

Modelle und Methoden zur Lösung von präemptiven Projektplanungsproblemen mit verallgemeinerten Anordnungsbeziehungen

Tobias Paetz

Abteilung für BWL, insbesondere Produktion und Logistik
Technische Universität Clausthal

22.05.2014

16. Doktorandenworkshop Nordost, Festenburg



TU Clausthal
Clausthal University of Technology

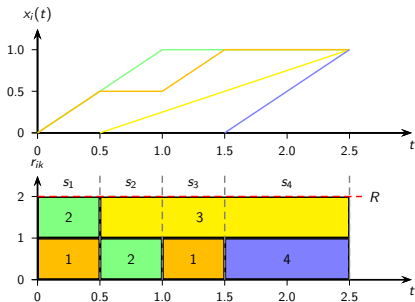
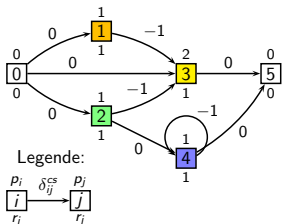
Agenda

- 1 Präemptives Projektplanungsproblem
 - Motivation
 - Literatur
 - Verallgemeinerte Anordnungsbeziehungen
 - Beschreibungsmodell
- 2 Strukturuntersuchung
 - Kanonische Form
 - Anzahl Zeitscheiben
- 3 Neue MILP-Formulierung
- 4 Preprocessing
 - Verschärfung von D^{CS}
 - Identifikation von unzulässigen Instanzen
 - Erweiterungen
- 5 Performanceanalyse
 - Nutzen der Unterbrechbarkeit
 - Experimentelle Ergebnisse
- 6 Ausblick

Motivation für die Arbeit

- Vielzahl von Maschinen-Scheduling-Problemen als Projektplanungsprobleme darstellbar
- **Maschinen-Scheduling: Unterbrechbarkeit** von Jobs/Operationen **intensiv behandelt** (siehe z. B. Baker 1974, French 1982, Błażewicz et al. 2007, Pinedo 2008, Brucker 2009, ...)
- **Projektplanung: Unterbrechbarkeit** von Aktivitäten bisher **kaum untersucht**
- Vielfältige **Anwendungsbereiche**:
 - Aggregierte Projektplanung
 - Smart Grid Scheduling
 - Termin- und Kapazitätsplanung
 - Multi-Processor Task Scheduling
 - ...

Einführendes Beispiel und Notation



Notation:

$V = \{1, \dots, n\}$ Menge (realer) Aktivitäten i
 \mathcal{R} Menge Ressourcen k
 R_k Kapazität von Ressource k
 r_{ik} Bedarf von Aktivität i an Ressource k

E Menge Kanten (i, j)
 p_i Dauer von Aktivität i
 δ_{ij}^{cs} Ende-Start Zeitabstand
 $x_i(t)$ Ausführungsanteil von i zum Zeitpunkt t

Literaturauswahl



Słowiński R (1980)

Two approaches to problems of resource allocation among project activities: A comparative study

The Journal of the Operational Research Society 31:711–723



Demeulemeester E, Herroelen W (1996)

An efficient optimal solution procedure for the preemptive resource-constrained project scheduling problem

European Journal of Operational Research 90:334–348



Damay J, Quilliot A, Sanlaville E (2007)

Linear programming based algorithms for preemptive and non-preemptive RCPSP

European Journal of Operational Research 182:1012–1022



Ballestin F, Valls V, Quintanilla S (2008)

Pre-emption in resource-constrained project scheduling

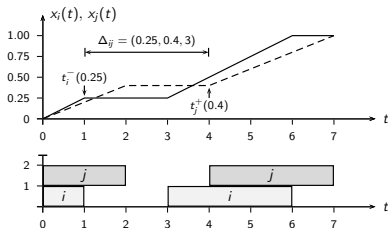
European Journal of Operational Research 189:1136–1152

Verallgemeinerte Anordnungsbeziehungen

Formulierung von $\Delta_{ij} = (\xi_i, \xi_j, \delta_{ij})$ mit $0 < \xi_i \leq 1$ und $0 \leq \xi_j < 1$

$$\underbrace{\sup\{t \mid x_j(t) \leq \xi_j\}}_{t_j^+(\xi_j)} \geq \underbrace{\min\{t \mid x_i(t) \geq \xi_i\}}_{t_i^-(\xi_i)} + \delta_{ij}$$

- $S_i = t_i^+(0)$ und $C_i = t_i^-(1)$



Semantische Mächtigkeit der verallgemeinerten Anordnungsbeziehungen

Aufgrund der generischen Formulierung lassen sich alle Fälle des klassischen RCPSP darstellen ...

