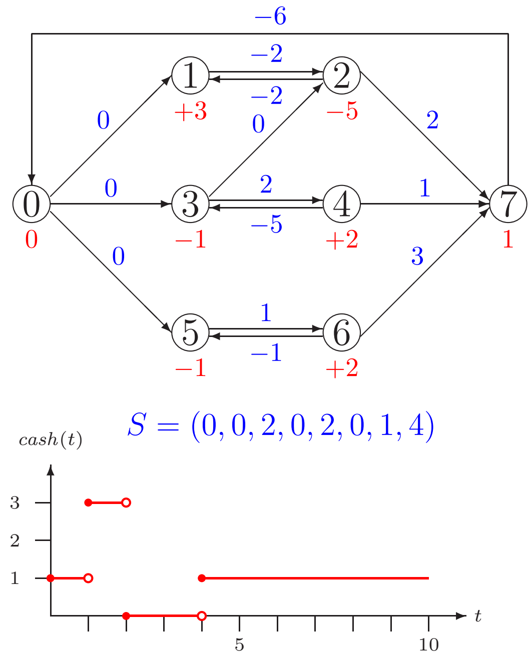
A Branch-and-Bound Algorithm for the Capital-Rationed Net Present Value Problem

Outline

1. Problem definition
2. Two IP formulations
3. Solution as inventory-constrained scheduling problem
4. Computational experience
5. Conclusions

1 Problem definition

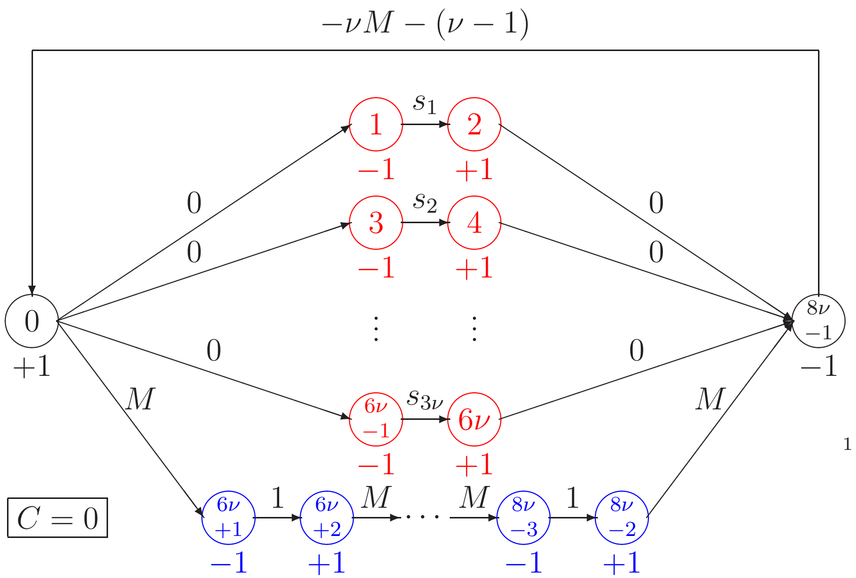
- $V = \{0, 1, \ldots, n, n+1\}$: Set of events
- $\delta_{ij} \in \mathbb{Z}$: Minimum time lag between i and j (if < 0, maximum time lag between j and i)
- $c_i^f \in \mathbb{Z}$: Cash flow associated with i
- $0 < \beta < 1$: Discount rate $-\ln \beta$
- $C \in \mathbb{Z}$: Minimum cash position
- $S_i \in \mathbb{R}_{\geq 0}$: Occurrence time of event $i \in V$



$$
(CNPV)\left\{
\begin{array}{lllr}
\text{Maximize} & \sum_{i \in V} c_i^f \beta^{S_i} & & (1) \\
\text{subject to} & S_j - S_i \geq \delta_{ij} & ((i,j) \in E) & (2) \\
 & S_0 = 0 & & (3) \\
 & \sum_{i \in V: S_i \leq t} c_i^f \geq C & (0 \leq t \leq \bar{d}) & (4)
\end{array}
\right.
$$

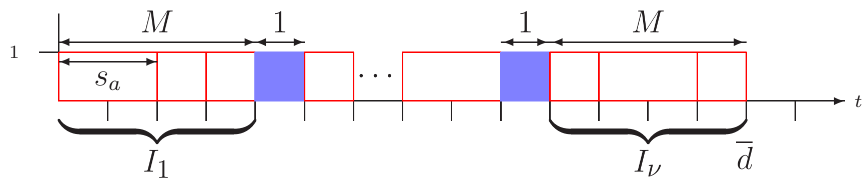
Feasibility problem NP-complete: Transformation from 3-PARTITION

