A Comparison of Relaxation-Based Enumeration Schemes in Production Scheduling

Ch. Schwindt, Clausthal University of Technology

supported by German Research Foundation DFG

Outline

1. Production scheduling problem
2. Generic scheduling model
3. Relaxation-based enumeration schemes
4. Avoiding redundancy
5. Performance analysis
6. Conclusions

1 Production scheduling problem

Operations

- Processing of production order (job) on machine
- Execution of chemical process (task) on processing unit
- Performance of activity in project using personnel and equipment

Temporal relationships

- Precedence constraints arising from process plans or recipes
- Release dates, deadlines
- Quarantine times, shelf life times

Resources

- Machinery, tools, manpower
- Storage facilities, intermediate products

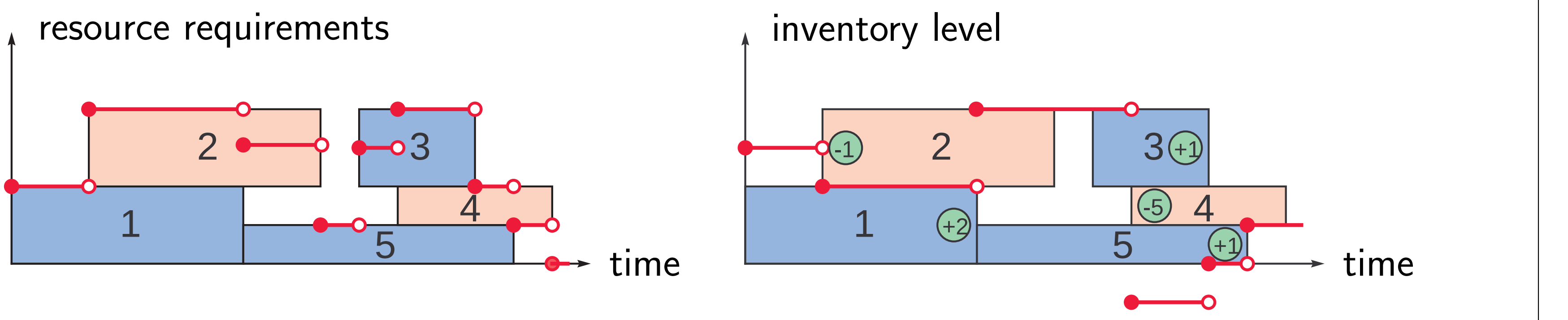
Problem: Determine production schedule (assignment of start times to operations) complying with temporal relationships and resource constraints

2 Generic scheduling model

Resource-constrained scheduling model

- Operations i with processing times p_i, including production start $i = 0$
- Temporal relationships: minimum and maximum time lags d_{ij}^{min} and d_{ij}^{max} between start times of operations i, j

- Manpower, machinery: renewable resources k with capacities R_k and requirements r_{ik}
- Storage facilities, intermediate products: cumulative resources l with minimum and maximum inventory levels $\underline{R}_l$ and $\overline{R}_l$ and requirements r_{il} (r_{0l}: initial stock)



Reduction to generic model

Replace operations by events

- Split each operation $i \neq 0$ in start and completion events $e = s(i)$ and $f = c(i)$
- Define time lags δ_{ef} between events e and f
  - ▷ Fixed processing times p_i: $\delta_{ef} = p_i$, $\delta_{fe} = -p_i$ with $e = s(i)$ and $f = c(i)$
  - ▷ Minimum and maximum time lags d_{ij}^{min} and d_{ij}^{max}: $\delta_{ef} = d_{ij}^{min}$, $\delta_{fe} = -d_{ij}^{max}$ with $e = s(i)$ and $f = s(j)$

Replace renewable resources by cumulative resources

- Renewable resources k: transform into cumulative resources l with $\underline{R}_l = 0$, $\overline{R}_l = R_k$, $r_{el} = r_{ik}$ for $e = s(i)$ and $r_{fl} = -r_{ik}$ for $f = c(i)$

