

WAS MUSS MAN WISSEN? OPERATIONS RESEARCH I

Patrick Elftmann und Matthias Sondermann
studium@sp3cialman.net

24. Oktober 2005

Vorbemerkung

Dies ist eine kleine Zusammenfassung des prüfungsrelevanten Stoffes der Vorlesung 'Operations Research I' bei Prof. Sebastian. Ich habe einfach mal versucht, die wichtigsten Sätze auszuformulieren und die wichtigsten Beweisideen zusammenzufassen. Dabei weise ich ausdrücklich daraufhin, dass es definitiv Lücken gibt, so dass man dieses Werk hier definitiv als unvollständig und falsch titulieren kann. Aber alles andere war nie mein Anspruch. Es soll einfach nur einen kleinen Überblick verschaffen mehr nicht.

Es erhebt keinerlei Anspruch auf Korrektheit und Vollständigkeit und stellt keine offizielle Veröffentlichung des Lehrstuhls dar. Bei Fehlern, Verbesserungsvorschlägen oder sonstigen Anregungen wird um eine Email an die angegebene Adresse gebeten.

Inhaltsverzeichnis

1 Grundlagen	2
1.1 Linearer Vektorraum	2
1.2 Konvexität	2
2 Theorie der linearen Programmierung	2
2.1 Grundmodell (GM)	3
2.2 Normalform (NF)	3
2.3 Grundmodell mit Gleichungsnebenbedingungen (GM*)	4
2.4 Basis des Modells	4
3 Simplex-Algorithmus	6
3.1 Ausgangsbasislösung	6
3.2 Aufnahmeregel	6
3.3 Eliminationsregel	7
3.4 Basistausch	7
3.5 Abbruchkriterium/Simplexkriterium	8
3.6 Der Algorithmus	8
3.7 Erweiterung	8
4 Dualität in der Linearen Programmierung	8
4.1 Duales Grundmodell	8
4.2 Schwache Dualität	9
4.3 Starke Dualität	10
4.4 Complementary-Slackness-Bedingung	10
4.5 Struktur- und Schlupfvariablen	11

5	Dualer Simplexalgorithmus	11
5.1	Ausgangsbasislösung	11
5.2	Eliminationsregel	11
5.3	Aufnahmeregel	12
5.4	Abbruchkriterium	12
6	Sensitivitätsanalysen	12
6.1	Veränderung eines Zielkoeffizienten	12
6.2	Veränderung einer Kapazität	13
7	Ganzzahlige Lineare Programmierung	13
7.1	Schnittebenenverfahren von Gomory	13
8	Graphen, Bäume, Netze, Netzpläne	14
8.1	Sätze aus der Graphentheorie	18
9	Netzplantechnik	20
9.1	Gantt-Chart	20
9.2	Begriffe der Netzplantechnik	21
9.3	Darstellung und Typen von Netzplänen	22
9.4	4 Stufen der Netztechnik	23
9.4.1	Strukturplanung	24
9.4.2	Zeitplanung	24
9.4.3	Kapazitätsplanung	26
9.4.4	Kostenplanung	26
9.5	CPM-Netzpläne	27
9.6	PERT-Netzpläne	30

1 Grundlagen

1.1 Linearer Vektorraum

- Ein Raum R^* heißt linearer Raum, wenn gilt:
 - $x, y \in R^* \leadsto z = x + y \in R^*$ (**komponentenweise Addition**)
 - $x \in R^*, \lambda \in \mathbb{R} \leadsto \lambda \cdot x \in R^*$ (**skalare Multiplikation**)
- Ein Raum R^* heißt **n-dimensional**, wenn er von n linear unabhängigen Vektoren erzeugt werden kann.
- z heißt **lineare Funktion**, wenn gilt:
 - $z : R^n \rightarrow R$
 - $z = c_1x_1 + \dots + c_nx_n + a = c^T x + a, c, x \in R^n, a \in \mathbb{R}$

1.2 Konvexität

- $M \subseteq R^n$ sei eine Punktmenge des R^n und jeder Punkt P_i von M wird durch einen n-dimensionalen Vektor x_i beschrieben.
- Eine Menge M heißt genau dann **konvex**, wenn für alle Punkte $P_1, P_2 \in M$ gilt, dass alle durch $x = \lambda x_1 + (1 - \lambda)x_2, 0 \leq \lambda \leq 1$, beschriebenen Punkte ebenfalls zu M gehören.
- Der Durchschnitt zweier konvexen Mengen ist konvex.
- Die Vereinigung zweier konvexer Mengen ist im Allgemeinen nicht konvex.
- Ein Punkt, der nicht echt ($\lambda = 0 \vee \lambda = 1$) auf dem Verbindungsvektor zweier Punkte von M liegt, heißt **Eckpunkt** von M .
- Eine konvexe Menge M heißt **konvexes Polyeder**, wenn sie endlich viele Eckpunkte besitzt.

2 Theorie der linearen Programmierung

verbale Problemformulierung

Ein Optimierungsproblem besteht darin eine Zielfunktion zu minimieren bzw. zu maximieren, wobei gewisse Nebenbedingungen zu berücksichtigen sind.

Ein lineares Optimierungsproblem zeichnet sich dadurch aus, dass die Zielfunktion eine lineare Funktion der Entscheidungsvariablen ist und dass die Nebenbedingungen in Form linearer Gleichungen oder Ungleichungen (oder Gleichungen und Ungleichungen) für die Entscheidungsvariablen gegeben sind.

2.1 Grundmodell (GM)

$$\begin{array}{ll|l} \max & z = c^T x & \text{(Zielfunktion)} \\ \text{unter} & Ax \leq b & \text{(Nebenbedingungen)} \\ & x \geq 0 & \text{(Nichtnegativitätsbedingungen)} \end{array}$$

$c \in \mathbb{R}^n$	Zielvektor / Vektor der Zielfunktionskoeffizienten
$x \in \mathbb{R}^n$	Entscheidungsvariablen (EV)
$A = A_{m,n}$	Koeffizientenmatrix
$b \in \mathbb{R}^m$	Kapazitätenvektor

2.2 Normalform (NF)

Die lineare Funktion

$$z = z(x_1, \dots, x_{n+m}) = c_1 x_1 + \dots + c_{n+m} x_{n+m}$$

ist unter Berücksichtigung der Gleichungen

$$\begin{array}{rcl} a_{11}x_1 + \dots + a_{1n}x_n + x_{n+1} & = & b_1 \\ \dots & & \\ a_{m1}x_1 + \dots + a_{mn}x_n + x_{n+m} & = & b_m \end{array}$$

zu maximieren. ($x_j \geq 0$, $j=1, \dots, n+m$ und $b_i \geq 0$, $i=1, \dots, m$)

Jedes lineare Optimierungsproblem läßt sich in die Normalform überführen:

- **fehlende Nichtnegativitätsbedingung:**
Ersetzung von x_i durch $x_i^* - x_i^{**}$ mit $x_i^*, x_i^{**} \geq 0$
- **Minimierung \rightarrow Maximierung:**
Multiplikation der Zielfunktion mit -1
- **negative Kapazität:**
Multiplikation der Nebenbedingung mit -1
- **Ungleichung \rightarrow Gleichung:**
Einführung einer Schlupfvariablen x_s . So ist
 $a_1 x_1 + \dots + a_n x_n \leq b$ äquivalent zu $a_1 x_1 + \dots + a_n x_n + x_s = b$ und
 $a_1 x_1 + \dots + a_n x_n \geq b$ äquivalent zu $a_1 x_1 + \dots + a_n x_n - x_s = b$
- **\geq und $= \rightarrow$ Hilfsvariablen:**
Einführung einer Variable $x_{n+1}, \dots, x_{n+m} \geq 0$ in jede Nebenbedingung, falls dort nicht schon eine Schlupfvariable addiert wird, mit zugehörigem Zielkoeffizient $-M$.

2.3 Grundmodell mit Gleichungsnebenbedingungen (GM*)

$$\begin{array}{l|l} \max & z = c^T x + 0^T s & \text{(Zielfunktion)} \\ \text{unter} & Ax + Is = b & \text{(Nebenbedingungen)} \\ & x, s \geq 0 & \text{(Nichtnegativitätsbedingungen)} \end{array}$$

mit $c, x \in \mathbb{R}^n$, $b, s \in \mathbb{R}^m$, $b \geq 0$, $A = A_{m,n}$, $I = I_m$

Andere Schreibweise:

$$\begin{array}{l|l} \max & z = c^T x & \text{(Zielfunktion)} \\ \text{unter} & Dx = b & \text{(Nebenbedingungen)} \\ & x \geq 0 & \text{(Nichtnegativitätsbedingungen)} \end{array}$$

mit $x = \begin{pmatrix} x \\ s \end{pmatrix}$, $c = \begin{pmatrix} c \\ 0 \end{pmatrix}$, $D = (A, I)$

2.4 Basis des Modells

- Jede nichtsinguläre $m \times m$ -Teilmatrix B von D heißt Basis des Modells \rightarrow Jedes aus m linear unabhängigen Spalten bestehende Teilsystem von D ist **Basis des Modells**.
- Bei $m+n$ Spalten existieren maximal $\binom{m+n}{m}$ verschiedene Basen.
- B heißt **zulässige Basis**, wenn gilt: $B^{-1}b \geq 0$
- **Basisdarstellung des Modells:**

$$\begin{array}{l|l} \max & z = (c_B^T, c_N^T) \begin{pmatrix} x_B \\ x_N \end{pmatrix} & \text{(Zielfunktion)} \\ \text{unter} & (B, N) \begin{pmatrix} x_B \\ x_N \end{pmatrix} = b & \text{(Nebenbedingungen)} \\ & x_B, x_N \geq 0 & \text{(Nichtnegativitätsbedingungen)} \end{array}$$

