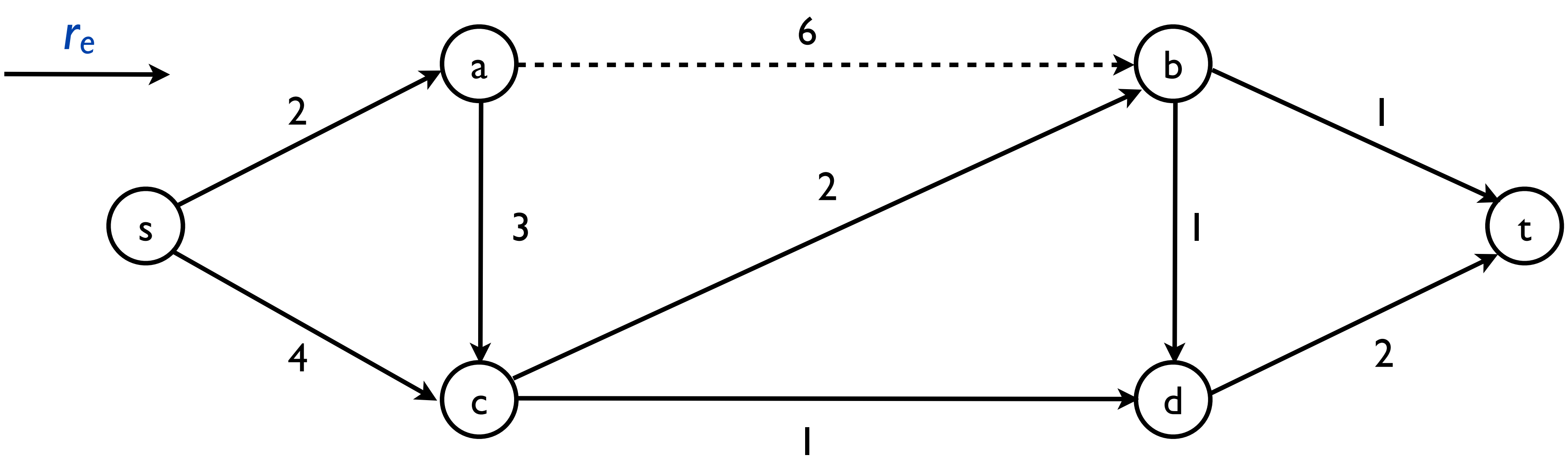


Resource-based Preprocessing

(Beasley and Christofides, 1989; Dumitrescu and Boland, 2003; Sellmann et al., 2007)

if $r^k(P^*_{si}) + r^k_e + r^k(P^*_{jt}) > U^k$ **then** remove arc $e = (i, j)$
where P^*_{si} and P^*_{jt} are shortest (k-th resource) paths.

Resource consumption of each arc. Upper resource bound $U = 7$.

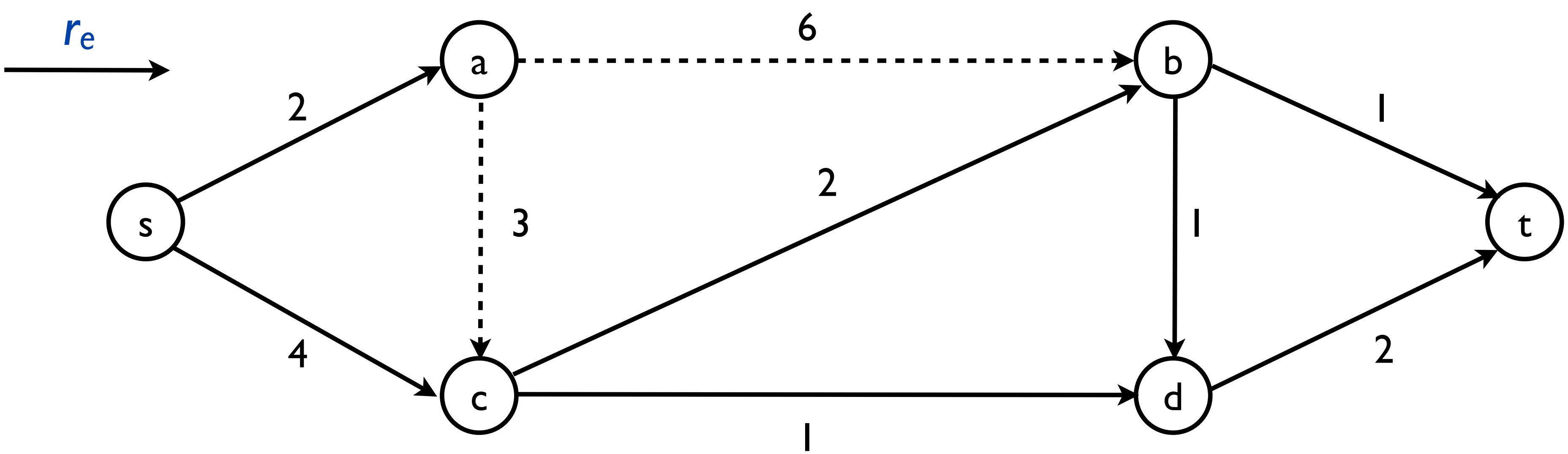


Resource-based Preprocessing

(Beasley and Christofides, 1989; Dumitrescu and Boland, 2003; Sellmann et al., 2007)

