

Vorlesung 09: Netzplantechnik

Grundlagen Operations Research

Prof. Dr. Knut Haase

Netzplantechnik

Ein Projekt planen



Was ist ein Projekt?

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

Netzplantechnik

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

Methoden im Überblick

Methoden im Überblick

- **CPM**: Critical Path Method (USA, 1956), **vorgangspfeilorientiert**.

Methoden im Überblick

- **CPM**: Critical Path Method (USA, 1956), **vorgangspfeilorientiert**.
- **MPM**: Metra Potential Method (Frankreich, 1957), **vorgangsknotenorientiert**.

Methoden im Überblick

- **CPM**: Critical Path Method (USA, 1956), **vorgangspfeilorientiert**.
- **MPM**: Metra Potential Method (Frankreich, 1957), **vorgangsknotenorientiert**.
- **PERT**: stochastische Vorgangs**dauern** (Polaris-Rakete).

Methoden im Überblick

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

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:

Vom Projekt zum Netzplan

Zwei Phasen der Netzplantechnik:

- **Strukturplanung:** Vorgänge, Dauern und Anordnungsbeziehungen erheben; daraus entsteht der Netzplan.

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.

Vom Projekt zum Netzplan

Zwei Phasen der Netzplantechnik:

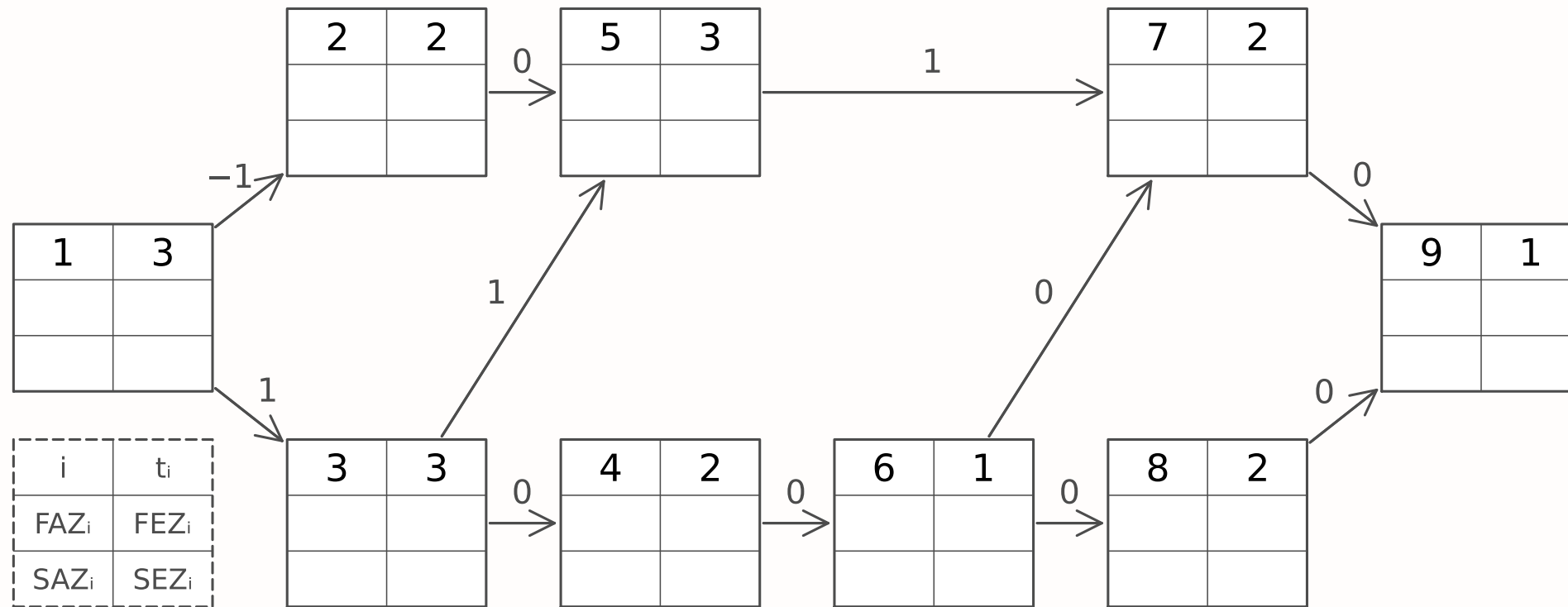
- **Strukturplanung:** Vorgänge, Dauern und Anordnungsbeziehungen erheben; daraus entsteht der Netzplan.
- **Zeitplanung:** früheste und späteste Zeitpunkte, Puffer und kritischen Weg berechnen.

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)
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 (oben i und t_i , die leeren Felder füllt gleich die Zeitplanung), Pfeile = Mindestabstände $d_{h,i}$.

Mindest- und Maximalabstände

Zwischen zwei Vorgängen h und i sind verschiedene Abstände möglich:

Mindest- und Maximalabstände

Zwischen zwei Vorgängen h und i sind verschiedene Abstände möglich:

- **Anfangsfolge** d_{hi}^A : Anfang h bis Anfang 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 .

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 .

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 .

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

Transformationsregeln

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

gegeben	Umrechnung	gegeben	Umrechnung
d_{hi}^A	$d_{hi} = d_{hi}^A - t_h$	$\bar{d}_{hi}^A$	$d_{ih} = -\bar{d}_{hi}^A - t_i$
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 **Rückwärtspfeile** und damit **Zyklen** im Netzplan.

Kurz geübt: Umrechnung

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

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?

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 Rückwärtspfeil von i nach h mit Bewertung -15 .

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 Rückwärtspfeil von i nach h mit Bewertung -15 .

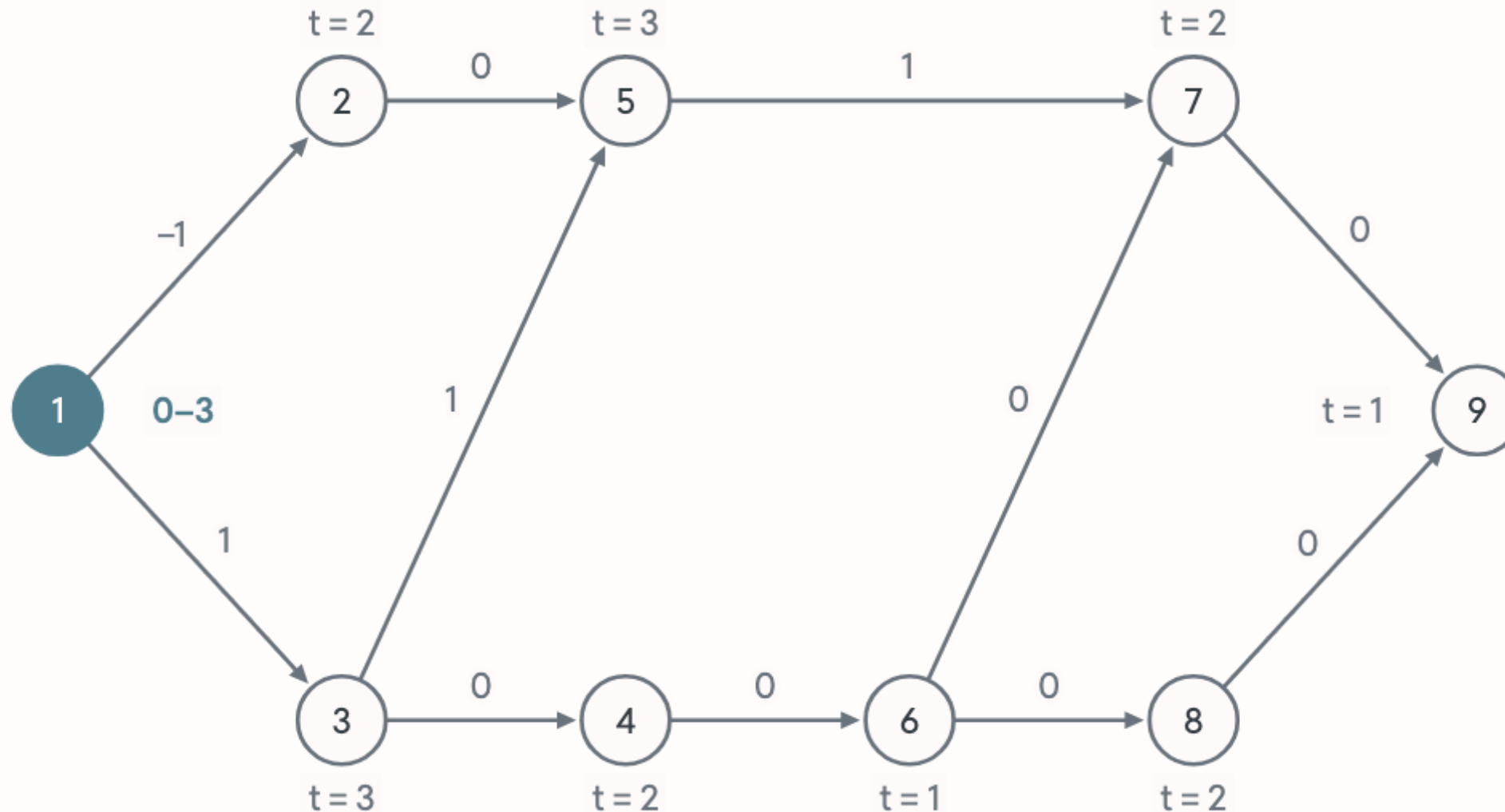


Gleich ausprobieren

Zurück zur Lagerhalle. Auf der nächsten Folie gehen Sie selbst Wege durch ihren Netzplan. Welcher Weg dauert am längsten?

Sie sind dran

Tippen Sie Pfeil für Pfeil einen Weg vom ersten zum letzten Vorgang. An jedem Vorgang steht seine Dauer t , auf dem Weg, wann er beginnt und endet. Finden Sie den Weg, der am längsten dauert.



Eigener Plan 3

Der Weg endet noch nicht bei Vorgang 9

Optimieren

Reset

Neu

Zeitplanung

Vorwärtsrechnung

Im Spiel haben Sie Wege einzeln durchgerechnet. Die Vorwärtsrechnung findet für jeden Vorgang den längsten Weg auf einmal, topologisch von $i = 0$ bis n :

$$\text{FAZ}_0 := 0$$

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

$$\text{FEZ}_i := \text{FAZ}_i + t_i$$

Vorwärtsrechnung

Im Spiel haben Sie Wege einzeln durchgerechnet. Die Vorwärtsrechnung findet für jeden Vorgang den längsten Weg auf einmal, topologisch von $i = 0$ bis n :

$$\text{FAZ}_0 := 0$$

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

$$\text{FEZ}_i := \text{FAZ}_i + t_i$$

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 ?

Kurz geübt: Vorwärts

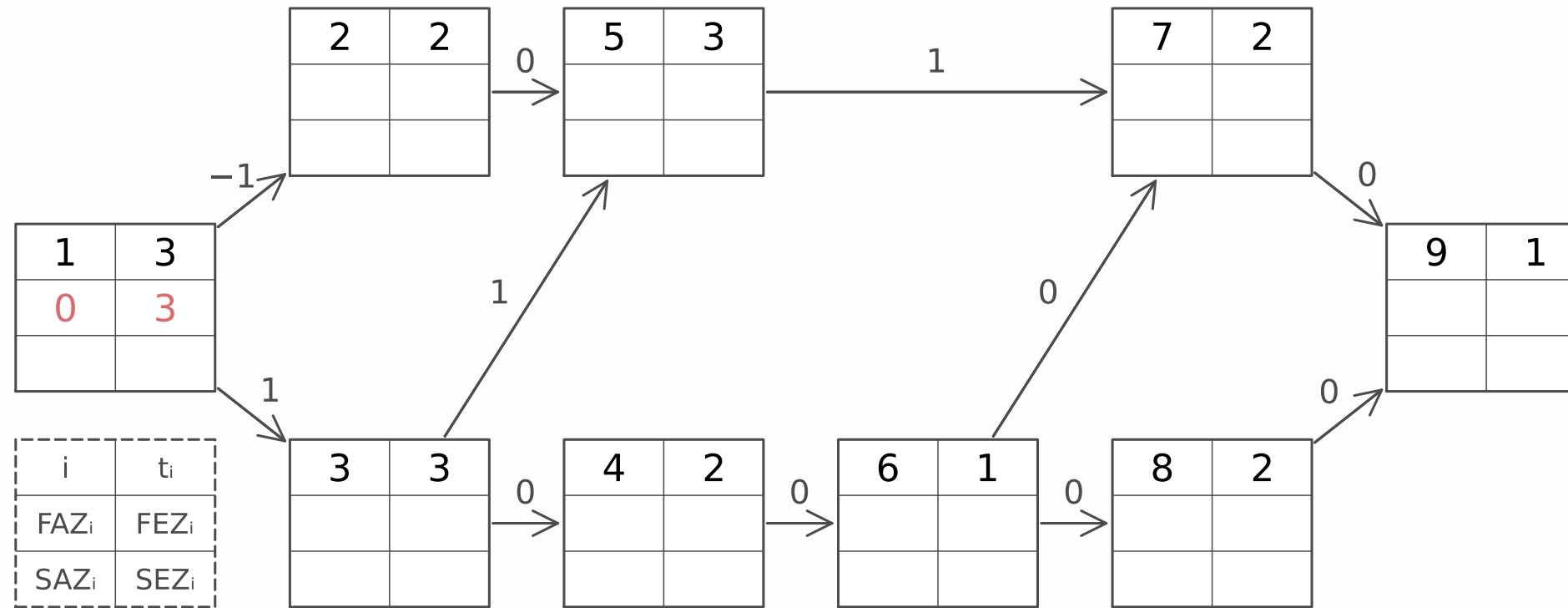
Vorgang i hat drei Vorgänger:

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

Frage: Wie groß ist FAZ_i ?