(Aufspaltung in Basisvariablen (BV) und Nichtbasisvariablen (NBV))

Multipliziert man das Gleichungssystem von links mit der Basisinversen B^{-1} , so erhält man die Basisdarstellung (BD) aller Lösungen, die $Dx=b$ erfüllen, in Bezug auf die Basis B

$$x_B = B^{-1}b - B^{-1}N x_N$$

Beziehungweise in Komponentenschreibweise für die i-te Basisvariable

$$x_{B_i} = b_i^* - \sum_{j=m+1}^{m+n} a_{ij}^* x_j$$

mit b_i^* als i -te Komponente von $B^{-1}b$ und a_{ij}^* als Element von $B^{-1}N$.
Um die Abhängigkeit der BV von den NBV zu untersuchen, schreibt man

$$x_B = x_B(x_N)$$

- Der Vektor, in dem alle Nichtbasisvariablen gleich Null gesetzt werden, heißt **Basislösung** (BL)

$$x = \begin{pmatrix} x_B(0) \\ 0 \end{pmatrix} = \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix}$$

Gilt zusätzlich $B^{-1}b \geq 0$, so heißt x_B **zulässige Basislösung** (ZBL).

- Die Ecken des Lösungsbereichs (LB) entsprechen den ZBL.
- Eine zulässige Lösung x ist genau dann eine Ecke des Lösungsbereichs, wenn x eine zulässige Basislösung ist.

- **Simplextheorem:**

Ein lineares Optimierungsproblem in Normalform nimmt sein Optimum an einer Ecke (ZBL) des Lösungsbereichs an, sofern es lösbar ist.

Beweisidee: Widerspruchsbeweis

Angenommen es existiert eine optimale Lösung \bar{x} , die keine ZBL ist. Dann kann \bar{x} als konvexe Linearkombination von den Eckpunkten x^1, \dots, x^p des Lösungsbereichs dargestellt werden

$$\bar{x} = \sum_{i=1}^p \lambda_i x^i \quad , \quad \sum_{i=1}^p \lambda_i = 1$$

Sei x^k diejenige unter den p Ecken, bei der der größte Zielfunktionswert angenommen wird

$$c^T x^k = \max_{1, \dots, p} \{c^T x^i\}$$

Weiterhin gilt

$$c^T \bar{x} = c^T \cdot \sum_{i=1}^p \lambda_i x^i = \sum_{i=1}^p \lambda_i (c^T x^i) \leq c^T x^k \sum_{i=1}^p \lambda_i = c^T x^k$$

Der Zielfunktionswert ist bei x^k mindestens so gross wie bei \bar{x} , der Eckpunkt x^k (ZBL) ist also ebenfalls eine optimale Lösung.

Allgemein:

Sind alle x^1, \dots, x^p optimale Lösungen, so auch alle Punkte der konvexen Hülle dieser Eckpunkte.

3 Simplex-Algorithmus

verbale Problemformulierung

Mit dem Simplex-Verfahren kann man für das Grundmodell (GM) der linearen Optimierung eine endliche Folge von Ecken des zulässigen Bereichs M von (GM) erhalten (bzw. von zulässigen Basislösungen von (GM)), wobei der Wert der Zielfunktion beim Übergang von einer Ecke zu einer anderen Ecke sich nicht verschlechtert. Die letzte Ecke dieser Folge ist optimal (falls (GM) überhaupt eine Lösung besitzt).

- **Voraussetzung:**

Es existiert eine zulässige Basislösung für das Modell.

- **Idee:** Basistausche

Iteration von Basistauschen, das heißt Tauschen von Basisvariablen gegen Nichtbasisvariablen, so dass sich der Zielfunktionswert nicht verschlechtert.

3.1 Ausgangsbasislösung

Ausgangsbasis sind die Spalten der Schlupfvariablen \rightarrow ZBL: $(x^T, s^T) = (0^T, b^T)$

3.2 Aufnahmeregel

Die Aufnahmeregel entscheidet, welche momentane NBV in die Basis aufgenommen werden soll. Dabei wird das heuristische Kriterium benutzt, diejenige NBV zu wählen, die den größten Zielfunktionswertzuwachs verspricht.

Formal soll also für die alte Basis B und die neue Basis B' gelten, dass

$$z = c_B^T x_B < c_{B'}^T x_{B'} = z' \text{ mit } x_B = B^{-1}b - B^{-1}N x_N$$

Die Basisdarstellung eingesetzt in $z = (c_B^T, c_N^T) \cdot \begin{pmatrix} x_B \\ x_N \end{pmatrix}$ ergibt

$$z = \underbrace{c_B^T B^{-1}b}_{z^*} - \underbrace{(-c_N^T + c_B^T B^{-1}N)}_{\Delta z^T x_N} \cdot x_N$$

z^* ist hierbei der Zielfunktionswert der aktuellen Basislösung x_B .

Δz_j gibt an, um wieviel sich der Zielfunktionswert verschlechtert, wenn x_j um eine Einheit erhöht wird ($\Delta z_j > 0$) \rightarrow Verbesserung des Zielfunktionswertes ist nur für diejenigen NBV x_j möglich, für die $\Delta z_j < 0$ ist.

Gibt es mehrere NBV mit $\Delta z_j < 0$, so wird diejenige gewählt, die die maximale Steigerung verspricht, also

$$\Delta z_l = \min\{\Delta z_j \mid \Delta z_j < 0\}$$

3.3 Eliminationsregel

Die Eliminationsregel entscheidet, welche momentane BV die Basis verlassen muss, um Platz für die durch die Aufnahme-regel bestimmte NBV zu schaffen. Der Wert der aufzunehmenden Variablen ist so zu wählen, dass alle Basisvariablen auch nach dem Basistausch noch einen positiven Wert besitzen. Es verläßt also diejenige Variable die Basis, die den minimalen Aufnahmewert der neuen Variable erzwingt.

B sei nichtentartete (alle $BV > 0$) zulässige Basis und o.B.d.A. enthalte B die ersten m Spalten von D. Dann gilt

$$x_{B_i} = b_i^* - \sum_{j=m+1}^{m+n} a_{ij}^* x_j$$

Sei x_l diejenige NBV, die BV werden soll. Dann ist $x_l = \theta$ und $x_j = 0$ für alle anderen NBV \curvearrowright

$$x_{B_i} = b_i^* - a_{il}^* \theta$$

Aufgrund der Nichtnegativitätsbedingung gilt:

- Ist $a_{il}^* \leq 0$, so kann nichts passieren.
- Ist $a_{il}^* > 0$, so muss gelten, dass $\theta \leq \frac{b_i^*}{a_{il}^*}$
- Sind alle $a_{il}^* \leq 0$, so existiert keine optimale Lösung, da θ beliebig groß gewählt werden kann, ohne eine Restriktion zu verletzen.

Wähle also diejenige BV, für die θ minimal ist, damit alle NNB erfüllt sind

$$\exists i_0 \text{ mit } \theta = \min \left\{ \frac{b_i^*}{a_{il}^*} \mid i = 1, \dots, m, a_{il}^* > 0 \right\} = \frac{b_{i_0}^*}{a_{i_0 l}^*}$$

2 Fälle:

- i_0 eindeutig \curvearrowright
Zu eliminierende BV eindeutig, Zulässigkeit der nächsten BL gesichert.
- i_0 nicht eindeutig \curvearrowright
Zulässigkeit der nächsten BL gesichert, aber in der nächsten ZBL existiert mindestens eine entartete ($BV=0$) Basisvariable.
Heuristik: Zu eliminierende BV zufällig wählen, um Berechnungszykel zu vermeiden.

3.4 Basistausch

Umformung des Gleichungssystems mittels Simplextableau.

Transformationsregeln (ähnlich dem Gauss-Algorithmus mit Pivotisierung):

- Dividiere die Pivotzeile durch das Pivoelement
- Pivotspalte wird zum Einheitsvektor e_k
- komponentenweise Transformation (**Kreisregel**)

3.5 Abbruchkriterium/Simplexkriterium

Eine optimale Lösung ist gefunden, wenn für alle Nichtbasisvariablen $\Delta z_j \geq 0$ gilt. Denn so existiert keine NBV, die durch Aufnahme in die Basis den Zielfunktionswert verbessern würde.

3.6 Der Algorithmus

```
start("Ausgangsbasislösung");
while(true)
{
    l = CalculateMinimumDelta();
    if(l>=0) STOP("optimum found");
    k = CalculateMinimumTheta();
    if(k==null) STOP("no solution");
    TableTransformation();
}
```

3.7 Erweiterung

Ist in der Koeffizientenmatrix die Einheitsmatrix nicht als Teilmatrix enthalten, so stellt sich die Frage nach der Ausgangsbasislösung. Abhilfe schafft das Einführen einer Hilfsvariablen x_H in denjenigen Restriktionen, die keine positive Schlupfvariable besitzen.

Das alte und neue Modell sind genau dann äquivalent, wenn alle $x_H = 0$ sind, also aus der Basis eliminiert werden. Dies wird sicher gestellt, indem jeder Hilfsvariablen ein sehr schlechter Zielkoeffizient $-M$ zugeordnet wird.

Kann ein x_H nicht aus der Basis eliminiert werden, so existiert keine zulässige Lösung des ursprünglichen Modells, falls keine Entartung vorliegt.