if $r^k(P^*_{si}) + r^k_e + r^k(P^*_{jt}) > U^k$ **then** remove arc $e = (i, j)$
where P^*_{si} and P^*_{jt} are shortest (k-th resource) paths.

Resource consumption of each arc. Upper resource bound $U = 7$.

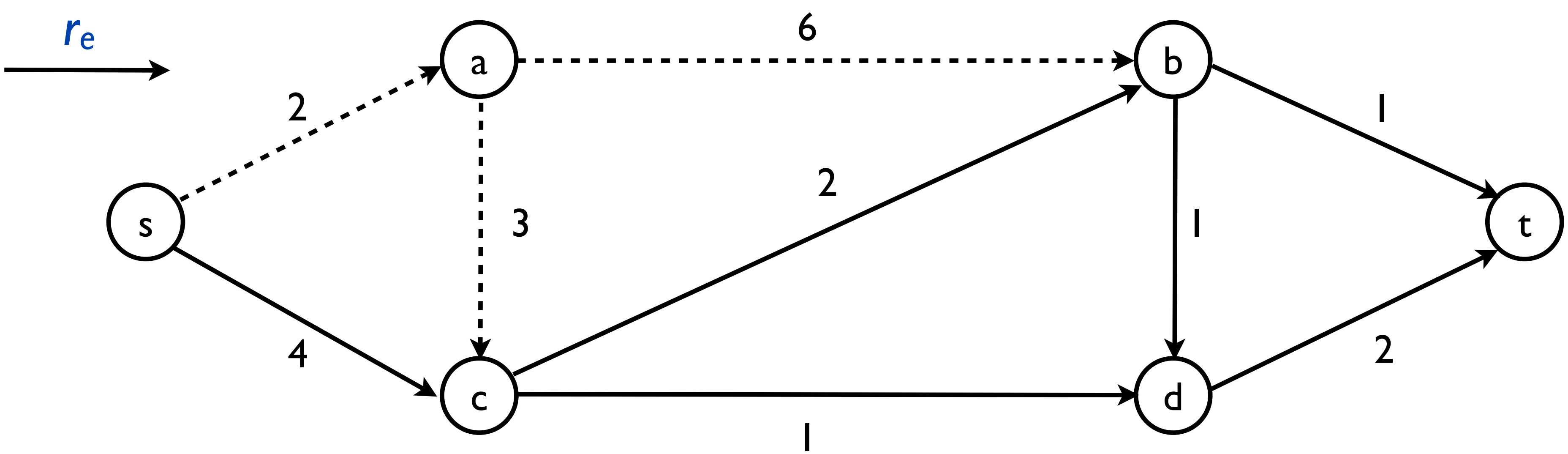


Resource-based Preprocessing

(Beasley and Christofides, 1989; Dumitrescu and Boland, 2003; Sellmann et al., 2007)

if $r^k(P^*_{si}) + r^k_e + r^k(P^*_{jt}) > U^k$ **then** remove arc $e = (i, j)$
where P^*_{si} and P^*_{jt} are shortest (k-th resource) paths.

Resource consumption of each arc. Upper resource bound $U = 7$.

