

# Vorlesung 08: Ganzzahlige Optimierung

Grundlagen Operations Research

Prof. Dr. Knut Haase

# Ganzzahligkeit

# Nur ganze Einheiten

# Was ist neu?

Für einige oder alle Variablen sind nur **ganze Zahlen** zulässig:

# Was ist neu?

Für einige oder alle Variablen sind nur **ganze Zahlen** zulässig:

- **Ganzzahlig:**  $X \geq 0$  und ganzzahlig (Stückzahlen, Fahrzeuge, ...).

# Was ist neu?

Für einige oder alle Variablen sind nur **ganze Zahlen** zulässig:

- **Ganzzahlig:**  $X \geq 0$  und ganzzahlig (Stückzahlen, Fahrzeuge, ...).
- **Binär:**  $X \in \{0, 1\}$  für Ja/Nein-Entscheidungen.

# Was ist neu?

Für einige oder alle Variablen sind nur **ganze Zahlen** zulässig:

- **Ganzzahlig:**  $X \geq 0$  und ganzzahlig (Stückzahlen, Fahrzeuge, ...).
- **Binär:**  $X \in \{0, 1\}$  für Ja/Nein-Entscheidungen.
- **Gemischt-ganzzahlig:** ganzzahlige **und** kontinuierliche Variablen.

# Was ist neu?

Für einige oder alle Variablen sind nur **ganze Zahlen** zulässig:

- **Ganzzahlig**:  $X \geq 0$  und ganzzahlig (Stückzahlen, Fahrzeuge, ...).
- **Binär**:  $X \in \{0, 1\}$  für Ja/Nein-Entscheidungen.
- **Gemischt-ganzzahlig**: ganzzahlige **und** kontinuierliche Variablen.

Lässt man die Ganzzahligkeit weg, entsteht die **LP-Relaxation**, ein gewöhnliches lineares Problem.

# Warum schwierig?

# Warum schwierig?

- Ganzzahlige Probleme gelten als **schwer**: kein Algorithmus mit polynomialem Aufwand für die optimale Lösung bekannt.

# Warum schwierig?

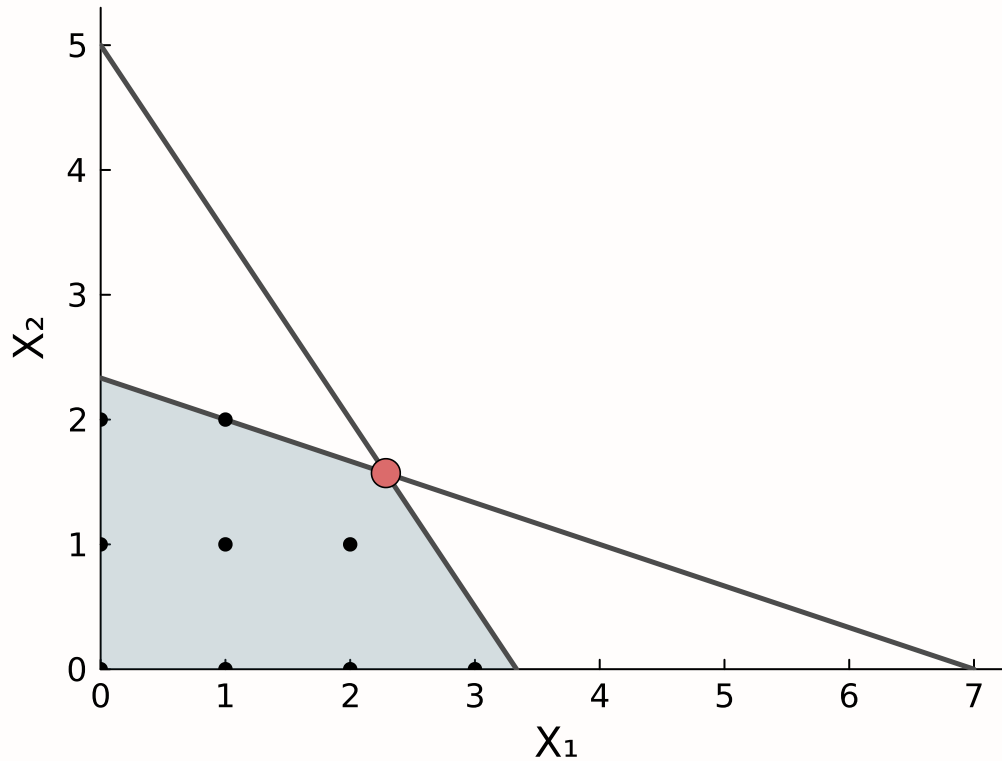
- Ganzzahlige Probleme gelten als **schwer**: kein Algorithmus mit polynomialem Aufwand für die optimale Lösung bekannt.
- Naiv: **vollständige Enumeration** aller ganzzahligen Möglichkeiten.

# Warum schwierig?

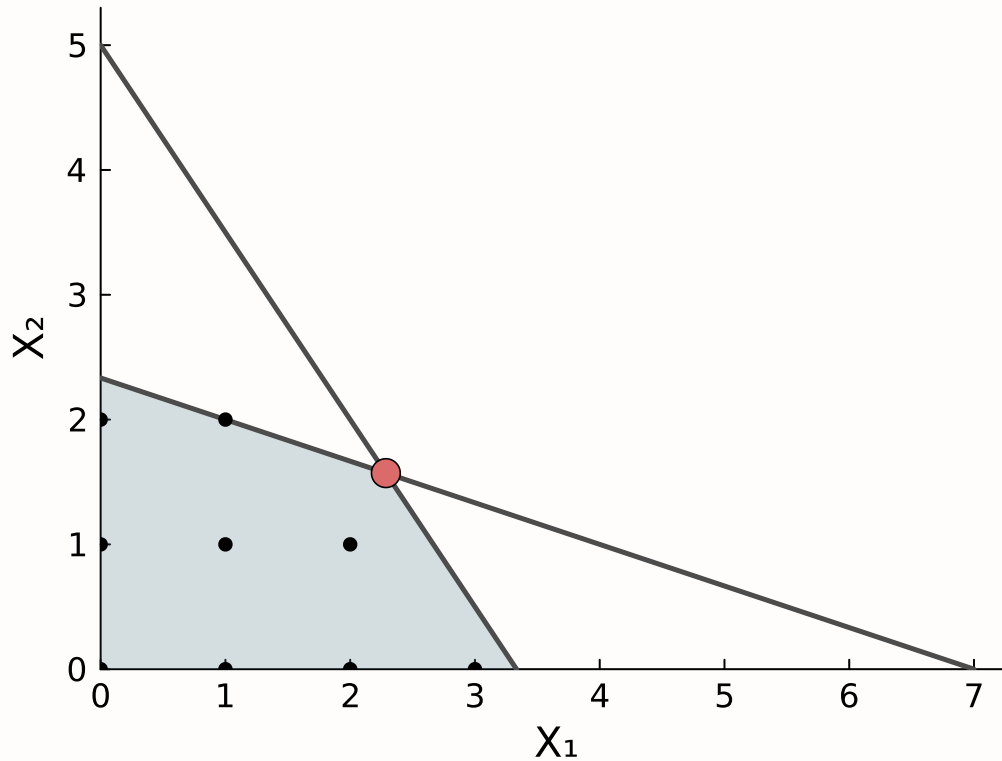
- Ganzzahlige Probleme gelten als **schwer**: kein Algorithmus mit polynomialem Aufwand für die optimale Lösung bekannt.
- Naiv: **vollständige Enumeration** aller ganzzahligen Möglichkeiten.
- In der Praxis: **Meta-Heuristiken** (Genetische Algorithmen, Tabu-Suche, ...) oder das exakte **Branch-&-Bound-Verfahren**.

# LP-Relaxation grafisch

Die schwarzen Punkte sind die **ganzzahligen** Kandidaten.



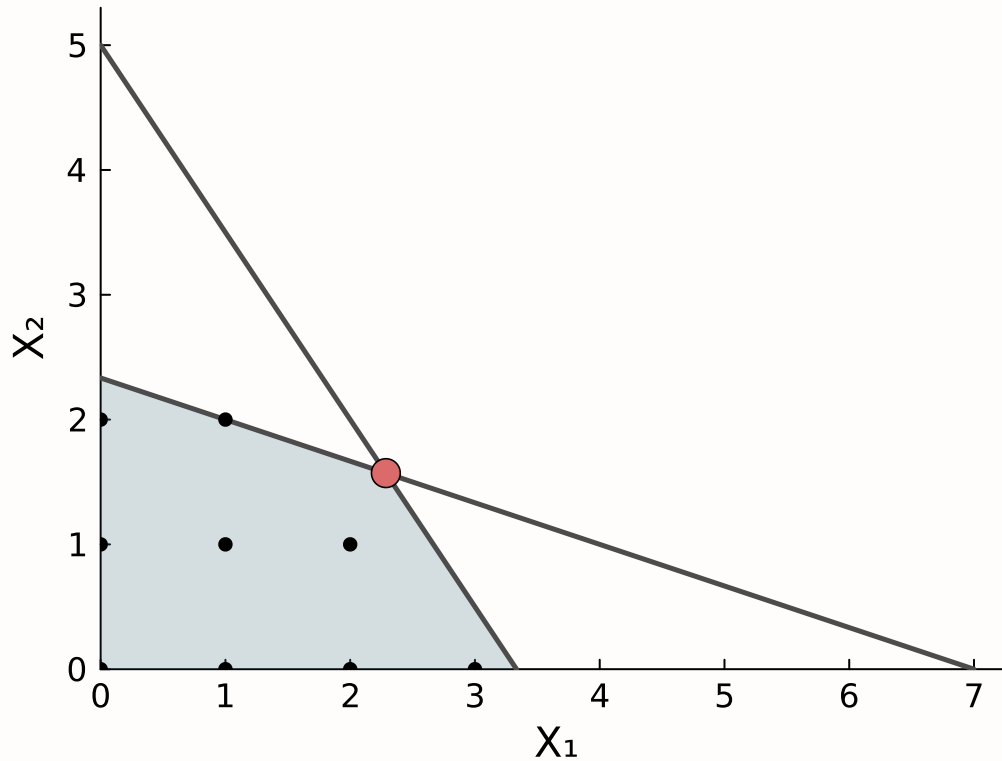
# LP-Relaxation grafisch



Die schwarzen Punkte sind die **ganzzahligen** Kandidaten.

Das LP-Optimum (2,29; 1,57) ist **nicht** ganzzahlig.

# LP-Relaxation grafisch



Die schwarzen Punkte sind die **ganzzahligen** Kandidaten.

Das LP-Optimum (2,29; 1,57) ist **nicht** ganzzahlig.

**Frage: Warum nicht einfach runden?**

Die Antwort folgt am Ende des Kapitels.

# Branch & Bound

# Ausgangspunkt $P_0$

$$\max F = X_1 + 2X_2$$

$$X_1 + 3X_2 \leq 7$$

$$3X_1 + 2X_2 \leq 10$$

$$X_1, X_2 \geq 0 \text{ und ganzzahlig}$$

# Ausgangspunkt $P_0$

$$\max F = X_1 + 2X_2$$

$$X_1 + 3X_2 \leq 7$$

$$3X_1 + 2X_2 \leq 10$$

$$X_1, X_2 \geq 0 \text{ und ganzzahlig}$$

LP-Relaxation  $P'_0: X_1^* = 2,286, X_2^* = 1,571, F_0^* = 5,429 \Rightarrow$  **obere Schranke**  $\bar{F}_0 = 5,429$ .

# Schranken (Bounding)

# Schranken (Bounding)

- Jede LP-Relaxation liefert eine obere Schranke für ihr Teilproblem.

# Schranken (Bounding)

- Jede LP-Relaxation liefert eine **obere Schranke** für ihr Teilproblem.
- Jede bekannte **ganzzahlige** Lösung liefert eine globale **untere Schranke**  $\underline{F}$ .

# Schranken (Bounding)

- Jede LP-Relaxation liefert eine **obere Schranke** für ihr Teilproblem.
- Jede bekannte **ganzzahlige** Lösung liefert eine globale **untere Schranke**  $\underline{F}$ .
- Start:  $X_1 = X_2 = 0$  ist ganzzahlig zulässig  $\Rightarrow \underline{F} = 0$ .

# Schranken (Bounding)

- Jede LP-Relaxation liefert eine **obere Schranke** für ihr Teilproblem.
- Jede bekannte **ganzzahlige** Lösung liefert eine globale **untere Schranke**  $\underline{F}$ .
- Start:  $X_1 = X_2 = 0$  ist ganzzahlig zulässig  $\Rightarrow \underline{F} = 0$ .
- Ist keine zulässige Lösung bekannt, startet man mit  $\underline{F} = -\infty$ .

# Schranken (Bounding)

- Jede LP-Relaxation liefert eine **obere Schranke** für ihr Teilproblem.
- Jede bekannte **ganzzahlige** Lösung liefert eine globale **untere Schranke**  $\underline{F}$ .
- Start:  $X_1 = X_2 = 0$  ist ganzzahlig zulässig  $\Rightarrow \underline{F} = 0$ .
- Ist keine zulässige Lösung bekannt, startet man mit  $\underline{F} = -\infty$ .

Ein Teilproblem  $P_i$  ist **ausgelotet**, wenn seine obere Schranke  $\bar{F}_i$  die untere Schranke  $\underline{F}$  nicht mehr übertrifft.

# Die Auslotungsregeln

Ein Teilproblem  $P_i$  wird **nicht weiter verzweigt**, wenn:

# Die Auslotungsregeln

Ein Teilproblem  $P_i$  wird **nicht weiter verzweigt**, wenn:

- **(3a)**  $\bar{F}_i \leq \underline{F}$ : es kann nichts Besseres mehr liefern.

# Die Auslotungsregeln

Ein Teilproblem  $P_i$  wird **nicht weiter verzweigt**, wenn:

- **(3a)**  $\bar{F}_i \leq \underline{F}$ : es kann nichts Besseres mehr liefern.
- **(3b)** die Relaxationslösung **ganzzahlig** ist und  $\underline{F}$  **verbessert** ( $\bar{F}_i > \underline{F}$ ): neue beste Lösung.

# Die Auslotungsregeln

Ein Teilproblem  $P_i$  wird **nicht weiter verzweigt**, wenn:

- **(3a)**  $\bar{F}_i \leq \underline{F}$ : es kann nichts Besseres mehr liefern.
- **(3b)** die Relaxationslösung **ganzzahlig** ist und  $\underline{F}$  **verbessert** ( $\bar{F}_i > \underline{F}$ ): neue beste Lösung.
- **(3c)** die Relaxation **unzulässig** ist: dann auch  $P_i$ .

# Die Auslotungsregeln

Ein Teilproblem  $P_i$  wird **nicht weiter verzweigt**, wenn:

- **(3a)**  $\bar{F}_i \leq \underline{F}$ : es kann nichts Besseres mehr liefern.
- **(3b)** die Relaxationslösung **ganzzahlig** ist und  $\underline{F}$  **verbessert** ( $\bar{F}_i > \underline{F}$ ): neue beste Lösung.
- **(3c)** die Relaxation **unzulässig** ist: dann auch  $P_i$ .

Im Baum stehen ✕ für ausgelotet (3a, 3c), ✓ für eine neue beste Lösung (3b) und ★ für die Lösung, die am Ende optimal ist.

# Auslotung bei Minimierung

Bei **Minimierung** tauschen obere und untere Schranke die Rollen:

# Auslotung bei Minimierung

Bei **Minimierung** tauschen obere und untere Schranke die Rollen:

- Jede bekannte ganzzahlige Lösung liefert eine globale **obere** Schranke  $\bar{F}$ .

# Auslotung bei Minimierung

Bei **Minimierung** tauschen obere und untere Schranke die Rollen:

- Jede bekannte ganzzahlige Lösung liefert eine globale **obere** Schranke  $\bar{F}$ .
- Jede Relaxation liefert eine **untere** Schranke  $\underline{F}_i$ .

# Auslotung bei Minimierung

Bei **Minimierung** tauschen obere und untere Schranke die Rollen:

- Jede bekannte ganzzahlige Lösung liefert eine globale **obere** Schranke  $\bar{F}$ .
- Jede Relaxation liefert eine **untere** Schranke  $F_{-i}$ .
- **(3a)**  $F_{-i} \geq \bar{F}$ : es kann nichts Besseres mehr liefern.

# Auslotung bei Minimierung

Bei **Minimierung** tauschen obere und untere Schranke die Rollen:

- Jede bekannte ganzzahlige Lösung liefert eine globale **obere** Schranke  $\bar{F}$ .
- Jede Relaxation liefert eine **untere** Schranke  $\underline{F}_i$ .
- **(3a)**  $\underline{F}_i \geq \bar{F}$ : es kann nichts Besseres mehr liefern.
- **(3b)** die Relaxationslösung ist **ganzzahlig** und **verbessert**  $\bar{F}$  ( $\underline{F}_i < \bar{F}$ ): neue beste Lösung.