- **Given**: Set $I = \{1, \ldots, 3\nu\}$ of indices a with sizes $s_a \in \mathbb{N}$ and bound $M \in \mathbb{N}$ such that
  - ▷ $\sum_{a \in I} s_a = \nu M$
  - ▷ $M/4 < s_a < M/2$ for all $a \in I$
- **Question**: Does there exist a partition $\{I_1, \ldots, I_\nu\}$ of I such that
  - ▷ $\sum_{a \in I_\mu} s_a = M$ for all $\mu = 1, \ldots, \nu$?



At any time, cash position is no greater than 1
Between S_i and S_{i+1} cash position equals 0 ($i = 1, 3, \ldots, 8\nu - 3$)
No event between i and $i + 1$ ($i = 1, 3, \ldots, 8\nu - 3$)
Time of occurrence of events $6\nu + 1, \ldots, 8\nu - 2$ fixed in advance
Within time M three pairs of events i and $i + 1$ must occur
($i = 1, 3, \ldots, 6\nu - 1$)

Feasible schedule:



2 Two IP formulations

Binary program by Doersch and Patterson (1977)

- $x_{it} \in \{0, 1\}$: 1 iff $S_i = t \in \mathbb{Z}_{\geq 0}$
- ES_i, LS_i: earliest and latest occurrence times of event $i \in V$

$$
(DP)\left\{
\begin{array}{llr}
\text{Maximize } \sum_{i \in V} \sum_{t=ES_i}^{LS_i} c_i^f \beta^t x_{it} & & (5) \\
\text{subject to } \sum_{t=ES_i}^{LS_i} x_{it} = 1 & (i \in V) & (6) \\
\sum_{t=ES_j}^{LS_j} t x_{jt} - \sum_{t=ES_i}^{LS_i} t x_{it} \geq \delta_{ij} & ((i,j) \in E) & (7) \\
\sum_{i \in V} \sum_{\tau=ES_i}^{\min(t, LS_i)} c_i^f x_{i\tau} \geq C & (t = 0, \ldots, \bar{d}) & (8) \\
x_{it} \in \{0, 1\} & (i \in V,\ t = ES_i, \ldots, LS_i) & (9)
\end{array}
\right.
$$

Order-theoretic mixed-binary program

- $y_i = \beta^{S_i} \geq 0$: Linearization of objective function by Grinold (1972)
- $z_{ij} \in \{0,1\}$: 1 iff $S_i \leq S_j$, i.e. $y_j \leq y_i$ (defines reflexive weak order in set V)
- $\varepsilon_i = \beta^{LS_i}(1-\beta)$

$$
(WO) \left\{
\begin{array}{lllr}
\text{Maximize} & \sum_{i \in V} c_i^f y_i & & (10) \\
\text{subject to} & y_j - \beta^{\delta_{ij}} y_i \leq 0 & ((i,j) \in E) & (11) \\
& y_0 = 1 & & (12) \\
& \varepsilon_j \leq y_j - y_i + z_{ij} \leq 1 & ((i,j) \in V \times V) & (13) \\
& \sum_{j \in V} c_j^f z_{ji} \geq C & (i \in V) & (14) \\
& y_i \geq 0 & (i \in V) & (15) \\
& z_{ij} \in \{0,1\} & ((i,j) \in V \times V) & (16)
\end{array}
\right.
$$

3 Solution as inventory-constrained scheduling problem

- Cumulative resource (Neumann and S., 1999): storage with safety stock and capacity
- Depleting events: starts of operations; replenishing events: completions of operations
- Schedule events such that inventory is constantly between safety stock and capacity
- Cash position: storage with safety stock C and infinite capacity
- Negative cash flows: depleting events; positive cash flows: replenishing events
- Shortage set: $F \subseteq V$ with $\sum_{i \in F} c_i^f < C$

Lemma. A schedule S satisfies cash position constraints (4) iff

▷ for each shortage set F,

▷ there exist two events $j \in F$ and $i \notin F$ with $c_j^f < 0$ and $c_i^f > 0$

▷ such that $S_j \geq S_i$.