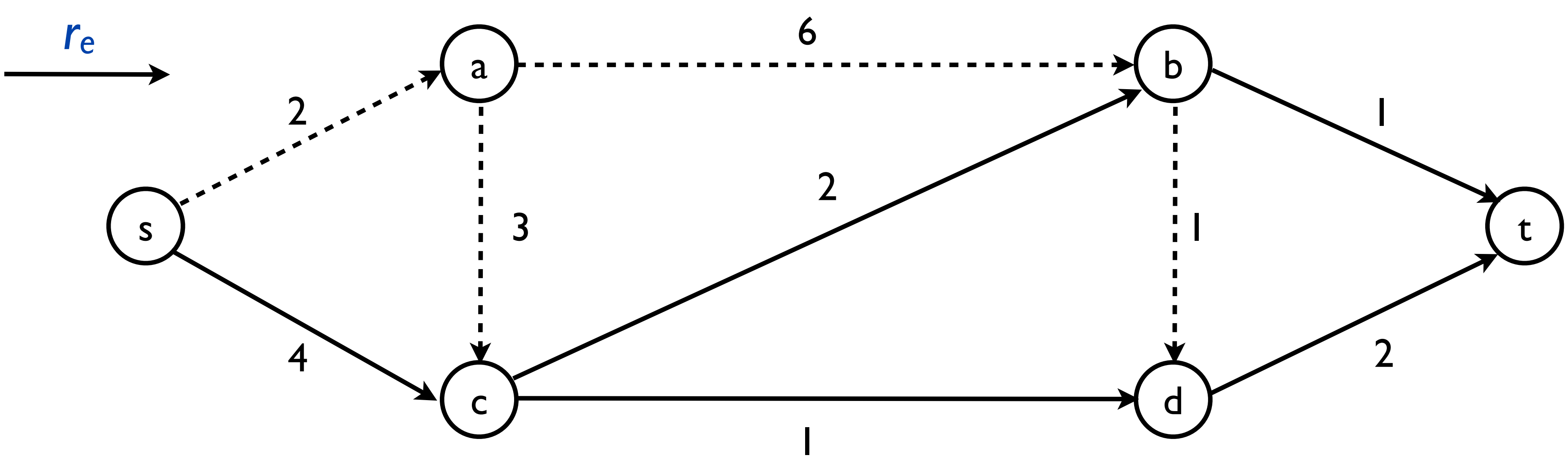


Resource-based Preprocessing

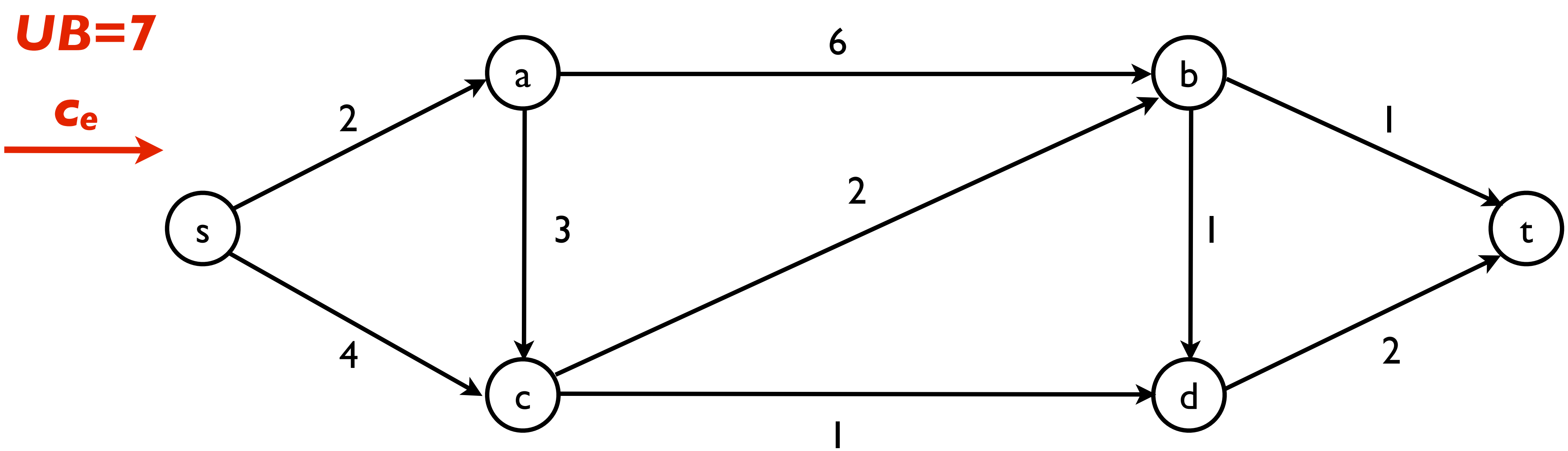
(Beasley and Christofides, 1989; Dumitrescu and Boland, 2003; Sellmann et al., 2007)

if $r^k(P^*_{si}) + r^k_e + r^k(P^*_{jt}) > U^k$ **then** remove arc $e = (i, j)$
where P^*_{si} and P^*_{jt} are shortest (k-th resource) paths.