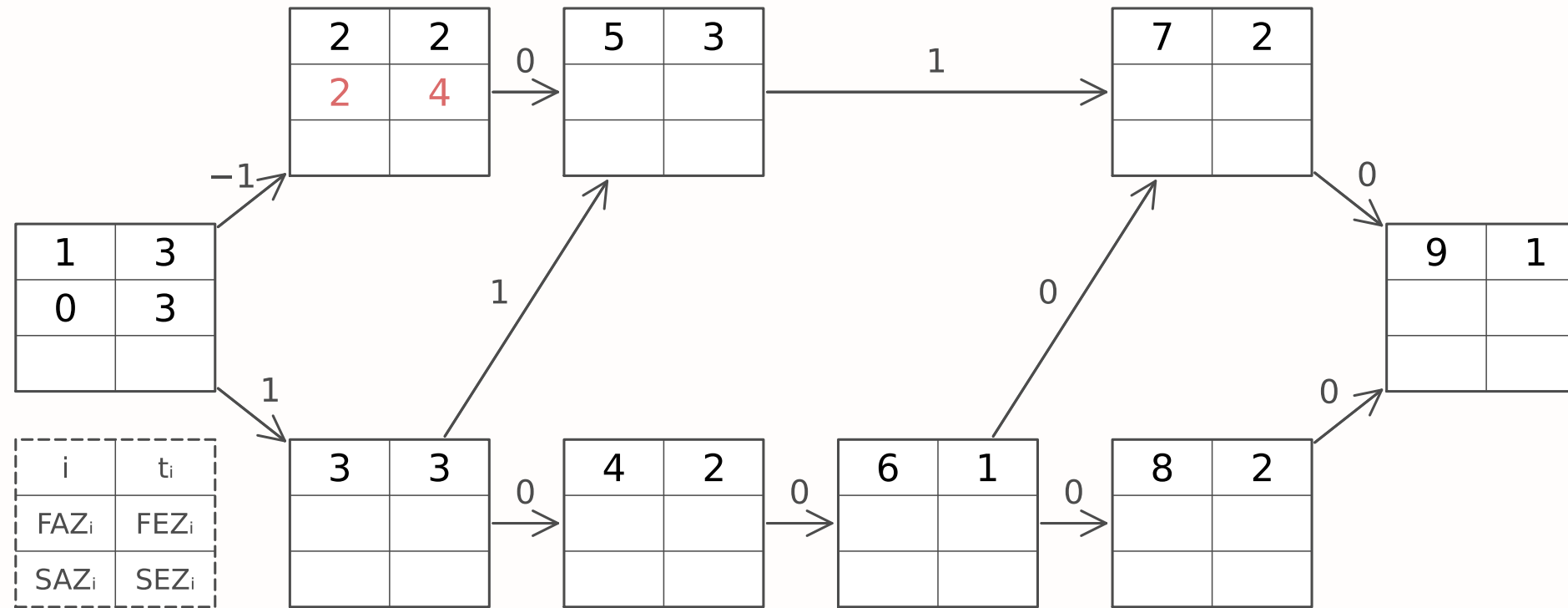
$$FAZ_i = \max\{6 + 3, 5 + 6, 12 - 4\} = \mathbf{11}$$

Vorwärts im Netzplan



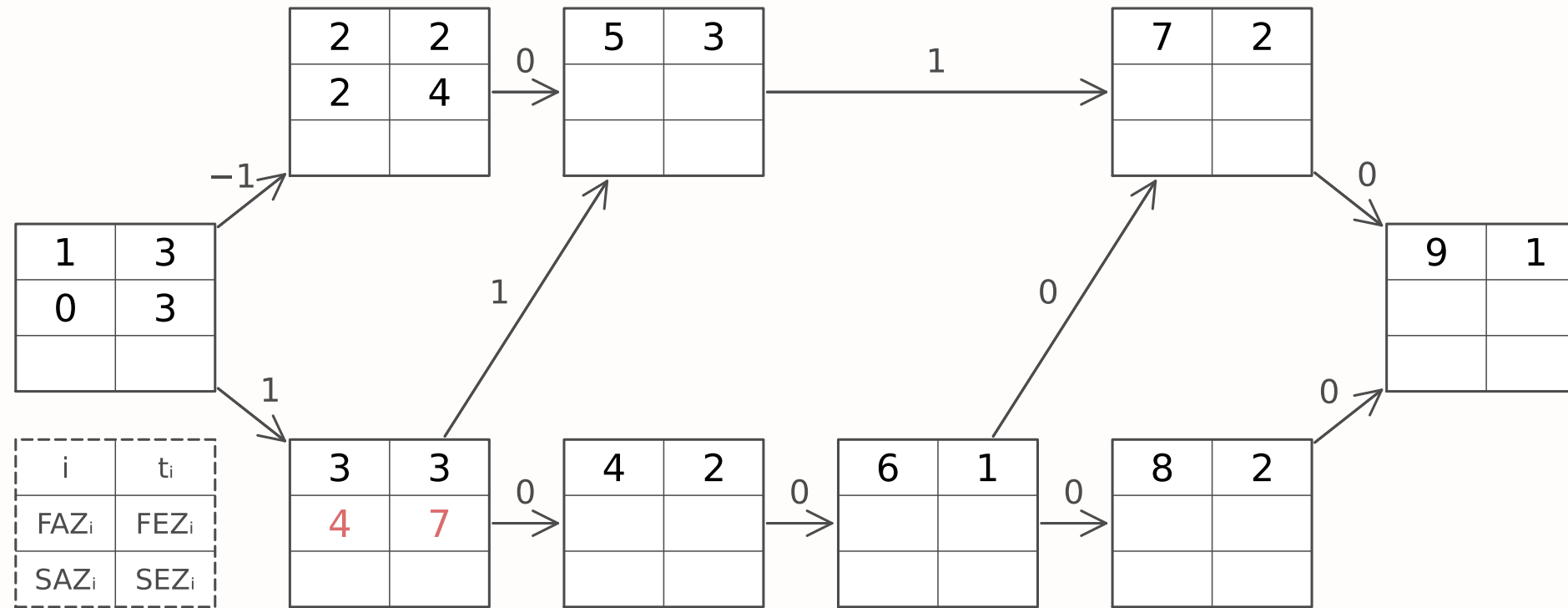
$$FAZ_1 = 0 \text{ (Projektstart)}, FEZ_1 = 0 + 3 = 3.$$

Vorwärts im Netzplan



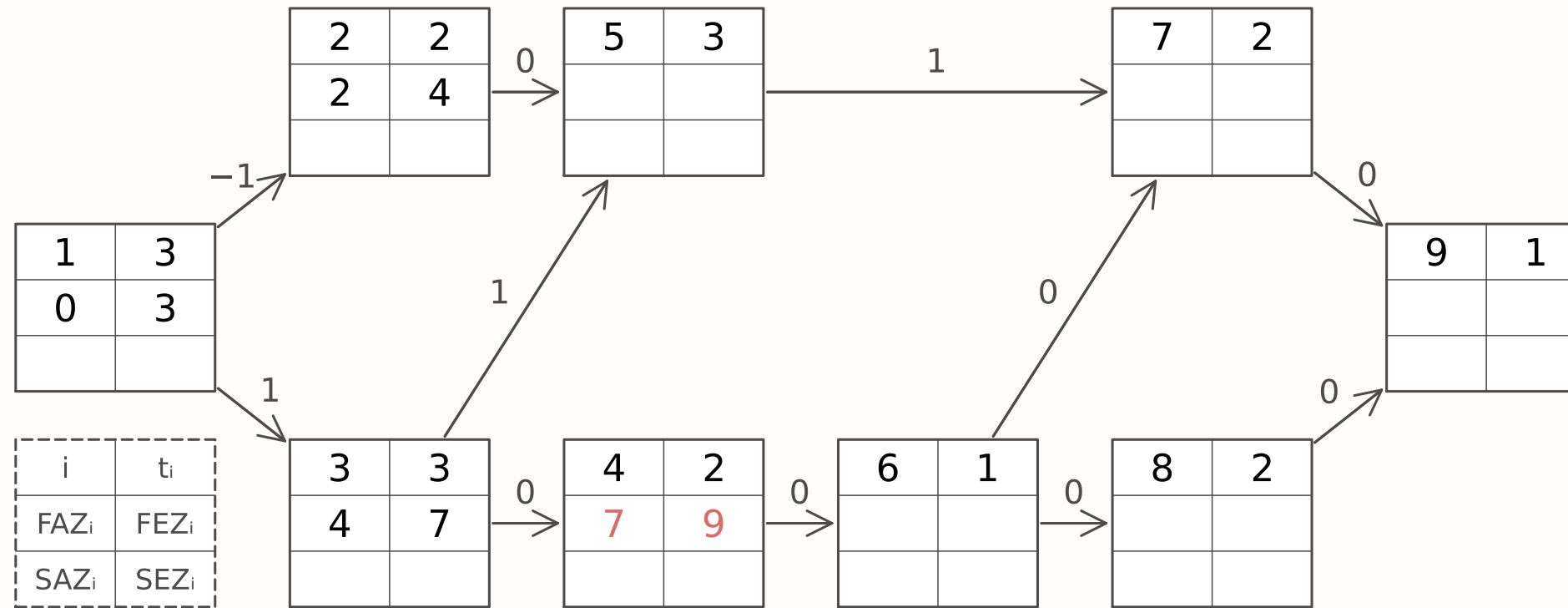
$$FAZ_2 = FEZ_1 + d_{1,2} = 3 - 1 = 2, FEZ_2 = 2 + 2 = 4.$$

Vorwärts im Netzplan



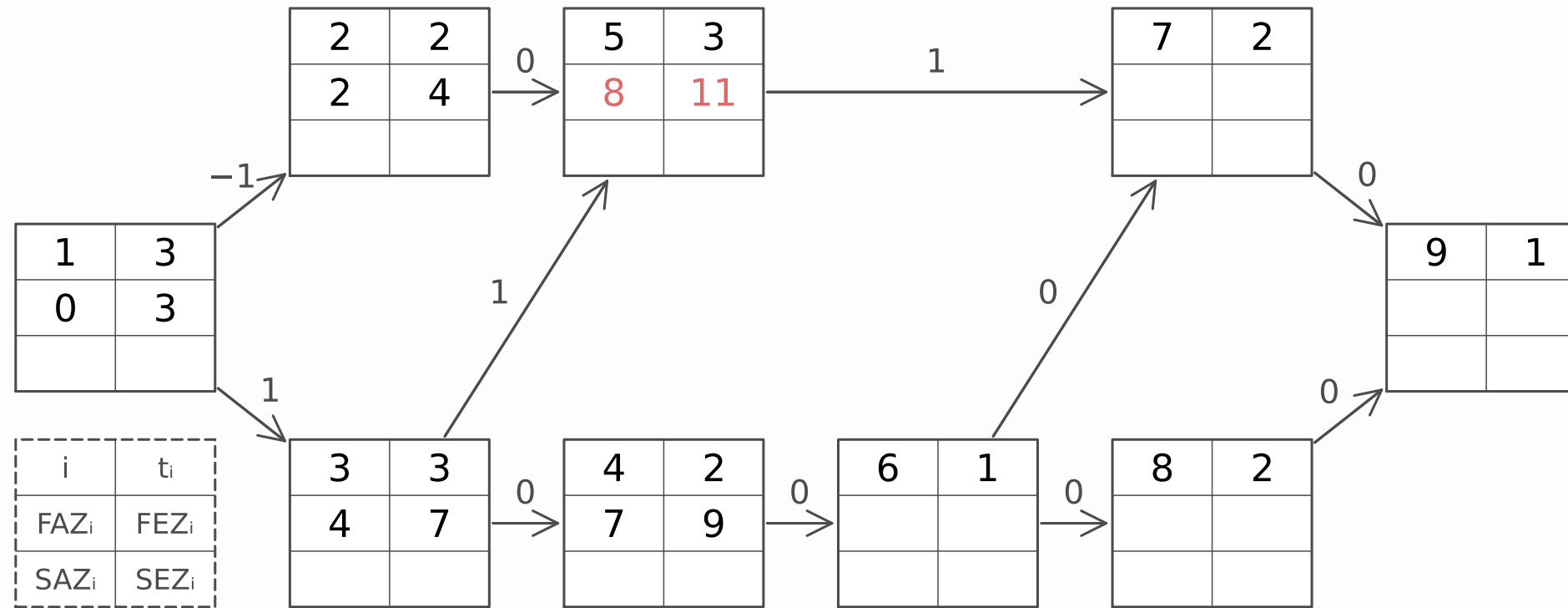
$$FAZ_3 = FEZ_1 + d_{1,3} = 3 + 1 = 4, FEZ_3 = 4 + 3 = 7.$$