Minimal delaying alternatives and minimal delaying modes

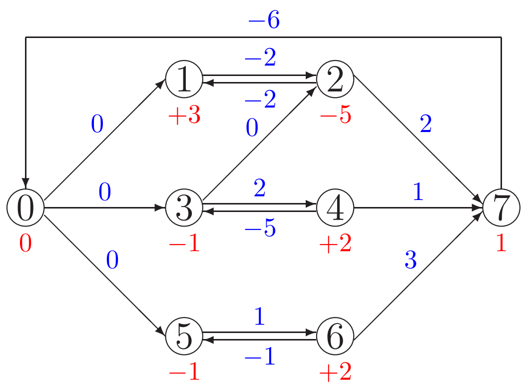
- Given shortage set F
- Minimal delaying alternative for F: $\subseteq$-minimal set $B \subseteq F$ with $\sum_{j \in F \setminus B} c_j^f \geq C$
- Minimal delaying mode for F: pair (i, B) of event $i \notin F$, $c_i^f > 0$ and minimal delaying alternative B for F

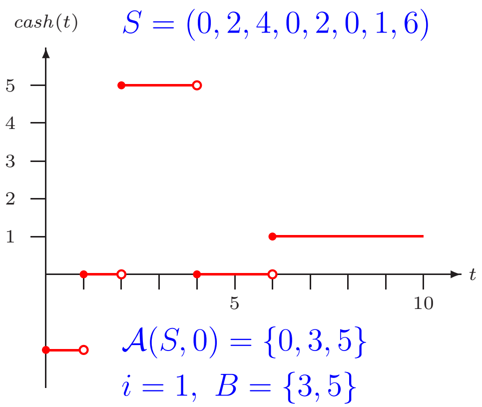
Lemma. Minimal delaying alternative B for shortage set F is $\subseteq$-minimal set containing an event j with $c_j^f < 0$ of each shortage set F' satisfying $\{i \in F' \mid c_i^f < 0\} \subseteq \{i \in F \mid c_i^f < 0\}$ and $\{i \in F' \mid c_i^f > 0\} \supseteq \{i \in F \mid c_i^f > 0\}$.

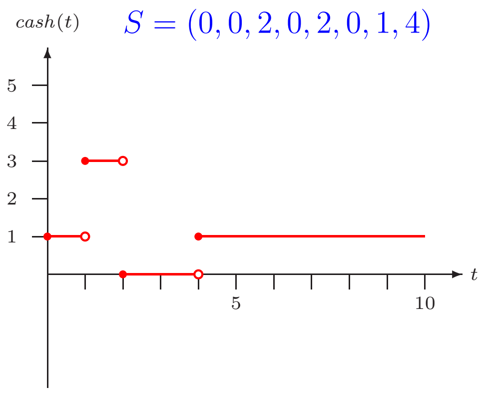
Theorem. Given shortage set F. For each feasible schedule S, there is a minimal delaying mode (i, B) for F with $S_j \geq S_i$ for all $j \in B$.

Branch-and-bound algorithm

- Disregard cash position constraints (4): Resource relaxation
- Enumeration node u: Resource relaxation on (expanded) project network
- Solve resource relaxation: schedule S
- Net present value of S is upper bound on net present values in subtree rooted at u
- Determine active shortage set $\mathcal{A}(S,t) := \{i \in V \mid S_i \leq t\}$ at some time t
- Introduce child node v for each minimal delaying mode (i,B) for $\mathcal{A}(S,t)$
- For each child node v: add arcs from $\{i\} \times B$ weighted with 0 to project network