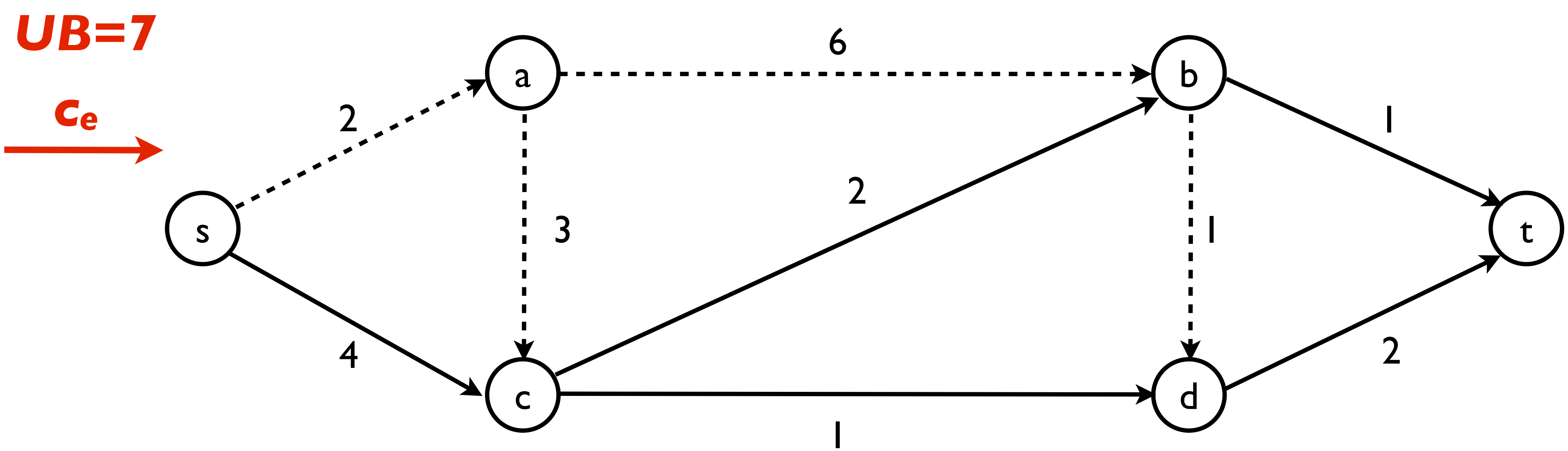
Resource consumption of each arc. Upper resource bound $U = 7$.



Cost-based Preprocessing

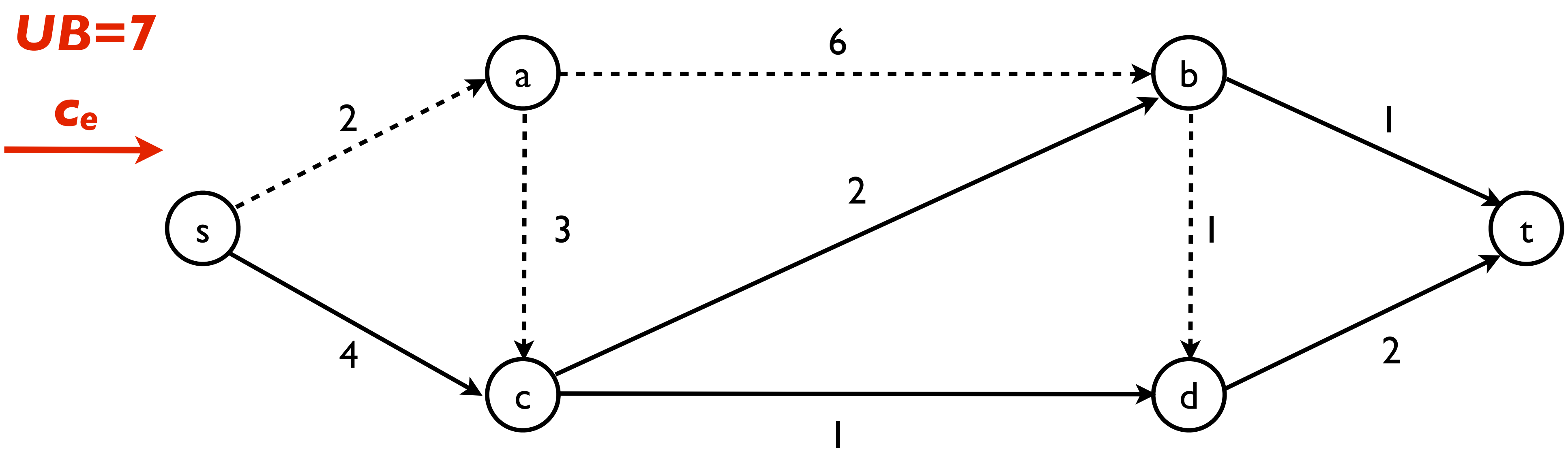


Cost-based Preprocessing



Cost-based Preprocessing

if $LB(c(P^*_{s \xrightarrow{e} t})) \geq UB$ **then** remove arc e
where $P^*_{s \xrightarrow{e} t}$ is a shortest path from s to t via arc e.