- Herkömmliche Anordnungsbeziehungen: $S_j \geq C_i \Rightarrow \Delta_{ij} = (1, 0, 0)$
- Ende-Start Mindestabstand: $S_j \geq C_i + \delta_{ij} \Rightarrow \Delta_{ij} = (1, 0, \delta_{ij})$
- Nicht-Unterbrechbarkeit von Aktivität i mit p_i : $\Delta_{ii} = (1, 0, -p_i)$
- Höchstabstand: $\delta_{ij} < 0$

Mindestabstand δ_{ij} zwischen i und j $\Delta_{ij} = (1, 0, \delta_{ij})$

$$\hat{=} t_j^+(\xi_j) \geq t_i^-(\xi_i) + \delta_{ij}$$

$$\Leftrightarrow t_i^-(\xi_i) \leq t_j^+(\xi_j) - \delta_{ij}$$

$$\hat{=} \text{Höchstabstand } -\delta_{ij} \text{ zwischen } j \text{ und } i \quad \Delta_{ji} = (1, 0, -\delta_{ij})$$

Beschreibungsmodell

Definiere binäre Funktion

$$y_i(t) := p_i \frac{d^+ x_i}{dt}(t) = \begin{cases} 1, & \text{wenn } i \text{ zum Zeitpunkt } t \text{ ausgeführt wird} \\ 0, & \text{sonst} \end{cases}$$

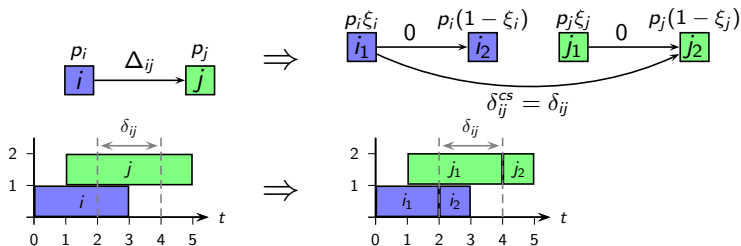
Modell für Problem $PS|pmtn, temp|C_{\max}$

$$(P) \quad \left\{ \begin{array}{l} \text{Minimiere} \quad C_{\max} \\ \text{u.d.N.} \quad C_{\max} \geq t_i^-(1) \quad (i \in V) \\ \quad \quad \quad \sum_{i \in V} r_{ik} y_i(t) \leq R_k \quad (k \in \mathcal{R}; t \geq 0) \\ \quad \quad \quad t_j^+(\xi_j) \geq t_i^-(\xi_i) + \delta_{ij} \quad ((i, j) \in E) \end{array} \right.$$

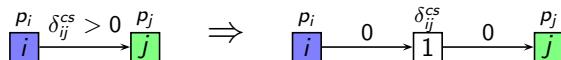
Kanonische Form des Problems

O. B. d. A. nehmen wir an, dass ...

- ... alle verallg. Anordnungsbeziehungen $\Delta_{ij} = (\xi_i, \xi_j, \delta_{ij})$ als **Ende-Start-Zeitabstände** $\Delta'_{ij} = (1, 0, \delta_{ij}^{cs})$ definiert werden



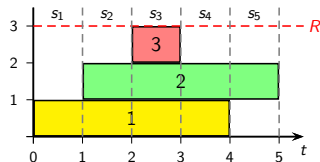
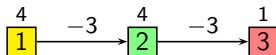
- ... alle Ende-Start-Zeitabstände δ_{ij}^{cs} **nichtpositiv** sind



Anzahl der Zeitscheiben

Satz

- 1 Für jede zulässige Instanz des Problems $PS|pmtn, prec|C_{\max}$ existiert ein optimaler Schedule x mit höchstens n Zeitscheiben, die positive Dauer aufweisen.
- 2 Für jede zulässige Instanz des Problems $PS|pmtn, temp|C_{\max}$ in kanonischer Form existiert ein optimaler Schedule x mit höchstens $2n - 1$ Zeitscheiben, die positive Dauer aufweisen.