Vorwärts im Netzplan



$$FAZ_4 = FEZ_3 + d_{3,4} = 7 + 0 = 7, FEZ_4 = 7 + 2 = 9.$$

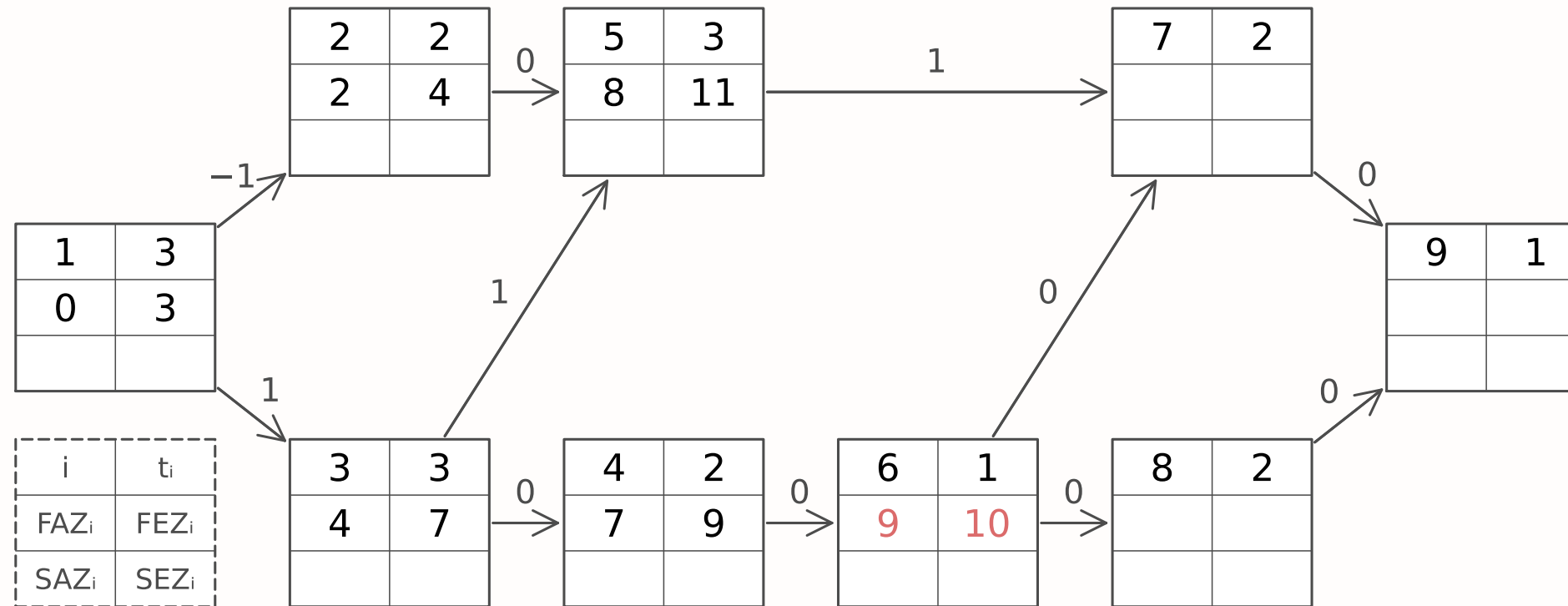
Vorwärts im Netzplan



$$\text{FAZ}_5 = \max\{\text{FEZ}_2 + d_{2,5}, \text{FEZ}_3 + d_{3,5}\} = \max\{4 + 0, 7 + 1\} = 8,$$

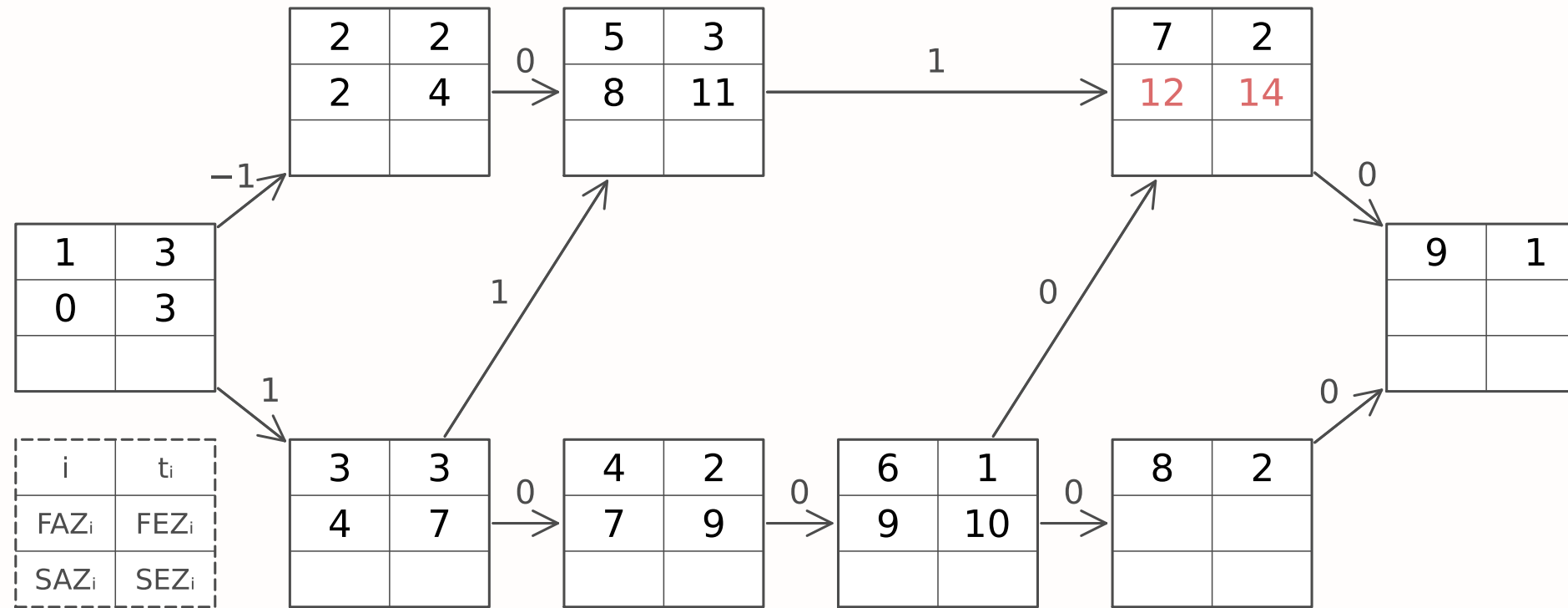
$$\text{FEZ}_5 = 8 + 3 = 11.$$

Vorwärts im Netzplan



$$FAZ_6 = FEZ_4 + d_{4,6} = 9 + 0 = 9, FEZ_6 = 9 + 1 = 10.$$

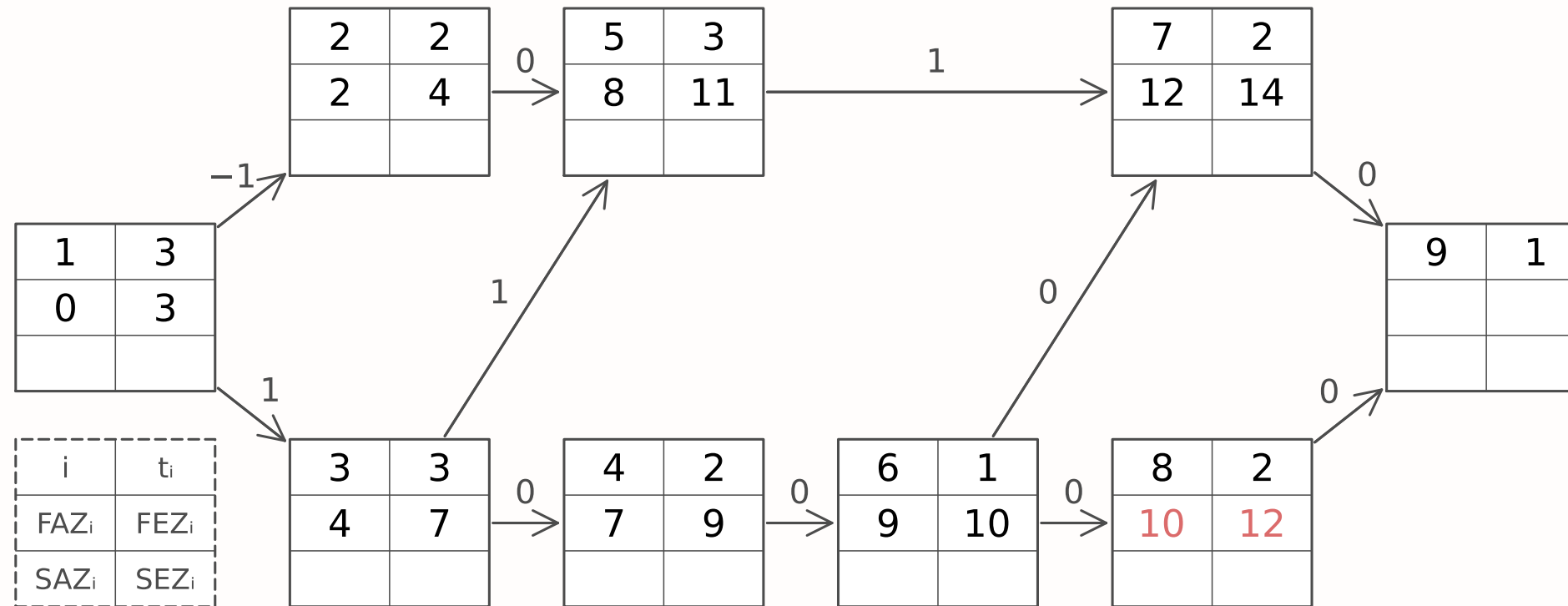
Vorwärts im Netzplan



$$\text{FAZ}_7 = \max\{\text{FEZ}_5 + d_{5,7}, \text{FEZ}_6 + d_{6,7}\} = \max\{11 + 1, 10 + 0\} = 12,$$

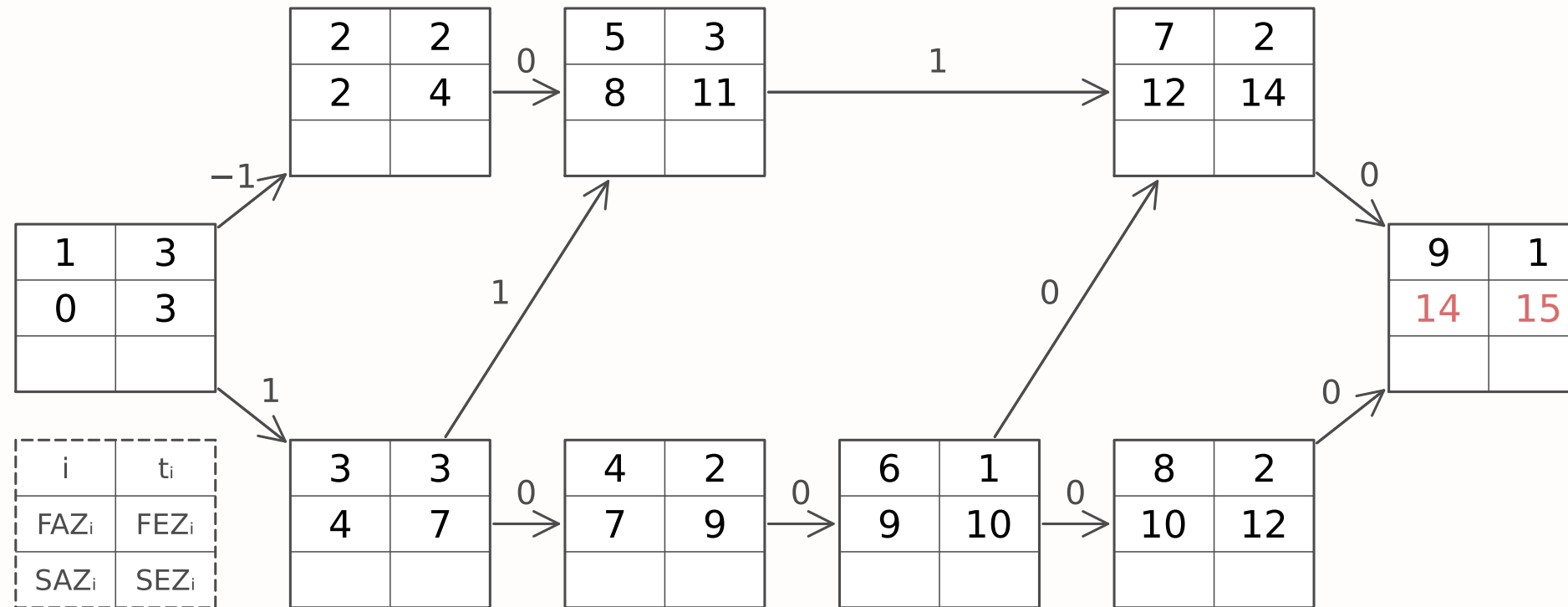
$$\text{FEZ}_7 = 12 + 2 = 14.$$

Vorwärts im Netzplan



$$FAZ_8 = FEZ_6 + d_{6,8} = 10 + 0 = 10, FEZ_8 = 10 + 2 = 12.$$

Vorwärts im Netzplan



$FAZ_9 = \max\{FEZ_7 + d_{7,9}, FEZ_8 + d_{8,9}\} = \max\{14 + 0, 12 + 0\} = 14,$
 $FEZ_9 = 14 + 1 = 15$: Projektdauer 15.

