

Vorlesung 09: Netzplantechnik

Grundlagen Operations Research

Netzplantechnik

Ein Projekt planen

Was ist ein Projekt?

Ein zeitlich, räumlich und sachlich **begrenztes** komplexes Arbeitsvorhaben mit einer bestimmten Zielsetzung.

i Netzplantechnik

Zeitliche Planung der **Vorgänge** eines Projekts: Bestimmung der **frühesten** und **spätesten** Anfangs- und Endzeitpunkte.

Methoden im Überblick

- **CPM**: Critical Path Method (USA, 1956), **vorgangspfeilorientiert**.
- **MPM**: Metra Potential Method (Frankreich, 1957), **vorgangsknotenorientiert**.
- **PERT**: stochastische Vorgangsdauern (Polaris-Rakete).
- **GERT / Simulation**: stochastische Netzstrukturen (F&E-Projekte).

Wir betrachten den **MPM**-Netzplan: Vorgänge sind **Knoten**, Pfeile tragen Mindestabstände.

Vom Projekt zum Netzplan

Zwei Phasen der Netzplantechnik:

- **Strukturplanung**: Vorgänge, Dauern und Anordnungsbeziehungen erheben; daraus entsteht der Netzplan.
- **Zeitplanung**: früheste und späteste Zeitpunkte, Puffer und kritischen Weg berechnen.

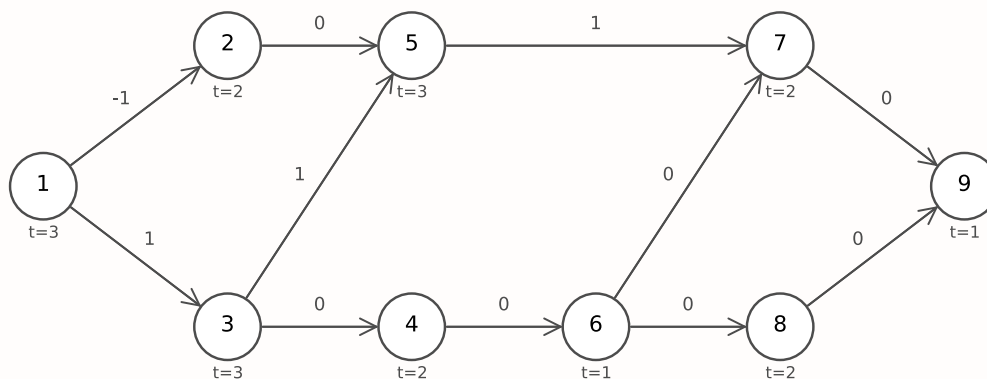
Mit einem künstlichen **Startvorgang** $i = 0$ und **Endvorgang** $i = n$ hat jedes Projekt eine eindeutige Quelle und Senke.

Beispiel: Bau einer Lagerhalle

i	Vorgang	t_i	Vorgänger ($d_{h,i}$)
1	Fundamente	3	–
2	Erdleitung	2	1 (–1)
3	Mauern	3	1 (1)

i	Vorgang	t_i	Vorgänger ($d_{h,i}$)
4	Dach	2	3 (0)
5	Boden	3	2 (0), 3 (1)
6	Tore einsetzen	1	4 (0)
7	Innen verputzen	2	5 (1), 6 (0)
8	Außen verputzen	2	6 (0)
9	Tor streichen	1	7 (0), 8 (0)

Der MPM-Netzplan



Knoten = Vorgänge (mit Dauer t_i), Pfeile = Mindestabstände $d_{h,i}$.

Mindest- und Maximalabstände

Zwischen zwei Vorgängen h und i sind verschiedene Abstände möglich:

- **Anfangsfolge** d_{hi}^A : Anfang h bis Anfang i .
- **Normalfolge** d_{hi} : Ende h bis Anfang i .
- **Endfolge** d_{hi}^E : Ende h bis Ende i .
- **Sprungfolge** d_{hi}^S : Anfang h bis Ende i .

Jeder Abstand existiert auch als **Maximalabstand** $\bar{d}_{hi}^A, \bar{d}_{hi}, \bar{d}_{hi}^E, \bar{d}_{hi}^S$: „höchstens so lange“.