Neue MILP-Formulierung

MILP-Modell für $PS|pmtn, temp|C_{\max}$ - Problem

$$\begin{array}{l}
 (MILP) \quad \left\{ \begin{array}{ll}
 \text{Minimiere} & C_{\max} = \sum_{\mu=1}^{\nu} z_{\mu} \\
 \text{u.d.N.} & 0 \leq z_{\mu} - p_i \Delta x_{i\mu} \leq p^{\max} \cdot (1 - y_{i\mu}) \quad (i \in V; \mu = 1, \dots, \nu) \\
 & 0 \leq \Delta x_{i\mu} \leq y_{i\mu} \quad (i \in V; \mu = 1, \dots, \nu) \\
 & \sum_{\mu=1}^{\nu} \Delta x_{i\mu} = 1 \quad (i \in V) \\
 & \sum_{i \in V} r_{ik} \cdot y_{i\mu} \leq R_k \quad (k \in \mathcal{R}; \mu = 1, \dots, \nu) \\
 & S_i \leq \sum_{\lambda=1}^{\mu-1} z_{\lambda} + UB \cdot (1 - y_{i\mu}) \quad (i \in V; \mu = 1, \dots, \nu) \\
 & C_i \geq \sum_{\lambda=1}^{\mu} z_{\lambda} - UB \cdot (1 - y_{i\mu}) \quad (i \in V; \mu = 1, \dots, \nu) \\
 & S_j \geq C_i + \delta_{ij}^{CS} \quad ((i, j) \in E) \\
 & y_{i\mu} \in \{0, 1\} \quad (i \in V; \mu = 1, \dots, \nu)
 \end{array} \right.
 \end{array}$$

- $y_{i\mu} = 1$: Aktivität i in Zeitscheibe μ ausgeführt
- z_{μ} : Dauer der Zeitscheibe μ
- S_j : untere Schranke für Startzeitpunkt der Aktivität j
- $\Delta x_{i\mu}$: Zuwachs des Ausführungsanteils von Aktivität i in Zeitscheibe μ ($= \frac{z_{\mu}}{p_i} \cdot y_{i\mu}$)
- C_j : obere Schranke für Endzeitpunkt der Aktivität j

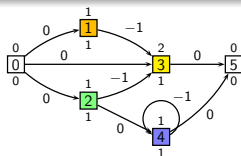
Preprocessing

- 1 Berechnung und Verschärfung der Distanzmatrix D^{CS}
- 2 Identifizierung von unzulässigen Instanzen
- 3 Weitere

Distanzmatrix D^{CS}

- Sei $d_{ij}^{CS} \in \mathbb{Z} \cup -\infty$ der transitive Ende-Start-Zeitabstand zwischen i und j induziert durch die Zeitabstände der Menge E
- d_{ij}^{CS} entspricht der minimalen Differenz $S_j - C_i$ in jedem zeit-zulässigen Schedule
- d_{ij}^{CS} lässt sich durch eine Adaption des Floyd-Warshall-Algorithmus als die längsten Wege im Projektnetzwerk berechnen
- Für alle Aktivitätenpaare $(i, j) \in V$ bildet d_{ij}^{CS} die Distanzmatrix D^{CS} :

$$D^{CS} = (d_{ij}^{CS})_{i,j \in V}$$



$$\Rightarrow D^{CS} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 2 \\ -\infty & -\infty & -\infty & -1 & -\infty & 1 \\ -\infty & -\infty & -\infty & -1 & 0 & 1 \\ -\infty & -\infty & -\infty & -\infty & -\infty & 0 \\ -\infty & -\infty & -\infty & -\infty & -1 & 0 \\ -\infty & -\infty & -\infty & -\infty & -\infty & 0 \end{pmatrix}$$

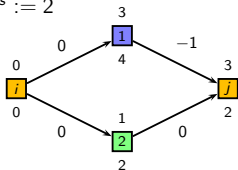
Verschärfung der Distanzmatrix I

Verschärfung von D^{CS} durch Berücksichtigung der Ressourcenbeschränkungen in Zeitabständen