Eliminate maximum inventory levels, normalize minimum inventory levels

- Maximum inventory levels $\overline{R}_l$: introduce cumulative resources l' with inventory levels $\underline{R}_{l'} = -\overline{R}_l$, $\overline{R}_{l'} = \infty$ and requirements $r_{el'} = -r_{el}$, put $\overline{R}_l := \infty$
- Minimum inventory levels $\underline{R}_l$: put $r_{0l} := r_{0l} - \underline{R}_l$, $\underline{R}_l := 0$

Generic scheduling model

Notation

V	Set of events e
$E \subseteq V \times V$	Temporal relation
$N = (V, E, \delta)$, D	Event-on-node network, distance matrix
$\mathcal{R}$	Set of cumulative resources
S_e, $S = (S_e)_{e \in V}$	Occurrence time of event e, schedule
$r_l(S,t) = \sum_{e \in V : S_e \leq t} r_{el}$	Inventory level of resource l at time t given schedule S
$f(S)$	Objective function, e.g., $f(S) = \max_{e \in V} S_e$
$\mathcal{S}$	Set of feasible schedules (feasible region)

Problem statement (Beck 2002, Neumann and S. 2002, Laborie 2003)

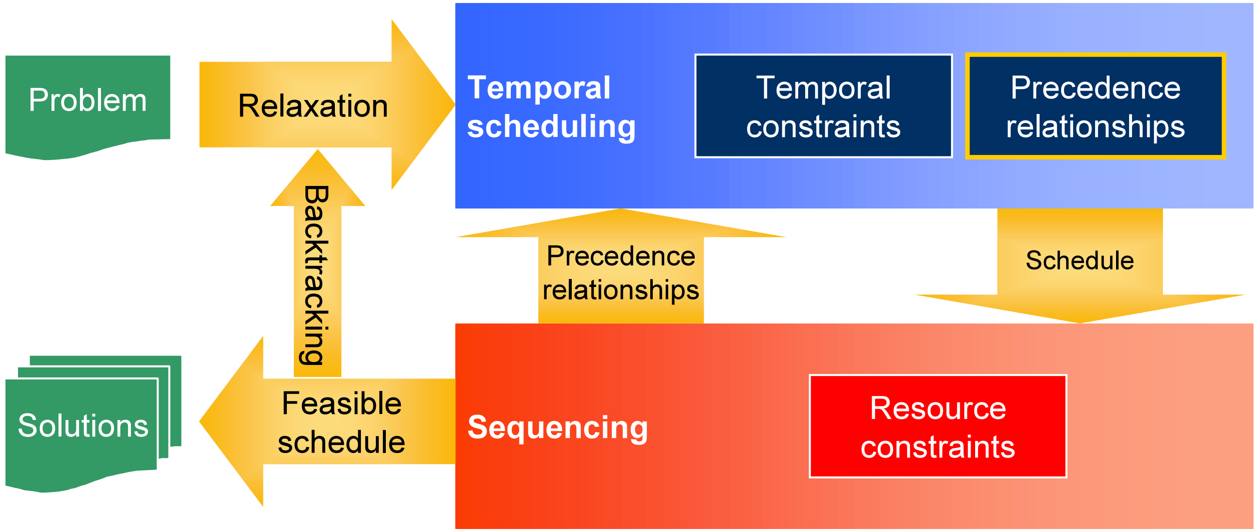
$$\left.\begin{array}{lll} \text{Minimize} & f(S) & \\ \text{subject to} & r_l(S,t) \geq 0 & (l \in \mathcal{R},\ t \geq 0) \\ & S_f - S_e \geq \delta_{ef} & ((e,f) \in E) \\ & S_0 = 0,\ S_e \geq 0 & (e \in V) \end{array}\right\} \quad \text{(PSP)}$$

3 Relaxation-based enumeration schemes

3.1 Basic scheme

Scheduling is (Bell and Park 1990) . . .