Rückwärtsrechnung

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

$$\text{SEZ}_n := \text{FEZ}_n$$

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

$$\text{SAZ}_i := \text{SEZ}_i - t_i$$

Rückwärtsrechnung

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

$$\text{SEZ}_n := \text{FEZ}_n$$

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

$$\text{SAZ}_i := \text{SEZ}_i - t_i$$

$\text{SAZ}_i / \text{SEZ}_i$: **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 ?

Kurz geübt: Rückwärts

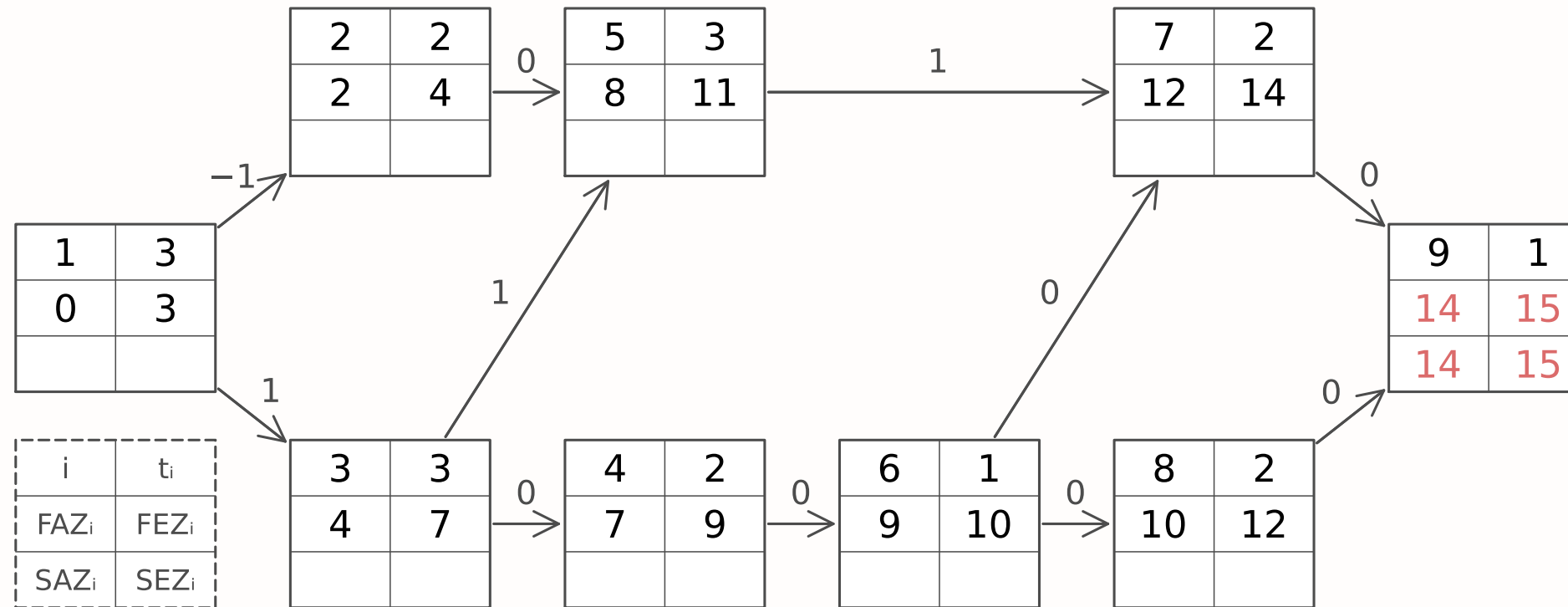
Vorgang i hat zwei Nachfolger:

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

Frage: Wie groß ist SEZ_i ?

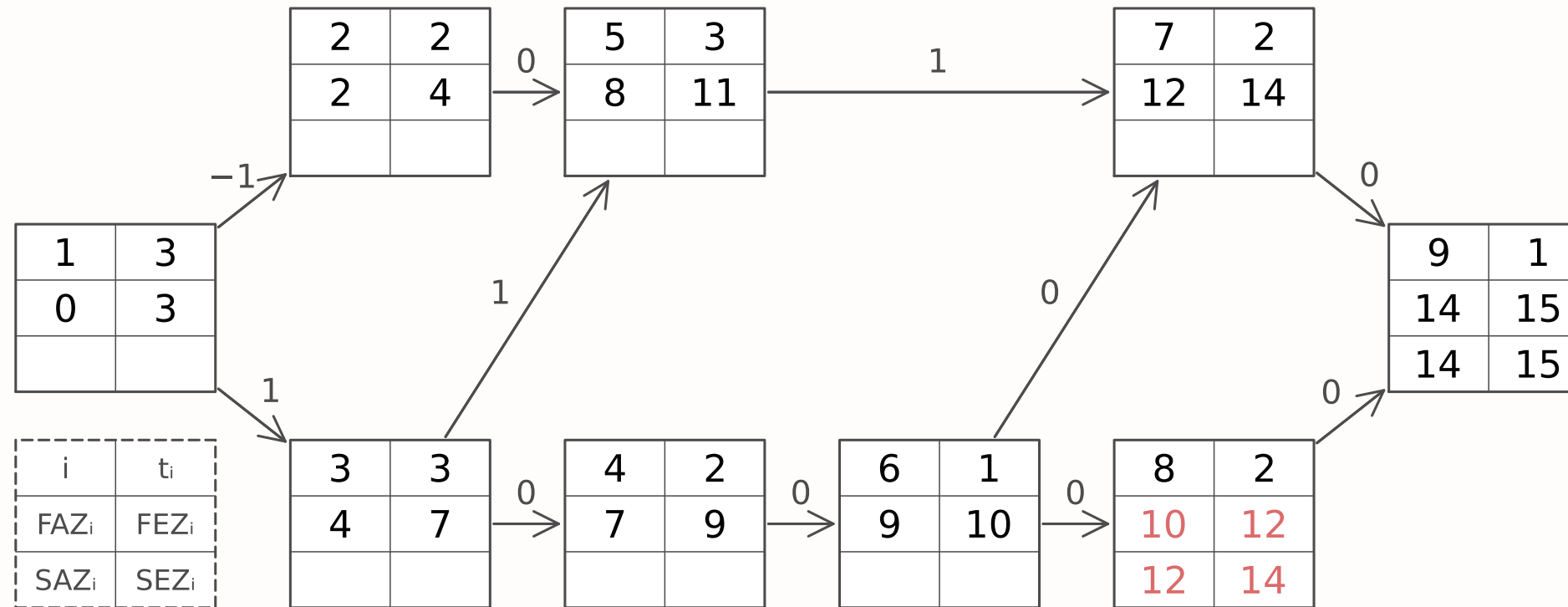
$$SEZ_i = \min\{14 - 2, 12 - (-2)\} = 12$$

Rückwärts im Netzplan



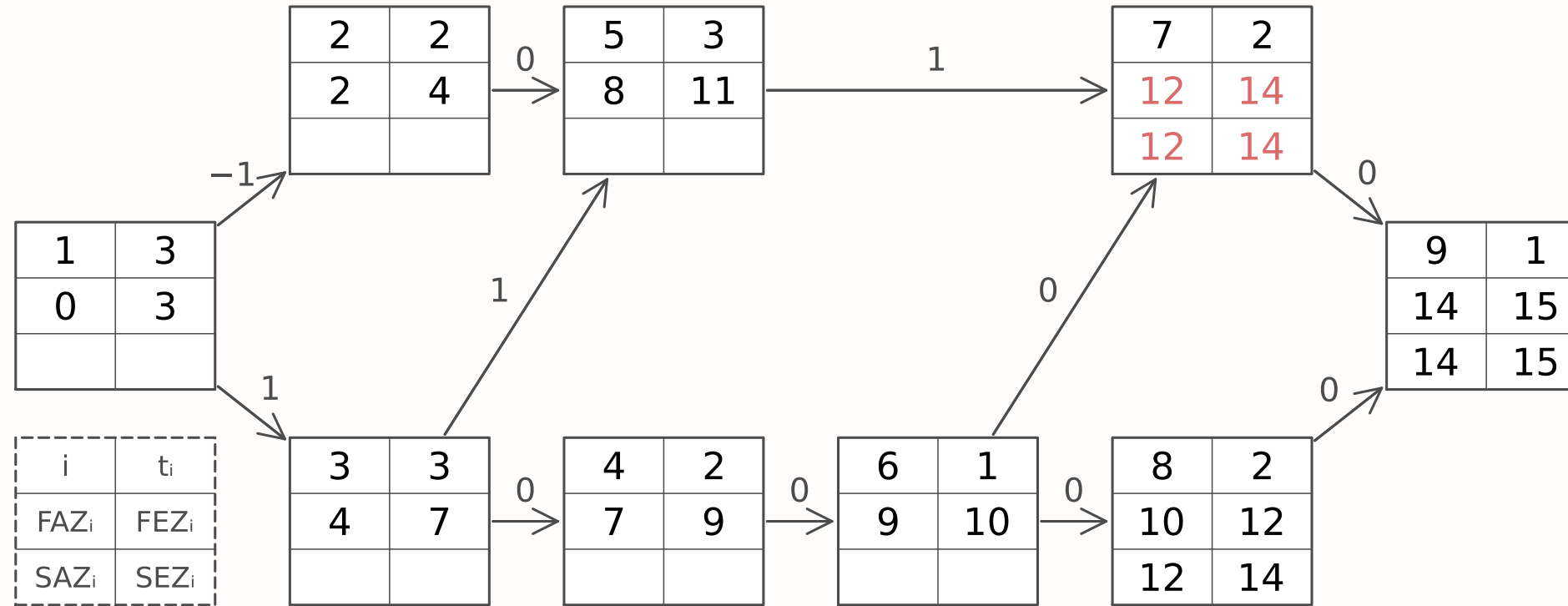
$SEZ_9 = FEZ_9 = 15$ (Projektende), $SAZ_9 = 15 - 1 = 14$.

Rückwärts im Netzplan



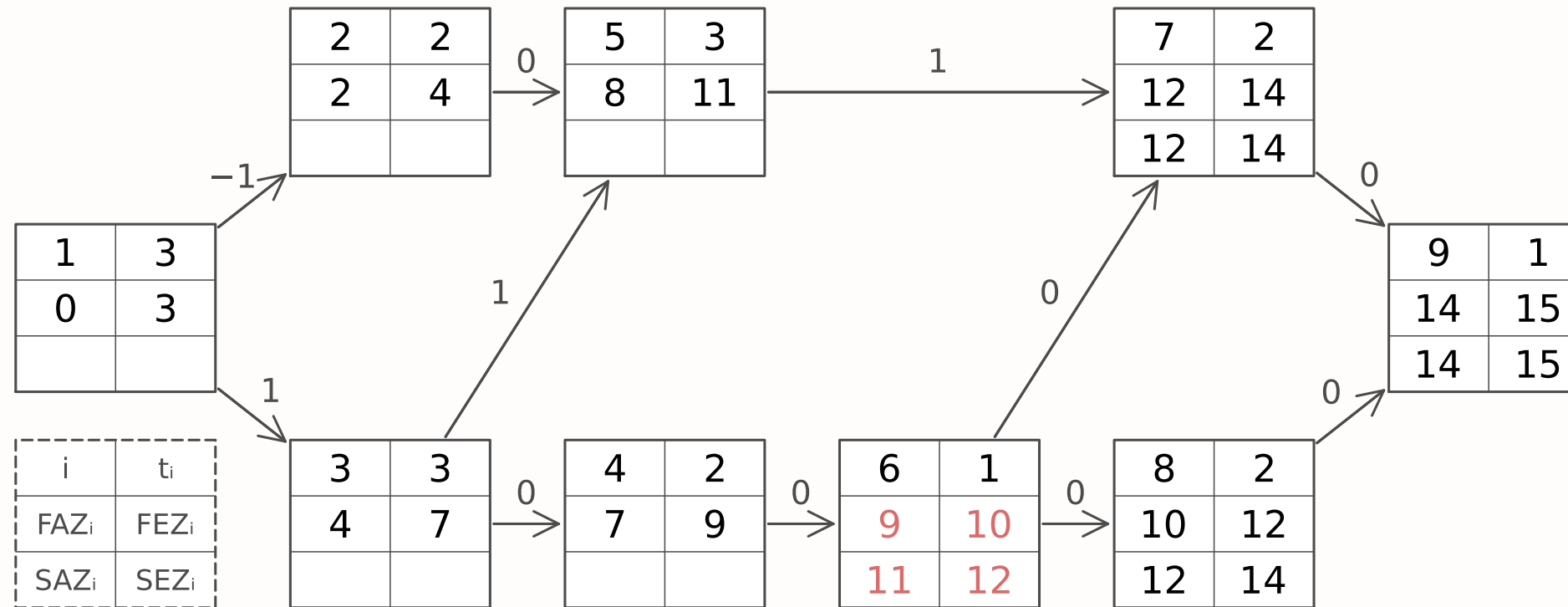
$$SEZ_8 = SAZ_9 - d_{8,9} = 14 - 0 = 14, SAZ_8 = 14 - 2 = 12.$$