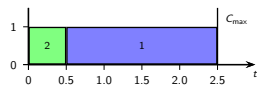
Ein-Maschinen-Preprocessing

- Bestimme **minimale Dauer** von Aktivitäten $h \in V$, die zwischen Ende von i und Beginn von j ausgeführt werden
- Löse für jede Ressource $k \in \mathcal{R}$ **Ein-Maschinen-Problem** $1|r_h, q_h, pmtn|C_{\max}$ mit
 - Release Date $r_h := d_{ih}^{CS}$
 - Nachlaufzeit $q_h := d_{hj}^{CS}$
 - Skalierter Ausführungsdauer $p_h := \frac{r_{hk}}{R_k} \cdot \min\{p_h, d_{ij}^{CS}, d_{ih}^{CS} + p_h, d_{hj}^{CS} + p_h\}$
- Setze $d_{ij}^{CS} := \max\{d_{ij}^{CS}, C_{\max}^*(i, j, k)\}$

$$d_{ij}^{CS} := 2$$



h	1	2
r_h	0	0
q_h	-1	0
p_h	$\frac{4}{4} \cdot 2 = 2$	$\frac{2}{4} \cdot 1 = 0,5$



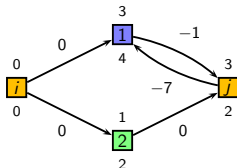
$$\Rightarrow d_{ij}^{CS} := 2,5$$

Verschärfung der Distanzmatrix II

Disjunktives Preprocessing

- Bestimme minimale Dauer von **inkompatiblen Aktivitätenpaaren** $(g, h) \in V$, die zwischen Ende von i und Start von j ausgeführt werden müssen
- Es gilt: Wenn g und h inkompatibel, dann $C_h - S_g \geq p_g + p_h$
- Untere Schranke für Zeitabstand** zwischen Ende von i und Start j :

$$d_{ij}^{low} := \min\{d_{ig}^{cs}, d_{ih}^{cs}\} + p_g + p_h + \min\{d_{gj}^{cs}, d_{hj}^{cs}\}$$
- Setze $d_{ij}^{cs} := \max\{d_{ij}^{cs}, d_{ij}^{low}\}$



$$d_{ij}^{low} (= 3) > d_{ij}^{cs} (= 2, 5) \Rightarrow d_{ij}^{cs} := 3$$

Identifikation unzulässiger Instanzen I

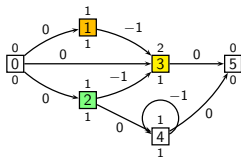
Konsistenztest

- Allgemeine **Zulässigkeitsbedingung**:

Sei $LB(U)$ für gegebene Menge von Aktivitäten $U \neq \emptyset$ eine untere Schranke für die minimale Projektdauer $C_{max}^*(U) = \max_{i \in U} C_i - \min_{j \in U} S_j$ zur Ausführung aller Aktivitäten $i \in U$. Instanz I ist nur zulässig, wenn gilt $\max_{\emptyset \neq U \subseteq V} (LB(U) + \min_{i,j \in U} d_{ij}^{CS}) \leq 0$

- **Untere Schranke für die Projektdauer** von minimalen Mengen:

Für gegebene Zahl $m \geq 2$, sei $U \subseteq V$ eine Menge von Aktivitäten, so dass jede Teilmenge $U' \subseteq U$ mit m Elementen verboten ist. Dann ist $LB(U) = \frac{1}{m-1} \sum_{i \in U} p_i$ eine untere Schranke für die Projektdauer $C_{max}^*(U)$



$$U = \{1, 2, 3\} \Rightarrow LB(U) = 2 \Rightarrow 2 + (-\infty) \leq 0$$

Identifikation unzulässiger Instanzen II

Zulässigkeitsbedingung basierend auf verbotenen Mengen:

- 1 Sei \mathcal{F}' eine Menge aller Aktivitätenmengen $U \subseteq V'$ mit $|U| \geq 2$ und ausschließlich paarweise inkompatiblen Aktivitäten. Dann ist Instanz I nur zulässig wenn gilt $\max_{U \in \mathcal{F}'} \left(\sum_{i \in U} p_i + \min_{i, j \in U} d_{ij}^{CS} \right) \leq 0$
- 2 Sei $\overline{\mathcal{F}}$ eine Menge aller verbotenen (Aktivitäten-)Mengen $U \subseteq V'$. Dann ist Instanz I nur zulässig, wenn gilt $\max_{U \in \overline{\mathcal{F}}} \left(\sum_{i \in U} p_i + (|U| - 1) \cdot \min_{i, j \in U} d_{ij}^{CS} \right) \leq 0$