Outline

1. Introduction
2. Non Linear Costs
3. Iterated Preprocessing
4. Lower bounding via Lagrangian Relaxation
5. Computational Results

Resource Constrained Shortest Paths (RCSP)

Arc-flow IP formulation with non linear costs $f(\cdot)_h$:

$$\min \quad \sum_{e \in A} w_e x_e + \sum_{k \in K} \sum_{h \in H} f_h\left(\sum_{e \in A} r_e^k x_e \right)$$

$$\text{s.t.} \quad \sum_{e \in \delta_i^+} x_e - \sum_{e \in \delta_i^-} x_e = b_i = \begin{cases} +1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases} \qquad \forall i \in N$$

$$\sum_{e \in A} r_e^k x_e \leq U^k \qquad \forall k \in K$$

$+$ side non linear constraints

$$x_e \in \{0, 1\} \qquad \forall e \in A.$$

Resource Constrained Shortest Paths (RCSP)

Arc-flow IP formulation with non linear costs $f(\cdot)$:

$$
\begin{aligned}
\min \quad & \sum_{e \in A} w_e x_e + f\Big(\sum_{e \in A} r_e^1 x_e\Big) \\
\text{s.t.} \quad & \sum_{e \in \delta_i^+} x_e - \sum_{e \in \delta_i^-} x_e = b_i = \begin{cases} +1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases} & \forall i \in N \\
& \sum_{e \in A} r_e^k x_e \leq U^k & \forall k \in K \\
& x_e \in \{0, 1\} & \forall e \in A.
\end{aligned}
$$

[G. Tsaggouris and C. Zaroliagis, ESA2004]

Resource Constrained Shortest Paths (RCSP)

Arc-flow IP formulation with non linear costs $f(\cdot)$:

$$
\begin{aligned}
\min \quad & \sum_{e \in A} w_e x_e + f\left(z\right) \\
\text{s.t.} \quad & \sum_{e \in \delta_i^+} x_e - \sum_{e \in \delta_i^-} x_e = b_i = \begin{cases} +1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases} && \forall i \in N \\
& \sum_{e \in A} r_e^k x_e \leq U^k && \forall k \in K \\
& \sum_{e \in A} r_e^1 x_e = z \\
& x_e \in \{0, 1\} && \forall e \in A.
\end{aligned}
$$

Lower Bounding: Lagrangian Relaxation

The arc-flow LP relaxation of RCSP with a super additive cost function $f(\cdot)$ is:

$$\min \quad \sum_{e \in A} w_e x_e + f(z) \tag{8}$$

$$\text{s.t.} \quad \sum_{e \in \delta_i^+} x_e - \sum_{e \in \delta_i^-} x_e = b_i \qquad \forall i \in N \tag{9}$$

$$\text{multiplier } \alpha_k \leq 0 \quad \rightarrow \quad \sum_{e \in A} r_e^k x_e \leq U^k \qquad \forall k \in K \tag{10}$$

$$\text{multiplier } \beta \leq 0 \quad \rightarrow \quad \sum_{e \in A} t_e x_e \leq z \tag{11}$$

$$x_e \geq 0 \qquad \forall e \in A. \tag{12}$$

Lower Bounding: Lagrangian Relaxation

It is possible to formulate the following Lagrangian dual:

$$\Phi(\alpha, \beta) = -\sum_{k \in K} \alpha_k U^k +$$

$$+ \min \sum_{e \in A} \left(w_e + \sum_{k \in K} \alpha_k r_e^k + \beta t_e \right) x_e + f(z) - \beta z$$

$$\text{s.t.} \quad \sum_{e \in \delta_i^+} x_e - \sum_{e \in \delta_i^-} x_e = b_i \qquad \forall i \in N$$

$$x_e \geq 0 \qquad \forall e \in A.$$

This problem decomposes into two subproblems and is solved via a **subgradient optimization algorithm**:

1. The x variables define a *shortest path problem*
2. The z variable defines an *unconstrained optimization problem*

Cost-based Preprocessing via Lagrangian Lower Bounds

if $LB(c(P^*_{s \xrightarrow{e} t})) \geq UB$ **then** remove arc e
where $P^*_{s \xrightarrow{e} t}$ is a shortest path from s to t via arc e.

$$c(P^*_{s \xrightarrow{e} t}) \geq \bar{w}(P^*_{s \xrightarrow{e} t}) + \min\{f(z) - \bar{\beta} z\}$$

[with reduced costs $\bar{w}_e = w_e + \sum_{k \in K} \bar{\alpha}_k r_e^k + \bar{\beta} t_e$]