4 Dualität in der Linearen Programmierung

4.1 Duales Grundmodell

Primales Modell (PM)

$$\begin{array}{ll|l} \max & z = c^T x & \text{(Zielfunktion)} \\ \text{unter} & Ax \leq b & \text{(Nebenbedingungen)} \\ & x \geq 0 & \text{(Nichtnegativitätsbedingungen)} \end{array}$$

Duales Modell (DM)

$$\begin{array}{ll|l}
\min & Z = b^T y & \text{(Zielfunktion)} \\
\text{unter} & A^T y \geq c & \text{(Nebenbedingungen)} \\
& y \geq 0 & \text{(Nichtnegativitätsbedingungen)}
\end{array}$$

- Die n-dimensionalen Vektoren x des primalen Lösungsbereiches werden in die m-dimensionalen Vektoren y des dualen Lösungsbereiches überführt.
- Der Zielvektor c des PM wird zum Kapazitätenvektor des DM.
- Der Kapazitätenvektor des PM wird zum Zielvektor des DM.
- Bei primaler Maximierung ist die duale Zielfunktion zu minimieren.
- Die transponierte primale Koeffizientenmatrix ist die Koeffizientenmatrix des DM.
- Bei \leq -Bedingungen des PM sind die dualen Variablen vorzeichenbeschränkt (≥ 0).
- Bei $=$ -Bedingungen des PM sind die dualen Variablen vorzeichenunbeschränkt.

Das duale Modell des dualen Modells ist das primale Modell.

4.2 Schwache Dualität

verbale Formulierung

Der Zielfunktionswert des primalen Modells ist immer kleiner oder gleich dem Zielfunktionswert des dualen Modells bzw. der Zielfunktionswert des dualen Modells ist immer größer oder gleich dem Zielfunktionswert des primalen Modells. Somit bilden die Zielfunktionswerte der Modelle obere bzw. untere Schranken für ihr duales Modell.

formale Formulierung

Ist x zulässige Lösung des primalen Modells und y zulässige Lösung des dualen Modells, so gilt

$$z = c^T x \leq b^T y = Z$$

Beweisidee:

- $z = c^T x = x^T c \leq x^T A^T y$, da $A^T y \geq c$
- $Z = b^T y = y^T b \geq y^T A x = x^T A^T y$, da $Ax \leq b \Leftrightarrow x^T A^T y \leq b^T y$
- zusammen ergibt sich: $z = c^T x \leq x^T A^T y \leq b^T y = Z$

4.3 Starke Dualität

verbale Formulierung

Haben beide Modelle zulässige Lösungen, so besitzen beide Probleme auch optimale Lösungen, und die optimalen Zielfunktionswerte der beiden Modelle sind gleich.

formale Formulierung

Gilt für ein Paar zulässiger Lösungen (x_0, y_0) die Gleichung $c^T x_0 = b^T y_0$, so ist es ein Paar von optimalen Lösungen.

Beweisidee:

$$c^T x \leq b^T y_0 = c^T x_0 \quad \curvearrowright \quad c^T x_0 = \max\{c^T x \mid Ax \leq b, x \geq 0\}$$

$$b^T y \geq c^T x_0 = b^T y_0 \quad \curvearrowright \quad b^T y_0 = \min\{b^T y \mid A^T y \geq c, y \geq 0\}$$

Die optimale Lösung des primalen Modells ist also gleich der optimalen Lösung des dualen Modells. Hat das primale Modell keine zulässige Lösung, so hat das duale Modell keine optimale Lösung.

Die Umkehrung gilt ebenfalls, das heißt für ein paar optimaler Lösungen (x_0, y_0) gilt die Gleichung $c^T x_0 = b^T y_0$.

4.4 Complementary-Slackness-Bedingung

verbale Formulierung

Ist eine Nebenbedingung vom Typ \leq oder \geq eines linearen Optimierungsproblem für eine optimale Lösung mit dem Ungleichungszeichen $<$ bzw. $>$ erfüllt, das heißt die Nebenbedingung hat Schlupf, dann ist die zugeordnete Nebenbedingung für eine optimale Lösung des dualen Problems mit dem Gleichheitszeichen erfüllt, hat also keinen Schlupf.

formale Formulierung

Seien (x, x_s) und (y, y_s) zulässige Lösungen des dualen Paares und x_s bzw. y_s die jeweiligen Schlupfvariablen des zugehörigen Modells, dann gilt:

(x, x_s) und (y, y_s) sind genau dann optimale Lösungen, wenn die Komplementaritätsbedingung

$$x^T y_s + x_s^T y = 0$$

gilt. Das heißt also das Produkt aus den zueinander gehörenden Struktur- und Schlupfvariablen der jeweiligen Modelle ist Null.

Beweisidee:

- Zulässigkeit erzwingt $Ax + I_m x_s = b$ und $A^T y - I_n y_s = c$

- Jeweiliges Transponieren und Multiplizieren mit x bzw. y ergibt:

$$\begin{aligned}x^T A^T y + x_s^T I_m y &= b^T y \\ y^T A x - y_s^T I_n x &= c^T x\end{aligned}$$

- Da $x^T A^T y = y^T A x$ gilt durch Subtraktion der beiden Gleichungen voneinander

$$x_s^T y + y_s^T x = b^T y - c^T x$$

bzw.

$$x_s^T y + y_s^T x = 0 \Leftrightarrow b^T y = c^T x$$

4.5 Struktur- und Schlupfvariablen

- Im primalen Endtableau sind die Δz_j unter den primalen Schlupfvariablen die Werte der Strukturvariablen der optimalen dualen Lösung. Die entsprechenden Werte der dualen Schlupfvariablen findet man in der Δz_j -Zeile unter den primalen Strukturvariablen.

- **Schattenpreise/Opportunitätskosten:**

Die Werte der dualen Strukturvariablen geben an, um wieviel sich der optimale Wert der Zielfunktion ändert, wenn die Kapazität der zugehörigen Restriktion aus dem primalen Modell um 1 erhöht wird.

Es wird also die "Knappheit" der jeweils zugeordneten Restriktion (Ressource) angegeben.

Je höher der Schattenpreis, desto knapper ist Ressource. Das heißt also, daß die Lockerung einer Restriktion mit hohem Schattenpreis einen höheren Zielfunktionswert zur Folge hat, als eine bei niedrigem Schattenpreis.

5 Dualer Simplexalgorithmus

5.1 Ausgangsbasislösung

Starte mit einer dual zulässigen und primal eventuell unzulässigen Lösung. Es gilt also

$$\Delta z_j \geq 0 \quad \forall j = 1, \dots, m+n$$

und

$$\exists i : x_{B_i} = b_i^* < 0$$

5.2 Eliminationsregel

Aus der Menge der dual zulässigen Lösungen ist eine primal zulässige zu bestimmen. Folglich muss eine Basisvariable eliminiert werden, für die die rechte Seite negativ ist. Gilt dies für mehrere Variablen, so liegt es nahe, das Minimum zu wählen, also

$$x_k = \min_i \{b_i^* \mid b_i^* < 0\}$$

5.3 Aufnahmeregel

Die duale Zulässigkeit soll bei der Suche nach einer primal zulässigen Basislösung erhalten bleiben, der Δ -Vektor soll also auch nach dem Basistausch wieder positiv sein.

Ein Schritt im dualen Simplexalgorithmus kann nur dann durchgeführt werden, wenn in der Pivotzeile k mindestens ein negativer Koeffizient existiert, da sonst b_k^* im nächsten Schritt nicht positiv werden kann. Ist kein Koeffizient negativ, so existiert keine primal und dual zulässige Lösung.

Die aufzunehmende Nichtbasisvariable wird bestimmt durch

$$\zeta = \frac{\Delta z_l}{a_{kl}^*} = \max \left\{ \frac{\Delta z_j}{a_{kj}^*} \mid j = 1, \dots, m+n, a_{kj}^* < 0 \right\}$$

Begründung/Herleitung:

- Δz_j des nachfolgenden Tableaus ergeben sich zu $\Delta z'_j = \Delta z_j - \frac{a_{kj}^*}{a_{kl}^*} \cdot \Delta z_l$
- Damit $\Delta z'_j$ positiv bleibt, muss $\Delta z_j \geq \frac{\Delta z_l}{a_{kl}^*} \cdot a_{kj}^*$ gelten
- Umgeformt ergibt sich $\frac{\Delta z_j}{a_{kj}^*} \leq \frac{\Delta z_l}{a_{kl}^*}$ mit $a_{kl}^*, a_{kj}^* < 0$ und $\Delta z_j, \Delta z_l \geq 0$

5.4 Abbruchkriterium

Der Algorithmus bricht ab, wenn die primale Zulässigkeit erreicht ist, denn dann ist aufgrund der erhalten gebliebenen dualen Zulässigkeit eine optimale Lösung gefunden. Oder er bricht ab, wenn keine weitere Iteration mehr möglich ist, da $a_{kj}^* \geq 0 \forall j$.

6 Sensitivitätsanalysen

Fragestellung:

Um wieviel dürfen die gegebenen Größen (c, b, A) um ihren Wert schwanken, ohne dass sich die optimale Lösung ändert?

Da eine optimale Lösung gefunden ist, wenn sie sowohl primal als auch dual zulässig ist, gilt $\Delta z_j \geq 0$ und $B^{-1}b \geq 0$.