Rückwärts im Netzplan



$$SEZ_7 = SAZ_9 - d_{7,9} = 14 - 0 = 14, SAZ_7 = 14 - 2 = 12.$$

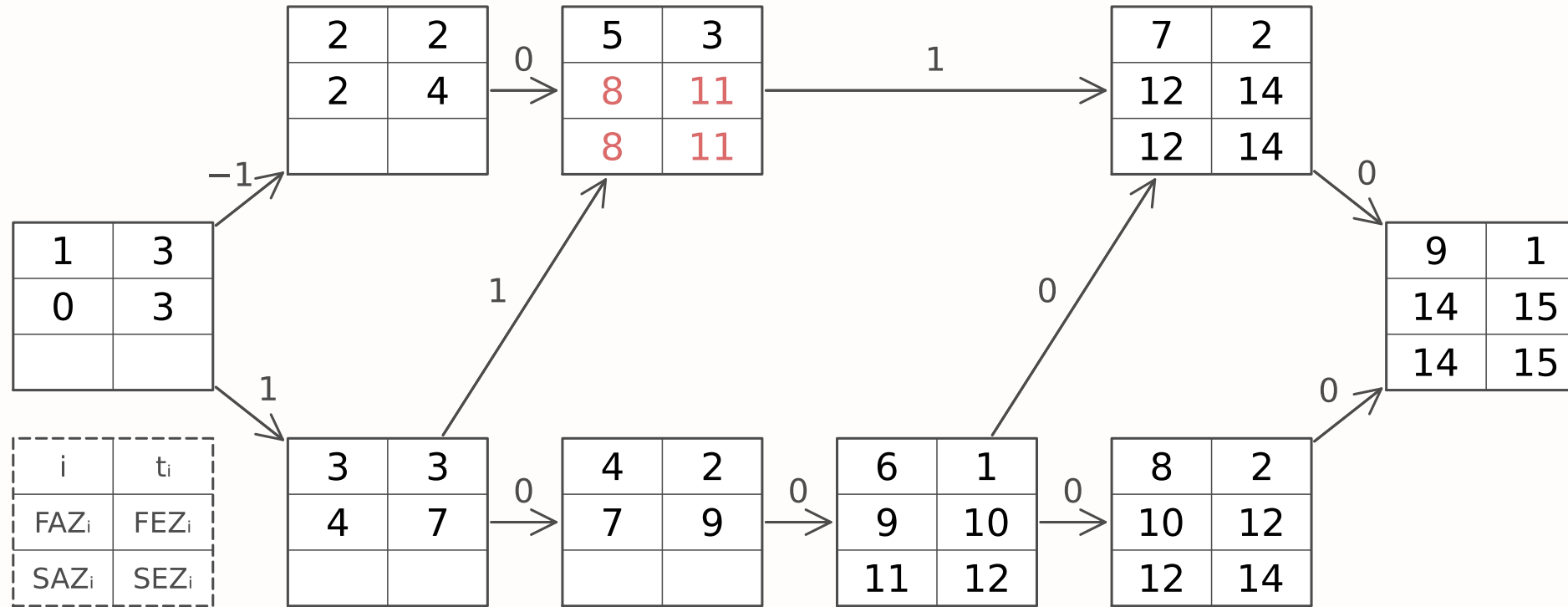
Rückwärts im Netzplan



$$SEZ_6 = \min\{SAZ_7 - d_{6,7}, SAZ_8 - d_{6,8}\} = \min\{12 - 0, 12 - 0\} = 12,$$

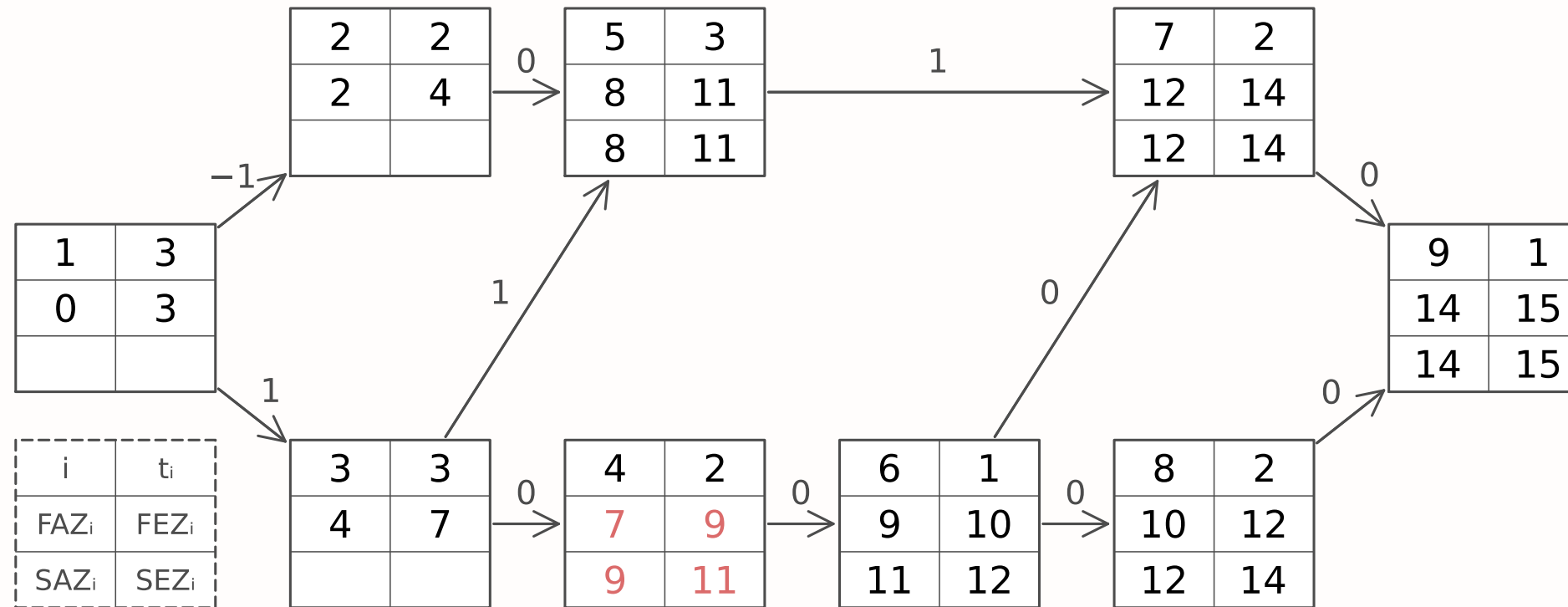
$$SAZ_6 = 12 - 1 = 11.$$

Rückwärts im Netzplan



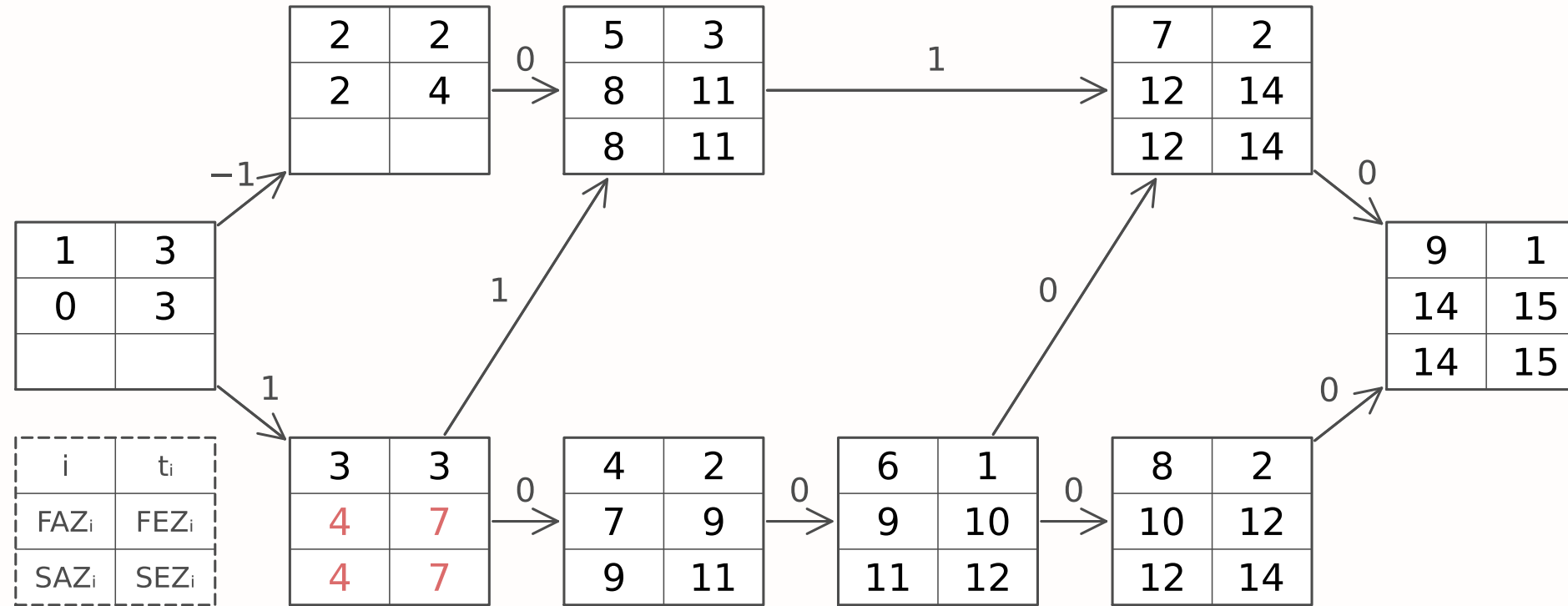
$$SEZ_5 = SAZ_7 - d_{5,7} = 12 - 1 = 11, SAZ_5 = 11 - 3 = 8.$$