Filter and Dive

Algorithm 1: $\textsc{FilterAndDive}(G, LB, UB, F^g, B^g, U^g)$

Input: $G = (N, A)$ directed graph and distance function $g(\cdot)$
Input: (LB, UB) lower and upper bounds on the optimal path
Input: F^g, B^g forward and backward shortest path tree as function of $g(\cdot)$
Input: U^g upper bound on the path length as function of $g(\cdot)$
Output: An optimum path, or updated UB, or a reduced graph

foreach $i \in N$ **do**
    if $F_i^g + B_i^g > U^g$ **then**
        $N \leftarrow N \setminus \{i\}$
    else
        foreach $e = (i, j) \in A$ **do**
            if $F_i^g + g(e) + B_j^g > U^g$ **then**
                $A \leftarrow A \setminus \{e\}$
            else
                if $\textsc{PathCost}(F_i^g, e, B_j^g) < UB \wedge \textsc{PathFeasible}(F_i^g, e, B_j^g)$ **then**
                    $P_{st}^* \leftarrow \textsc{MakePath}(F_i^g, e, B_j^g)$;
                    Update UB and store P_{st}^*;
                    if $LB \geq UB$ **then**
                        return P_{st}^* (that is an optimum path)
                    else
                        $A \leftarrow A \setminus \{e\}$

FILTER...

...DIVE

check for side constraints

Near Shortest Path Enumeration

After reaching a fixpoint, if $LB < UB$ then, we apply a **near shortest path** enumeration algorithm (Carlyle et al., 2008).

We compute shortest reversed distances for every resource and for reduced costs

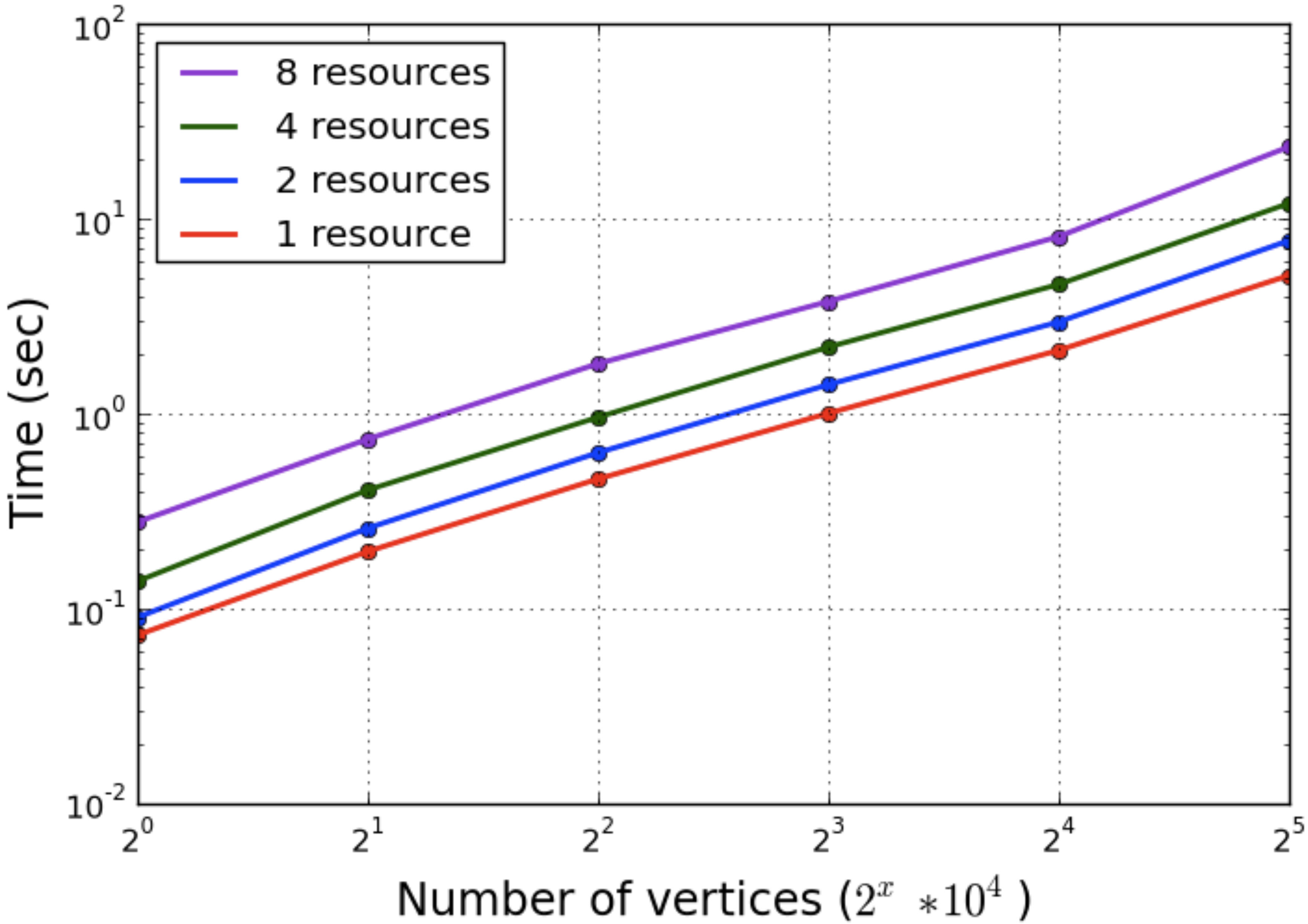
Then we perform a depth-first search from s. When a vertex i is visited, the algorithm backtracks if