Transformationsregeln

Alles wird in **Normalfolgen** umgerechnet; Maximalabstände werden zu **negativen** Mindestabständen in **Gegenrichtung**:

gegeben	Umrechnung	gegeben	Umrechnung
d_{hi}^A	$d_{hi} = d_{hi}^A - t_h$	$\bar{d}_{hi}^A$	$d_{ih} = -\bar{d}_{hi}^A - t_i$

gegeben	Umrechnung	gegeben	Umrechnung
d_{hi}	d_{hi}	$\bar{d}_{hi}$	$d_{ih} = -\bar{d}_{hi} - t_h - t_i$
d_{hi}^E	$d_{hi} = d_{hi}^E - t_i$	$\bar{d}_{hi}^E$	$d_{ih} = -\bar{d}_{hi}^E - t_h$
d_{hi}^S	$d_{hi} = d_{hi}^S - t_h - t_i$	$\bar{d}_{hi}^S$	$d_{ih} = -\bar{d}_{hi}^S$

So entstehen **Rückwärtspeile** und damit **Zyklen** im Netzplan.

Kurz geübt: Umrechnung

Zwischen h ($t_h = 2$) und i ($t_i = 3$) gilt der maximale Normalabstand $\bar{d}_{hi} = 10$.

Frage: **Welcher Pfeil entsteht daraus im Netzplan?**

$$d_{ih} = -\bar{d}_{hi} - t_h - t_i = -10 - 2 - 3 = -15$$

Ein Rückwärtspeil von i nach h mit Bewertung -15 .

Zeitplanung

Vorwärtsrechnung

Früheste Zeitpunkte, topologisch von $i = 0$ bis n :

$$\text{FAZ}_0 := 0, \quad \text{FAZ}_i := \max_{h \in \mathcal{V}(i)} \{\text{FEZ}_h + d_{hi}\}, \quad \text{FEZ}_i := \text{FAZ}_i + t_i$$

FAZ_i / FEZ_i : **frühester** Anfangs- / Endzeitpunkt von Vorgang i .

Kurz geübt: Vorwärts

Vorgang i hat drei Vorgänger:

Vorgänger h	FEZ_h	d_{hi}
a	6	3
b	5	6
c	12	-4

Frage: **Wie groß ist FAZ_i ?**

$$\text{FAZ}_i = \max\{6 + 3, 5 + 6, 12 - 4\} = 11$$

Rückwärtsrechnung

Späteste Zeitpunkte, von $i = n$ zurück bis 0:

$$\text{SEZ}_n := \text{FEZ}_n, \quad \text{SEZ}_i := \min_{j \in \mathcal{N}(i)} \{\text{SAZ}_j - d_{ij}\}, \quad \text{SAZ}_i := \text{SEZ}_i - t_i$$

SAZ_i / SEZ_i : **spätester** Anfangs- / Endzeitpunkt von Vorgang i .

Kurz geübt: Rückwärts

Vorgang i hat zwei Nachfolger:

Nachfolger j	SAZ_j	d_{ij}
a	14	2
b	12	-2

Frage: **Wie groß ist SEZ_i ?**

$$SEZ_i = \min\{14 - 2, 12 - (-2)\} = 12$$

Der Terminplan

Erst die Vorwärts-, dann die Rückwärtsrechnung:

i	1	2	3	4	5	6	7	8	9
FAZ_i	0	2	4	7	8	9	12	10	14
FEZ_i	3	4	7	9	11	10	14	12	15

i	1	2	3	4	5	6	7	8	9
FAZ_i	0	2	4	7	8	9	12	10	14
FEZ_i	3	4	7	9	11	10	14	12	15
SAZ_i	0	6	4	9	8	11	12	12	14
SEZ_i	3	8	7	11	11	12	14	14	15

i	1	2	3	4	5	6	7	8	9
FAZ_i	0	2	4	7	8	9	12	10	14
FEZ_i	3	4	7	9	11	10	14	12	15
SAZ_i	0	6	4	9	8	11	12	12	14
SEZ_i	3	8	7	11	11	12	14	14	15

Vorwärts, z. B. $FAZ_5 = \max\{FEZ_2 + 0, FEZ_3 + 1\} = \max\{4, 8\} = 8$.

Rückwärts, z. B. $SEZ_3 = \min\{SAZ_4 - 0, SAZ_5 - 1\} = \min\{9, 7\} = 7$.

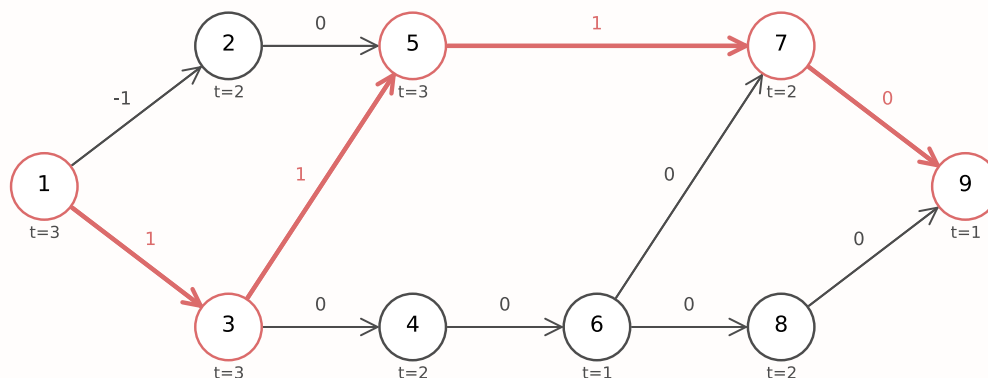
Projektdauer $FEZ_9 = 15$; bei 1, 3, 5, 7, 9 gilt $FAZ_i = SAZ_i$, dazu gleich mehr.