Rückwärts im Netzplan



$$SEZ_4 = SAZ_6 - d_{4,6} = 11 - 0 = 11, SAZ_4 = 11 - 2 = 9.$$

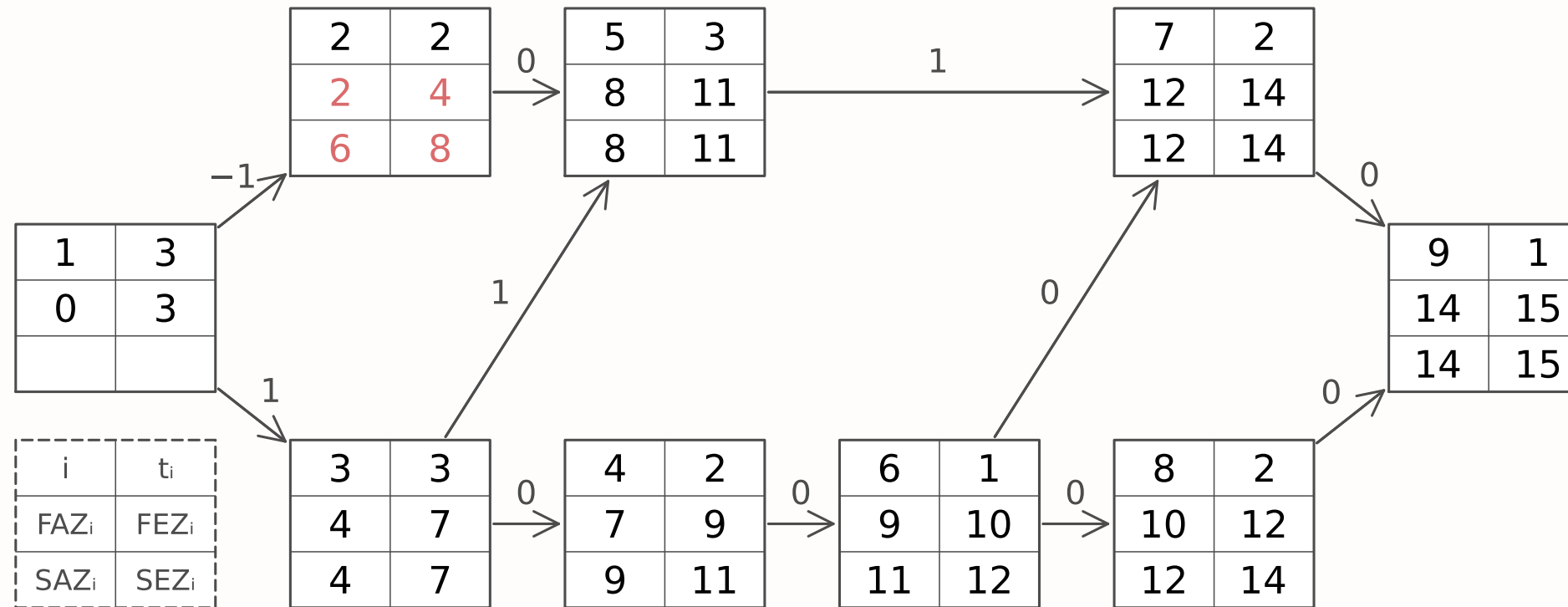
Rückwärts im Netzplan



$$SEZ_3 = \min\{SAZ_4 - d_{3,4}, SAZ_5 - d_{3,5}\} = \min\{9 - 0, 8 - 1\} = 7,$$

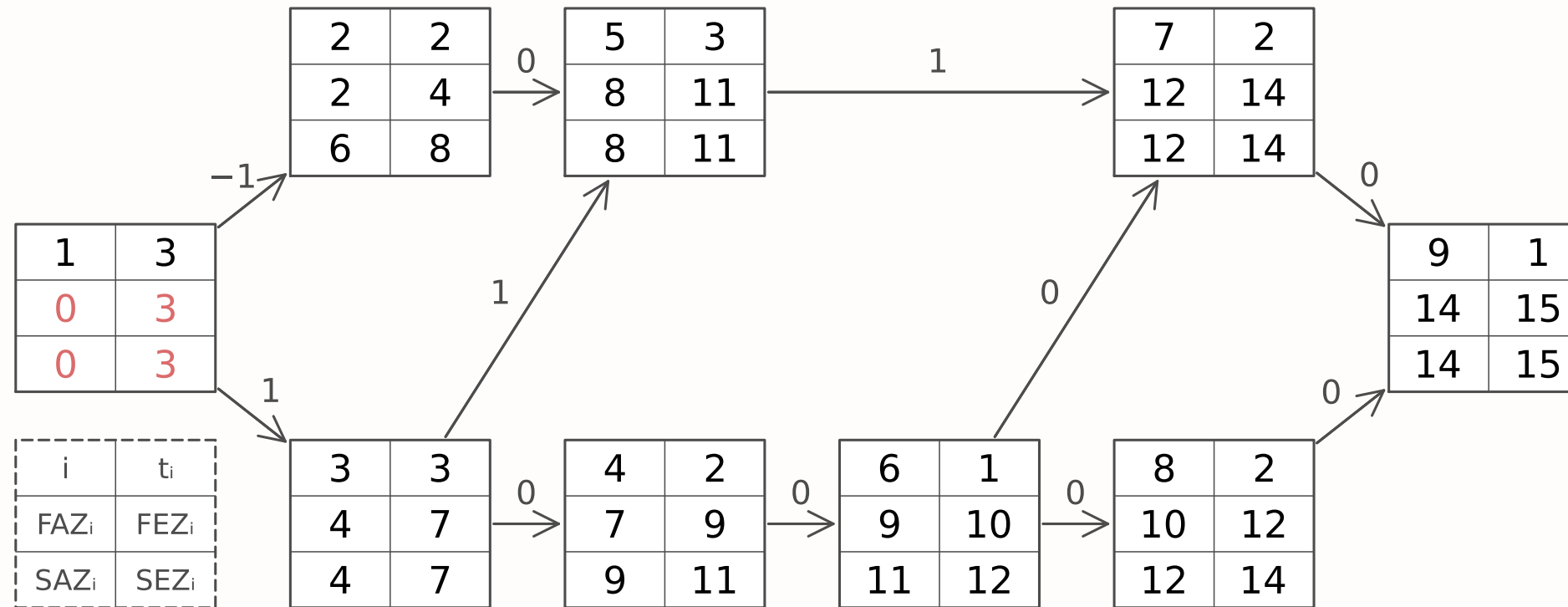
$$SAZ_3 = 7 - 3 = 4.$$

Rückwärts im Netzplan



$$SEZ_2 = SAZ_5 - d_{2,5} = 8 - 0 = 8, SAZ_2 = 8 - 2 = 6.$$

Rückwärts im Netzplan



$$SEZ_1 = \min\{SAZ_2 - d_{1,2}, SAZ_3 - d_{1,3}\} = \min\{6 - (-1), 4 - 1\} = 3,$$

$$SAZ_1 = 3 - 3 = 0.$$

Der Terminplan

Alle Werte als Tabelle:

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

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

Der Terminplan

Alle Werte als Tabelle:

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

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



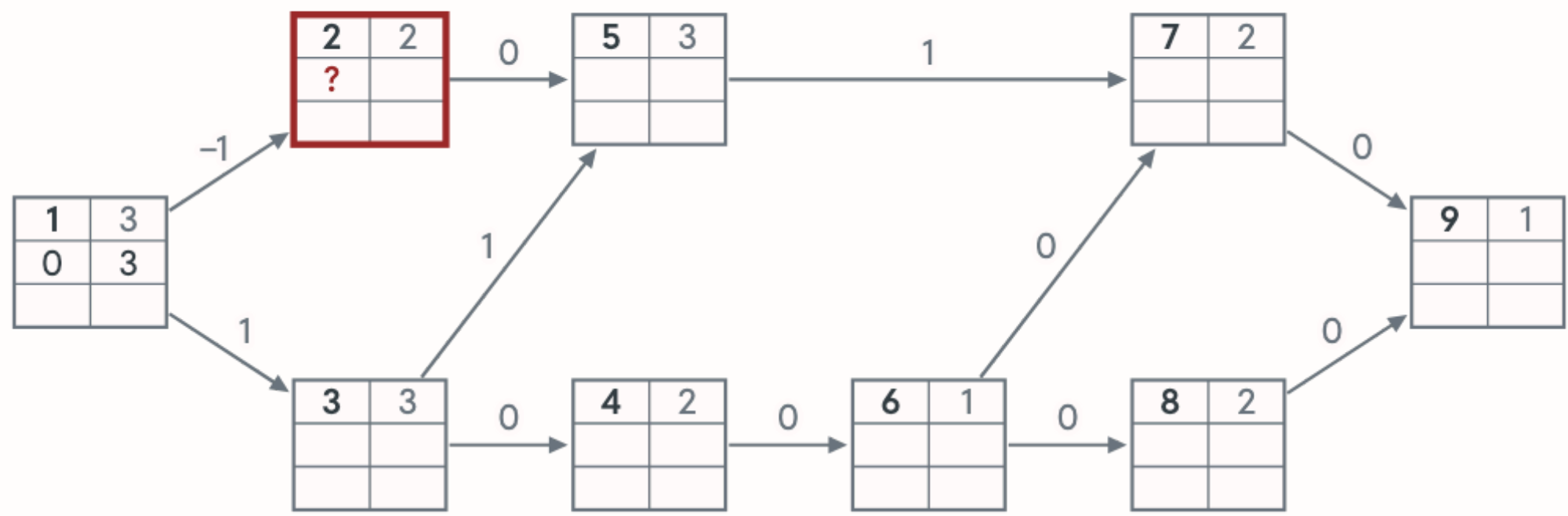
Gleich ausprobieren

Auf der nächsten Folie rechnen Sie selbst, Vorgang für Vorgang, erst vorwärts, dann rückwärts. Unter „Neu“ kommen weitere Netzpläne: Welche Vorgänge sind dort kritisch?

Sie sind dran

Bestimmen Sie Vorgang für Vorgang erst FAZ (vorwärts: Maximum von FEZ + d über die Vorgänger), dann SEZ (rückwärts: Minimum von SAZ – d über die Nachfolger). $FEZ = FAZ + t$ und $SAZ = SEZ - t$ trägt das Spiel ein.

Vorwärts: $FAZ = \text{Maximum von } FEZ + d \text{ über die Vorgänger}$



$FAZ_2 =$ 1 2 3 4



Eigener Plan
0 Abweichungen

Noch 16 Werte zu bestimmen, als Nächstes FAZ_2

Optimieren

Reset

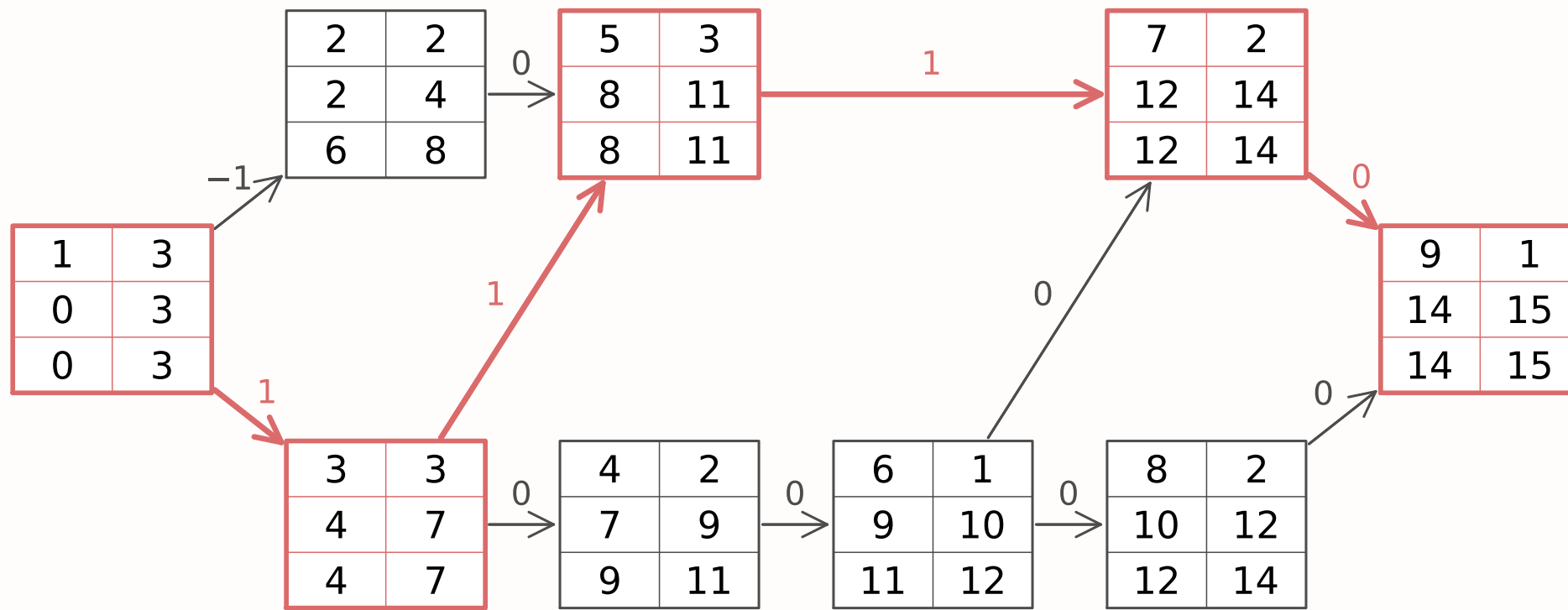
Neu

Kritischer Weg

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:

Pufferzeiten

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

- **Gesamte Pufferzeit:** $GP_i = SAZ_i - FAZ_i$.

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

Weitere Pufferzeiten

- **Freie Rückwärtspufferzeit:** $FRP_i = SAZ_i - \max_{h \in \mathcal{V}(i)} \{SEZ_h + d_{hi}\}$:
Spielraum, wenn alle **Vorgänger spätestmöglich** enden.

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.

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

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.

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 ($UP_2 = 4$).

Zyklen

Wenn Maximalabstände gelten

Enthält der Netzplan Zyklen (z. B. durch **Maximalabstände**), versagt die topologische Reihenfolge. Dann rechnet ein **FIFO-Verfahren** wie bei den kürzesten Wegen (Vorlesung 6):

Wenn Maximalabstände gelten

Enthält der Netzplan Zyklen (z. B. durch **Maximalabstände**), versagt die topologische Reihenfolge. Dann rechnet ein **FIFO-Verfahren** wie bei den kürzesten Wegen (Vorlesung 6):

- erst **vorwärts**, dann **rückwärts**;

Wenn Maximalabstände gelten

Enthält der Netzplan Zyklen (z. B. durch **Maximalabstände**), versagt die topologische Reihenfolge. Dann rechnet ein **FIFO-Verfahren** wie bei den kürzesten Wegen (Vorlesung 6):

- erst **vorwärts**, dann **rückwärts**;
- vorwärts wird geprüft, ob Zeitpunkte zu **klein** sind, rückwärts, ob sie zu **groß** sind;

Wenn Maximalabstände gelten

Enthält der Netzplan Zyklen (z. B. durch **Maximalabstände**), versagt die topologische Reihenfolge. Dann rechnet ein **FIFO-Verfahren** wie bei den kürzesten Wegen (Vorlesung 6):

- erst **vorwärts**, dann **rückwärts**;
- vorwärts wird geprüft, ob Zeitpunkte zu **klein** sind, rückwärts, ob sie zu **groß** sind;
- wer aktualisiert wird, kommt hinten in die Liste, sofern er nicht schon darin steht.

Wenn Maximalabstände gelten

Enthält der Netzplan Zyklen (z. B. durch **Maximalabstände**), versagt die topologische Reihenfolge. Dann rechnet ein **FIFO-Verfahren** wie bei den kürzesten Wegen (Vorlesung 6):

- erst **vorwärts**, dann **rückwärts**;
- vorwärts wird geprüft, ob Zeitpunkte zu **klein** sind, rückwärts, ob sie zu **groß** sind;
- wer aktualisiert wird, kommt hinten in die Liste, sofern er nicht schon darin steht.

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: $\bar{d}_{3,8}^E = 6$.

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?