- defining precedence relationships between events competing for same resources (Sequencing: hard)
- optimizing objective function subject to prescribed time lags and established precedence relationships (Temporal scheduling: tractable)



3.2 Resolving inventory shortages

- Schedule $\hat{S}$ not resource-feasible: determine some time $t \geq 0$ with $r_l(\hat{S}, t) < 0$
- Determine set $A := \{e \in V \mid \hat{S}_e > t,\ r_{el} > 0\}$
- Compute minimal delaying alternatives B: $\subseteq$-minimal set of events f with $\hat{S}_f \leq t$ and $r_l(\hat{S}, t) - \sum_{f \in B} r_{fl} \geq 0$
- Add precedence relationships between sets A and B
  - ▷ Release dates: Fest et al. (1999)

$$S_f \geq \min_{e \in A} \hat{S}_e \qquad (f \in B)$$

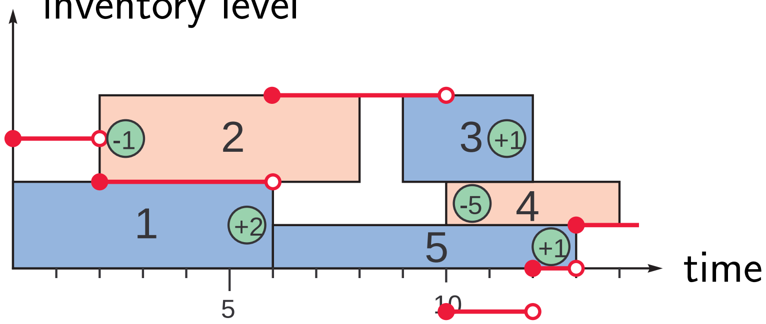
  - ▷ Ordinary precedence constraints (branch over all $e \in A$): De Reyck, Herroelen (1998)

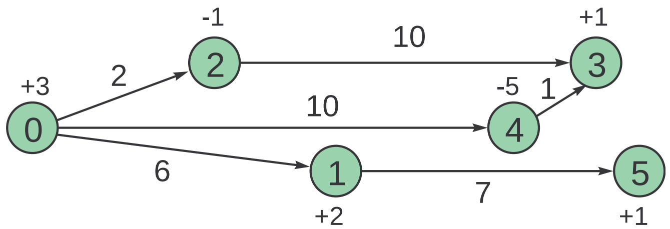
$$S_f \geq S_e \qquad (f \in B)$$

  - ▷ Disjunctive precedence constraints: Neumann et al. (2001)

$$\min_{f \in B} S_f \geq \min_{e \in A} S_e$$

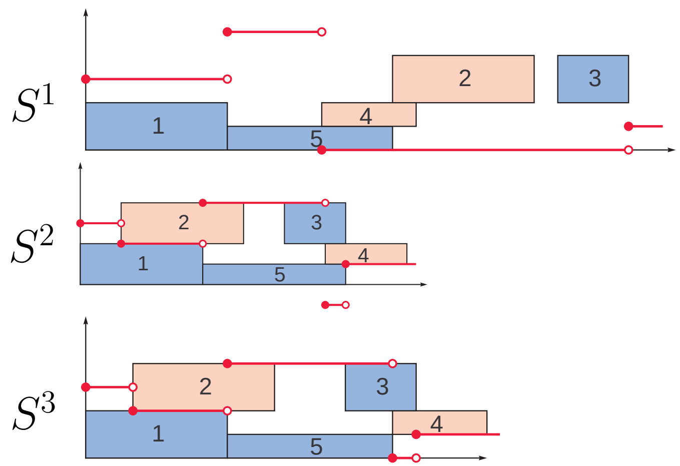
Example





Inventory shortage at time $t = 10$, $A = \{3, 5\}$, $B_1 = \{2\}$, $B_2 = \{4\}$, $\min_{e \in A} \hat{S}_e = 12$

RD	$S_2 \geq 12$	Schedule S^1
	$S_4 \geq 12$	Schedule S^2
OPC	$S_2 \geq S_5$	Schedule S^1
	$S_4 \geq S_5$	Schedule S^3
	$S_2 \geq S_3$	—
	$S_4 \geq S_3$	—
DPC	$S_2 \geq \min\{S_3, S_5\}$	Schedule S^1
	$S_4 \geq \min\{S_3, S_5\}$	Schedule S^3