6.1 Veränderung eines Zielkoeffizienten

Verändert man einen Zielkoeffizienten c_i zu $c'_i = c_i + \lambda$, so ist im optimalen Endtableau in der Δz_j -Spalte abzulesen, welche Werte λ annehmen darf, ohne daß sich dabei die optimale Lösung ändert. (Bei der Durchführung des Simplexalgorithmus wird $\lambda = 0$ gesetzt, aber als Variable "mitgenommen")

6.2 Veränderung einer Kapazität

Verändert man einen Koeffizienten b_i im Kapazitätenvektor zu $b'_i = b_i + \lambda$, so ist in

$$x_B(0) = B^{-1}b_\lambda \geq 0$$

abzulesen, welche Werte λ annehmen darf, ohne daß sich dabei die optimale Lösung ändert.

7 Ganzzahlige Lineare Programmierung

Bei der ganzzahligen Optimierung sind die Variablen nur ganzzahlige und nicht mehr beliebige reelle Werte, beispielsweise in der Produktions-Programmplanung bei der Produktion von Stückgütern. Ganzzahlige Variablen und damit ganzzahlige Optimierungsprobleme treten auf, wenn kein Kontinuum von Alternativen vorliegt.

7.1 Schnittebenenverfahren von Gomory

Bei einer Indizierung, so daß die ersten m Variablen genau die Basisvariablen sind, ergibt sich die Basisdarstellung der i -ten Basisvariable zu

$$x_{B_i} = b_i^* - \sum_{j=m+1}^n a_{ij}^* x_j$$

Eine Aufspaltung der nichtganzzahligen b_i^* und a_{ij}^* in ihren ganzzahligen und positiv gebrochenen Anteil ergibt

$$b_i^* = \lfloor b_i^* \rfloor + h_i$$

$$a_{ij}^* = \lfloor a_{ij}^* \rfloor + h_{ij}$$

Eingesetzt in die obige Gleichung ergibt

$$x_{B_i} = \lfloor b_i^* \rfloor + h_i - \sum_{j=m+1}^n (\lfloor a_{ij}^* \rfloor + h_{ij}) \cdot x_j$$

Aufspalten in einen ganzzahligen und nicht notwendig ganzzahligen Teil ergibt

$$x_{B_i} = \underbrace{\lfloor b_i^* \rfloor - \sum_{j=m+1}^n \lfloor a_{ij}^* \rfloor x_j}_{\text{ist ganzzahlig}} + \underbrace{h_i - \sum_{j=m+1}^n h_{ij} x_j}_{\text{muss ganzzahlig sein}}$$

Da $h_{ij} x_j \geq 0$, weil h_{ij} und x_j jeweils ≥ 0 sind, gilt

$$h_i - \sum_{j=m+1}^n h_{ij} x_j < 1$$

Und somit aufgrund der Ganzzahligkeit

$$h_i - \sum_{j=m+1}^n h_{ij}x_j \leq 0$$

Fügt man noch eine (Gomory-)Variable x_G als Schlupfvariable hinzu, so entsteht die Gomory-Restriktion

$$x_G - \sum_{j=m+1}^n h_{ij}x_j = -h_i$$

Da in der Basislösung alle Nichtbasisvariablen gleich 0 sind, gilt also $x_G = -h_i$

Idee des Algorithmus:

- Hinzufügen der Gomoryrestriktion zum optimalen Tableau des relaxierten Problems (Die Gomoryvariable wird also Basisvariable mit Wert $-h_i$)
- Dadurch bleibt die duale Zulässigkeit erhalten, die primale wird aber durch $-h_i < 0$ verletzt
- Deshalb muß die Gomoryvariable aus der Basis eliminiert werden \curvearrowright Duale Simplexiteration mit Gomoryrestriktion als Pivotzeile
- Folglich wird die primale Unzulässigkeit beseitigt

Algorithmus:

1. Löse relaxiertes Problem mit dem Simplexalgorithmus
2. if(Lösung ganzzahlig) STOPP
3. Aufspalten der Komponenten der nichtganzzahligen Lösungen
4. Bildung der Gomory-Restriktion für die Variable mit größte
5. Hinzufügen der Gomory-Restriktion zum Tableau
6. Durchführung einer Iteration des dualen Simplexalgorithmus
Restriktion als Pivotzeile

8 Graphen, Bäume, Netze, Netzpläne

Graph

Ein Graph G besteht aus einer Menge V (Knoten, vertices) und einer Menge E (Kanten, edges) mit $E \neq \emptyset$, $V \cap E = \emptyset$ sowie einer auf V definierten Abbildung Ψ (Inzidenz), die jedem $v \in V$ eine Menge von zwei, möglicherweise identischen, Elementen $e, e' \in E$ zuordnet. Ist das jedem $v \in V$ zugeordnete Paar von Elementen aus E geordnet, so sind die Kanten gerichtet und man spricht von einem gerichteten Graphen, anderenfalls handelt

es sich um einen ungerichteten Graphen.

Kanten sind normalerweise durch genau zwei Knoten begrenzt. Die Kante ist dann inzident zu diesen Knoten. Kanten, die eine gemeinsamen Knoten haben, nennt man dagegen adjazent zueinander. Auch zwei Knoten, die durch eine Kante verbunden sind, heißen adjazent oder benachbart zueinander. Sind zwei Kanten inzident zu denselben zwei Knoten, so spricht man von parallelen Kanten.

Eine Kante, die mit einem einzigen Knoten inzidiert, wird als Schlinge bezeichnet.

Kantenfolge, Kantenzug

Eine alternierende Folge von Knoten und Kanten eines Graphen, in der jede vorkommende Kante mit den beiden benachbarten Knoten inzidiert, heißt Kantenfolge.

Tritt in einer solchen Kantenfolge jede Kante nur einmal auf, so spricht man von einem Kantenzug

Grad, Valenz

Unter dem Grad oder der Valenz eines Knoten in einem ungerichteten Graphen versteht man die Anzahl zu diesem Knoten inzidenten Kanten. Hierbei zählen Schlingen doppelt. Bei gerichteten Graphen unterscheidet man entsprechend zwischen dem positiven Grad oder Ausgangsgrad eines Knoten (Zahl der ausgehenden Pfeile) und dem negativen Grad oder Eingangsgrad (Zahl der eingehenden Pfeile) eines Knoten.

Teilgraph, Untergraph

Ein Graph (gerichtet oder ungerichtet) G' mit der Knotenmenge V' , der Kantenmenge E' und der Inzidenzabbildung Ψ' heißt Teilgraph eines Graphen mit der Knotenmenge V , der Kantenmenge E und der Inzidenzabbildung Ψ , wenn $V' \subseteq V$, $E' \subseteq E$ sowie $\Psi = \Psi'$ für alle $e \in E'$. G' heißt echter Teilgraph, wenn $E' \subset E$ gilt.

G' ist dann Untergraph von G , wenn G' Teilgraph von G ist und wenn jede Kante aus E , die zwei Knoten aus V' verbindet, auch zu E' gehört.

Zusammenhängender Graph

Ein Graph G heißt dann zusammenhängend, wenn jeweils zwei beliebige Knoten von G durch eine Kantenfolge zu verbinden sind. Andernfalls heißt G nicht-zusammenhängend.

Alle Knoten eines Graphen G , die durch eine Kantenfolge mit einem Knoten v verbunden sind, definieren mit v einen Untergraphen, der Komponente von G genannt wird. Jeder nicht-zusammenhängende Graph G besteht aus mehreren derartigen (zusammenhängenden) Komponenten. Besteht eine solche Komponente aus einem einzigen Knoten, so bezeichnet man sie als isolierte Knoten.

Regulärer Graph

Haben in einem Graphen G alle Knoten die gleiche Valenz r , so sprechen wir von einem regulären Graphen vom Grad r .

Vollständiger Graph

Ein vollständiger Graph ist ein regulärer Graph, in dem je zwei Knoten entweder durch eine Kante (ungerichteter Graph) oder durch ein Paar gegenläufiger Pfeile (gerichteter Graph) verbunden sind.

Schlichter Graph

Ein schlichter Graph G ist ein (gerichteter oder ungerichteter) Graph ohne parallele Kanten und ohne Schlingen.

Für (endliche) vollständige schlichte Graphen mit n Knoten ist die Kantenzahl fest:
Ein ungerichteter vollständiger Graph mit n Knoten hat $\frac{n(n-1)}{2}$ Kanten, ein ebensolcher gerichteter Graph $n(n-1)$ Kanten.

Nachfolger-, Vorgängerknoten

In einem gerichteten Graphen G heißt ein Knoten v_j unmittelbar nachfolgender Knoten von v_i , falls eine gerichtete Kante von v_i nach v_j existiert. Der Knoten v_i ist entsprechend unmittelbar vorangehender Knoten von Knoten v_j .

Vorgänger, Nachfolger

In einem gerichteten Graphen heißt ein Pfeil e mit $\Psi(e) = (v_i, v_j)$ unmittelbarer Vorgänger des Pfeils e' , falls $\Psi(e') = (v_j, v_h)$ ist. Entsprechend ist e' unmittelbarer Nachfolger von e .

Quelle, Senke

In einem gerichteten Graphen heißt ein Knoten mit negativem Grad (Valenz) von Null eine Quelle und ein Knoten mit einer positiven Valenz von Null eine Senke.

Schleife, Zyklus, Masche

In einem gerichteten Graphen bezeichnet man als Schleife einen gerichteten Kantenzug, bei dem der Anfangsknoten gleich dem Endknoten ist. Handelt es sich bei dem Kantenzug um einen Weg, d.h. werden die Kanten in Pfeilrichtung durchlaufen, spricht man von einem Zyklus. Besteht ein solcher Zyklus nur aus zwei Kanten, so bezeichnet man ihn als Masche.