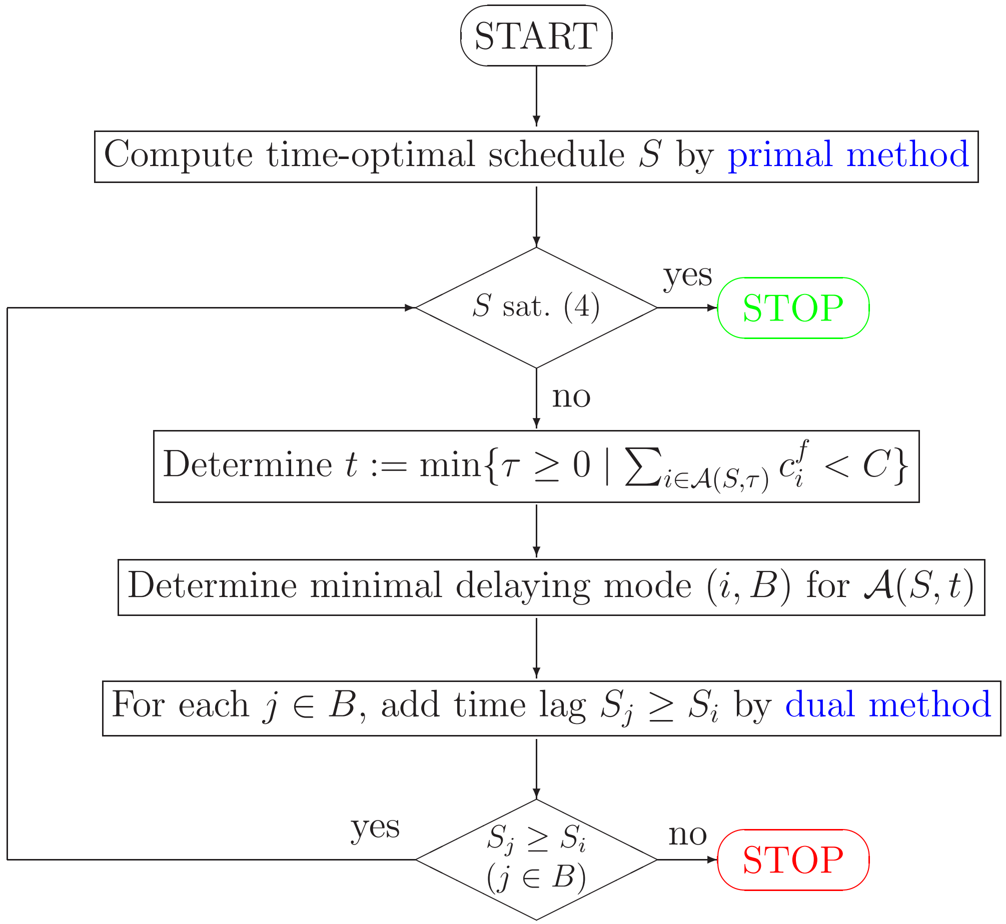




Solving the resource relaxation

- **Primal method**
  - Objective function convexifiable
  - First-order steepest-ascent algorithm iterating time-feasible schedules
  - Ascent directions normalized by maximum norm
  - Direction finding phase performed in $O(n)$ time
- **Dual method**
  - First-order flattest-descent algorithm
  - Flattest-descent directions increase $S_j - S_i$ for all $j \in B$
  - Direction finding problem decomposes into two independent subproblems
  - Subproblems can be solved in $O(n)$ time

Schedule-generation scheme



4 Computational experience

- ProGen/max test set: 630 instances with $n = 10, 20, 50, 100, 200, 500, 1000$ events
- $OS \in \{0.25, 0.5, 0.75\}$, $c_i^f \in \{-10, -9, \ldots, 9, 10\}$, $\overline{d} = 2.0ES_{n+1}$, $\beta = 0.99$
- $RS = 0.0$, i.e. $C = \min(0, \sum_{i\in V} c_i^f)$
- IP's solved by CPLEX 6.0, branch-and-bound coded in ANSI C
- Pentium PC with 333 MHz and 128 MB RAM, time limit: n seconds

	IP Doersch&Patterson				*MIP Weak order*				*Cumulative resources*			
	p_{opt}	p_{ins}	p_{feas}	p_{unk}	p_{opt}	p_{ins}	p_{feas}	p_{unk}	p_{opt}	p_{ins}	p_{feas}	p_{unk}
$n = 10$	72.2	2.2	4.4	21.1	77.8	22.2	0.0	0.0	77.8	22.2	0.0	0.0
$n = 20$	20.0	3.3	28.9	47.8	62.2	21.1	2.2	14.4	64.4	35.6	0.0	0.0
$n = 50$	0.0	0.0	3.3	96.7	11.1	3.3	20.0	65.6	66.7	21.1	2.2	10.0
$n = 100$	0.0	0.0	0.0	100.0	0.0	0.0	0.0	100.0	68.9	10.0	2.2	18.9
$n = 200$									61.1	6.7	4.4	27.8
$n = 500$									64.4	1.1	7.8	26.7
$n = 1000$									70.0	0.0	2.2	27.8

5 Conclusions

- Capital-rationed net present value problem
- Feasibility problem NP-complete
- Two IP formulations
  - Time-indexed model by Doersch and Patterson
  - Order-theoretic model of polynomial size
- Formulation as inventory-constrained scheduling problem
- Relax inventory constraints
- Solve relaxations by efficient feasible-directions methods
- Enumerate sets of precedence constraints between replenishing and depleting events
- Branch-and-bound performs well on standard test set