Kritischer Weg & Puffer

Der kritische Weg

i Kritischer Weg

Der **längste** Weg im Netzplan. Für jeden **kritischen** Vorgang gilt $FAZ_i = SAZ_i$ (kein Spielraum).



Kritische Vorgänge

Der kritische Weg ist $1 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9$.

! Konsequenz

Verzögert sich ein kritischer Vorgang, verzögert sich das **gesamte Projekt**.

Pufferzeiten

Zeitspannen, um die ein Vorgang verschoben werden kann, **ohne** die Projektdauer zu erhöhen:

- **Gesamte Pufferzeit:** $GP_i = SAZ_i - FAZ_i$.
- **Freie Pufferzeit:** $FP_i = \min_{j \in \mathcal{N}(i)} \{FAZ_j - d_{ij}\} - FEZ_i$: Spielraum, wenn **alle Nachfolger** frühestmöglich beginnen.

Weitere Pufferzeiten

- **Freie Rückwärtspufferzeit:** $FRP_i = SAZ_i - \max_{h \in \mathcal{V}(i)} \{SEZ_h + d_{hi}\}$: Spielraum, wenn alle **Vorgänger** spätestmöglich enden.
- **Unabhängige Pufferzeit:** $UP_i = \max\{0, FP_i + FRP_i - GP_i\}$: Spielraum **unabhängig** von allen anderen Vorgängen.

Es gilt stets $UP_i \leq FP_i, FRP_i \leq GP_i$.

Puffer im Beispiel

i	1	2	3	4	5	6	7	8	9
GP_i	0	4	0	2	0	2	0	2	0
FP_i	0	4	0	0	0	0	0	2	0
FRP_i	0	4	0	2	0	0	0	0	0
UP_i	0	4	0	0	0	0	0	0	0

Genau die Vorgänge mit $GP_i = 0$ sind **kritisch**: 1, 3, 5, 7, 9.

Nur Vorgang 2 (Erdleitung) ist völlig frei verschiebbar ($UP_2 = 4$).

Netzpläne mit Zyklen

Wenn Maximalabstände gelten

Enthält der Netzplan Zyklen (z. B. durch **Maximalabstände**), versagt die topologische Reihenfolge.

Dann hilft ein **FIFO-Verfahren** wie bei den kürzesten Wegen: erst Vorwärts-, dann Rückwärtsrechnung, Knoten bei jeder Aktualisierung erneut in die FIFO-Liste aufnehmen.

Voraussetzung: alle Zyklen haben **nichtpositive Länge**. Ein Zyklus positiver Länge macht den Plan **unzulässig**.

Beispiel: Maximalabstand

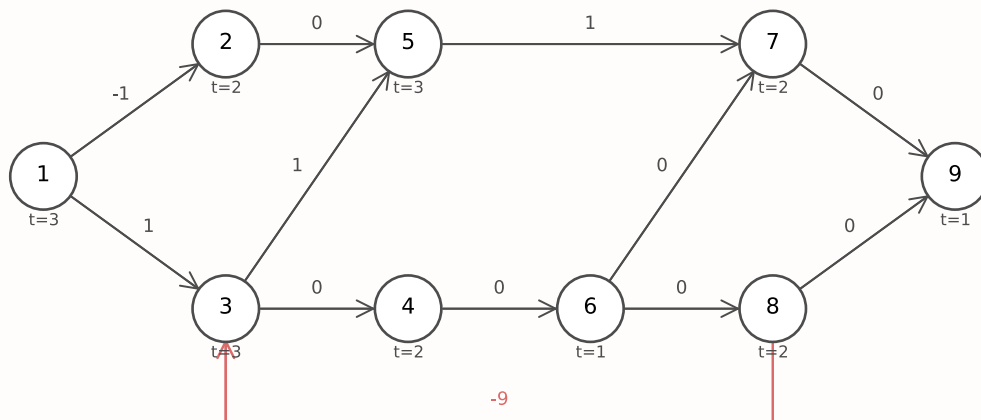
Neue Vorgabe für die Lagerhalle: Vom **Ende des Mauerns** (3) bis zur **Fertigstellung des Außenputzes** (8) dürfen höchstens **6 ZE** vergehen: $\bar{d}_{3,8}^E = 6$.