- Algorithmisch abgeschwächte Version mit Beschränkung auf **zwei- und drei-elementige Mengen** von U
- Aus beiden Bedingungen sind **drei Tests** abgeleitet
- Prüfung auf Unzulässigkeit in **polynomialer Zeit**

Erweiterung

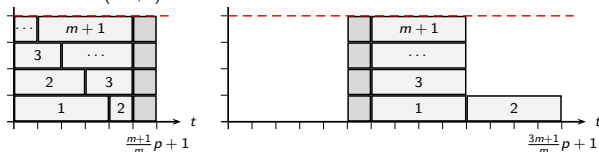
- Fixierungen von Variablen
- Bestimmung unterer Schranke durch Spaltengenerierung
- Variable Neighborhood Search

Proposition:

Es existieren zulässige Instanz I des Problems $Pm|temp, pmtn|C_{\max}$ mit $m > 2$ mit Nutzen der Unterbrechbarkeit $\gamma = \frac{2}{3} - \frac{2}{9m+3} - \varepsilon$ für ein beliebig kleines $\varepsilon > 0$.

Instanz I :

- m parallele Prozessoren
- $i \in V_1 = \{1, \dots, m+1\}$ mit $p = p_i = a \cdot m$, wobei $a \in \mathbb{N}$ und $r_i = 1$
- $i \in V_2 = \{m+2, \dots, 2m+1\}$ mit $p_i = 1$ und $r_i = 1$
- Dummy-Aktivität $2m+2$ mit $p_{(2m+2)} = \frac{m+1}{m}p$ und $r_{(2m+2)} = 0$
- Zeitabstände:
 - $i, j \in V_1 : \delta_{ij}^{cs} = -2p$
 - $i \in V_2 : \delta_{(2m+2)i}^{cs} = 0$ und $\delta_{i0}^{cs} = -\left(\frac{m+1}{m}p + 1\right)$



γ als Funktion $\gamma(p)$ der Dauern der Aktivitäten $i \in V_1$: $\lim_{p \rightarrow \infty} \gamma(p) = \frac{2}{3} - \frac{2}{9m+3}$

Experimentelle Ergebnisse I

- PC mit 3.16 GHz und 3 GB RAM, Betriebssystem Windows 7
- Modell (*MILP*) umgesetzt in GAMS 24.0.1, CPLEX 12.5 als MILP-Solver
- Preprocessing in C# umgesetzt
- Zeitlimit Solver: 300 Sekunden pro Instanz

Testset für Probleme des Typs $PS|prec, pmtn|C_{\max}$

Ergebnisse des MILP Modell für KSD-30 Instanzen

	P_{opt}	P_{opt}^{fnd}	Δ_{opt}^{\emptyset}	Δ_{nonp}	Δ_{nonp}^{min}	P_{imp}	n_{int}
$RS = 0.2$	10.0 %	31.7 %	2.5 %	-0.6 %	-8.8 %	46.7 %	14.2
$RS = 0.5$	40.0 %	60.0 %	0.7 %	-1.7 %	-7.3 %	57.5 %	11.4
$RS = 0.7$	80.0 %	87.5 %	0.2 %	-0.9 %	-6.7 %	27.5 %	8.2
$RS = 1.0$	100.0 %	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0
Total	57.5 %	69.8 %	0.8 %	-0.8 %	-8.8 %	32.9 %	8.5

Experimentelle Ergebnisse II

Testset für Probleme des Typs $PS|temp, pmtn|C_{max}$

Ergebnisse des MILP Modell für UBO-10 Instanzen

	P_{opt}	P_{inf}	P_{feas}	P_{opt}^{fnd}	Δ_{nonp}	P_{imp}	n_{int}
$RS = 0.0$	20.0 %	20.0 %	60.0 %	43.3 %	-1.9 %	36.7 %	1.4
$RS = 0.25$	23.3 %	10.0 %	66.7 %	50.0 %	-2.7 %	43.3 %	2.8
$RS = 0.5$	56.7 %	13.3 %	30.0 %	60.0 %	-1.9 %	33.3 %	1.8
Total	33.3 %	14.4 %	52.2 %	51.1 %	-2.2 %	37.8 %	2.0