# Auslotung bei Minimierung

Bei **Minimierung** tauschen obere und untere Schranke die Rollen:

- Jede bekannte ganzzahlige Lösung liefert eine globale **obere** Schranke  $\bar{F}$ .
- Jede Relaxation liefert eine **untere** Schranke  $F_{-i}$ .
- **(3a)**  $F_{-i} \geq \bar{F}$ : es kann nichts Besseres mehr liefern.
- **(3b)** die Relaxationslösung ist **ganzzahlig** und **verbessert**  $\bar{F}$  ( $F_{-i} < \bar{F}$ ): neue beste Lösung.
- **(3c)** unverändert: Relaxation unzulässig, dann auch  $P_i$ .

# Relaxationen

Die LP-Relaxation ist nur **eine** Möglichkeit, Schranken zu gewinnen:

# Relaxationen

Die LP-Relaxation ist nur **eine** Möglichkeit, Schranken zu gewinnen:

- **LP-Relaxation:** Ganzzahligkeit weglassen (unser Standard).

# Relaxationen

Die LP-Relaxation ist nur **eine** Möglichkeit, Schranken zu gewinnen:

- **LP-Relaxation:** Ganzzahligkeit weglassen (unser Standard).
- **Weglassen von Nebenbedingungen:** jedes entspannte Problem schrankt.

# Relaxationen

Die LP-Relaxation ist nur **eine** Möglichkeit, Schranken zu gewinnen:

- **LP-Relaxation:** Ganzzahligkeit weglassen (unser Standard).
- **Weglassen von Nebenbedingungen:** jedes entspannte Problem schrankt.
- **Lagrange-Relaxation:** verletzte Restriktionen wandern bestraft in die Zielfunktion.

# Verzweigen (Branching)

Wähle eine **nicht ganzzahlige** Variable und verzweige, um den nicht ganzzahligen Bereich auszuschließen:

$$P_1 : P_0 \text{ und } X_1 \leq 2 \qquad P_2 : P_0 \text{ und } X_1 \geq 3$$

# Verzweigen (Branching)

Wähle eine **nicht ganzzahlige** Variable und verzweige, um den nicht ganzzahligen Bereich auszuschließen:

$$P_1 : P_0 \text{ und } X_1 \leq 2 \quad P_2 : P_0 \text{ und } X_1 \geq 3$$

Übliche Regel: die Variable mit dem **geringsten Abstand** zur nächsten ganzen Zahl, hier  $X_1$  ( $0,286 < 0,429$ ). Einfache Alternative: die nicht ganzzahlige Variable mit dem **kleinsten Index**.

# Verzweigen (Branching)

Wähle eine **nicht ganzzahlige** Variable und verzweige, um den nicht ganzzahligen Bereich auszuschließen:

$$P_1 : P_0 \text{ und } X_1 \leq 2 \quad P_2 : P_0 \text{ und } X_1 \geq 3$$

Übliche Regel: die Variable mit dem **geringsten Abstand** zur nächsten ganzen Zahl, hier  $X_1$  ( $0,286 < 0,429$ ). Einfache Alternative: die nicht ganzzahlige Variable mit dem **kleinsten Index**.

Die optimale ganzzahlige Lösung von  $P_0$  steckt in  $P_1$  **oder**  $P_2$ .

# Suchstrategien

Frage: Welches offene Teilproblem als Nächstes?

# Suchstrategien

Frage: Welches offene Teilproblem als Nächstes?

- **Reine Tiefensuche:** einen Ast bis zum Ende, schnell eine erste Lösung.

# Suchstrategien

Frage: Welches offene Teilproblem als Nächstes?

- **Reine Tiefensuche:** einen Ast bis zum Ende, schnell eine erste Lösung.
- **Tiefensuche + MUB:** nach vollständiger Verzweigung den Ast mit **größter** oberer Schranke (Maximum Upper Bound).

# Suchstrategien

Frage: Welches offene Teilproblem als Nächstes?

- **Reine Tiefensuche:** einen Ast bis zum Ende, schnell eine erste Lösung.
- **Tiefensuche + MUB:** nach vollständiger Verzweigung den Ast mit **größter** oberer Schranke (Maximum Upper Bound).
- **Breitensuche:** unter **allen** offenen Knoten den mit größter Schranke.

# Suchstrategien

Frage: Welches offene Teilproblem als Nächstes?

- **Reine Tiefensuche:** einen Ast bis zum Ende, schnell eine erste Lösung.
- **Tiefensuche + MUB:** nach vollständiger Verzweigung den Ast mit **größter** oberer Schranke (Maximum Upper Bound).
- **Breitensuche:** unter **allen** offenen Knoten den mit größter Schranke.

**Backtracking** (beide Tiefensuchen, am Ende eines Astes): zurück zum vorangegangenen Knoten, dessen nächstes Teilproblem betrachten, sofern noch nicht ausgelotet.

# Suchstrategien bei Minimierung

Bei **Minimierung** drehen sich die Suchstrategien mit:

# Suchstrategien bei Minimierung

Bei **Minimierung** drehen sich die Suchstrategien mit:

- **Tiefensuche + MLB** (verzweigte Tiefensuche): nach vollständiger Verzweigung den Ast mit **kleinster unterer** Schranke (Minimum Lower Bound).

# Suchstrategien bei Minimierung

Bei **Minimierung** drehen sich die Suchstrategien mit:

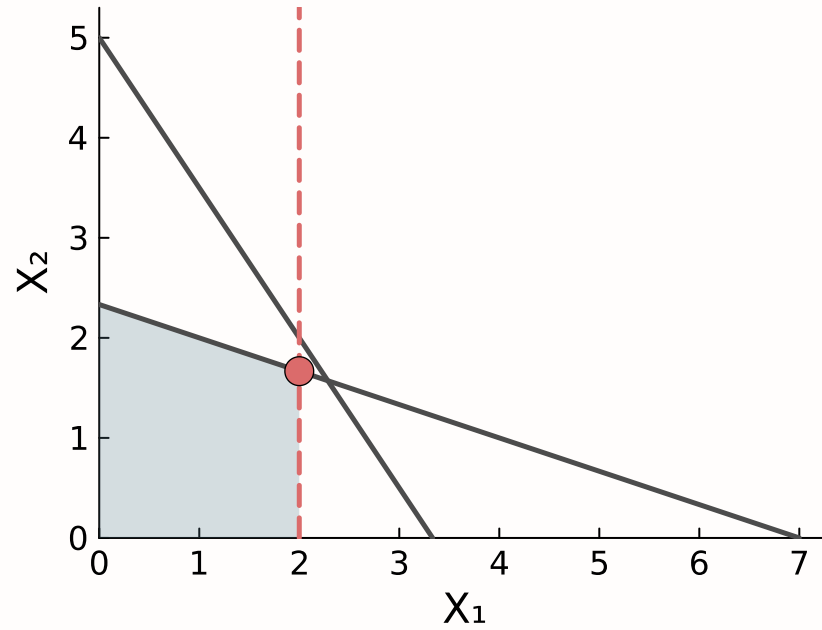
- **Tiefensuche + MLB** (verzweigte Tiefensuche): nach vollständiger Verzweigung den Ast mit **kleinster unterer** Schranke (Minimum Lower Bound).
- **Breitensuche**: unter **allen** offenen Knoten den mit kleinster unterer Schranke.

# Suchstrategien bei Minimierung

Bei **Minimierung** drehen sich die Suchstrategien mit:

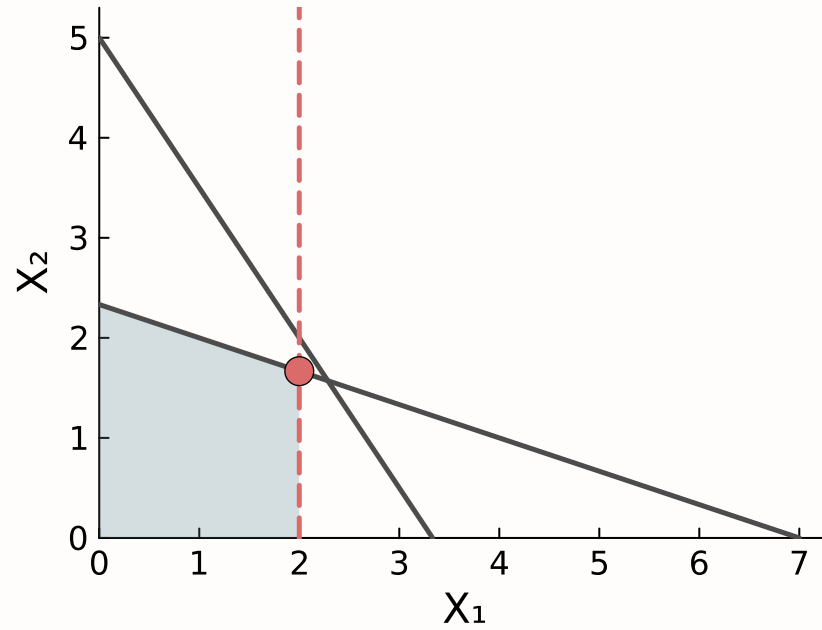
- **Tiefensuche + MLB** (verzweigte Tiefensuche): nach vollständiger Verzweigung den Ast mit **kleinster unterer** Schranke (Minimum Lower Bound).
- **Breitensuche**: unter **allen** offenen Knoten den mit kleinster unterer Schranke.
- Ohne bekannte Lösung startet  $\bar{F} = +\infty$ .

# Teilprobleme $P_1$ und $P_2$

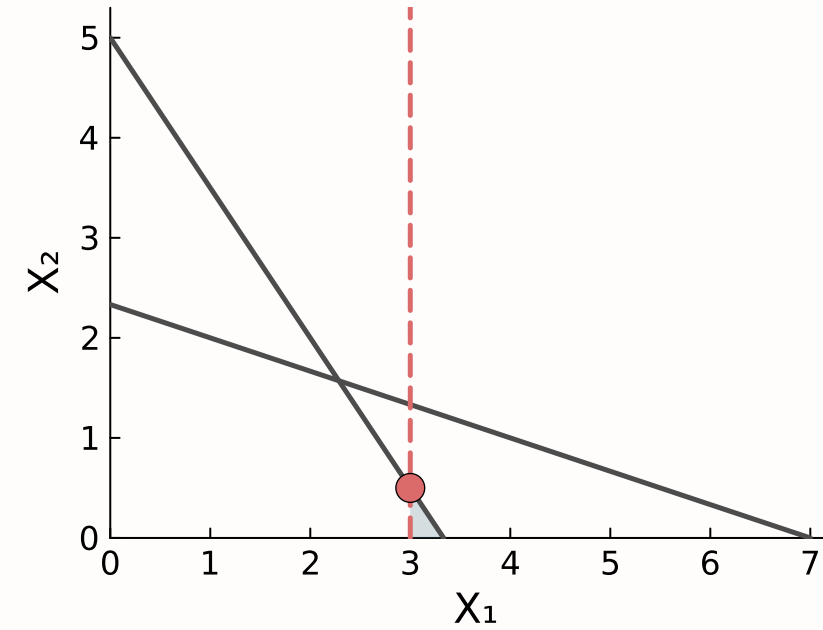


$P_1$  ( $x_1 \leq 2$ ):  $(2, 1,67), \bar{F}_1 = 5,33$

# Teilprobleme $P_1$ und $P_2$



$P_1$  ( $X_1 \leq 2$ ):  $(2, 1,67), \bar{F}_1 = 5,33$



$P_2$  ( $X_1 \geq 3$ ):  $(3, 0,5), \bar{F}_2 = 4$

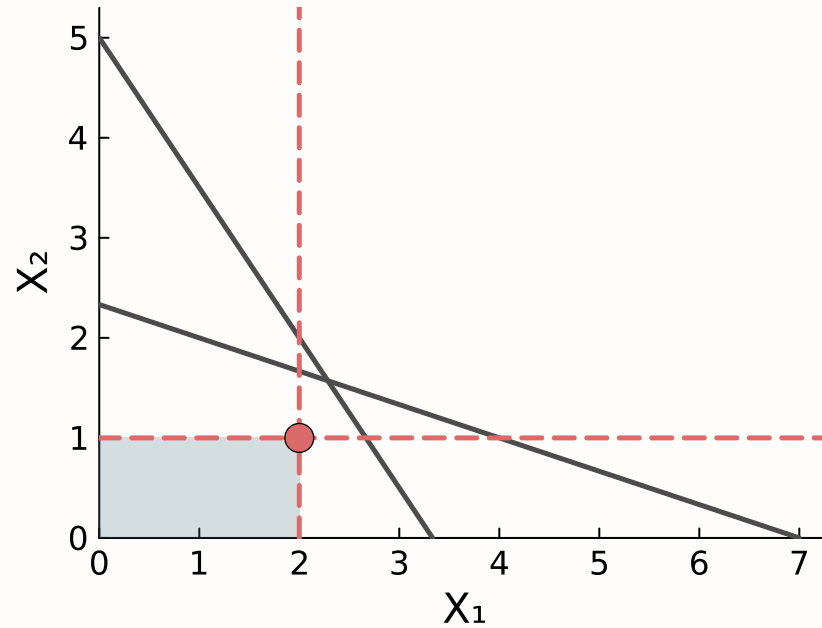
# Weiter mit dem besten Ast

$\bar{F}_1 = 5,33 > \bar{F}_2 = 4 \Rightarrow$  verzweige zuerst  $P_1$  (Maximum-Upper-Bound).

In  $P_1$  ist  $X_2 = 1,667$  nicht ganzzahlig:

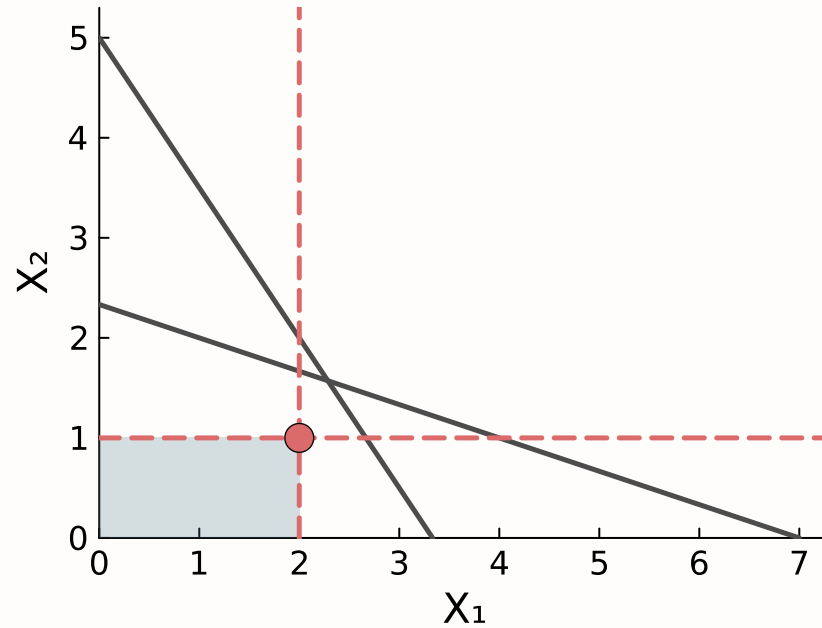
$$P_3 : P_1 \text{ und } X_2 \leq 1 \qquad P_4 : P_1 \text{ und } X_2 \geq 2$$

# Teilprobleme $P_3$ und $P_4$

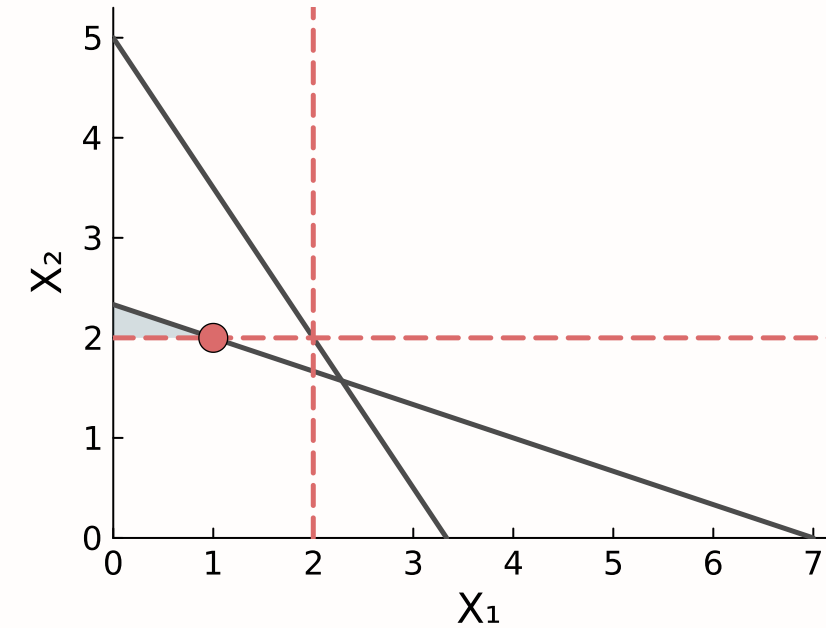


$P_3 : (2, 1), F_3 = 4, \text{ ganzzahlig} \Rightarrow \underline{F} = 4$

# Teilprobleme $P_3$ und $P_4$



$P_3 : (2, 1), F_3 = 4$ , **ganzzahlig**  $\Rightarrow \underline{F} = 4$



$P_4 : (1, 2), F_4 = 5$ , **ganzzahlig**  $\Rightarrow \underline{F} = 5$

# $P_2$ wird ausgelotet

$P_3$  und  $P_4$  sind ganzzahlig gelöst. Die beste untere Schranke ist jetzt  $\underline{F} = 5$ .

$$\bar{F}_2 = 4 \leq \underline{F} = 5 \Rightarrow P_2 \text{ ausgelotet}$$

# $P_2$ wird ausgelotet

$P_3$  und  $P_4$  sind ganzzahlig gelöst. Die beste untere Schranke ist jetzt  $\underline{F} = 5$ .

$$\bar{F}_2 = 4 \leq \underline{F} = 5 \Rightarrow P_2 \text{ ausgelotet}$$

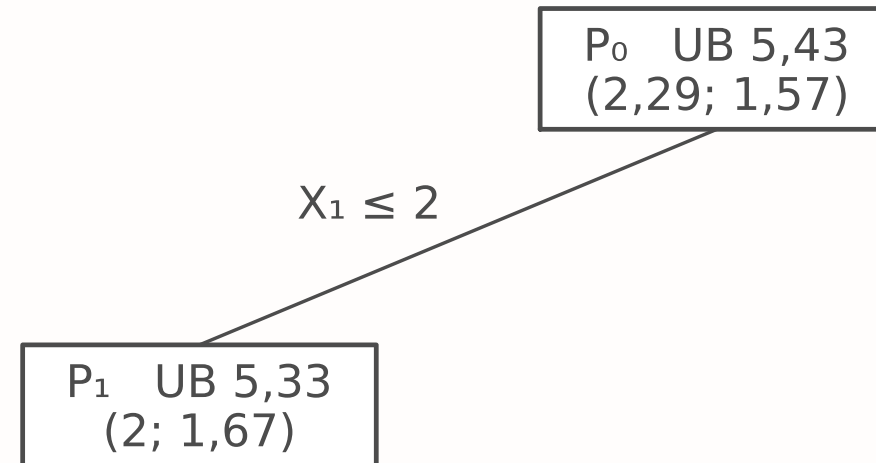
$P_2$  kann **keine** bessere Lösung mehr enthalten. Es muss nicht weiter untersucht werden.