Frage: **Welcher Pfeil entsteht daraus?**

Transformationsregel $d_{ih} = -\bar{d}_{hi}^E - t_h$:

$$d_{8,3} = -6 - t_3 = -6 - 3 = -9$$

Der erweiterte Netzplan



Der Zyklus $3 \rightarrow 4 \rightarrow 6 \rightarrow 8 \rightarrow 3$ hat die Länge $3 + 0 + 2 + 0 + 1 + 0 + 2 - 9 = -1 \leq 0$: der Plan bleibt **zulässig**.

Neue späteste Zeitpunkte

Die Vorwärtsrechnung ändert sich nicht ($\text{FEZ}_8 - 9 = 3 < 4$), aber **rückwärts** zieht der Pfeil die späten Zeitpunkte zusammen:

i	1	2	3	4	5	6	7	8	9
SAZ_i (ohne)	0	6	4	9	8	11	12	12	14
SAZ_i (mit Zy- klus)	0	6	4	8	8	10	12	11	14

$\text{SEZ}_8 = \min\{\text{SAZ}_9 - 0, \text{SAZ}_3 - (-9)\} = \min\{14, 13\} = 13$: die Puffer von 4, 6, 8 schrumpfen auf 1.

Kontrolle mit Julia

Projektdauer als LP

Die Projektdauer ist der längste Weg, als LP formuliert:

```
using JuMP, HiGHS
```

```
t = Dict{1=>3,2=>2,3=>3,4=>2,5=>3,6=>1,7=>2,8=>2,9=>1}
# (Vorgänger h, Vorgang i, Mindestabstand d_hi)
kanten = [(1,2,-1),(1,3,1),(3,4,0),(2,5,0),(3,5,1),(4,6,0),
```

```

(5,7,1),(6,7,0),(6,8,0),(7,9,0),(8,9,0)]

modell = Model(HiGHS.Optimizer); set_silent(modell)
@variable(modell, S[keys(t)] >= 0) # Anfangszeitpunkte (FAZ)
@variable(modell, C >= 0) # Projektdauer
@constraint(modell, [(h,i,d) in kanten], S[i] >= S[h] + t[h] + d)
@constraint(modell, [i in keys(t)], C >= S[i] + t[i])
@objective(modell, Min, C)

optimize!(modell)
println("Projektdauer = ", objective_value(modell))

```

```

Projektdauer = 15.0

```

Vorwärtsrechnung als LP

Frühestmögliche Zeitpunkte: **minimiere** die Summe aller Endzeitpunkte:

```

tv = [3, 2, 3, 2, 3, 1, 2, 2, 1]

modell = Model(HiGHS.Optimizer); set_silent(modell)
@variable(modell, FAZ[1:9] >= 0)
@constraint(modell, FAZ[1] == 0)
@constraint(modell, [(h,i,d) in kanten], FAZ[i] >= FAZ[h] + tv[h] + d)
@objective(modell, Min, sum(FAZ[i] + tv[i] for i in 1:9)) # min Σ FEZ
optimize!(modell)
println("FAZ = ", round.(Int, value.(FAZ)))

```

```

FAZ = [0, 2, 4, 7, 8, 9, 12, 10, 14]

```

Rückwärtsrechnung als LP

Spätestmögliche Zeitpunkte: **maximiere**, bei fixiertem Projektende:

```

modell = Model(HiGHS.Optimizer); set_silent(modell)
@variable(modell, SAZ[1:9] >= 0)
@constraint(modell, SAZ[9] + tv[9] == 15) # SEZ_9 = FEZ_9*
@constraint(modell, [(h,i,d) in kanten], SAZ[i] >= SAZ[h] + tv[h] + d)
@objective(modell, Max, sum(SAZ))
optimize!(modell)
println("SAZ = ", round.(Int, value.(SAZ)))

```

```

SAZ = [0, 6, 4, 9, 8, 11, 12, 12, 14]

```

Beide Vektoren stimmen mit dem **Terminplan** von vorhin überein.

Zusammenfassung

Das Wichtigste auf einen Blick

- **MPM-Netzplan**: **Vorwärtsrechnung** liefert früheste, **Rückwärtsrechnung** späteste Zeitpunkte.
- Der **kritische Weg** ist der längste Weg; kritische Vorgänge haben $GP_i = 0$.
- **Pufferzeiten** (GP, FP, FRP, UP) zeigen den Spielraum unkritischer Vorgänge.
- **Maximalabstände** werden zu negativen Rückwärtspfeilen; die Zyklen löst ein **FIFO-Verfahren**.

Literatur

Literaturverzeichnis

Bibliografie