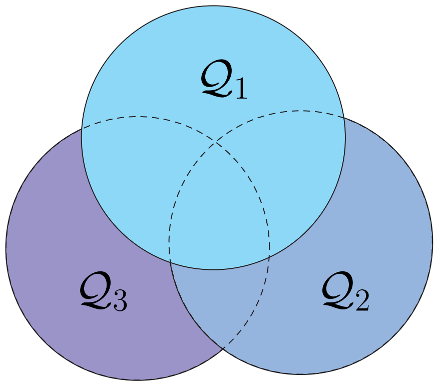


4 Avoiding redundancy

4.1 Partitioning the feasible region

- Consider enumeration node u with search space $\mathcal{Q}$
- Compute minimal delaying alternatives $B_1, \ldots, B_\nu$
- Define disjunctive precedence constraints $\min_{f \in B_\mu} S_f \geq \min_{e \in A} S_e$ belonging to sets

$$\mathcal{P}_\mu := \{S \in \mathcal{Q} \mid \min_{f \in B_\mu} S_f \geq \min_{e \in A} S_e\}$$



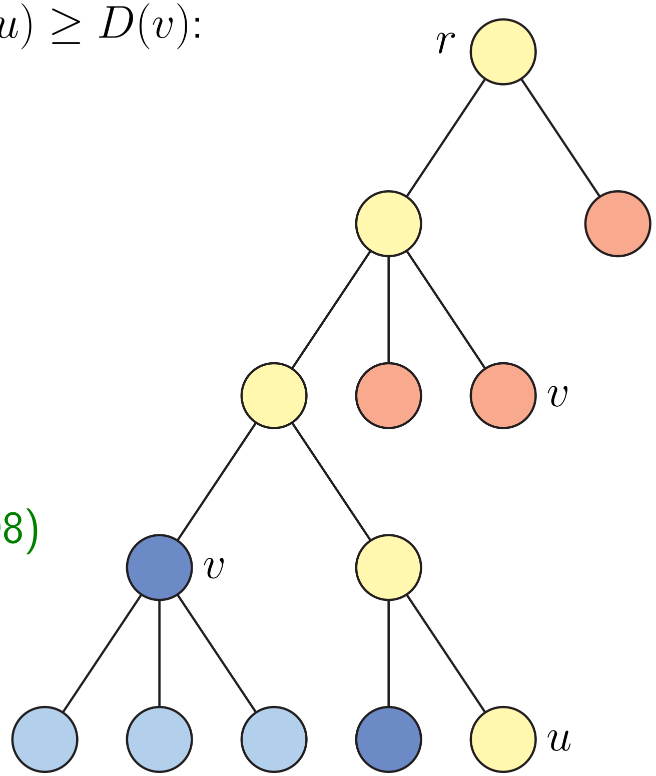
- Enumerate child nodes $v_1, \ldots, v_\nu$ with search spaces

$$\mathcal{Q}_\mu := \mathcal{P}_\mu \setminus [\cup_{\lambda=1}^{\mu-1} \mathcal{P}_\lambda]$$

- $\cup_{\mu=1}^{\nu}(\mathcal{Q}_\mu \cap \mathcal{S}) = \mathcal{Q} \cap \mathcal{S}$ and $\mathcal{Q}_\lambda \cap \mathcal{Q}_\mu = \emptyset$ for all $\lambda \neq \mu$
- Construction of sets $\mathcal{Q}_\mu$
  - ▷ Introduce disjunctive precedence constraint $\min_{f \in B_\mu} S_f \geq \min_{e \in A} S_e$ at node v_μ
  - ▷ Introduce reverse constraint $\min_{e \in A} S_e \geq \min_{f \in B_\mu} S_f + 1$ at all nodes $v_{\mu+1}, \ldots, v_\nu$

4.2 Generalized subset dominance

- Release dates, ordinary precedence constraints: time lags δ_{ef}
- Associate a distance matrix $D(u)$ with each enumeration node u
- Node u dominated by node v if $\mathcal{Q}(u) \subseteq \mathcal{Q}(v)$, i.e., $D(u) \geq D(v)$: Neumann, Zimmermann (2002)
- Perform depth-first search: enumeration nodes v
  - ▷ on active path from root r to active node u
  - ▷ bud nodes
  - ▷ fully explored (all descendant nodes explored)
- Generalized subset dominance rule: fathom node u if
  - ▷ there exists bud node v with $D(v) \leq D(u)$: S. (1998)
  - ▷ there exists fully explored node v with distance one from active path and $D(v) \leq D(u)$: De Reyck, Herroelen (1998)
- Each search space $\mathcal{Q}(u)$ explored only once