Digraph

Ein endlicher, gerichteter, schlichter Graph heißt Digraph.

Darstellungsformen von Graphen

1. Darstellung durch Abbildungen (Funktionen)
2. Darstellung durch Kanten- oder Pfeillisten
3. Darstellung mit Hilfe einer Adjazenzmatrix
4. Darstellung mit Hilfe einer Inzidenzmatrix

Adjazenzmatrix

Für einen Digraphen mit m Knoten ist die Adjazenzmatrix eine $m \times m$ Matrix A , deren Elemente a_{ij} wie folgt definiert sind.

$$a_{ij} = \begin{cases} 1 & , \text{ falls Kante von } v_i \text{ nach } v_j \text{ existiert} \\ 0 & , \text{ sonst} \end{cases} \quad \forall i, j = 1, \dots, m$$

Zeilensumme = positive Valenz

Spaltensumme = negative Valenz

Inzidenzmatrix

Für einen Digraphen mit m Knoten und n Kanten ist die Inzidenzmatrix eine $m \times n$ Matrix. Jeder Zeile ist ein Knoten und jeder Spalte eine Kante zugeordnet. Die Elemente h_{ij} dieser Matrix sind wie folgt definiert:

$$h_{ij} = \begin{cases} 1 & , \text{ falls Kante von } e_j \text{ von Knoten } v_i \text{ ausgeht} \\ -1 & , \text{ falls Kante in } e_j \text{ in Knoten } v_i \text{ einmündet} \\ 0 & , \text{ sonst} \end{cases} \quad \forall i = 1, \dots, m, \forall j = 1, \dots, n$$

Mit der Inzidenzmatrix kann man im Gegensatz zur Adjazenzmatrix auch parallele Kanten beschreiben.

Ungerichteter Baum

Ein zyklensfreier zusammenhängender ungerichteter Graph wird als ungerichteter Baum bezeichnet.

Wald

Ein nicht-zusammenhängender zyklensfreier Graph wird als Wald bezeichnet. Ein Baum B mit $m > 1$ Knoten hat folgende Eigenschaften:

1. Zwei beliebige Knoten von B sind durch genau eine Kantenfolge miteinander verbunden.
2. Hat B m Knoten, so enthält er $n = m - 1$ Kanten.
3. Verbindet man zwei beliebige Knoten von B miteinander durch eine zusätzliche Kante, so entsteht genau ein Kreis (Zyklus).
4. Ein Baum hat mindestens zwei Endknoten mit der Valenz 1.

Gerüst

Ein zusammenhängender, alle Knoten eines gegebenen zusammenhängenden Graphen G enthaltender Teilgraph mit minimaler Kantenzahl heißt Gerüst H von G .

Netz, Netzwerk

Ein zyklensfreier Digraph mit genau einer Quelle und einer Senke heißt Netz. Ordnet man durch eine Abbildung E (Menge der Kanten) in den Raum \mathbb{R}^b den Kanten dieses Netzes Gewichte zu, so bezeichnet man das bewertete Netz als Netzwerk.

8.1 Sätze aus der Graphentheorie

- Handschlaglemma: Die Anzahl der Ecken ungeraden Grades ist gerade.

Beweis:

Sei $G = (V, E)$ ein Graph mit n Knoten und m Kanten. Jede Kante, auch jede Schlinge liefert zum Gesamtgrad des Graphen $d(G) = \sum_{i=1}^n d(i)$ den Beitrag 2.

$$\Rightarrow d(G) = 2m = \sum_{d(i) \text{ gerade}} d(i) + \sum_{d(i) \text{ ungerade}} d(i) \quad (\text{gerade} = \text{gerade} + X)$$

$\Rightarrow G$ enthält gerade Anzahl von Knoten ungeraden Grades

- Ein Graph $G = (V, E)$ mit $|V| \geq 2$ ohne Schlingen enthält mindestens 2 Knoten gleichen Grades.

Beweis:

Angenommen alle Knotengrade wären verschieden. Dann könnte man die Knotengrade wie folgt absteigend sortieren:

$d(1) > \dots > d(|V|)$. Angenommen $d(1) = |V| - 1$, dann ist Knoten 1 zu allen Knoten adjazent. Widerspruch zu Knoten $d(|V|) = 0$.

\Rightarrow Es existieren zwei Knoten gleichen Grades.

- Ein Graph $G = (V, E)$ mit Knotengrad $d(i) \geq 2 \forall v \in V$ besitzt mindestens einen Kreis.

Beweis:

Falls G Schlinge oder parallele Kanten hat, ist die Behauptung klar.

Ist G schlicht sei $W = (i_0, e_1, \dots, e_v, e_i)$ ein längster Weg. Da $d(i) \geq 2$, muss i_0 zu einem i_j mit $1 \leq j \leq p$ adjazent sein und damit hat man einen Kreis gefunden.

- Jeder Baum (zusammenhängender Graph ohne Kreis) $T = (V, E)$ mit $|V| \geq 2$ besitzt mindestens 2 Knoten mit Grad 1.

Beweis:

Aus dem Beweis des Handschlaglemmas folgt für den Gesamtgrad des Baumes T mit n Knoten:

$$\sum_{i \in T} d(i) = 2|E| = 2(n-1) = 2n - 2$$

Da ein Baum ein zusammenhängender Graph ist, gilt: $d(i) \geq 1 \forall i \in V(T)$.

Angenommen nur ein Knoten hätte Grad 1. Dann hätten $n-1$ Knoten mindestens Grad 2, und der Gesamtgrad des Baumes wäre mindestens $2n-1$. Daraus folgt die Behauptung.

- In einem Graphen $G = (V, E)$ gilt: $e \in E$ ist eine Brücke ($e \in E$ heißt Brücke gdw. die Anzahl der Komponenten von $G \setminus \{e\} = \text{Anzahl der Komponenten von } G+1$) gdw. e nicht in einem Kreis von G liegt.

Beweis: -

- Ein zusammenhängender Graph $G = (V, E)$ mit $V \neq \emptyset, E \neq \emptyset$ ist eulersch (enthält eine Eulertour) gdw. jeder Knoten geraden Grad hat.

Beweis:

' \Rightarrow '

Sei T eine Eulertour in G . Bewegt man sich entlang der Eulertour, so liefert jeder Durchgang durch einen Knoten den Beitrag 2 zum Grad dieses Knotens. Dies gilt auch für den Startknoten, denn dieser ist gleichzeitig auch Endknoten. Da G zusammenhängend ist, wird jeder Knoten mind. einmal durchlaufen und wegen $E(G) = E(T)$ werden alle $V \in G$ berücksichtigt. Daher hat jeder Knoten geraden Grad ≥ 2 .

' \Leftarrow '

Hat jeder Knoten geraden Grad, so gilt wegen des Zusammenhangs $d(i, G) \geq 2 \forall i \in V(G)$

Induktion über $|E|$:

I.A.:

$m=1$: Besteht G aus einem Knoten und einer Schlinge, so hat G eine Eulertour.

$m=2$: Wegen $d(i, G) \geq 2$ besitzt G nach c) einen Kreis

Gilt $E(G) = E(C)$, dann ist C eine Eulertour. Falls nicht, so ist der Graph $G' = (V(G), E(G) \setminus E(C))$ ein möglicherweise nicht zusammenhängender Teilgraph von G mit weniger als $|E|$ Kanten, in dem immer noch jede Ecke geraden Grad hat.

Daher besitzt jede Komponente von G' , die mindestens einen Knoten enthält nach Voraussetzung eine Eulertour. Wegen des Zusammenhanges von G ergibt sich nun wie folgt eine Eulertour: Beginne mit beliebigen Knoten von C , folge der Kante von C bis zu einem ersten Knoten einer Komponente G' , durchlaufe die Eulertour an dieser Komponente usw ...

9 Netzplantechnik

9.1 Gantt-Chart

Ein Planungsverfahren, das auf den Beginn der wissenschaftlichen Betriebswirtschaftslehre zurückgeht, ist das Verfahren von Henry L. Gantt. Die heute noch praktizierten konventionellen Planungsmethoden zur *zeitlichen Festlegung von Arbeitsabläufen* sind in der Hauptsache Weiterentwicklungen des Ganttschen Planungsbogen. Hierbei werden gewöhnlich auf der horizontalen Achse die Zeit und auf der vertikalen Achse die durchzuführenden Arbeitsgänge aufgezeichnet. Werden geplante Abläufe und wirklich erfolgte Abläufe voneinander unterscheidbar festgehalten, so kann die Gantt-Chart auch als *Kontrollinstrument des Arbeitsablaufes* angesehen werden.

Die Vorteile einer solchen Darstellung sind:

- Länge der Balken steht in Beziehung zur Zeit
- Das Diagramm selbst wird zur Aufzeichnung des Arbeitsfortschrittes verwendet
- Das Gantt-Schema ist auch für Nichtspezialisten sehr informativ; der geplante Arbeitsfortschritt kann jederzeit mit dem erreichten Ergebnis verglichen werden. Ihr Anwendungsbereich ist sehr weit entwickelt

Die offensichtlichen Nachteile sind:

- simultane Struktur- und Zeitplanung
- geringe Aussagefähigkeit bezüglich der Reichenfolge bzw. Verknüpfung der Vorgänge
- feste Terminierung alle Vorgänge, die zu einem Projekt gehören
- geringe Anpassungsmöglichkeit an veränderte Bedingungen bei der Überwachung
- Begrenzung der Anzahl der Vorgänge aus Gründen der Übersichtlichkeit
- keine Möglichkeit des Einsatzes von Datenverarbeitungsanlagen

Der heutige Anwendungsbereich des Ganttschen Planungsbogens liegt nach dem Abschluß einer Netzplanuntersuchung hauptsächlich zur Verdeutlichung der Ergebnisse für Führungskräfte.