Ergebnisse des MILP Modell für UBO-20 Instanzen

	P_{opt}	P_{inf}	P_{feas}	P_{opt}^{fnd}	Δ_{nonp}	P_{imp}	n_{int}
$RS = 0.0$	3.3 %	16.7 %	80.0 %	13.3 %	-3.0 %	63.3 %	3.6
$RS = 0.25$	16.7 %	6.7 %	76.7 %	23.3 %	-1.3 %	60.0 %	5.7
$RS = 0.5$	50.0 %	6.7 %	43.3 %	50.0 %	-0.5 %	46.7 %	4.6
Total	23.3 %	10.0 %	66.7 %	28.9 %	-1.4 %	56.7 %	4.6

Zusammenfassung und geplante Aufgaben

Zusammenfassung

- Ressourcenbeschränkte Schedulingprobleme mit unterbrechbaren Aktivitäten und verallgemeinerten Anordnungsbeziehungen
- Kompaktes Beschreibungsmodell und MILP-Formulierung
- Kanonisches Problem mit nichtpositiven Ende-Start-Zeitabständen
- Verbesserung der Performance durch Preprocessing
- Relativ kleine Optimalitätslücke für kleine Probleminstanzen

Nächste Aufgaben

1. Serielles Schedule-Generation-Schema basierend auf dem Verfahren von Franck und Neumann (1997)
2. Schreiben, schreiben, schreiben, ...

Vielen Dank für Ihre Aufmerksamkeit!

Schwindt C, Paetz T (2014)

Continuous Preemption Problems

In: Schwindt C, Zimmermann J (eds.) Handbook on Project Management and Scheduling. Springer, Berlin, to appear

Kontakt:

Tobias Paetz

tobias.paetz@tu-clausthal.de

Operations Management Group

Clausthal University of Technology

Literatur



Baker KR (1974)
Introduction to Sequencing and Scheduling
John Wiley, New York



Błażewicz J, Ecker KH, Pesch E, Schmidt G, Węglarz J (2007)
Handbook of Scheduling
Springer, Berlin



Brucker P (2009)
Scheduling Algorithms, 5th ed.
Springer, Berlin



Damay J, Quilliot A, Sanlaville E (2007)
Linear programming based algorithms for preemptive and non-preemptive RCPSP
European Journal of Operational Research 182:1012–1022



Demeulemeester E, Herroelen W (1996)
An efficient optimal solution procedure for the preemptive resource-constrained project scheduling problem
European Journal of Operational Research 90:334–348

Literatur



Franck B, Neumann K (1997)

Resource-constrained project scheduling with time windows: Structural questions and priority-rule methods

Report WIOR-492, Institute for Economic Theory and Operations Research, University of Karlsruhe



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



French S (1982)

Sequencing and Scheduling: An Introduction to the Mathematics of the Job-Shop

Ellis Horwood, Chichester



Kolisch R, Sprecher A (1996)

PSPLIB: A project scheduling library

European Journal of Operational Research 99:205–216

Literatur



Neumann K, Schwindt C, Zimmermann J (2003)

Project Scheduling with Time Windows and Scarce Resources, 2nd ed.

Springer, Berlin



Pinedo M (2008)

Scheduling: Theory, Algorithms and Systems, 3rd ed.

Springer, New York



Richter LK, Yano CA (1986)

A comparison of heuristics for preemptive resource-constrained project scheduling

Technical Report, 86-39, University of Michigan, Ann Arbor



Schwindt C, Paetz T (2014)

Continuous Preemption Problems

In: Schwindt C, Zimmermann J (eds.) Handbook on Project Management and Scheduling. Springer, Berlin, to appear



Słowiński R (1980)

Two approaches to problems of resource allocation among project activities: A comparative study

The Journal of the Operational Research Society 31:711–723

Literatur



Vanhoucke M (2008)

Setup times and fast tracking in resource-constrained project scheduling
Computers & Industrial Engineering 4:1062–1070