5 Performance analysis

Test bed

- Test set from literature with 90 instances comprising 50 events and 10 resources each
- Pentium IV PC with 1.8 GHz clock pulse and 512 MB RAM, time limit 10 seconds
- Branch-and-bound algorithms for makepan problem coded under MS Visual C++ 6.0
  - ▷ RD(-SSD): release dates (+ subset dominance)
  - ▷ OPC(-SSD): ordinary precedence constraints (+ subset dominance)
  - ▷ DPC(-PFR): disjunctive precedence constraints (+ partitioning of feasible region)

Computational results

	RD	RD-SSD	OPC	OPC-SSD	DPC	DPC-PFR
Number instances solved	71	79	74	79	87	90
Number of nodes explored	49045	15784	7383	1229	1110	204
CPU time total [ms]	2117	1304	2047	1622	413	254
CPU time first solution [ms]	2	1	140	81	8	59

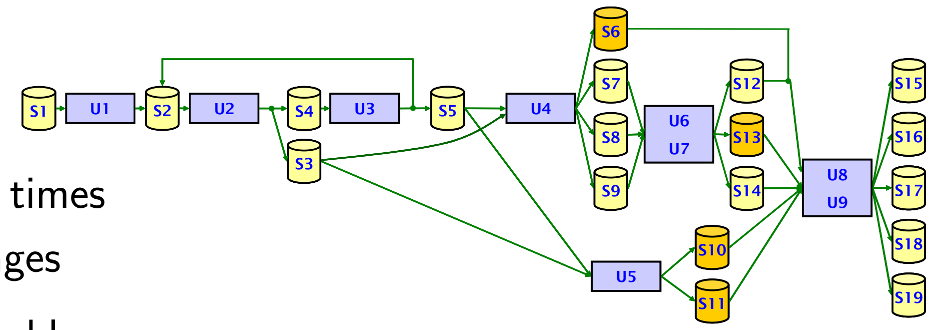
6 Conclusions

Summary

- Production scheduling problem
- Generic scheduling model with cumulative resources
- Different relaxation-based enumeration schemes
  - ▷ Release dates
  - ▷ Ordinary precedence constraints
  - ▷ Disjunctive precedence constraints
- Avoid redundancy by partitioning feasible region or subset dominance

Further research

- Integration of further constraints
  - ▷ Sequence-dependent changeover times
  - ▷ Multi-purpose intermediate storages
- Application to process scheduling problems



Cumulative resources

- Carlier J, Rinnooy Kan AHG (1982) Scheduling subject to nonrenewable resource constraints. Operations Research Letters 1: 52–55
- Beck JC (2002) Heuristics for scheduling with inventory: Dynamic focus via constraint criticality. Journal of Scheduling 5: 43–69
- Neumann K, Schwindt C (2002) Project scheduling with inventory constraints. Mathematical Methods of Operations Research 56: 513–533
- Laborie P (2003) Algorithms for propagating resource constraints in AI planning and scheduling: Existing approaches and new results. Artificial Intelligence 143: 151–188

Enumeration schemes

- Bell CE, Park K (1990) Solving resource-constrained project scheduling problems by A^* search. Naval Research Logistics 37: 61–84
- Fest A, Möhring RH, Stork F, Uetz M (1999) Resource-constrained project scheduling with time windows: A branching scheme based on dynamic release dates. Technical Report 596, Technical University of Berlin
- Franck B, Neumann K, Schwindt C (2001) Truncated branch-and-bound, schedule construction, and schedule-improvement procedures for resource-constrained project scheduling. OR Spektrum 23: 297-324

Redundancy avoidance

- De Reyck B, Herroelen WS (1998) A branch-and-bound procedure for the resource-constrained project scheduling problem with generalized precedence relations. European Journal of Operational Research 111: 152–174
- Schwindt C (1998) Verfahren zur Lösung des ressourcenbeschränkten Projektdauerminimierungsproblems. Shaker, Aachen
- Neumann K, Zimmermann J (2002) Exact and truncated branch-and-bound procedures for resource-constrained project scheduling with discounted cash flows and general temporal constraints. Central European Journal of Operations Research 10: 357–380