# Der Branch-&-Bound-Baum

|                               |
|-------------------------------|
| $P_0$ UB 5,43<br>(2,29; 1,57) |
|-------------------------------|

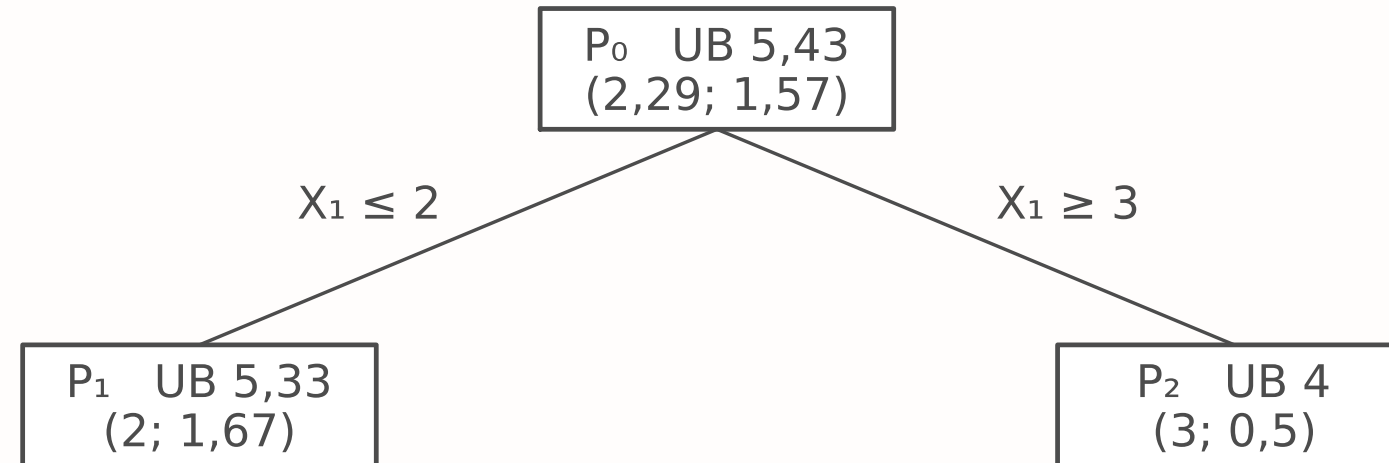
$P_0$  : LP-Optimum nicht ganzzahlig, UB ( $= \bar{F}$ ) 5,43, verzweigen nach  $X_1$  .

# Der Branch-&-Bound-Baum



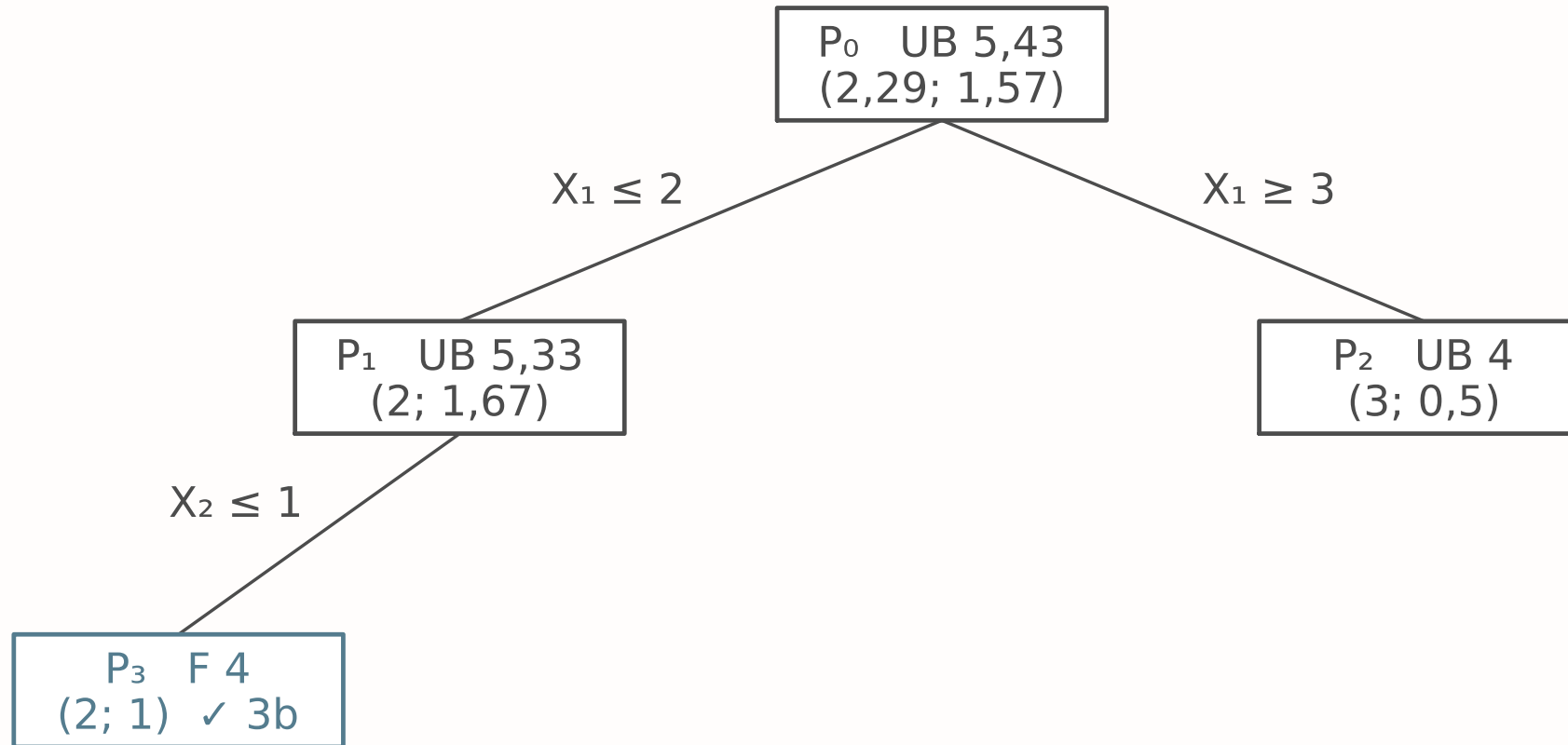
$P_1$  mit  $X_1 \leq 2$ :  $(2, 1,67)$ ,  $\bar{F}_1 = 5,33 > \underline{F} = 0$ , nicht ganzzahlig: offen.

# Der Branch-&-Bound-Baum



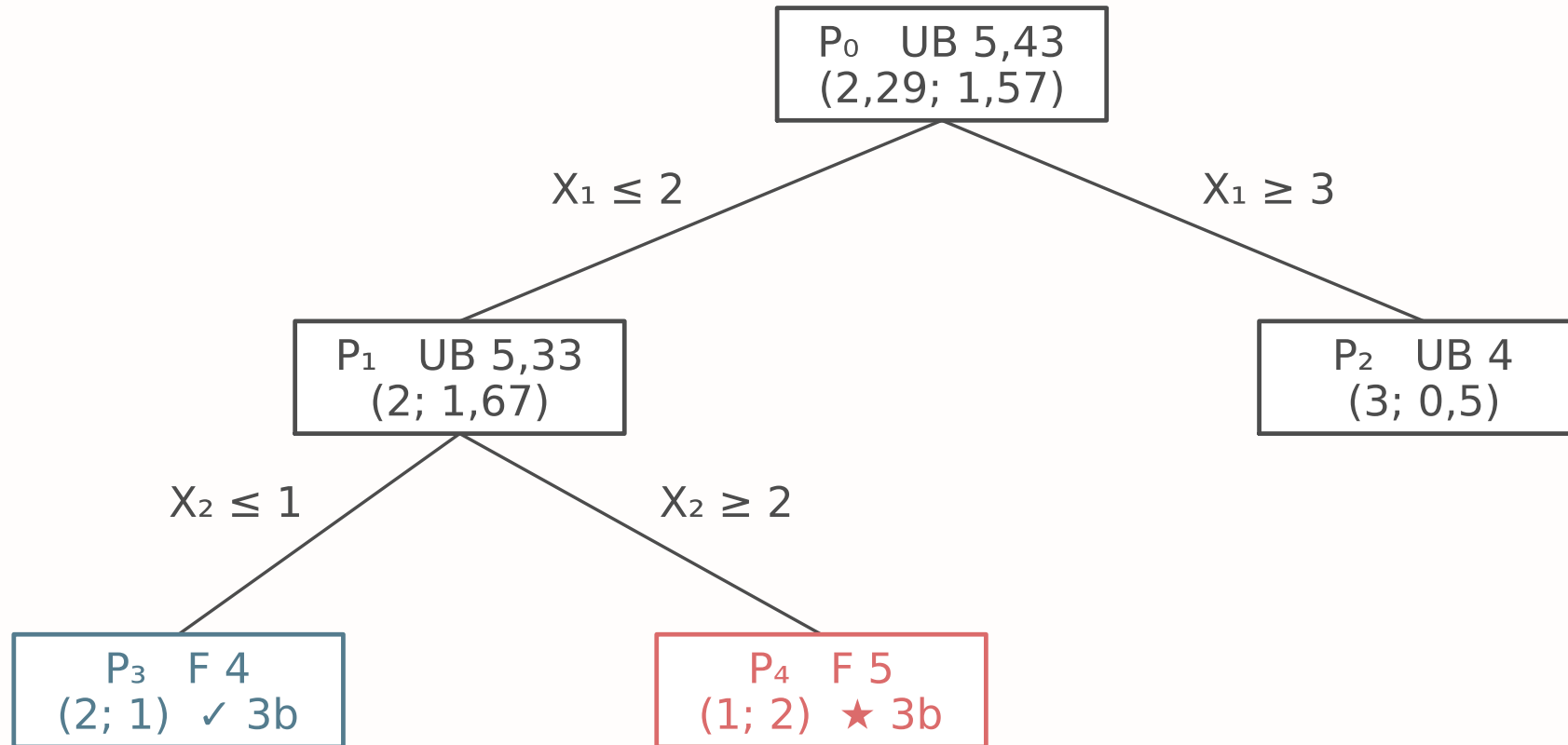
$P_2$  mit  $X_1 \geq 3$ :  $(3, 0,5)$ ,  $\bar{F}_2 = 4 > \underline{F} = 0$ : offen.  $P_1$  verspricht mehr (MUB),  $P_2$  wartet.

# Der Branch-&-Bound-Baum



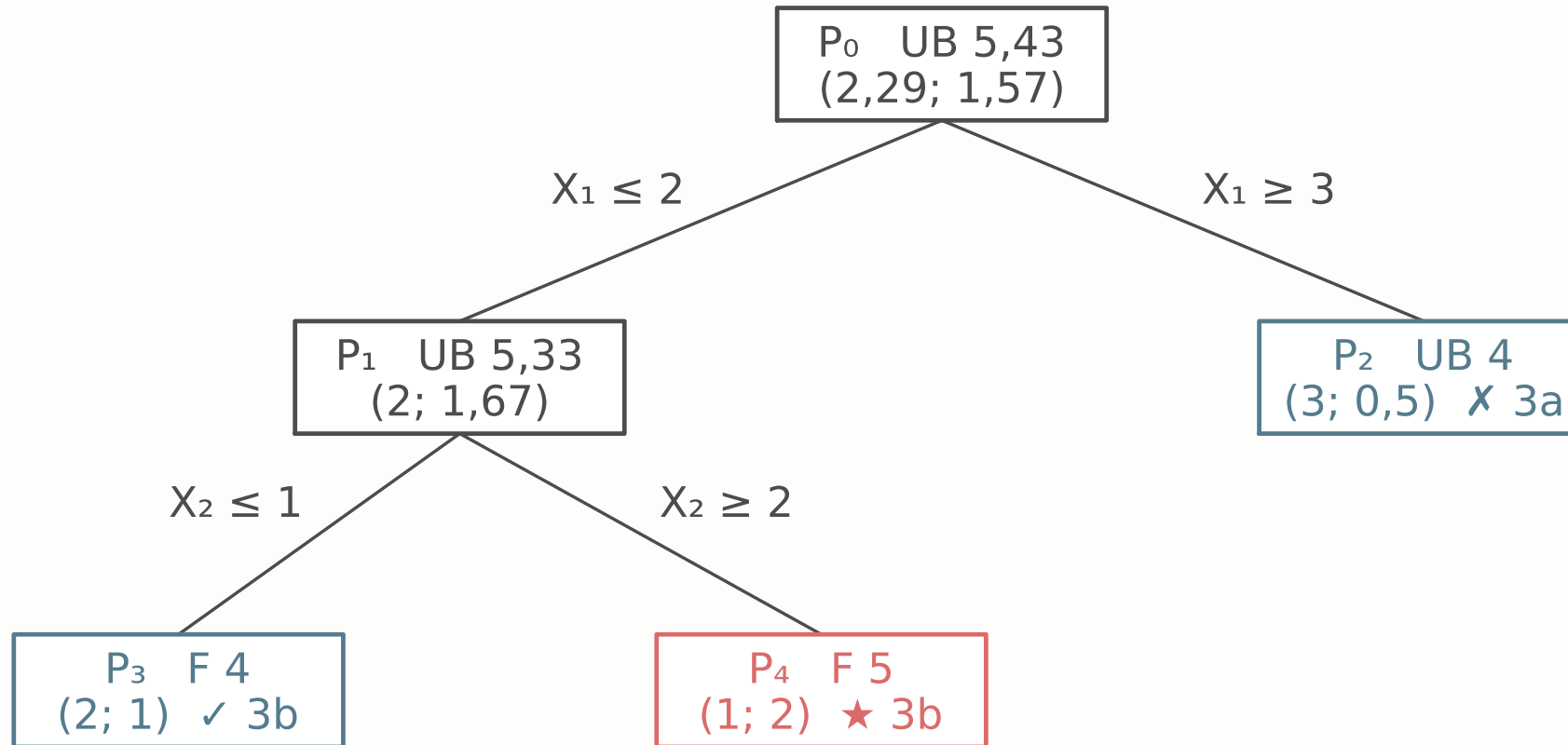
$P_3$  mit  $X_2 \leq 1$ :  $(2, 1)$  ganzzahlig,  $4 > \underline{F} = 0$  (3b)  $\Rightarrow \underline{F} = 4$ .

# Der Branch-&-Bound-Baum



$P_4$  mit  $X_2 \geq 2$ : (1, 2) ganzzahlig,  $5 > \underline{F} = 4$  (3b)  $\Rightarrow \underline{F} = 5$ .

# Der Branch-&-Bound-Baum



$P_2$  ausgelotet (3a), denn  $\bar{F}_2 = 4 \leq \underline{F} = 5$ .

# Optimale Lösung

Alle Teilprobleme sind ausgelotet. Die beste ganzzahlige Lösung ist optimal:

! Optimale Lösung

$$X_1^* = 1, \quad X_2^* = 2, \quad F_0^* = 5$$

# Optimale Lösung

Alle Teilprobleme sind ausgelotet. Die beste ganzzahlige Lösung ist optimal:

## ! Optimale Lösung

$$X_1^* = 1, \quad X_2^* = 2, \quad F_0^* = 5$$

Beachten Sie: das **gerundete** LP-Optimum (2,2) ist gar nicht zulässig. Runden führt oft in die Irre.