Beispiel: Maximalabstand

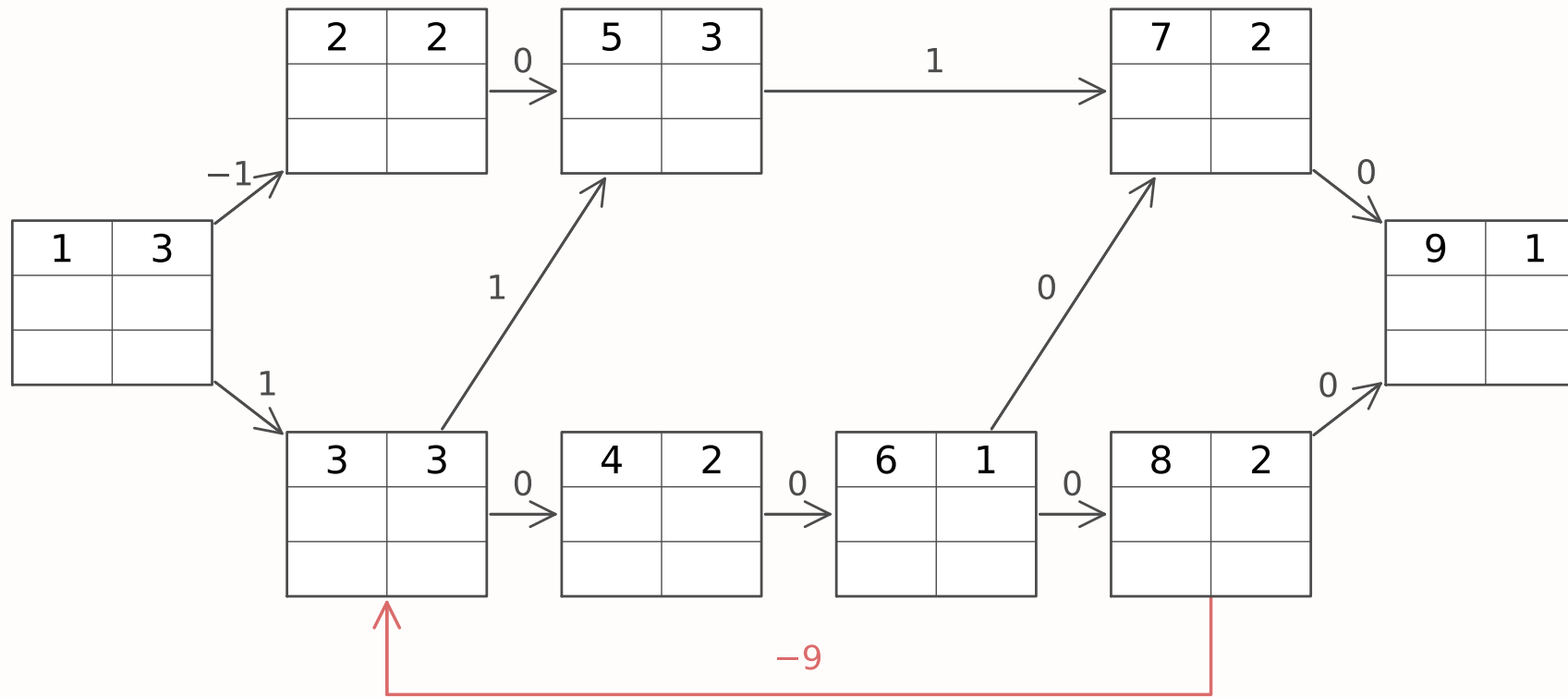
Neue Vorgabe für die Lagerhalle: Vom **Ende des Mauerns** (3) bis zur **Fertigstellung des Außenputzes** (8) dürfen höchstens **6 ZE** vergehen: $\bar{d}_{3,8}^E = 6$.

Frage: Welcher Pfeil entsteht daraus?

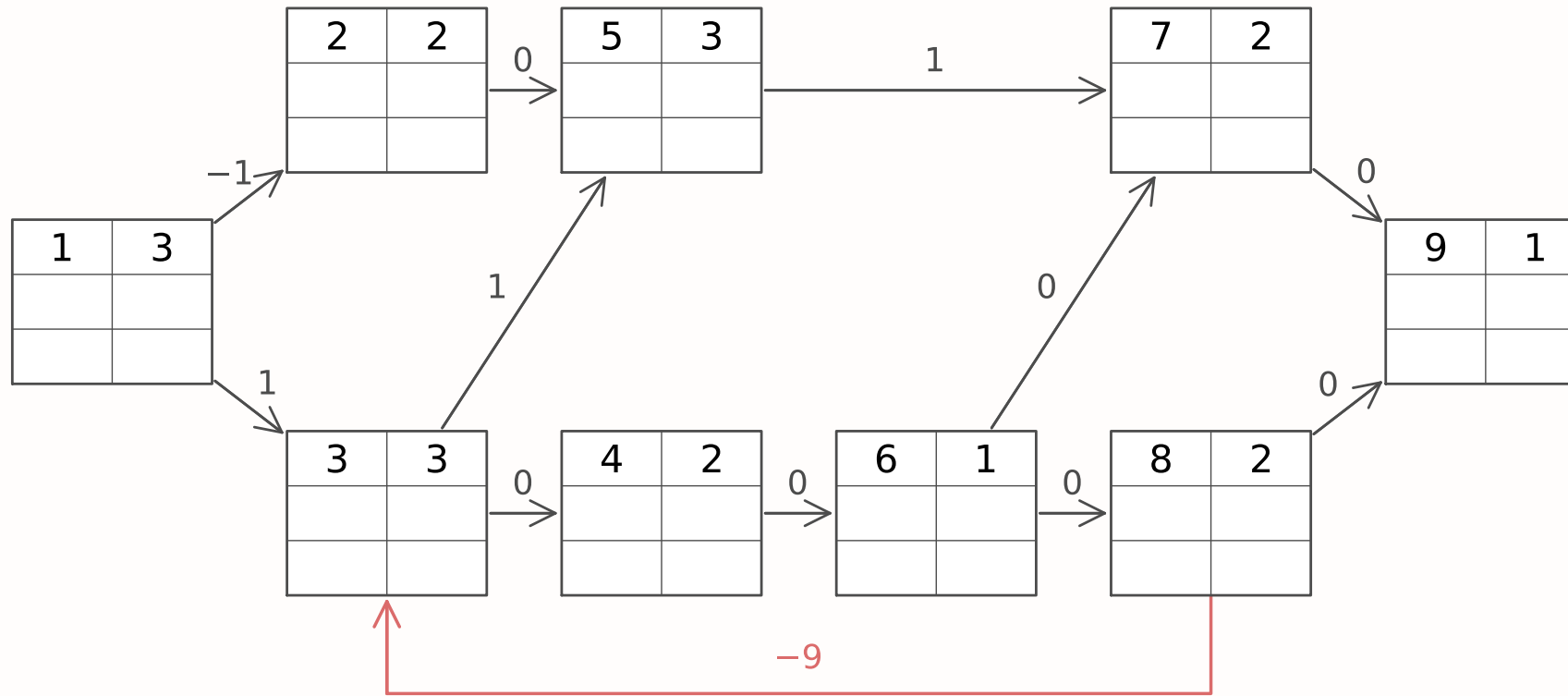
Transformationsregel $d_{ih} = -\bar{d}_{hi}^E - t_h$:

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

Der erweiterte Netzplan



Der 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**.

FIFO vorwärts

Liste mit Kopf H und Ende T . Start: $FAZ_1 := 0$, $FEZ_1 := t_1$, alle anderen $FAZ_j := -\infty$; $H = T = 1$.

FIFO vorwärts

Liste mit Kopf H und Ende T . Start: $FAZ_1 := 0$, $FEZ_1 := t_1$, alle anderen $FAZ_j := -\infty$; $H = T = 1$.

- Für jeden Nachfolger $j \in \mathcal{N}(H)$: ist $FAZ_j < FEZ_H + d_{Hj}$?

FIFO vorwärts

Liste mit Kopf H und Ende T . Start: $FAZ_1 := 0$, $FEZ_1 := t_1$, alle anderen $FAZ_j := -\infty$; $H = T = 1$.

- Für jeden Nachfolger $j \in \mathcal{N}(H)$: ist $FAZ_j < FEZ_H + d_{Hj}$?
- Falls ja: $FAZ_j := FEZ_H + d_{Hj}$, j hinten einreihen (falls nicht bereits enthalten).

FIFO vorwärts

Liste mit Kopf H und Ende T . Start: $FAZ_1 := 0$, $FEZ_1 := t_1$, alle anderen $FAZ_j := -\infty$; $H = T = 1$.

- Für jeden Nachfolger $j \in \mathcal{N}(H)$: ist $FAZ_j < FEZ_H + d_{Hj}$?
- Falls ja: $FAZ_j := FEZ_H + d_{Hj}$, j hinten einreihen (falls nicht bereits enthalten).
- Ist $H = T$: Liste leer $\Rightarrow$ **Abbruch**.

FIFO vorwärts

Liste mit Kopf H und Ende T . Start: $FAZ_1 := 0$, $FEZ_1 := t_1$, alle anderen $FAZ_j := -\infty$; $H = T = 1$.

- Für jeden Nachfolger $j \in \mathcal{N}(H)$: ist $FAZ_j < FEZ_H + d_{Hj}$?
- Falls ja: $FAZ_j := FEZ_H + d_{Hj}$, j hinten einreihen (falls nicht bereits enthalten).
- Ist $H = T$: Liste leer $\Rightarrow$ **Abbruch**.
- Sonst $H :=$ nächster Knoten der Liste, $FEZ_H := FAZ_H + t_H$.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ _{j}	0	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
FEZ _{j}	3	–	–	–	–	–	–	–	–

Start: FAZ₁ = 0, FEZ₁ = 3, alle anderen FAZ _{j} = $-\infty$; $H = T = 1$.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ _{j}	0	2	4	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
FEZ _{j}	3	–	–	–	–	–	–	–	–

$H = 1$: FAZ₂ = 3 – 1 = 2, 2 stellt sich an; FAZ₃ = 3 + 1 = 4, 3 stellt sich an.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ _{j}	0	2	4	$-\infty$	4	$-\infty$	$-\infty$	$-\infty$	$-\infty$
FEZ _{j}	3	4	–	–	–	–	–	–	–

$H = 2$: FEZ₂ = 2 + 2 = 4; FAZ₅ = 4 + 0 = 4, 5 stellt sich an.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ_j	0	2	4	7	8	$-\infty$	$-\infty$	$-\infty$	$-\infty$
FEZ_j	3	4	7	–	–	–	–	–	–

$H = 3$: $FEZ_3 = 4 + 3 = 7$; $FAZ_4 = 7 + 0 = 7$, 4 stellt sich an;
 $FAZ_5 = 7 + 1 = 8$ statt 4: **korrigiert**, 5 wartet schon.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ_j	0	2	4	7	8	$-\infty$	12	$-\infty$	$-\infty$
FEZ_j	3	4	7	–	11	–	–	–	–

$H = 5$: $FEZ_5 = 8 + 3 = 11$; $FAZ_7 = 11 + 1 = 12$, 7 stellt sich an.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ _{j}	0	2	4	7	8	9	12	$-\infty$	$-\infty$
FEZ _{j}	3	4	7	9	11	–	–	–	–

$H = 4$: FEZ₄ = 7 + 2 = 9; FAZ₆ = 9 + 0 = 9, 6 stellt sich an.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ_j	0	2	4	7	8	9	12	$-\infty$	14
FEZ_j	3	4	7	9	11	–	14	–	–

$H = 7: FEZ_7 = 12 + 2 = 14; FAZ_9 = 14 + 0 = 14$, 9 stellt sich an.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ _{j}	0	2	4	7	8	9	12	10	14
FEZ _{j}	3	4	7	9	11	10	14	–	–

$H = 6$: $FEZ_6 = 9 + 1 = 10$; für 7: $10 + 0 = 10 \leq FAZ_7 = 12$, keine Änderung; $FAZ_8 = 10 + 0 = 10$, 8 stellt sich an.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ _{j}	0	2	4	7	8	9	12	10	14
FEZ _{j}	3	4	7	9	11	10	14	–	15

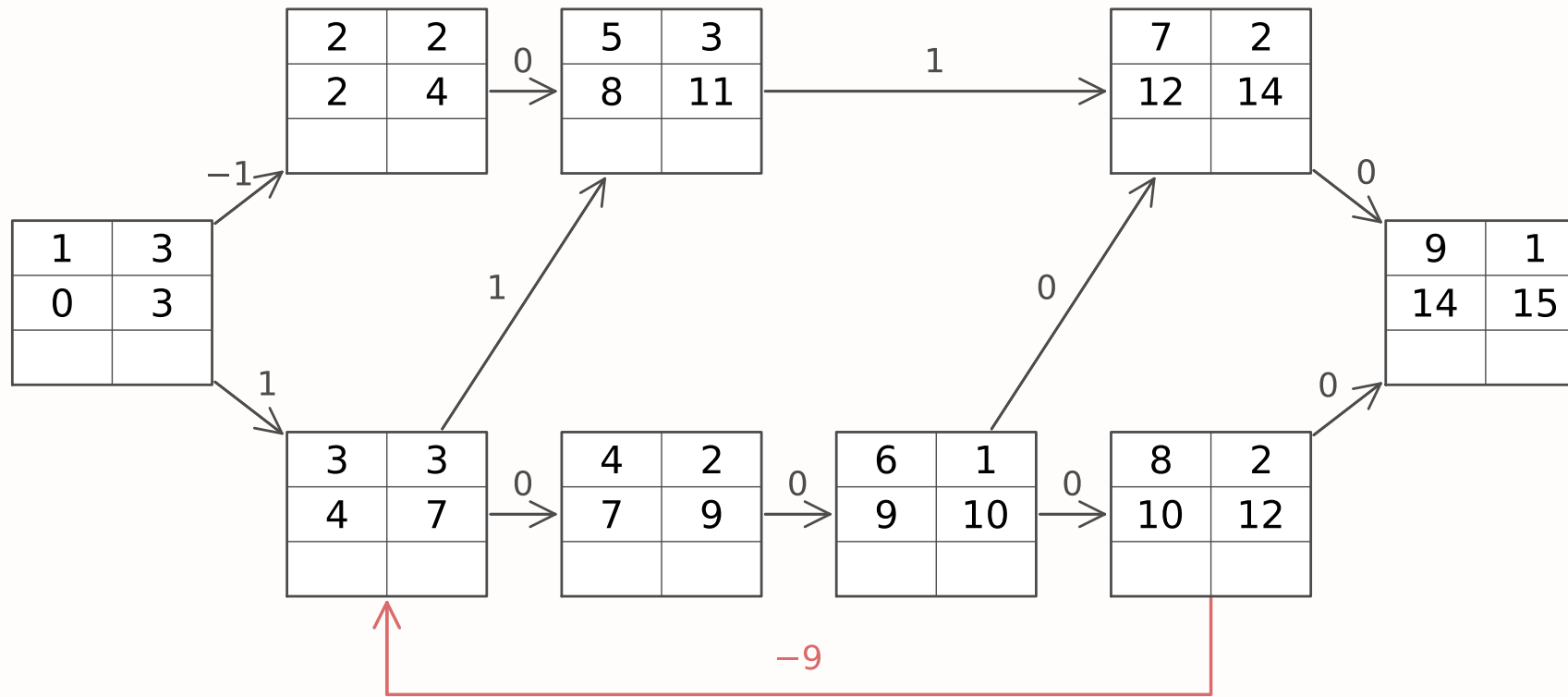
$H = 9$: FEZ₉ = 14 + 1 = 15; 9 hat keine Nachfolger.