1. for any resource k, the consumption of P_{si} plus the reversed (resource) distance to t exceeds U^k
2. the reduced cost of P_{si} plus the reversed (reduced cost) distance to t exceeds UB
3. the cost $c(P_{si}) \geq UB$

Outline

1. Introduction
2. Non Linear Costs
3. Iterated Preprocessing
4. Lower bounding via Lagrangian Relaxation
5. **Computational Results**

Constrained Path Solver: Scalability



Resource and Cost-based Preprocessing

Non linear costs: **extra allowances**

Each row gives the averages over 16 instances, with 7 resources.

- Δ is percentage of removed arcs
- Gap is $\frac{UB - Opt}{Opt} \times 100$

Graphs		Resource		Reduced Cost			Exact
n	m	Time	Δ	Time	Δ	Gap	Time
4137	135506	0.77	22.5%	3.12	30.2%	0.0%	75.1
2835	132468	0.59	40.3%	2.35	45.4%	0.0%	30.6
3792	134701	0.92	30.2%	2.87	37.4%	0.0%	69.3

Crew Scheduling: Real Life Instances

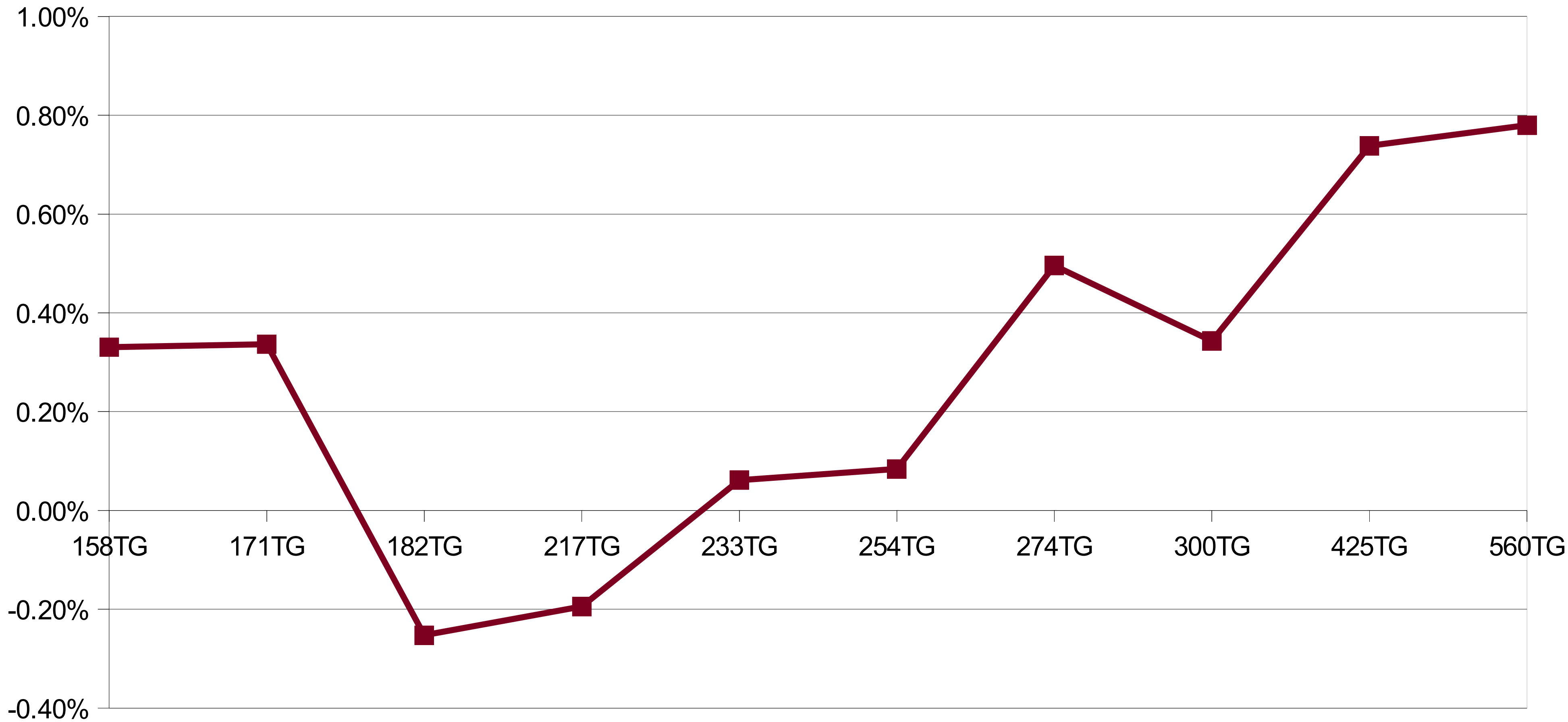
Instance	Pieces	Glob. Const.	Depots
158TG	684	4	3
171TG	802	6	6
182TG	846	7	7
217TG	967	8	8
233TG	1067	8	8
254TG	1169	10	10
274TG	1240	11	11
300TG	1369	12	12
425TG	1865	32	16
560TG	2314	21	10

Non linear component: step-wise on single resource

Side constraints: non linear constraint on break distribution

M.A.I.O.R.