# Kontrolle mit Julia

# Ganzzahlig mit JuMP

Ganzzahligkeit ist in JuMP nur ein Zusatz. HiGHS führt Branch & Bound automatisch aus:

```
1 using JuMP, HiGHS
2
3 modell = Model(HiGHS.Optimizer); set_silent(modell)
4 @variable(modell, X1 >= 0, Int)          # ganzzahlig
5 @variable(modell, X2 >= 0, Int)
6 @objective(modell, Max, X1 + 2X2)
7 @constraint(modell, X1 + 3X2 <= 7)
8 @constraint(modell, 3X1 + 2X2 <= 10)
9
10 optimize!(modell)
11 println("F* = ", objective_value(modell),
12         " mit X1 = ", value(X1), ", X2 = ", value(X2))
```

# Ganzzahlig mit JuMP

Ganzzahligkeit ist in JuMP nur ein Zusatz. HiGHS führt Branch & Bound automatisch aus:

```
1 using JuMP, HiGHS
2
3 modell = Model(HiGHS.Optimizer); set_silent(modell)
4 @variable(modell, X1 >= 0, Int)           # ganzzahlig
5 @variable(modell, X2 >= 0, Int)
6 @objective(modell, Max, X1 + 2X2)
7 @constraint(modell, X1 + 3X2 <= 7)
8 @constraint(modell, 3X1 + 2X2 <= 10)
9
10 optimize!(modell)
11 println("F* = ", objective_value(modell),
12         " mit X1 = ", value(X1), ", X2 = ", value(X2))
```

F\* = 5.0 mit X1 = 1.0, X2 = 2.0

# Binäres Beispiel

# Auftragsauswahl

Ein Fuhrunternehmen kann vier Aufträge annehmen. Welche maximieren den Erlös? Kapazität: **30 t** und **50 Paletten**.

| Auftrag     | 1   | 2   | 3   | 4   |
|-------------|-----|-----|-----|-----|
| Gewicht [t] | 10  | 15  | 12  | 14  |
| Paletten    | 12  | 30  | 16  | 10  |
| Erlös       | 200 | 350 | 250 | 180 |

# Das Binärmodell

Mit  $X_j \in \{0, 1\}$  (Auftrag  $j$  annehmen?):

$$\max F = \sum_j e_j X_j$$

$$\sum_j g_j X_j \leq G \quad (\text{Gewicht})$$

$$\sum_j p_j X_j \leq P \quad (\text{Paletten})$$

$$X_j \in \{0, 1\}$$

# Das Binärmodell

Mit  $X_j \in \{0, 1\}$  (Auftrag  $j$  annehmen?):

$$\max F = \sum_j e_j X_j$$

$$\sum_j g_j X_j \leq G \quad (\text{Gewicht})$$

$$\sum_j p_j X_j \leq P \quad (\text{Paletten})$$

$$X_j \in \{0, 1\}$$

**LP-Relaxation:** statt  $X_j \in \{0, 1\}$  nur  $0 \leq X_j \leq 1$ .

# LP-Relaxation lösen

Die LP-Relaxation ist nicht ganzzahlig:

$$X^* = (0, 3, 1, 1, 0), \quad F_0^* = 660$$

# LP-Relaxation lösen

Die LP-Relaxation ist nicht ganzzahlig:

$$X^* = (0,3, 1, 1, 0), \quad F_0^* = 660$$

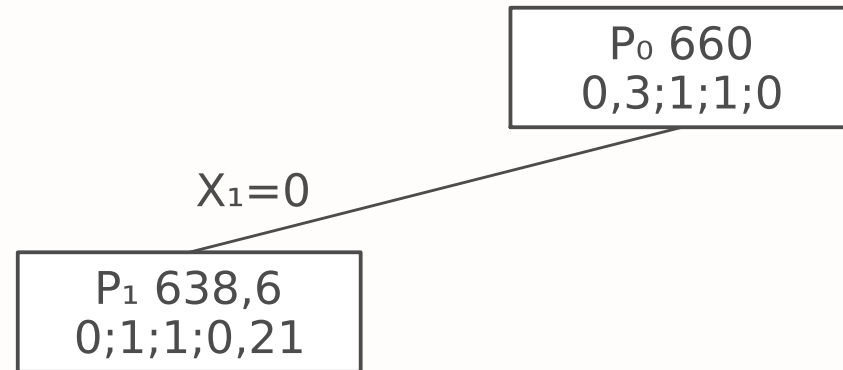
$X_1 = 0,3$  ist nicht ganzzahlig  $\Rightarrow$  verzweige nach  $X_1$ , und zwar, da binär, in die Äste  $X_1 = 0$  und  $X_1 = 1$ .

# Reine Tiefensuche

|                        |
|------------------------|
| $P_0$ 660<br>0,3;1;1;0 |
|------------------------|

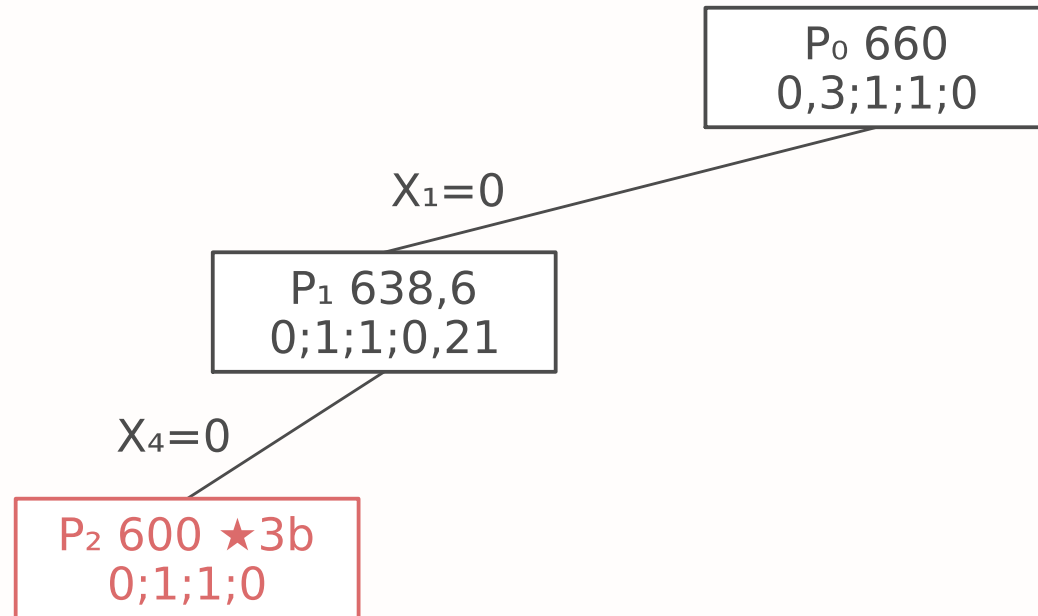
$P_0 : 660 > \underline{F} = 0, X_1$  nicht ganzzahlig  $\Rightarrow$  offen, verzweige nach  $X_1$ ,  
immer  $X_j = 0$  zuerst.

# Reine Tiefensuche



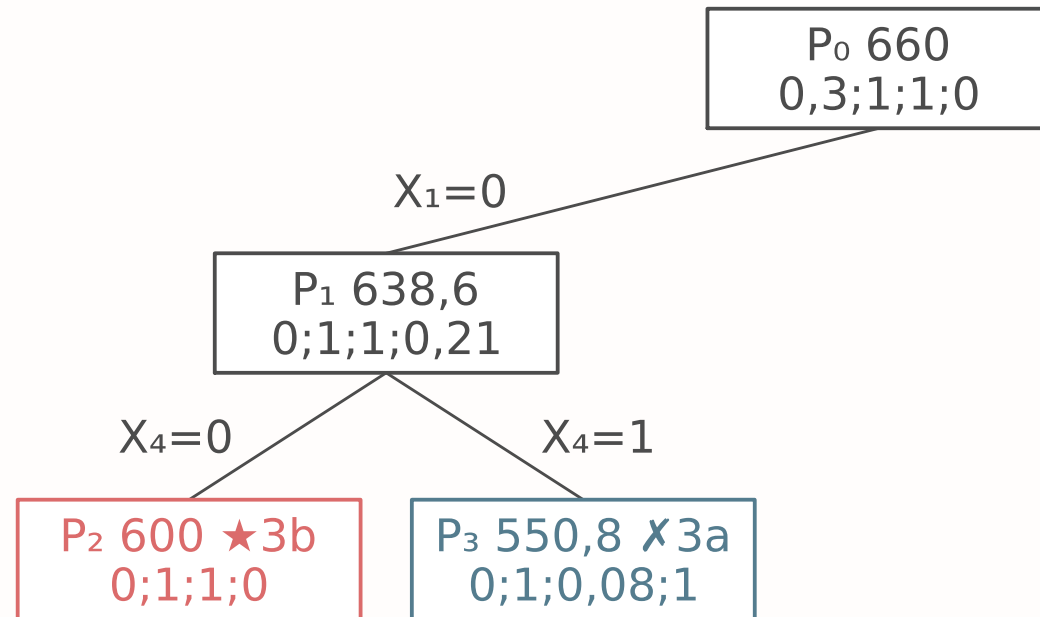
$P_1$  mit  $X_1 = 0$ :  $638,6 > \underline{F} = 0$ ,  $X_4$  nicht ganzzahlig  $\Rightarrow$  offen, verzweige nach  $X_4$ .

# Reine Tiefensuche



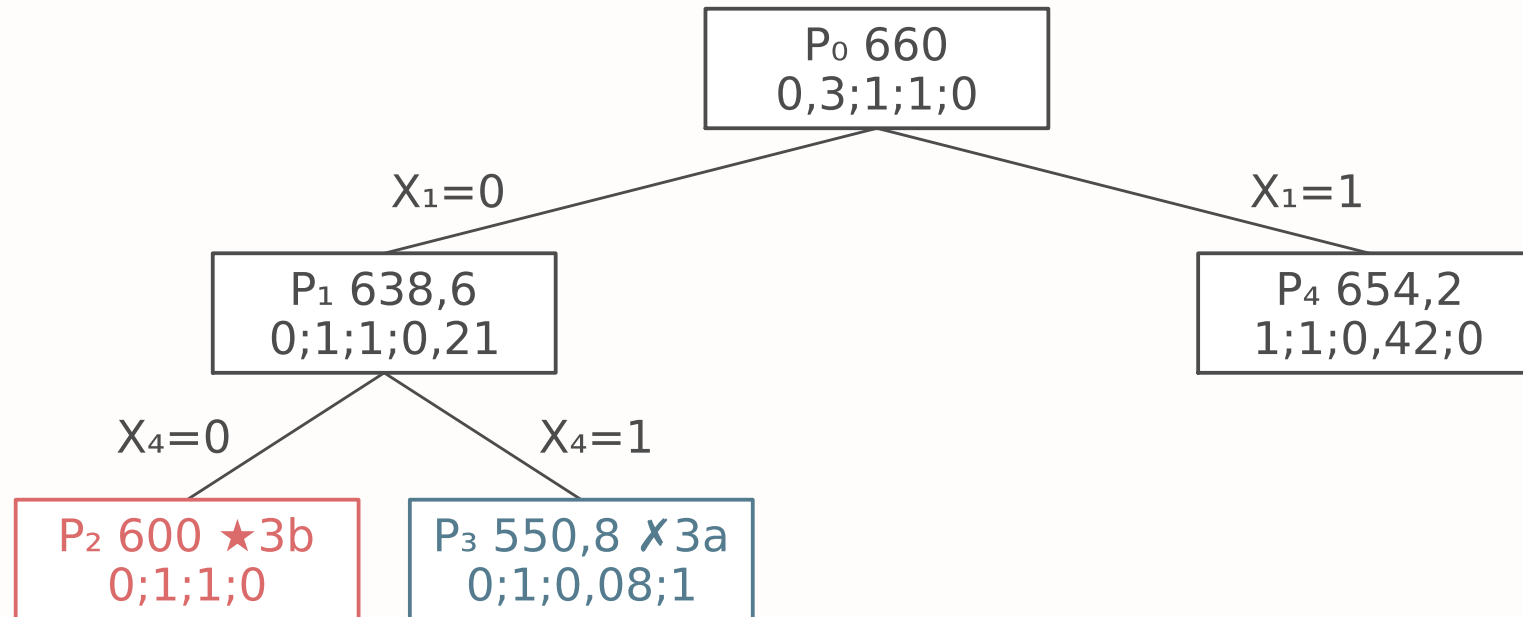
$P_2$  mit  $X_4 = 0$ :  $(0, 1, 1, 0)$  ganzzahlig,  $600 > \underline{F} = 0 \text{ (3b)} \Rightarrow \underline{F} = 600$ .

# Reine Tiefensuche



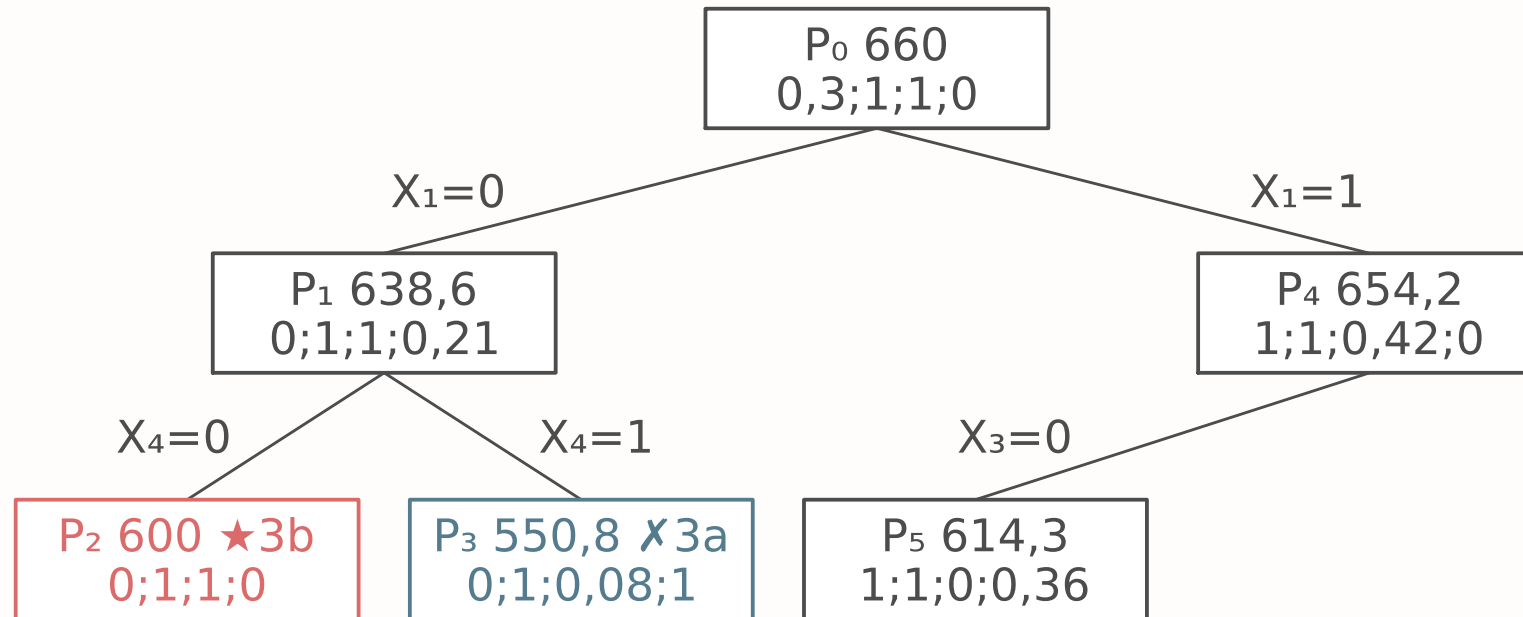
**Backtracking** zu  $P_1$ , Ast  $X_4 = 1$ :  $P_3$  mit  $550,8 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a).