Vorwärts mit der FIFO-Liste

j	1	2	3	4	5	6	7	8	9
FAZ _{j}	0	2	4	7	8	9	12	10	14
FEZ _{j}	3	4	7	9	11	10	14	12	15

$H = 8$: $FEZ_8 = 10 + 2 = 12$; für 9: $12 + 0 = 12 \leq FAZ_9 = 14$, keine Änderung; für 3: $12 - 9 = 3 \leq FAZ_3 = 4$, keine Änderung. Liste leer: **Abbruch.**

Ergebnis vorwärts



Der Rückwärtspfeil hat vorwärts nichts geändert ($12 - 9 = 3 \leq 4$): Die frühesten Zeitpunkte sind dieselben wie ohne Zyklus.

FIFO rückwärts

Start: $SEZ_n := FEZ_n$, $SAZ_n := SEZ_n - t_n$, alle anderen $SEZ_j := \infty$;
 $H = T = n$.

FIFO rückwärts

Start: $SEZ_n := FEZ_n$, $SAZ_n := SEZ_n - t_n$, alle anderen $SEZ_j := \infty$;
 $H = T = n$.

- Für jeden Vorgänger $j \in \mathcal{V}(H)$: ist $SEZ_j > SAZ_H - d_{jH}$?

FIFO rückwärts

Start: $SEZ_n := FEZ_n$, $SAZ_n := SEZ_n - t_n$, alle anderen $SEZ_j := \infty$;
 $H = T = n$.

- Für jeden Vorgänger $j \in \mathcal{V}(H)$: ist $SEZ_j > SAZ_H - d_{jH}$?
- Falls ja: $SEZ_j := SAZ_H - d_{jH}$, j hinten einreihen (falls nicht bereits enthalten).

FIFO rückwärts

Start: $SEZ_n := FEZ_n$, $SAZ_n := SEZ_n - t_n$, alle anderen $SEZ_j := \infty$;
 $H = T = n$.

- Für jeden Vorgänger $j \in \mathcal{V}(H)$: ist $SEZ_j > SAZ_H - d_{jH}$?
- Falls ja: $SEZ_j := SAZ_H - d_{jH}$, j hinten einreihen (falls nicht bereits enthalten).
- Ist $H = T$: Liste leer $\Rightarrow$ **Abbruch**.

FIFO rückwärts

Start: $SEZ_n := FEZ_n$, $SAZ_n := SEZ_n - t_n$, alle anderen $SEZ_j := \infty$;
 $H = T = n$.

- Für jeden Vorgänger $j \in \mathcal{V}(H)$: ist $SEZ_j > SAZ_H - d_{jH}$?
- Falls ja: $SEZ_j := SAZ_H - d_{jH}$, j hinten einreihen (falls nicht bereits enthalten).
- Ist $H = T$: Liste leer $\Rightarrow$ **Abbruch**.
- Sonst $H :=$ nächster Knoten der Liste, $SAZ_H := SEZ_H - t_H$.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	∞	∞	∞	∞	∞	∞	∞	∞	15
SAZ_j	–	–	–	–	–	–	–	–	14

Start: $SEZ_9 = FEZ_9 = 15$, $SAZ_9 = 15 - 1 = 14$, alle anderen $SEZ_j = \infty$;
 $H = T = 9$.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	∞	∞	∞	∞	∞	∞	14	14	15
SAZ_j	–	–	–	–	–	–	–	–	14

$H = 9$: $SEZ_7 = 14 - 0 = 14$, 7 stellt sich an; $SEZ_8 = 14 - 0 = 14$, 8 stellt sich an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ _{j}	∞	∞	∞	∞	11	12	14	14	15
SAZ _{j}	–	–	–	–	–	–	12	–	14

$H = 7$: $SAZ_7 = 14 - 2 = 12$; $SEZ_5 = 12 - 1 = 11$, 5 stellt sich an;
 $SEZ_6 = 12 - 0 = 12$, 6 stellt sich an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	∞	∞	∞	∞	11	12	14	14	15
SAZ_j	–	–	–	–	–	–	12	12	14

$H = 8$: $SAZ_8 = 14 - 2 = 12$; für 6: $12 - 0 = 12 \geq SEZ_6 = 12$, keine Änderung.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	∞	8	7	∞	11	12	14	14	15
SAZ_j	–	–	–	–	8	–	12	12	14

$H = 5$: $SAZ_5 = 11 - 3 = 8$; $SEZ_2 = 8 - 0 = 8$, 2 stellt sich an;
 $SEZ_3 = 8 - 1 = 7$, 3 stellt sich an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	∞	8	7	11	11	12	14	14	15
SAZ_j	–	–	–	–	8	11	12	12	14

$H = 6$: $SAZ_6 = 12 - 1 = 11$; $SEZ_4 = 11 - 0 = 11$, 4 stellt sich an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ _{j}	7	8	7	11	11	12	14	14	15
SAZ _{j}	–	6	–	–	8	11	12	12	14

$H = 2$: $SAZ_2 = 8 - 2 = 6$; $SEZ_1 = 6 - (-1) = 7$, 1 stellt sich an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	3	8	7	11	11	12	14	13	15
SAZ_j	–	6	4	–	8	11	12	12	14

$H = 3$: $SAZ_3 = 7 - 3 = 4$; $SEZ_1 = 4 - 1 = 3$ statt 7: **korrigiert**, 1 wartet schon; $SEZ_8 = 4 - (-9) = 13$ statt 14, 8 stellt sich **erneut** an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	3	8	7	11	11	12	14	13	15
SAZ_j	–	6	4	9	8	11	12	12	14

$H = 4$: $SAZ_4 = 11 - 2 = 9$; für 3: $9 - 0 = 9 \geq SEZ_3 = 7$, keine Änderung.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	3	8	7	11	11	12	14	13	15
SAZ_j	0	6	4	9	8	11	12	12	14

$H = 1: SAZ_1 = 3 - 3 = 0$; 1 hat keine Vorgänger.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ _{j}	3	8	7	11	11	11	14	13	15
SAZ _{j}	0	6	4	9	8	11	12	11	14

$H = 8$: $SAZ_8 = 13 - 2 = 11$; $SEZ_6 = 11 - 0 = 11$ statt 12, 6 stellt sich **erneut** an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ _{j}	3	8	7	10	11	11	14	13	15
SAZ _{j}	0	6	4	9	8	10	12	11	14

$H = 6$: $SAZ_6 = 11 - 1 = 10$; $SEZ_4 = 10 - 0 = 10$ statt 11, 4 stellt sich **erneut** an.

Rückwärts mit der FIFO-Liste

Jetzt greift der Zyklus: Zu den Vorgängern von 3 gehört auch 8.

j	1	2	3	4	5	6	7	8	9
SEZ_j	3	8	7	10	11	11	14	13	15
SAZ_j	0	6	4	8	8	10	12	11	14

$H = 4$: $SAZ_4 = 10 - 2 = 8$; für 3: $8 - 0 = 8 \geq SEZ_3 = 7$, keine Änderung. Liste leer: **Abbruch**.

Neue späteste Zeitpunkte

Die Vorwärtsrechnung ändert sich nicht ($FEZ_8 - 9 = 3 < 4$), 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 Zyklus)	0	6	4	8	8	10	12	11	14

Neue späteste Zeitpunkte

Die Vorwärtsrechnung ändert sich nicht ($FEZ_8 - 9 = 3 < 4$), 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 Zyklus)	0	6	4	8	8	10	12	11	14

$SEZ_8 = \min\{SAZ_9 - 0, SAZ_3 - (-9)\} = \min\{14, 13\} = 13$: die Puffer von 4, 6, 8 schrumpfen auf 1.

Kontrolle mit Julia

Projektdauer als LP

Die Projektdauer ist der längste Weg, als LP formuliert:

```
1 using JuMP, HiGHS
2
3 t = Dict{1=>3,2=>2,3=>3,4=>2,5=>3,6=>1,7=>2,8=>2,9=>1}
4 # (Vorgänger h, Vorgang i, Mindestabstand d_hi)
5 kanten = [(1,2,-1),(1,3,1),(3,4,0),(2,5,0),(3,5,1),(4,6,0),
6           (5,7,1),(6,7,0),(6,8,0),(7,9,0),(8,9,0)]
7
8 modell = Model(HiGHS.Optimizer); set_silent(modell)
9 @variable(modell, S[keys(t)] >= 0) # Anfangszeitpunkte (FAZ)
10 @variable(modell, C >= 0) # Projektdauer
11 @constraint(modell, [(h,i,d) in kanten], S[i] >= S[h] + t[h] + d)
12 @constraint(modell, [i in keys(t)], C >= S[i] + t[i])
13 @objective(modell, Min, C)
14
15 optimize!(modell)
16 println("Projektdauer = ", objective_value(modell))
```

Projektdauer als LP

Die Projektdauer ist der längste Weg, als LP formuliert:

```
1 using JuMP, HiGHS
2
3 t = Dict{1=>3,2=>2,3=>3,4=>2,5=>3,6=>1,7=>2,8=>2,9=>1}
4 # (Vorgänger h, Vorgang i, Mindestabstand d_hi)
5 kanten = [(1,2,-1),(1,3,1),(3,4,0),(2,5,0),(3,5,1),(4,6,0),
6           (5,7,1),(6,7,0),(6,8,0),(7,9,0),(8,9,0)]
7
8 modell = Model(HiGHS.Optimizer); set_silent(modell)
9 @variable(modell, S[keys(t)] >= 0) # Anfangszeitpunkte (FAZ)
10 @variable(modell, C >= 0) # Projektdauer
11 @constraint(modell, [(h,i,d) in kanten], S[i] >= S[h] + t[h] + d)
12 @constraint(modell, [i in keys(t)], C >= S[i] + t[i])
13 @objective(modell, Min, C)
14
15 optimize!(modell)
16 println("Projektdauer = ", objective_value(modell))
```

Projektdauer = 15.0

Vorwärtsrechnung als LP

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

```
1 tv = [3, 2, 3, 2, 3, 1, 2, 2, 1]
2
3 modell = Model(HiGHS.Optimizer); set_silent(modell)
4 @variable(modell, FAZ[1:9] >= 0)
5 @constraint(modell, FAZ[1] == 0)
6 @constraint(modell, [(h,i,d) in kanten], FAZ[i] >= FAZ[h] + tv[h] + d)
7 @objective(modell, Min, sum(FAZ[i] + tv[i] for i in 1:9)) # min Σ FEZ
8 optimize!(modell)
9 println("FAZ = ", round.(Int, value.(FAZ)))
```

Vorwärtsrechnung als LP

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

```
1 tv = [3, 2, 3, 2, 3, 1, 2, 2, 1]
2
3 modell = Model(HiGHS.Optimizer); set_silent(modell)
4 @variable(modell, FAZ[1:9] >= 0)
5 @constraint(modell, FAZ[1] == 0)
6 @constraint(modell, [(h,i,d) in kanten], FAZ[i] >= FAZ[h] + tv[h] + d)
7 @objective(modell, Min, sum(FAZ[i] + tv[i] for i in 1:9)) # min Σ FEZ
8 optimize!(modell)
9 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:

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

Rückwärtsrechnung als LP

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

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

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

Rückwärtsrechnung als LP

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

```
1 modell = Model(HiGHS.Optimizer); set_silent(modell)
2 @variable(modell, SAZ[1:9] >= 0)
3 @constraint(modell, SAZ[9] + tv[9] == 15)           # SEZ_9 = FEZ_9*
4 @constraint(modell, [(h,i,d) in kanten], SAZ[i] >= SAZ[h] + tv[h] + d)
5 @objective(modell, Max, sum(SAZ))
6 optimize!(modell)
7 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

Das Wichtigste auf einen Blick

- MPM-Netzplan: **Vorwärtsrechnung** liefert früheste, **Rückwärtsrechnung** späteste Zeitpunkte.

Das Wichtigste auf einen Blick

- MPM-Netzplan: **Vorwärtsrechnung** liefert früheste, **Rückwärtsrechnung** späteste Zeitpunkte.
- Der **kritische Weg** ist der längste Weg; kritische Vorgänge haben $GP_i = 0$.

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.

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

Domschke, Wolfgang, Andreas Drexl, Robert Klein, und Armin Scholl. 2015. *Einführung in Operations Research*. 9. Aufl. Springer Gabler. <https://doi.org/10.1007/978-3-662-48216-2>.