9.2 Begriffe der Netzplantechnik

Projekt

Ein Projekt ist ein Vorhaben mit einem definierten Anfang und einem definierten Ende. (DIN 69901), charakterisiert durch "Einmaligkeit der Bedingungen in ihrer Gesamtheit" z.B.

1. Zielvorgaben
2. zeitliche, finanzielle, personelle oder andere Restriktionen
3. Abgrenzung gegenüber anderen Vorhaben
4. projektspezifische Organisation

Projektplanung

Projektplanung ist ein wesentlicher Teil des über den gesamten Projektzeitraum reichenden Projektmanagements.

Teilaufgaben der Projektplanung sind:

- Planung von Teil- oder Unterzielen
- Ableitung von Teilaufgaben des Projekts
- Planung der Ablauforganisation
- Planung des Aufwands für Personal, Finanzen, Sachmittel, Raumbedarf
- Planung des Projektinformations- und Dokumentationssystems

Zum Zwecke der Darstellung, Analyse und Steuerung werden die zu planenden oder steuernden Komplexe in Elemente zerlegt. Die Gesamtheit des zu Planenden wird als Projekt bezeichnet. Dieses wird in zwei Arten von Elementen zerlegt:

- **Vorgang:**
Zeitbeanspruchendes Element des Projektes
(Projektteile mit definierbarem Anfang und Ende)
- **Ereignis:**
Definierter Zustand im Projektablauf
(Zeitpunkte, an denen Projektteile beginnen oder enden)

9.3 Darstellung und Typen von Netzplänen

Netzpläne sind im wesentlichen gerichtete Graphen mit Bewertungen auf Kanten und Knoten. Die Kanten drücken Anordnungsbeziehungen aus, die Knoten stehen für Ereignisse. Je nach Typ des Netzplans und vorliegender Konstellation können auch mehrere Ereignisse in einem Knoten zusammengefaßt werden.

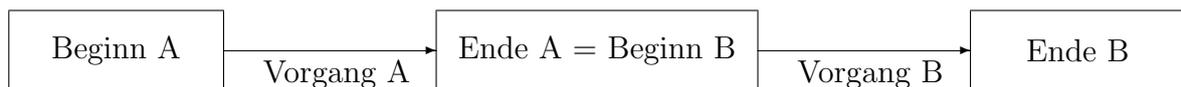
Die drei Grundtypen sind:

Vorgangspfeilnetzpläne, Vorgangsknotennetzpläne und Ereignisknotennetzpläne.

Vorgangspfeilnetze

vorgangsorientierter Netzplan in Pfeildarstellung

Im Mittelpunkt der Betrachtungen stehen bei Vorgangspfeilnetzplänen Vorgänge. Sie werden als Pfeile dargestellt. Ereignisse treten lediglich in Form von Anfangs- und Endzuständen von Vorgängen auf. Sie sind immer dann angebracht, wenn man besonders an den einzelnen Arbeitsgängen interessiert ist.



Der bekannteste Typus dieser Gruppe ist der **CPM-Netzplan**.

(CPM = Critical Path Method)

Vorgangsknotennetze

vorgangsorientierter Netzplan in Kreisdarstellung

Beschreibt man dagegen die Ereignisse in ihrem Inhalt, sozusagen als Kontrollpunkte im Projektablauf, so bleiben die Vorgänge, die zu den einzelnen Ereignissen führen, im wesentlichen undefiniert, man spricht dann von 'ereignisorientierten Netzplänen', die hauptsächlich zu Kontrollzwecken Anwendung finden. Ereignisorientierte Netzpläne enthalten i.a. weniger Informationen als gleich große vorgangsorientierte Netzpläne, da die Zahl der im vorgangsorientierten Netzplan beschriebenen Kanten (Pfeile) meist größer ist als die Zahl der im Netzplan beschriebenen Knoten (Kreise).

In dieser Gruppe von Netzplänen werden Anfangs- und Endzustand eines Vorganges zu einem Knoten zusammengefaßt und lediglich die Abfolge zwischen den Vorgängen durch Pfeile dargestellt. Für die beiden oben dargestellten Vorgänge A und B ergibt sich folgende Darstellung:



Ein Typ dieser Gruppe ist der **MPM-Netzplan**.

(MPM = Metra-Potential-Methode)

Ereignisknotennetze

ereignisorientierter Netzplan in Pfeildarstellung

Bei Ereignisknotennetzen werden ähnlich wie bei Vorgangspfeilnetzen Ereignisse durch Knoten dargestellt sowie Abhängigkeiten zwischen den Ereignissen durch Pfeile. Die Pfeile brauchen jedoch nicht exakt als Vorgänge spezifiziert zu werden. Der Schwerpunkt der Betrachtung liegt auf den Ereignissen, die auch näher benannt werden. Bekanntester Vertreter dieses Netzplantyps ist **PERT** (Program Evaluation and Review Technique). PERT verwendet bei der Zeitanalyse neben der zu erwartenden Zeitdauer auch optimistische und pessimistische Zeitabschätzungen.

9.4 4 Stufen der Netztechnik

Vier Stufen (Phasen) der Netztechnik lassen sich unterscheiden:

1. Die Strukturanalyse

Darstellung der Abhängigkeitsbeziehungen der Vorgänge auf graphische Weise (Netzplan), tabellarisch oder durch eine Matrix (Inzidenz- oder Adjazenzmatrix)
Davor Zerlegung in Teilprojekte, Vorgänge oder Ereignisse definieren;
Reihenfolgeproblem

2. Zeitanalyse

Bestimmung der frühesten und spätesten Anfangs- und Endzeiten der einzelnen Vorgänge, der Gesamtprojektdauer und verschiedener Pufferzeiten.

3. Kapazitätsanalyse (Ressource)

Sind die zur Durchführung der geplanten Vorgänge notwendigen Betriebsmittel nicht in ausreichendem Maße vorhanden, so entstehen Engpässe, die zur Erhöhung der Gesamtprojektdauer gegenüber der in der zweiten Stufen errechneten führen.
Ziel ist die Bestimmung der optimalen Reihenfolge der Vorgänge, d.h. der Reihenfolge, in der die Vorgänge durchzuführen sind, damit unter Berücksichtigung der vorhandenen Kapazitäten und der bestehenden technologischen Vorgangsabhängigkeiten eine minimale Projektdauer erreicht wird.

4. Kostenanalyse

Kann die Dauer der Vorgänge (z.B. durch Überstunden) verkürzt werden, so läßt sich nicht mehr eine eindeutige Projektdauer berechnen. Man kann jedoch dann in Phase 4 die Vorgangsdauer ermitteln, die zu der Projektdauer führt, bei der die geringsten Gesamtkosten auftreten.

Oder:

Kann man die Dauer der Vorgänge beeinflussen, so kann man bei fester Projektdauer die Vorgangsdauern bestimmen, die minimale Gesamtkosten erfordern.

Die Phasen 1 und 2 sind reine Darstellungs- oder Berechnungsphasen, die Phasen 3 und 4 Optimierungsphasen. Wesentliche Entscheidungen fallen aber schon in Stufe 1.

9.4.1 Strukturplanung

1. Jeder Vorgang beginnt mit einem Ereignis und endet mit einem nachfolgenden Ereignis.
2. Müssen ein oder mehrere Vorgänge beendet sein, bevor ein weiterer beginnen kann, so enden diese alle im Anfangsereignis des nachfolgenden Vorganges.
3. Können mehrere Vorgänge beginnen, nachdem ein vorausgegangener beendet ist, so beginnen diese alle im Endereignis des vorangegangenen Vorgangs.
4. Haben zwei Vorgänge gemeinsame Anfangs- und Endereignisse, dann wird die eindeutige Kennzeichnung durch einen Scheinvorgang hergestellt. Der Scheinvorgang stellt keinen realen Vorgang dar. Er wird jedoch wie ein normaler Vorgang behandelt, besitzt aber die Zeitdauer Null.
5. Enden und beginnen in einem Ereignis mehrere Vorgänge, die nicht alle voneinander abhängig sind, wird der richtige Ablauf ebenfalls mit Hilfe von Scheinvorgängen hergestellt.
6. Kann ein Vorgang beginnen, bevor der vorhergehende vollständig beendet ist, kann der vorhergehende unterteilt werden.
7. In einem Netz dürfen keine Schleifen auftreten.

9.4.2 Zeitplanung

Die zeitliche Berechnung des Netzplanes erfolgt in drei Stufen:

1. Die Errechnung frühester und spätester Zeiten für jeden Knoten (Ereignis).
2. Die Errechnung frühester und spätester Anfangs- und Endzeiten für alle Pfeile (Vorgänge).
3. Die Errechnung der Puffer und die Bestimmung des kritischen Weges.

- zu 1) **Errechnung von Ereigniszeiten**

Die Errechnung der frühestmöglichen Ereigniszeiten FZ_i entspricht der Bestimmung längster Wege vom Anfangsknoten (Quelle) zu allen anderen Knoten. Die früheste Zeit des Endknoten (Senke) ergibt dann die kürzestmögliche Projektdauer, wenn man mit Null bei der Quelle beginnt.

Die spätesten (spätest-zulässigen) Ereigniszeiten SZ_i erhält man, indem man von der Senke beginnend die längsten Wege zu allen Knoten berechnet.