# Reine Tiefensuche



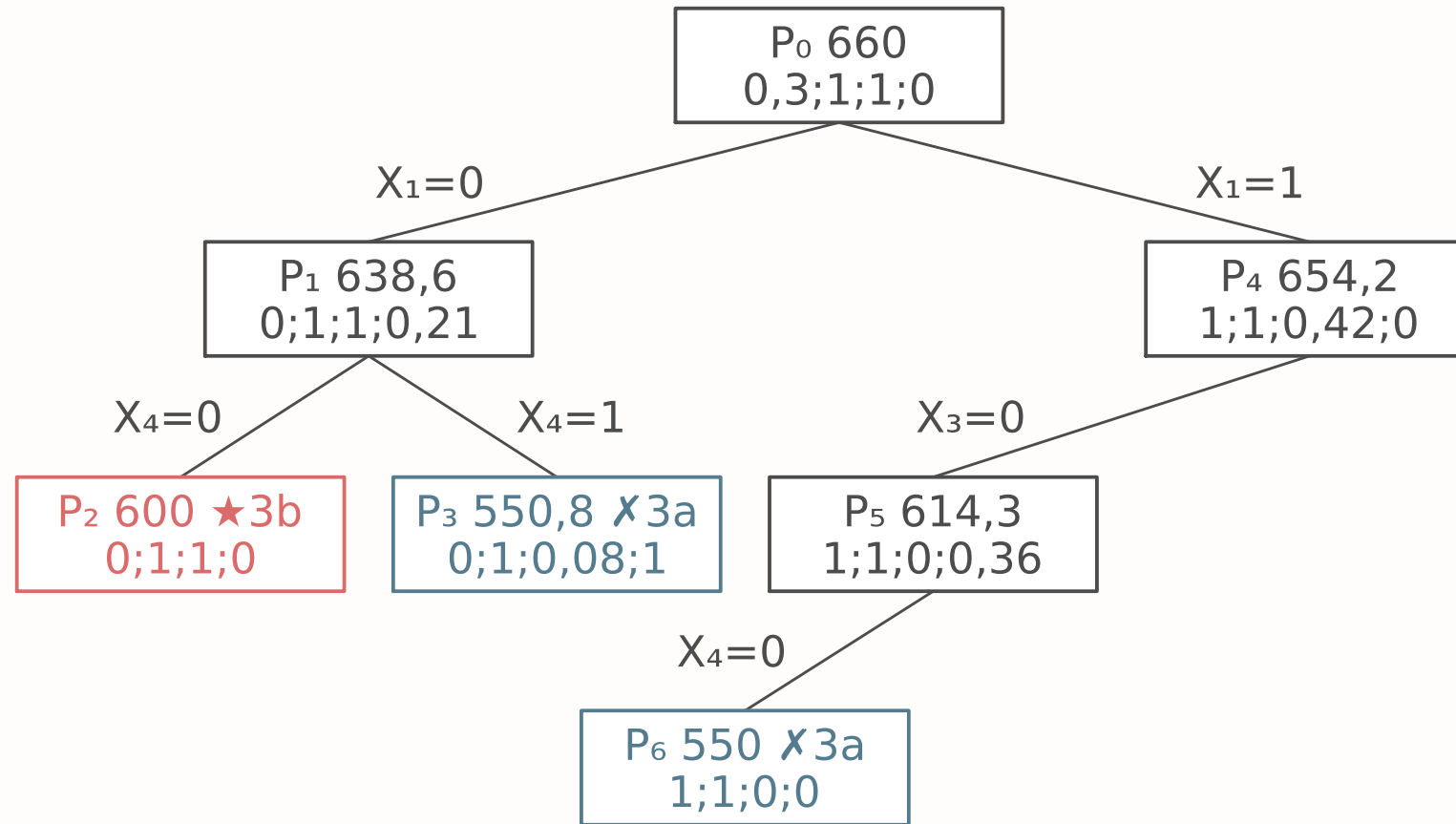
Zurück zu  $P_0$ , Ast  $X_1 = 1$ :  $P_4$  mit  $654,2 > \underline{F} = 600$ ,  $X_3$  nicht ganzzahlig  $\Rightarrow$  offen, verzweige nach  $X_3$ .

# Reine Tiefensuche



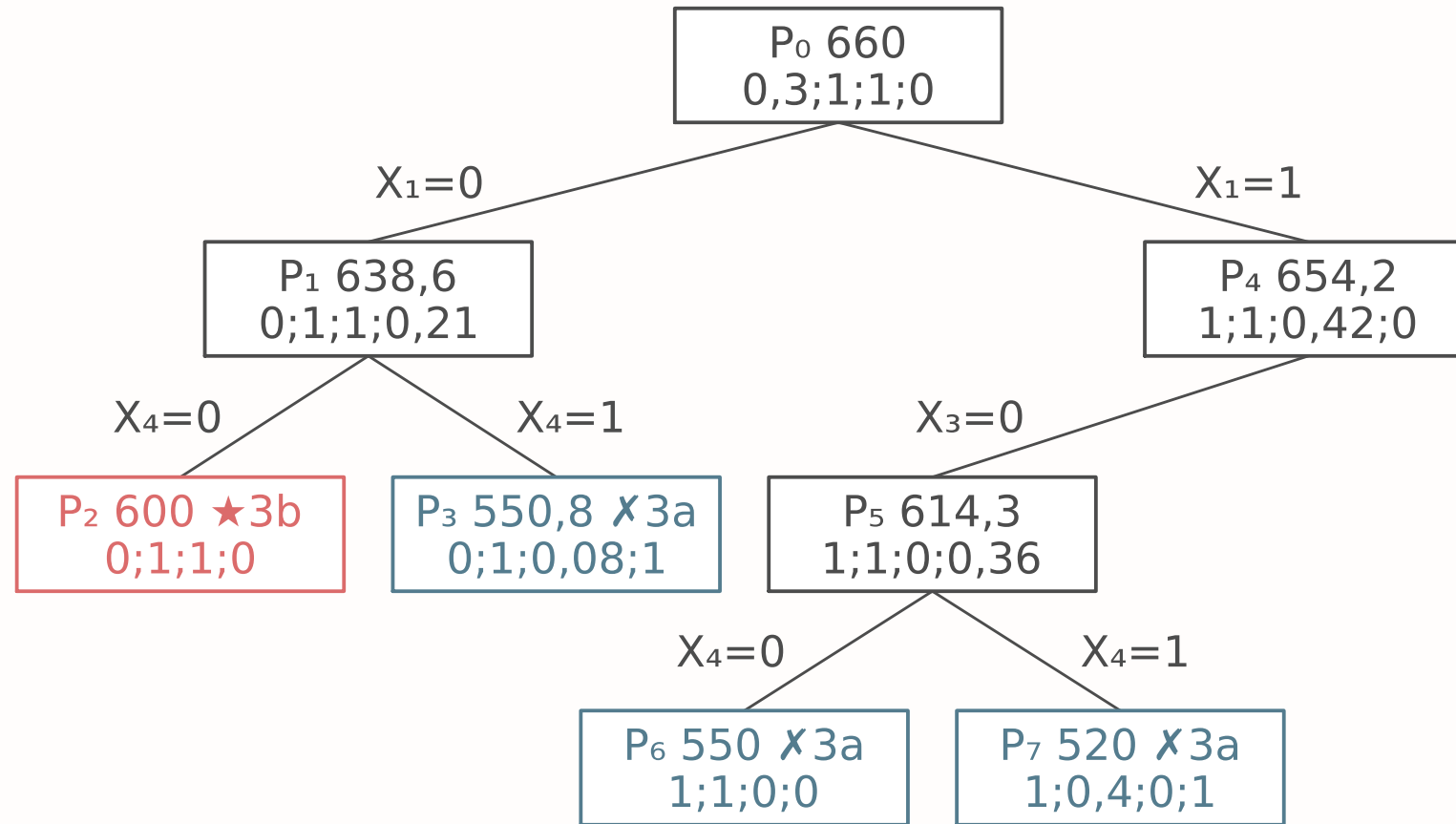
$P_5$  mit  $X_3 = 0$ :  $614,3 > \underline{F} = 600$ ,  $X_4$  nicht ganzzahlig  $\Rightarrow$  offen, verzweige nach  $X_4$ .

# Reine Tiefensuche



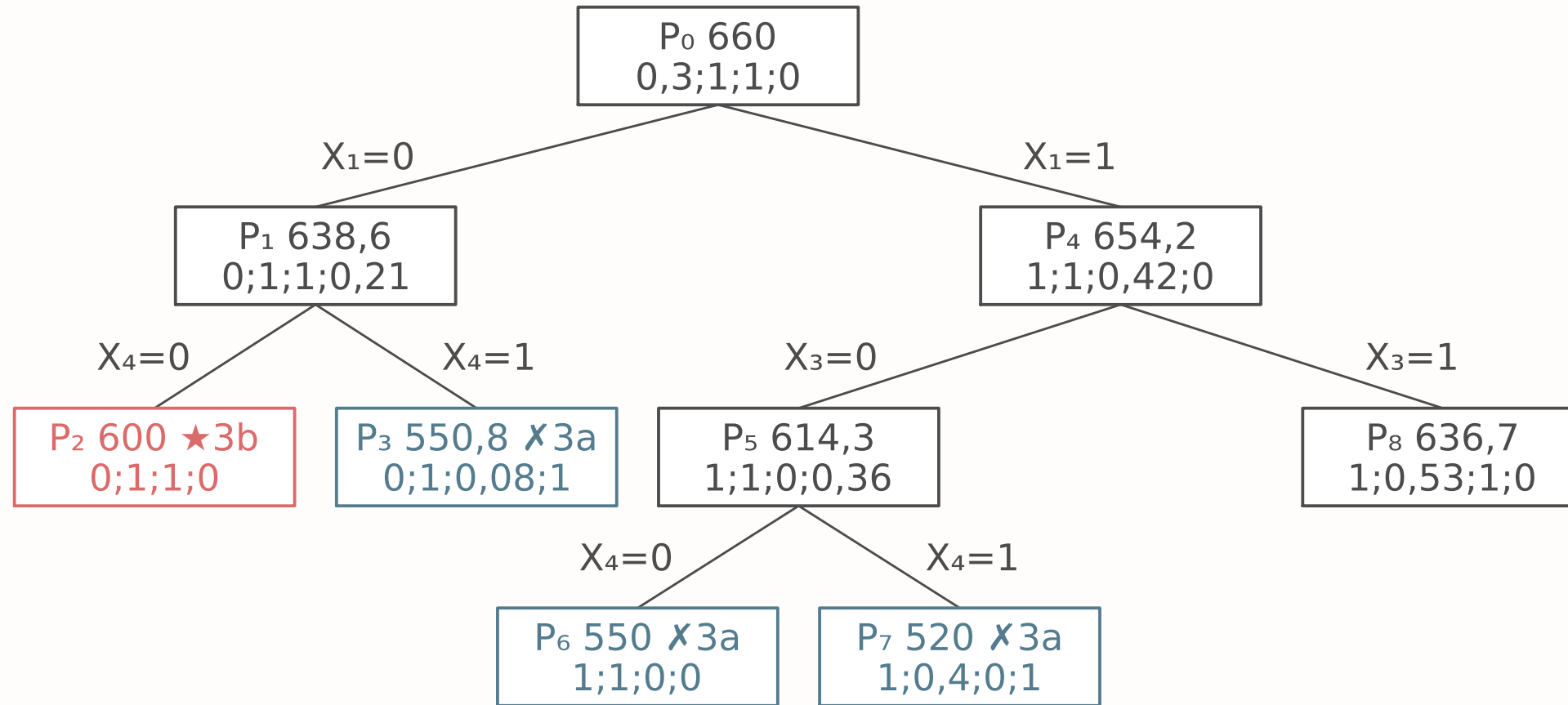
$P_6$  mit  $X_4 = 0$ :  $(1, 1, 0, 0)$  ganzzahlig, aber  $550 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a).

# Reine Tiefensuche



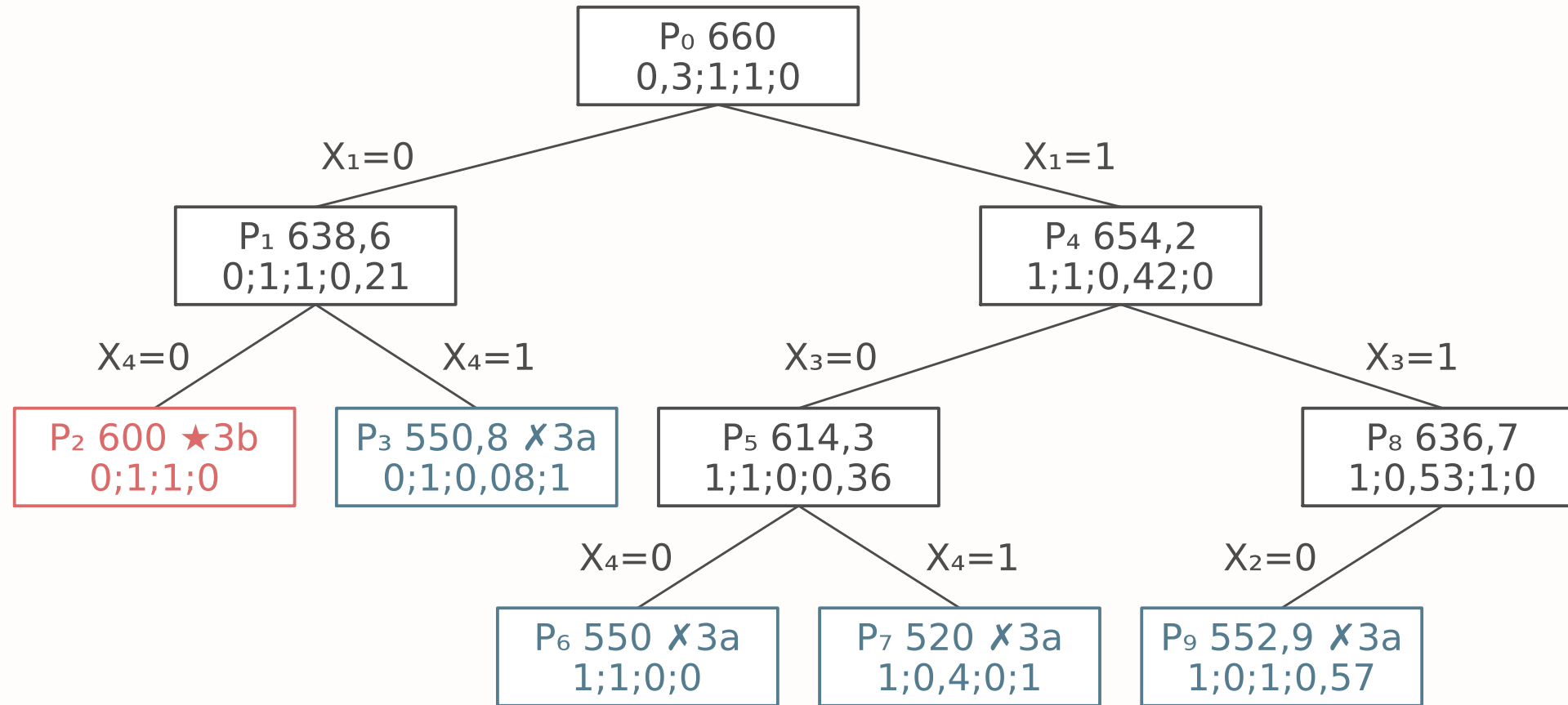
Zurück zu  $P_5$ , Ast  $X_4 = 1$ :  $P_7$  mit  $520 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a).

# Reine Tiefensuche



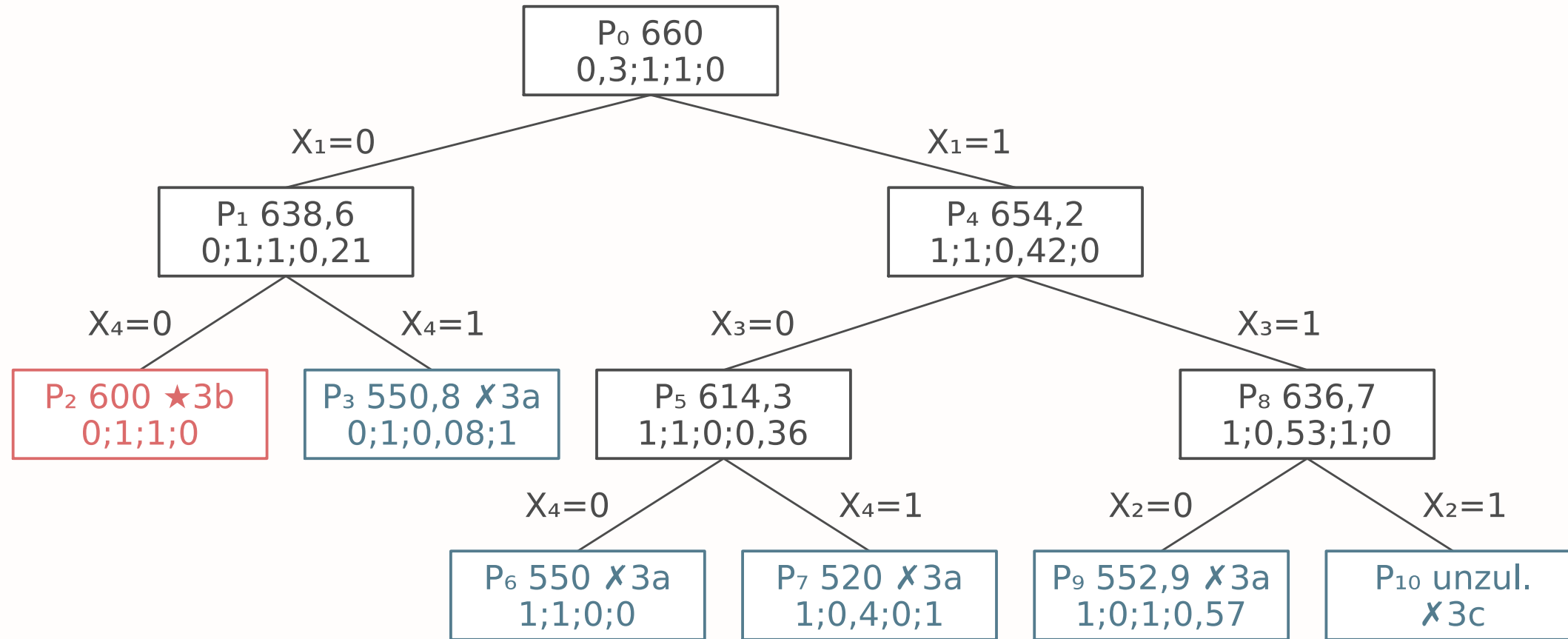
Zurück zu  $P_4$ , Ast  $X_3 = 1$ :  $P_8$  mit  $636,7 > \underline{F} = 600$ ,  $X_2$  nicht ganzzahlig  $\Rightarrow$  offen, verzweige nach  $X_2$ .