Impact on Column Generation-based Heuristic

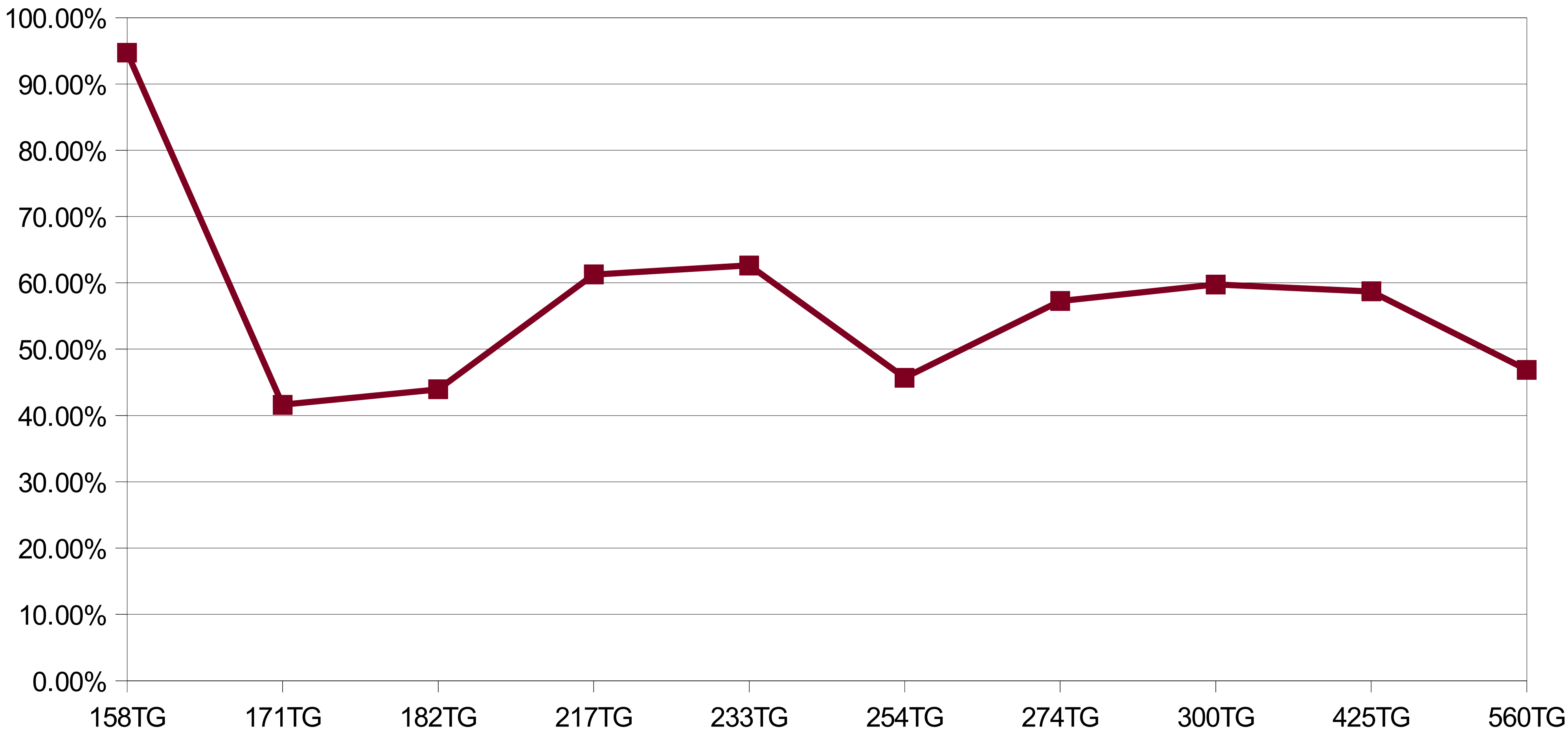


Labeling-heuristic vs. Exact New Algorithm

Difference of **cost solution** obtained via column generation

M.A.I.O.R.

Impact on Column Generation-based Heuristic



Labeling-heuristic vs. Exact New Algorithm

Difference of **run time** obtained via column generation

M.A.I.O.R.