- zu 2) **Errechnung von frühesten und spätesten Vorgangszeiten**

Bezeichnet man mit dem Index i die Anfangsknoten und mit dem Index j die Endknoten von Vorgängen, so kann man unter Verwendung der bereits errechneten Ereigniszeiten die Vorgangszeiten wie folgt bestimmen:

Frühestmöglicher Beginn:	$FA_{ij} = FZ_i$
Frühestmögliches Ende:	$FE_{ij} = FZ_i + d_{ij}$
Spätestmöglicher Beginn:	$SA_{ij} = SZ_j - d_{ij}$
Spätestmögliches Ende:	$SE_{ij} = SZ_j$
Maximal verfügbare Zeit:	$SZ_j - FZ_i$

- zu 3) **Errechnung der Puffer und des kritischen Weges**

Unter Pufferzeiten versteht man die Zeitreserve, um die ein Vorgang verschoben werden kann bzw. seine Dauer ausgedehnt werden kann, ohne dass der Endtermin des Projektes beeinflusst wird.

Man unterscheidet drei Arten von Pufferzeiten:

1. **Gesamtpufferzeit** (total float):

$$GP_{ij} = SZ_j - FZ_i - d_{ij}$$

Die Gesamtpufferzeit gibt an, um wieviel sich ein Vorgang bestenfalls verschieben lässt. Alle Vorgänge längs des kritischen Weges haben die Gesamtpufferzeit $GP = 0$.

2. **Freie Pufferzeit** (free float):

$$FP_{ij} = FZ_j - FZ_i - d_{ij}$$

Die Freie Pufferzeit gibt an, um wieviel Zeiteinheiten der Vorgang (i,j) verschoben werden kann, ohne dass der frühestmögliche Beginn des nachgeordneten Vorgangs beeinflusst wird.

3. **Unabhängige Pufferzeit** (independent float):

$$UP_{ij} = \max\{0, FZ_j - SZ_i - d_{ij}\}$$

Die Unabhängige Pufferzeit gibt den Zeitraum an, in dem sich der Vorgang (i,j) verschieben lässt, wenn alle vorgeordneten Vorgänge (h,i) zum spätmöglichsten Abschluß enden und alle nachgeordneten Vorgänge (j,k) zum frühestmöglichen Anfang beginnen sollen.

- **Kritischer Pfad**

Wenn die maximal für den Vorgang verfügbare Zeit gleich dessen Dauer ist:

$$SZ_j - FZ_i = d_{ij}$$

wird der Vorgang als kritisch bezeichnet. Zieht sich eine ununterbrochene Folge von kritischen Vorgängen vom Beginn bis zum Endergebnis des Projektes, wird diese Folge als kritischer Pfad bezeichnet (critical path). Die Verzögerung eines kritischen

Vorganges verlängert im gleichen Maße die Dauer des gesamten Projektes.
Für kritische Vorgänge gilt:

$$FZ_i = SZ_i; FZ_j = SZ_j$$

d.h. frühestmögliche und spätestmögliche Zeiten fallen zusammen.

1. Jeder Vorgang auf einem nicht-kritischen Pfad weist als Gesamtpuffer den gesamten Puffer eines Pfades aus. Wenn ein beliebiger Vorgang auf diesem Pfad seinen Gesamtpuffer verbraucht, so werden alle verbleibenden Vorgänge dieses Pfades kritisch.
2. Die Summe der Freien Vorgangspuffer auf diesem Pfad ist gerade gleich der zur Verfügung stehenden Pfadpufferzeit. Jeder Vorgang kann also seinen Freien Puffer verbrauchen, ohne den Status der anderen Vorgänge dieses Pfades zu beeinflussen. Allerdings sind diese Puffer nicht sehr sinnvoll auf die einzelnen Vorgänge verteilt.

Will man die Vorgangspuffer zum Steuern des Projektablaufes verwenden, empfiehlt es sich deshalb, die zur Verfügung stehenden Freien Puffer sinnvoll auf die Vorgänge zu verteilen.

9.4.3 Kapazitätsplanung

Bei dem Standardverfahren der Netzplantechnik werden die Vorgangszeiten unter dem Gesichtspunkt eines wirtschaftlichen Produktionsmitteleinsatzes geschätzt bzw. errechnet.

Die Reihenfolge, in der die ursprünglich simultanen Vorgänge ablaufen, beeinflusst wesentlich die zu erreichende Projektzeit. Das Problem besteht hauptsächlich darin, die optimale Reihenfolge der Vorgänge zu finden, d.h. diejenige Reihenfolge zu bestimmen, die zu der kürzesten Projektdauer führt. Die Ermittlung optimaler oder nahoptimaler Reihenfolgen kann auf vier verschiedene Weisen durchgeführt werden:

1. durch Enumeration, d.h. Durchrechnung aller möglichen Reihenfolgen
2. durch verschiedene Näherungsverfahren
3. durch Entscheidungsbaumverfahren
4. durch lineares Programmieren.

9.4.4 Kostenplanung

Man unterscheidet 3 relevante Kosten:

1. Vorgangskosten:

zu denen im wesentlichen die variablen Kosten der Vorgangsdurchführung gehören, die vorgangsdauerabhängig sind (also, z.B. Löhne, verschleißabhängige Wertminderungen und Reparaturen etc.).

2. Projektkosten:

unter denen man im die wesentlichen Kosten versteht, die während der Dauer des Projektes in annähernd konstanter Höhe anfallen.

3. Verlängerungskosten:

die dadurch entstehen, dass sich die Projektdauer über einen vorgeplanten Zeitpunkt hinaus verlängert (z.B. Konventionalstrafen, Mietausfall, entgangener Gewinn ect.). Die Kostenbetrachtung vollzieht sich in 4 Stufen:

- (a) Bestimmung der Zeit-Kosten-Relation der einzelnen Vorgänge
- (b) Bestimmung der Zeit-Kosten-Relation der Projekt- und Verlängerungskosten
- (c) Bestimmung der Gesamtkosten-Projektzeit-Relation
- (d) Bestimmung der optimalen Projektdauer und des entsprechenden Netzplanes

9.5 CPM-Netzpläne

Ein CPM-Netzplan ist aus folgenden Elementen aufgebaut: *Vorgänge* werden mit Pfeilen (gerichteten Kanten) dargestellt. Beginn und Ende eines Vorgangs bilden die Ereignisse, die durch die Knoten am Beginn und Ende jeder Kante symbolisiert werden. Das Ende eines Vorgangs und der Beginn eines nachfolgenden Vorganges werden in der Regel durch denselben Knoten dargestellt. Der Knoten, der den Beginn des Projektes markiert, heißt auch Startereignis, der Knoten, bei dessen Erreichen das Projekt fertiggestellt ist, heißt *Zielereignis*. *Vorgänger* bzw. *Nachfolger* eines Vorganges nennt man solche Vorgänge, die dem betrachteten Vorgang unmittelbar vorausgehen bzw. nachfolgen. Damit ein Vorgang beginnen kann, müssen alle seine Vorgänger abgeschlossen sein. Scheinvorgänge dienen dazu, logische Abhängigkeiten zwischen Vorgängen darzustellen. Scheinvorgänge haben die Dauer Null.

Die Knoten (Ereignisse) des Netzplanes werden mit fortlaufenden Nummern $0, 1, 2, \dots, i, j, \dots, n$ bezeichnet. Die Nummer des Startereignisses sei stets 0, die des Zielereignisses stets n. Das Ereignis j möge (zeitlich) nach dem Ereignis i liegen. Falls für einen Vorgang Startknoten i und Endknoten j bekannt sind, wird der Vorgang einfach mit ij angesprochen.

Zeitanalyse im CPM-Netzplan

Für jeden Knoten i sowie für alle Vorgänge ij werden folgende Zeitangaben in den Netzplan eingetragen:

- FZ_i : Frühester Zeitpunkt, an dem Ereignis i eintreten kann
- SZ_i : Spätester Zeitpunkt, an dem Ereignis i eintreten kann
- d_{ij} : Dauer des Vorgangs ij mit Startereignis i und Endereignis j

Aus den frühesten bzw. spätesten Ereigniszeitpunkten läßt sich der kritische Weg im Netzplan leicht bestimmen. Der kritische Weg ist der zeitlängste Weg vom Start- zum Zielknoten und determiniert die Dauer des gesamten Projektes. Für alle Knoten i entlang des kritischen Weges gilt: $FZ_i = SZ_i$.

Jeder Vorgang auf dem kritischen Weg heißt ebenfalls kritisch, weil eine Verzögerung jedes kritischen Vorgangs bereits die Gesamtprojektdauer verlängert. Umgekehrt kann bei gleichbleibender Netzplanstruktur eine Projektbeschleunigung nur dann erreicht werden, wenn (auch) kritische Vorgänge verkürzt werden. Darüber hinaus kann man aus den frühesten bzw. spätesten Ereigniszeitpunkten FZ_i bzw. SZ_i auch sogenannte 'Pufferzeiten' für alle Vorgänge ermitteln. Pufferzeiten geben Auskunft über tolerierbare Verzögerungen bei der Dauer bzw. beim Beginn von Vorgängen. Die Gesamtpufferzeit (GP) eines Vorganges ist diejenige Zeitspanne, die ein Vorgang gegenüber seinem frühestmöglichem Beginn (bzw. vorgesehener Dauer) verzögert werden kann, ohne daß notwendigerweise eine Verzögerung des Projektes erfolgt. Die Freie Pufferzeit (FP) ist diejenige Zeitspanne, um die ein Vorgang gegenüber seiner frühesten Lage verschoben werden kann, ohne die früheste Lage anderer Vorgänge zu beeinflussen. Die Unabhängige Pufferzeit (UP) ist diejenige Zeitspanne, um die ein Vorgang verschoben werden kann, wenn seine Vorereignisse in spätestem und seine Nachereignisse sich in frühester Lage befinden.