# Reine Tiefensuche



$P_9$  mit  $X_2 = 0$ :  $552,9 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a).

# Reine Tiefensuche



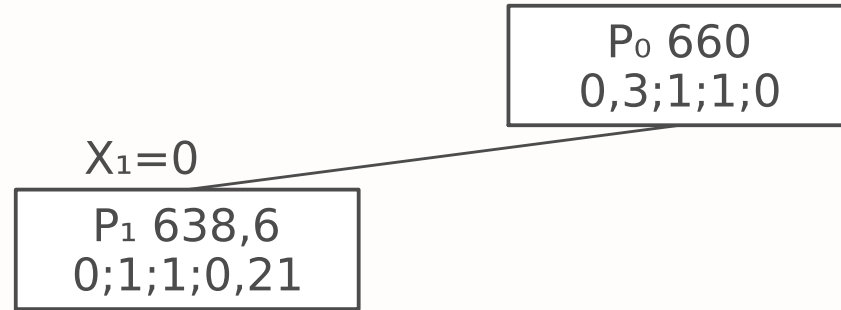
Zurück zu  $P_8$ , Ast  $X_2 = 1$ :  $P_{10}$  unzulässig (3c), Gewicht  $37 > 30$ .  $P_2$  bleibt optimal.

# Tiefensuche + MUB

|                        |
|------------------------|
| $P_0$ 660<br>0,3;1;1;0 |
|------------------------|

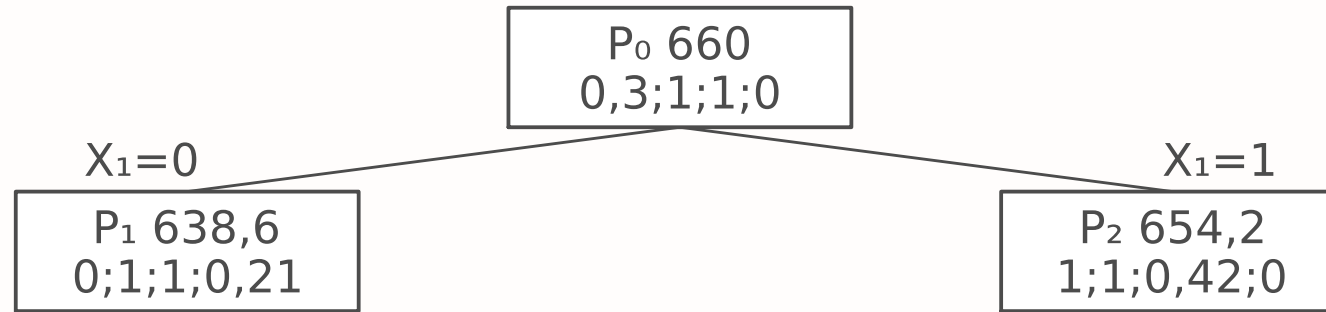
$P_0 : 660 > \underline{F} = 0, X_1$  nicht ganzzahlig  $\Rightarrow$  offen, verzweige nach  $X_1$   
und löse **beide** Äste.

# Tiefensuche + MUB



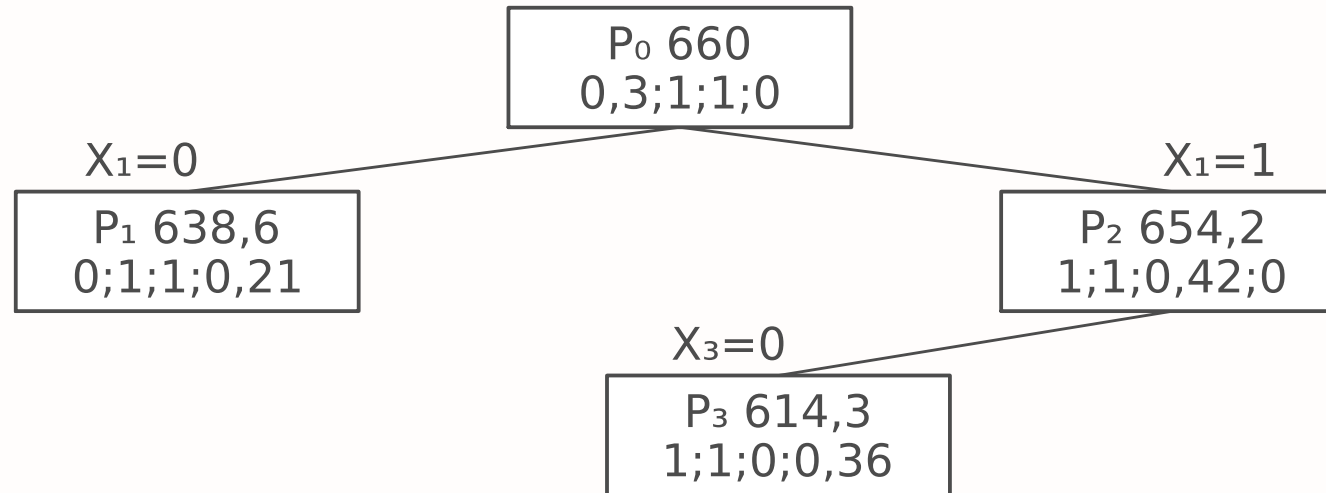
$P_1$  mit  $X_1 = 0$ :  $638,6 > \underline{F} = 0$ ,  $X_4$  nicht ganzzahlig  $\Rightarrow$  offen.

# Tiefensuche + MUB



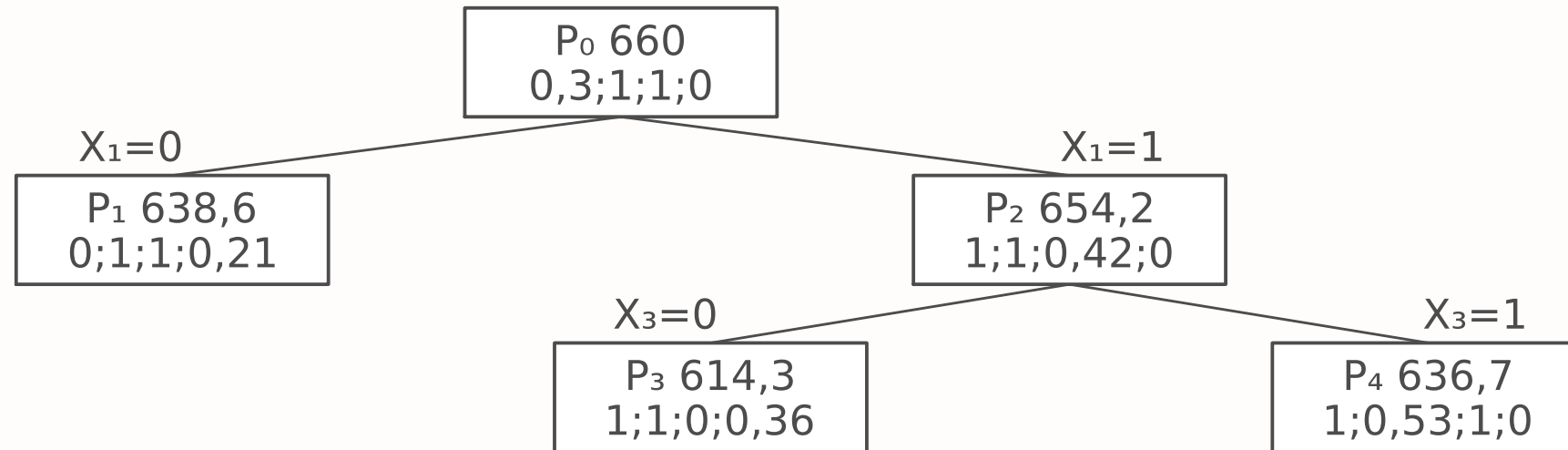
$P_2$  mit  $X_1 = 1$ :  $654,2 > \underline{F} = 0$ ,  $X_3$  nicht ganzzahlig  $\Rightarrow$  offen. MUB:  
 $654,2 > 638,6$ , weiter mit  $P_2$ .

# Tiefensuche + MUB



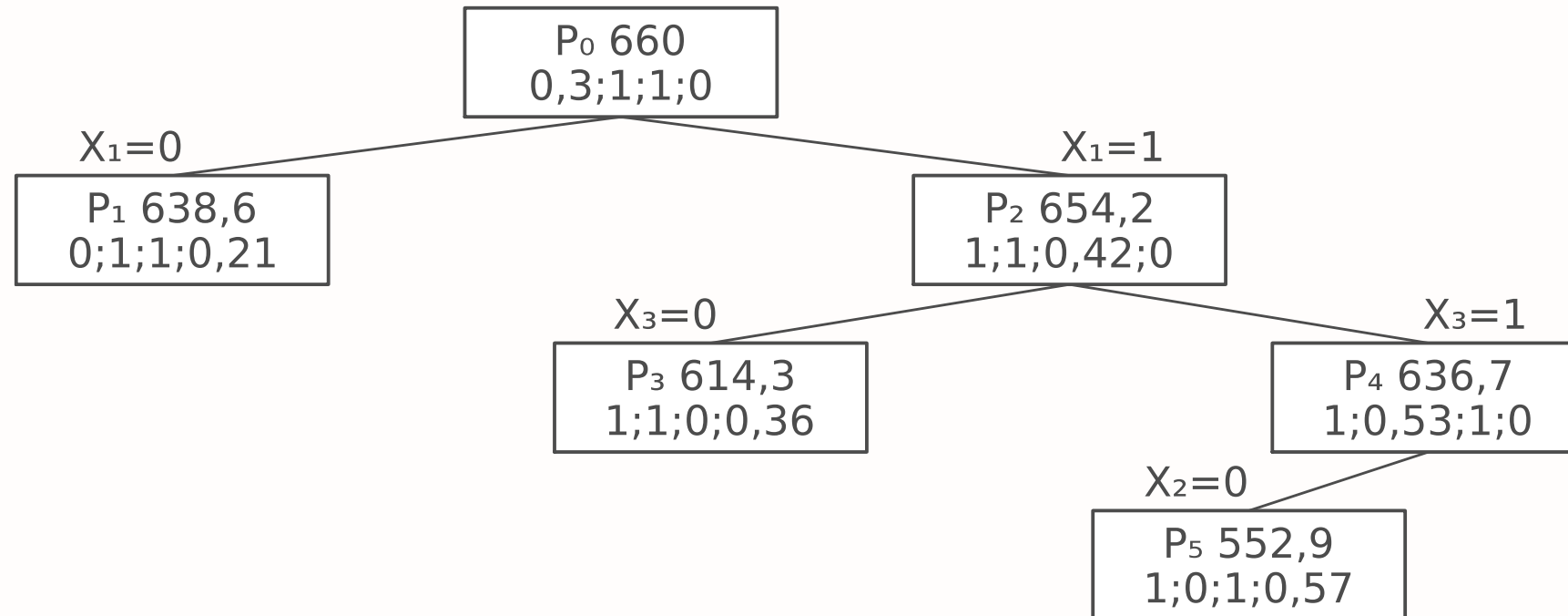
$P_3$  mit  $X_3 = 0$ :  $614,3 > \underline{F} = 0$ ,  $X_4$  nicht ganzzahlig  $\Rightarrow$  offen.

# Tiefensuche + MUB



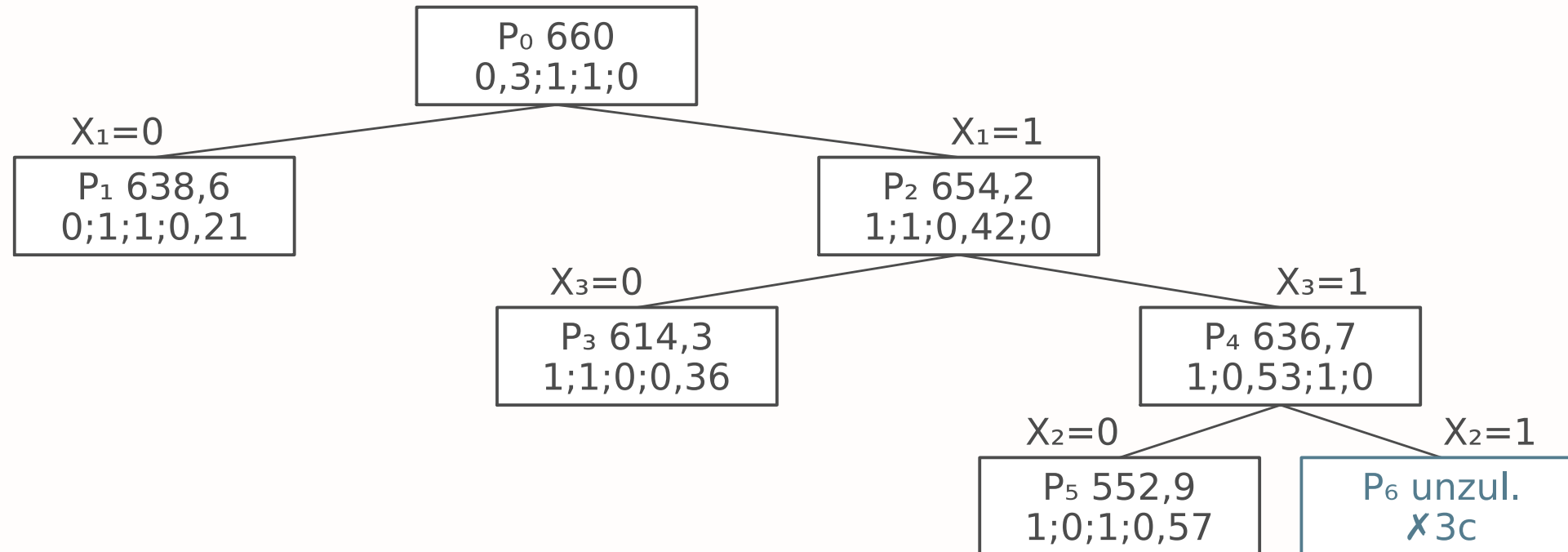
$P_4$  mit  $X_3 = 1$ :  $636,7 > \underline{F} = 0$ ,  $X_2$  nicht ganzzahlig  $\Rightarrow$  offen. MUB:  
 $636,7 > 614,3$ , weiter mit  $P_4$ .

# Tiefensuche + MUB



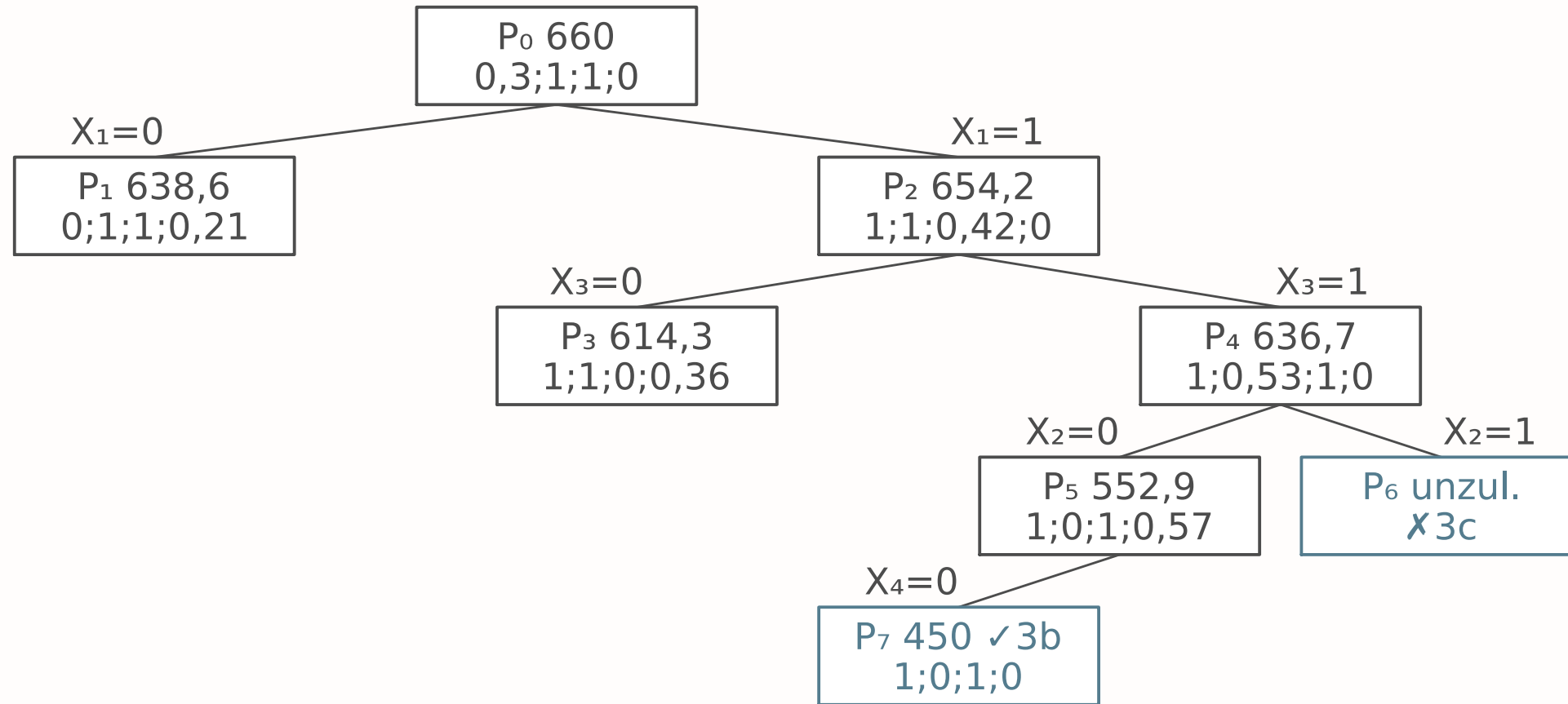
$P_5$  mit  $X_2 = 0$ :  $552,9 > \underline{F} = 0$ ,  $X_4$  nicht ganzzahlig  $\Rightarrow$  offen.