Algorithmus zum Erstellen eines CPM-Netzplanes

1. Sämtliche Vorgänge samt Vorgangsdauer auflisten.
2. Für jeden Vorgang alle Vorgänger ermitteln.
3. Vorgänge so sortieren, daß alle Vorgänger eines jeden Vorgangs vor diesem Vorgang liegen. Ist eine solche Sortierung unmöglich, so sind die 'Vorgängerbedingungen' widersprüchlich
4. Zeichne einen Startknoten 0
5. Wiederhole für alle Vorgänge V in der Reihenfolge der Sortierung (3):
 - (a) Falls der betrachtete Vorgang V keinen Vorgänger hat, lasse man ihn beim Startknoten beginnen und gehe zu (d).
 - (b) Falls der betrachtete Vorgang genau einen Vorgänger hat, lasse ihn beim Endknoten des betreffenden Vorgangs beginnen und gehe zu (d).
 - (c) Falls der betrachtete Vorgang mehr als einen Vorgänger hat, so zeichne man von den bisher erstellten Endknoten Scheinvorgänge (mit Zeitdauer 0) derart zu einem neuen Anfangsknoten, daß in dem neuen Anfangsknoten genau alle Vorgänger des zu zeichnenden Vorganges abgeschlossen sind. Gehe zu (d).

- (d) Schreibe zum gezeichneten Vorgang seine Dauer und schließe ihn mit einem Endknoten ab.
6. Führe alle Endknoten, von denen noch kein Pfeil (Vorgang oder Scheinvorgang) ausgeht, durch Scheinvorgänge in einem Zielknoten n (Projektende) zusammen.
7. Ermittlung der frühesten Ereigniszeitpunkte FZ_i für sämtliche Knoten i :
- Startknoten 0 erhält $FZ_0 = 0$
 - Wiederhole für alle Knoten j :
 - Falls j Endknoten des echten Vorganges ij mit Dauer d_{ij} , so ist $FZ_j := FZ_i + d_{ij}$.
 - Falls j Endknoten mehrerer Scheinvorgänge (die in den Knoten s_1, s_2, \dots, s_k beginnen) ist, so ist sein frühester Ereigniszeitpunkt $FZ_j := \max FZ_{s_i}$
8. Ermittlung der spätesten Ereigniszeitpunkte SZ_i für alle Knoten i :
- Das Zielereignis n erhält $SZ_n = FZ_n$; für den Zielknoten sind frühester und spätester Ereigniszeitpunkt gleich.
 - Wiederhole für alle Knoten i ($i = n-1, \dots, 0$), in absteigender Reihenfolge:
 - Falls i Anfangsknoten eines einzigen Vorganges ij ist, so ist $SZ_i = SZ_j - d_{ij}$. (Für Scheinvorgänge ist $d_{ij} = 0$).
 - Falls i Anfangsknoten mehrerer (Schein-)Vorgänge, die in den Knoten s_1, s_2, \dots, s_k enden, so ist sein spätester Ereigniszeitpunkt $SZ_i = \min (SZ_{s_j} - d_{ij})$. Unter den in Frage kommenden Zeitpunkten ist bei der Rückwärtsrechnung der früheste (kleinste) zu nehmen, weil die Fertigstellung aller in Knoten i beginnenden Vorgänge bis zu den Zeitpunkten SZ_j gewährleistet sein muß.
9. Die Anzahl von Scheinvorgängen kann auch reduziert werden. Jeder Knoten i , von dem nur ein einziger Scheinvorgang ij ausgeht, kann mit dem Endknoten j dieses Scheinvorganges in folgender Weise verschmolzen werden:
- Die Scheinkante ij wird eliminiert
 - für alle übrigen in j ankommenden (ausgehenden) Kanten wird i zum Endpunkt (Ausgangspunkt).
 - Die FZ_i bzw. SZ_i erhalten die Werte von FZ_j bzw. SZ_j .
 - Knoten j ist zu streichen.

Durch Reduktion der Scheinkanzahl wird auch die Information über die Zeitabläufe im Netzplan komprimiert und damit weniger übersichtlich.

9.6 PERT-Netzpläne

- In der Praxis ist die Schätzung von Projektdauern eines Projektes mit großen Unsicherheiten verbunden. Deshalb ist es zweckmäßig, die Vorgangsdauern nicht mehr als deterministische Größen, sondern als Zufallsgrößen anzusehen. Hierfür ist die Netzplantechnik-Methode PERT (Program Evaluation and Review Technique) entwickelt worden.
- PERT verwendet ein Ereignisknotennetz
- Die Vorgangsdauern werden bei PERT als betaverteilte Zufallsgrößen angesehen
- Die Parameter der Beta-Verteilung für eine Vorgangsdauer D werden mit Hilfe folgenden Zeitschätzungen für D bestimmt:
 - Optimistischer Schätzwert OD von D (Dauer des Vorgangs unter besonders günstigen Bedingungen)
 - Pessimistischer Schätzwert PD von D (Dauer des Vorgangs unter besonders ungünstigsten Bedingungen, die nur im Fall höherer Gewalt überschritten werden kann)
 - Häufigste Vorgangsdauer HD von D (Dauer des Vorgangs unter normalen Bedingungen)

- Für den Erwartungswert ('Mittelwert') MD der Vorgangsdauer D erhält man dann

$$MD = \frac{OD + 4HD + PD}{6}$$

und für die Varianz (Streuung) VD der Vorgangsdauer D verwendet man die Näherungsformel

$$VD = \frac{(PD - OD)^2}{36}$$

- Die frühesten und spätesten Eintrittstermine FZ_i und SZ_i der Ereignisse i stellen jetzt Zufallsgrößen dar.
- Die Erwartungswerte MFZ_i und MSZ_i dieser Zufallsgrößen bestimmt man bei PERT wie bei CPM, nur ersetzt man die Größen D_{ij} , FZ_i und SZ_i durch MD_{ij} , MFZ_i und MSZ_i .
- Diese Vorgehensweise führt jedoch zu fehlerhaften Ergebnissen, da man hierbei in unzulässiger Weise die Operationen 'Erwartungswertbildung' und 'Maximierung' bzw. 'Minimierung' vertauscht.
- Als Varianz der Länge eines Weges W in einem PERT-Netzplan legt man die Summe der Varianzen der Dauer der einzelnen Vorgänge des Weges W fest. Dies ist im

allgemeinem nur richtig, wenn die einzelnen Vorgänge von W unkorreliert sind (was der Fall, ist wenn die einzelnen Vorgangsdauern von W voneinander unabhängig sind).

- Auf diese Weise kann man (näherungsweise) die Erwartungswerte und Varianzen aller ereignisbezogenen Termine FZ_i und SZ_i ermitteln.
- Als **kritischer Weg** wird bei PERT ein WEG von der Quelle 1 zur Senke n im PERT-Netzplan mit der größten erwarteten Weglänge MFZ_n bezeichnet. Als Erwartungswert und Varianz der kürzesten Projektdauer werden Erwartungswert MFZ_n bzw. Varianz VFZ_n von FZ_n verwendet.
- Zentraler Grenzwertsatz der Wahrscheinlichkeitsrechnung: Die Summe von unabhängigen Zufallsgrößen ist, wenn die Anzahl der Summanden groß ist, unter in der Praxis im allgemeinen erfüllten Voraussetzungen näherungsweise normalverteilt.
- In Anlehnung an den Satz nimmt man bei PERT die Zufallsgröße FZ_i (für große i) und SZ_i (für kleine i) als normalverteilt mit den Erwartungswerten MFZ_i bzw. MSZ_i und den Varianzen VFZ_i und VSZ_i an (obwohl die einzelnen Vorgänge eines Weges in der Regel nicht voneinander unabhängig sind und nicht nur jeweils ein Weg diese Zeitpunkte bestimmt)
- Insbesondere ist dann die kürzeste Projektdauer normalverteilt mit dem Erwartungswert MFZ_n und der Varianz VFZ_n
- Da Ereignisse im Vordergrund stehen, werden nicht Pufferzeiten, sondern der Schlupf je Ereignis berechnet

$$SL(i) = SZ(i) - FZ(i)$$

- Varianz des Schlupfs:

$$\sigma^2[SL(i)] = \sigma^2[FZ(i)] + \sigma^2[SZ(i)]$$

- Mit Hilfe dieser Wahrscheinlichkeitsverteilungen kann man die Wahrscheinlichkeit von Terminüberschreitungen bzw. das Risiko für die Einhaltung gewisser vorgegebener Termine ermitteln (also Größen, die für die Projektüberwachung wichtig sind)
- **Nachteil von CPM und PERT:** Mit Hilfe der in den Netzplantechnik-Methoden CPM und PERT verwendeten zyklensfreien Vorgangspfeilnetze (sowie auch mit MPM-Netzplänen) können nur Projekte modelliert werden, deren zeitlicher Ablauf von vornherein eindeutig festgelegt ist. Hierbei wird während einer Projektausführung jedes Ereignis des Projektes genau einmal realisiert, und es treten während des Projektablaufs keine Rücksprünge zu bereits zuvor einmal ausgeführten Vorgängen auf.