# Tiefensuche + MUB



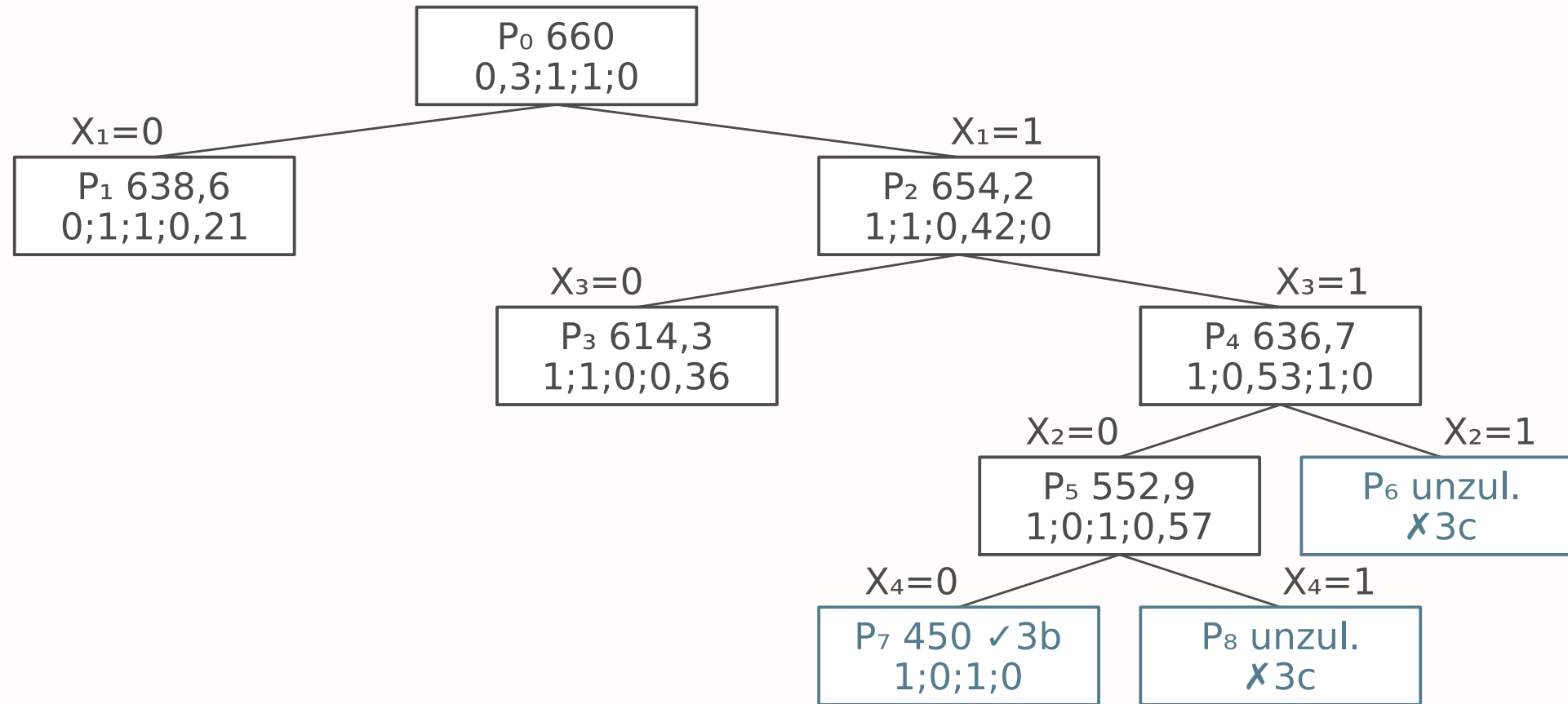
$P_6$  mit  $X_2 = 1$ : unzulässig (3c), Gewicht  $37 > 30$ . Weiter mit  $P_5$ , verzweige nach  $X_4$ .

# Tiefensuche + MUB



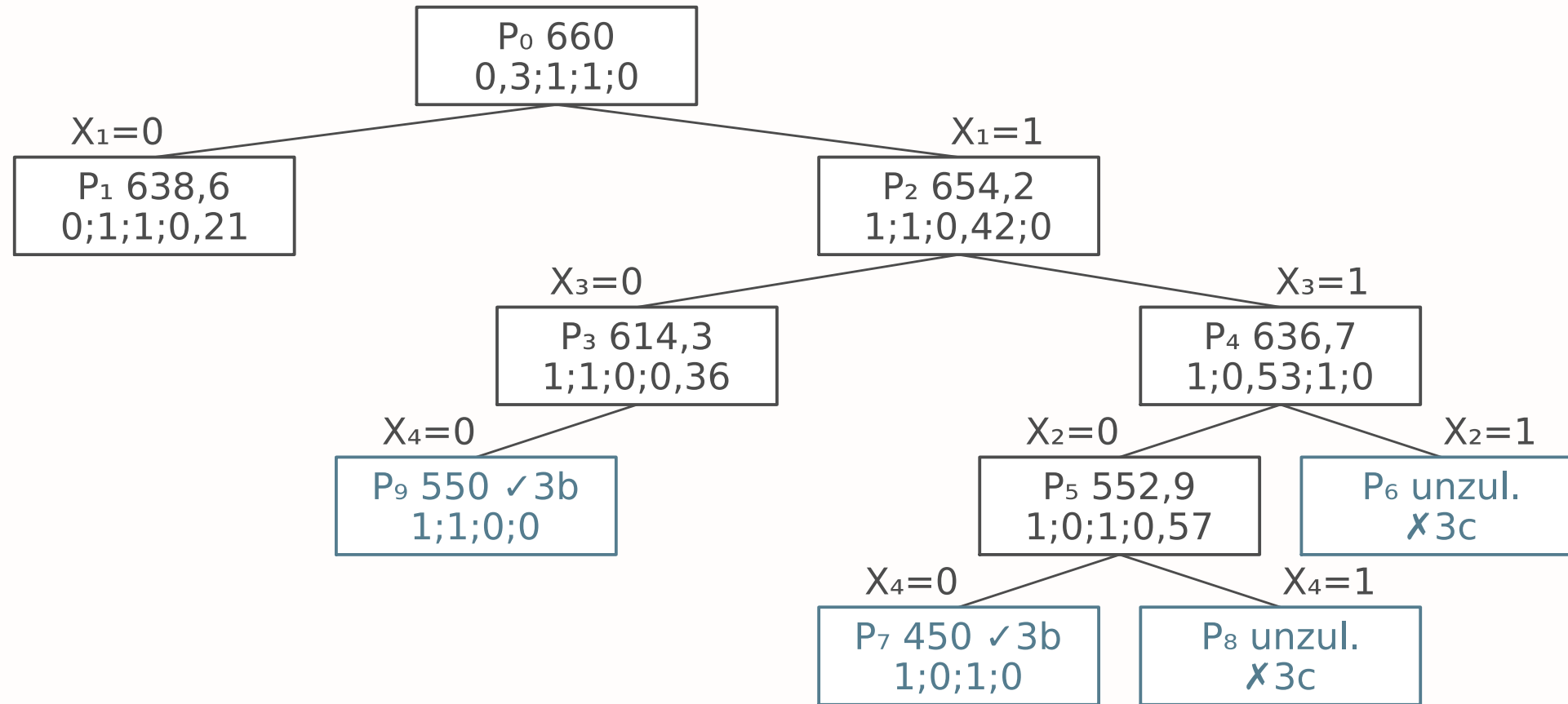
$P_7$  mit  $X_4 = 0$ :  $(1,0,1,0)$  ganzzahlig,  $450 > \underline{F} = 0$  (3b)  $\Rightarrow \underline{F} = 450$ .  $P_1$ ,  $P_3$  bleiben offen.

# Tiefensuche + MUB



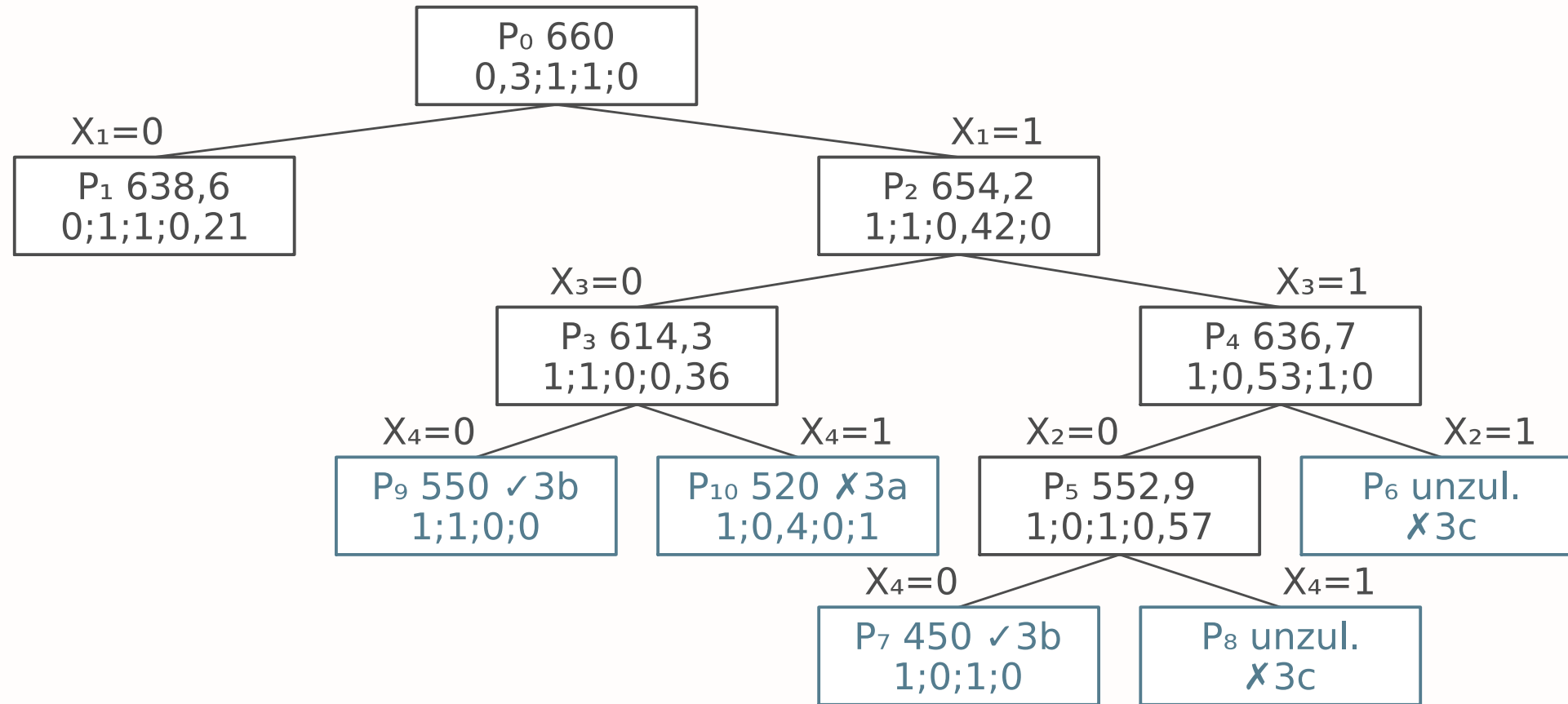
$P_8$  mit  $X_4 = 1$ : unzulässig (3c), Gewicht  $36 > 30$ . **Backtracking** zum tiefsten offenen Knoten,  $P_3$ .

# Tiefensuche + MUB



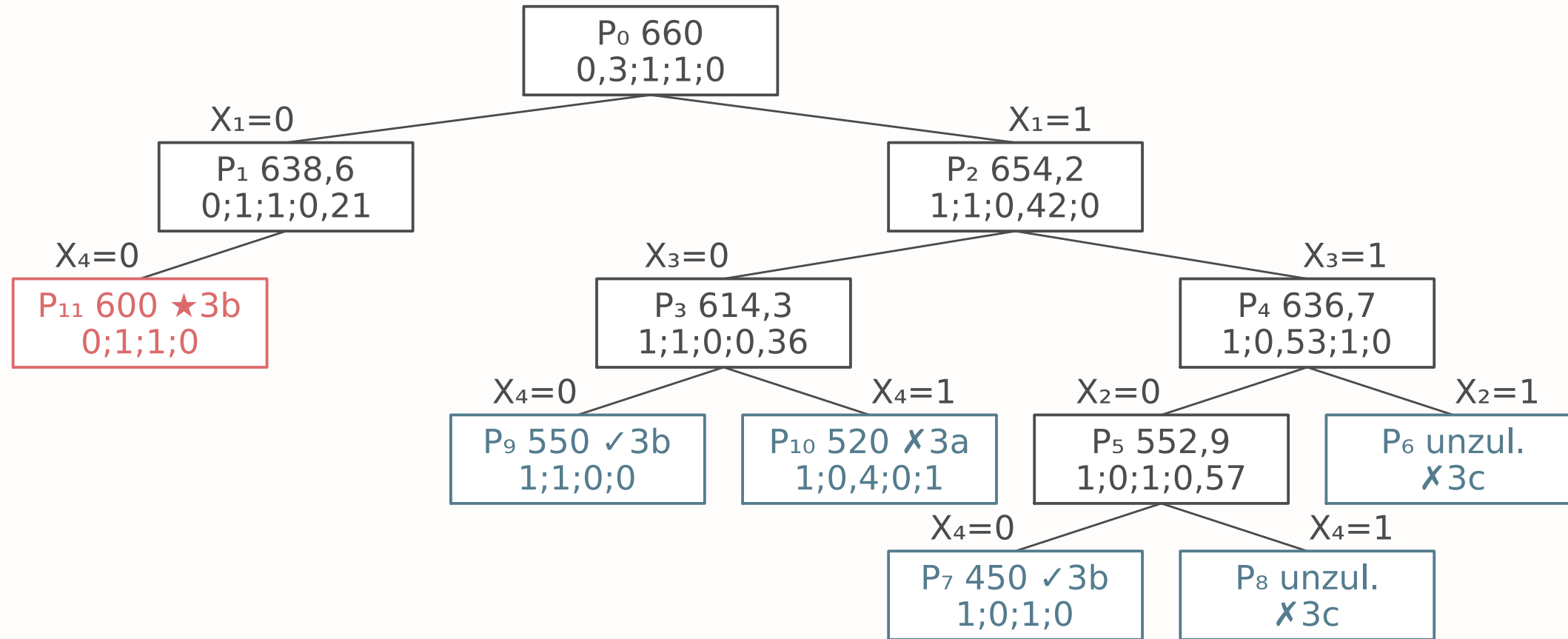
$P_3$ , Ast  $X_4 = 0$ :  $P_9$  mit  $(1, 1, 0, 0)$  ganzzahlig,  $550 > \underline{F} = 450$  (3b)  
 $\Rightarrow \underline{F} = 550$ .

# Tiefensuche + MUB



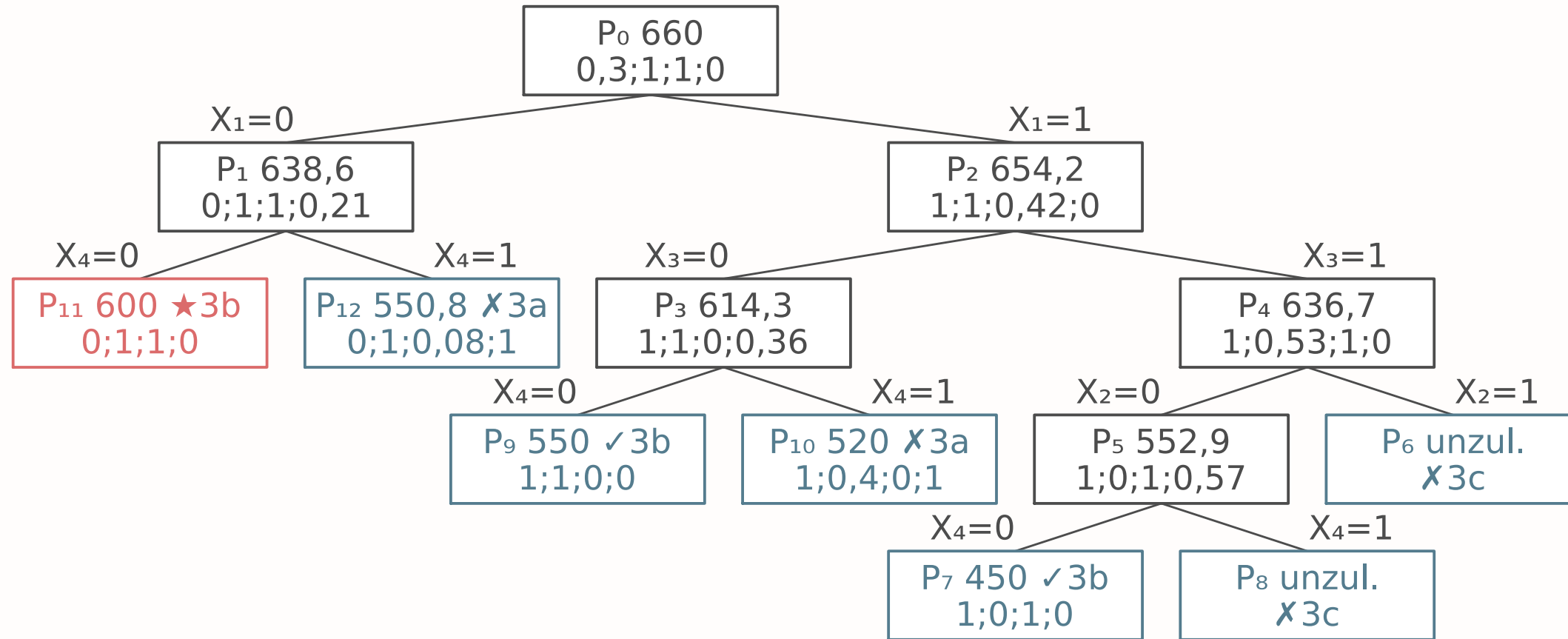
$P_{10}$  mit  $X_4 = 1$ :  $520 \leq \underline{F} = 550 \Rightarrow$  ausgelotet (3a). **Backtracking** zu  $P_1$ :  $638,6 > 550$ , offen.

# Tiefensuche + MUB



$P_1$ , Ast  $X_4 = 0$ :  $P_{11}$  mit  $(0, 1, 1, 0)$  ganzzahlig,  $600 > \underline{F} = 550$  (3b)  
 $\Rightarrow \underline{F} = 600$ .

# Tiefensuche + MUB



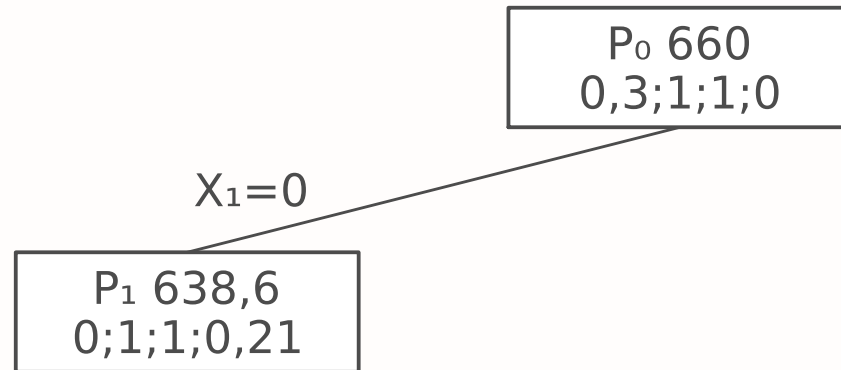
$P_{12}$  mit  $X_4 = 1$ :  $550,8 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a). Nichts mehr offen,  $P_{11}$  ist optimal.

# Breitensuche

|                        |
|------------------------|
| $P_0$ 660<br>0,3;1;1;0 |
|------------------------|

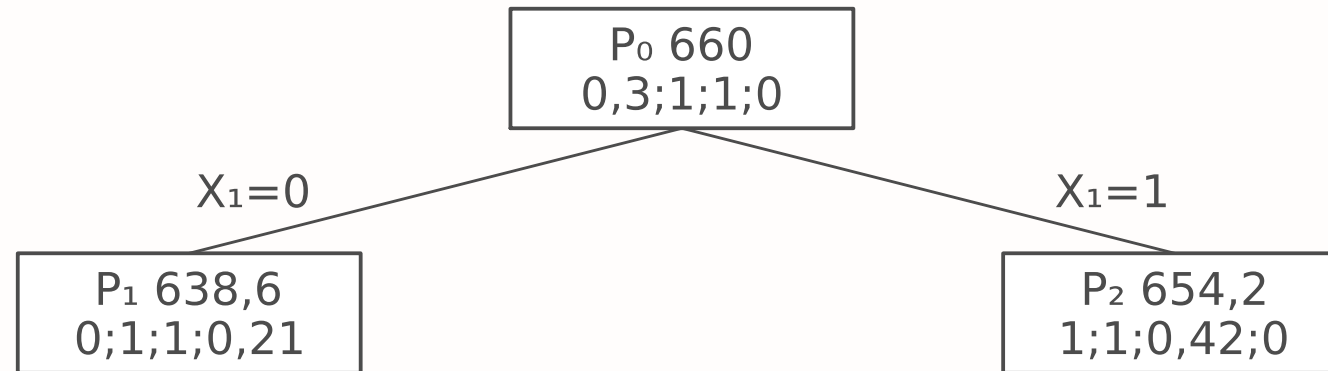
$P_0 : 660 > \underline{F} = 0, X_1$  nicht ganzzahlig  $\Rightarrow$  offen, verzweige nach  $X_1$   
und löse **beide** Äste.

# Breitensuche



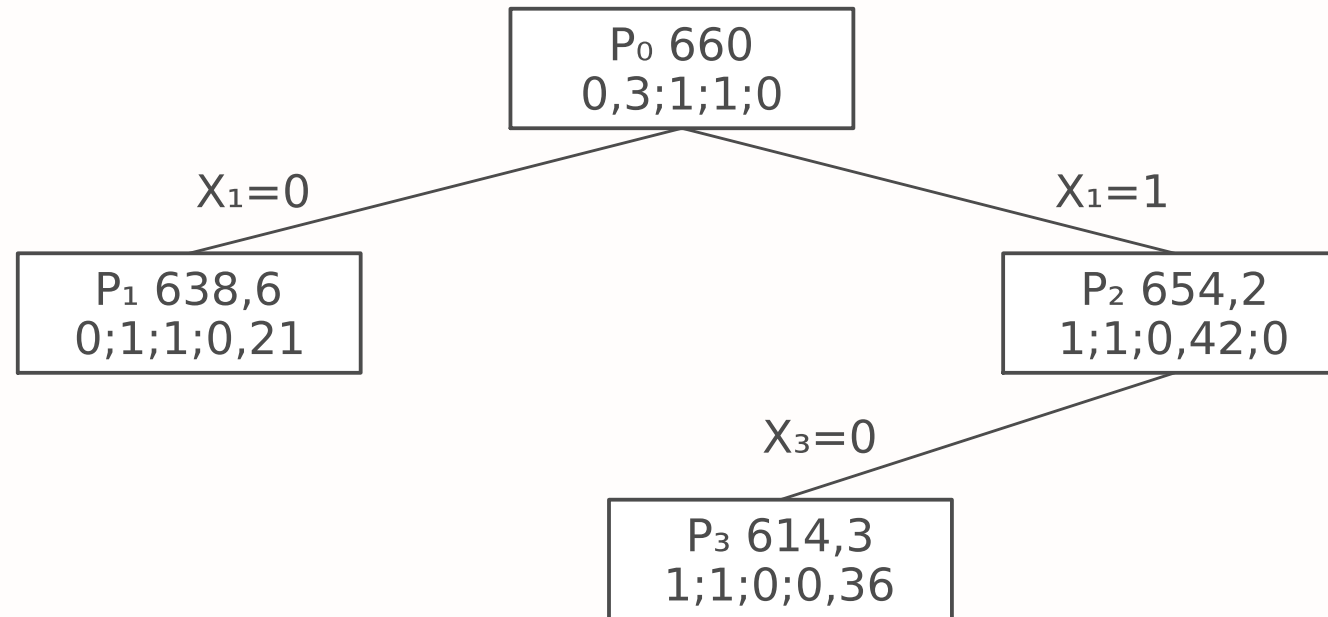
$P_1$  mit  $X_1 = 0$ :  $638,6 > \underline{F} = 0$ ,  $X_4$  nicht ganzzahlig  $\Rightarrow$  offen.

# Breitensuche



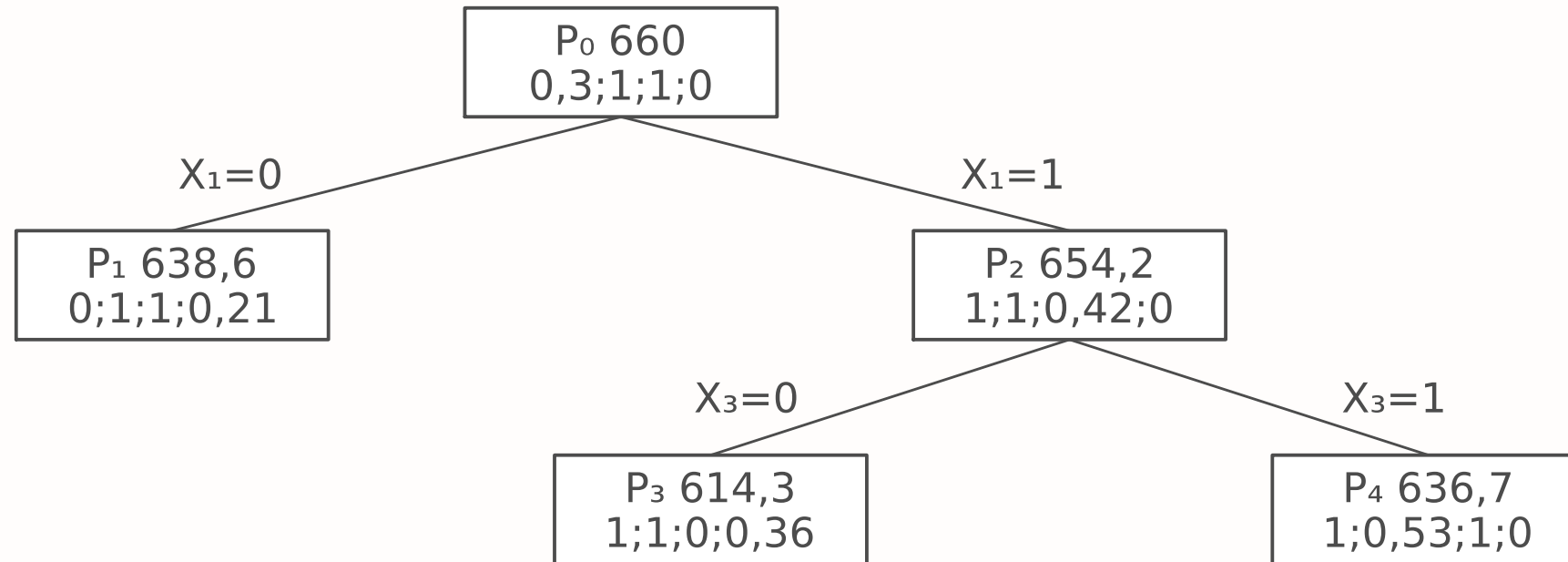
$P_2$  mit  $X_1 = 1$ :  $654,2 > \underline{F} = 0$ ,  $X_3$  nicht ganzzahlig  $\Rightarrow$  offen. Größte Schranke aller offenen:  $P_2$ .

# Breitensuche



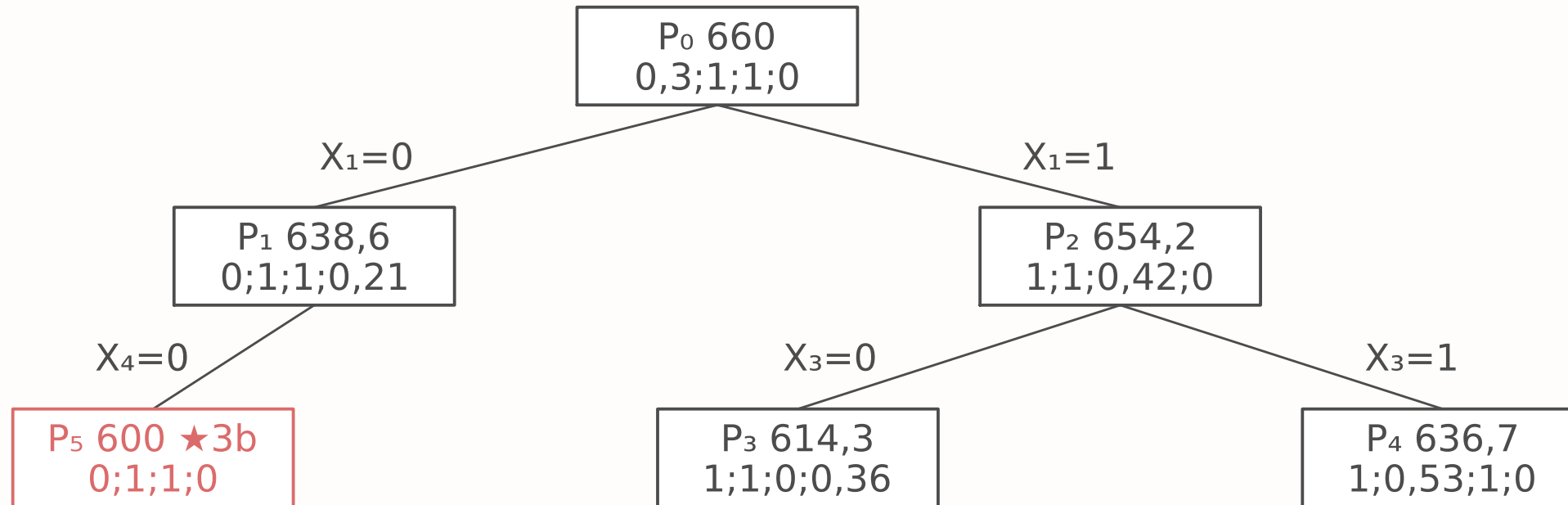
$P_3$  mit  $X_3 = 0$ :  $614,3 > \underline{F} = 0$ ,  $X_4$  nicht ganzzahlig  $\Rightarrow$  offen.

# Breitensuche



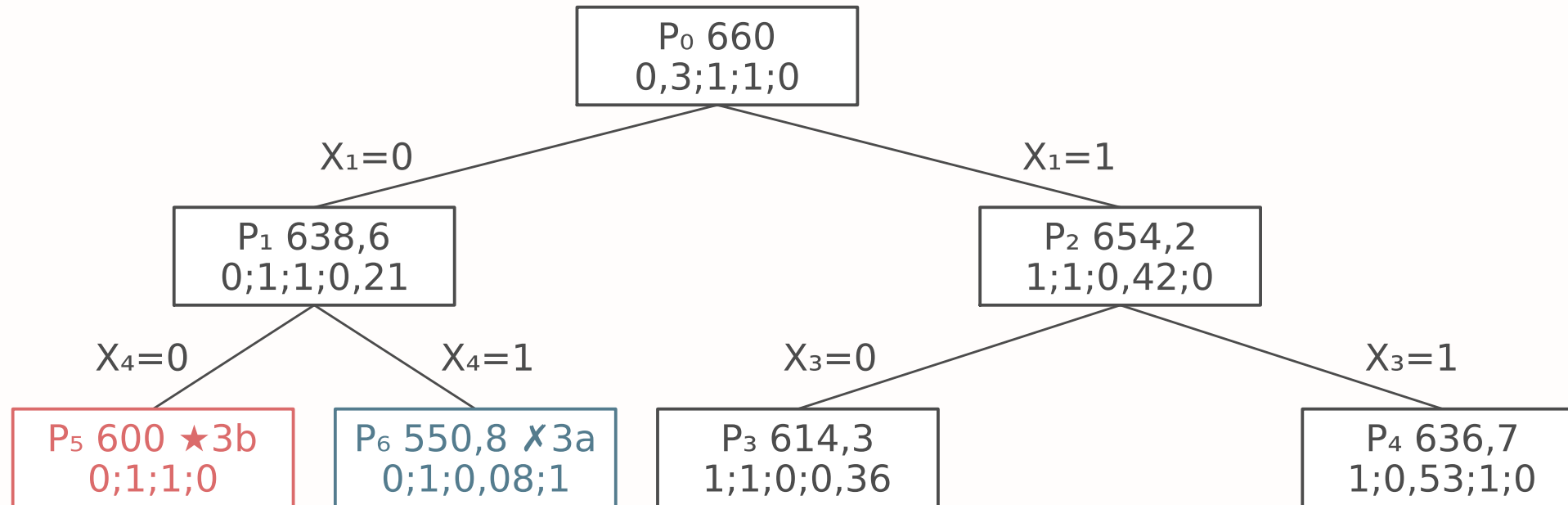
$P_4$  mit  $X_3 = 1$ :  $636,7 > \underline{F} = 0$ ,  $X_2$  nicht ganzzahlig  $\Rightarrow$  offen. Größte Schranke **aller** offenen hat  $P_1$ .

# Breitensuche



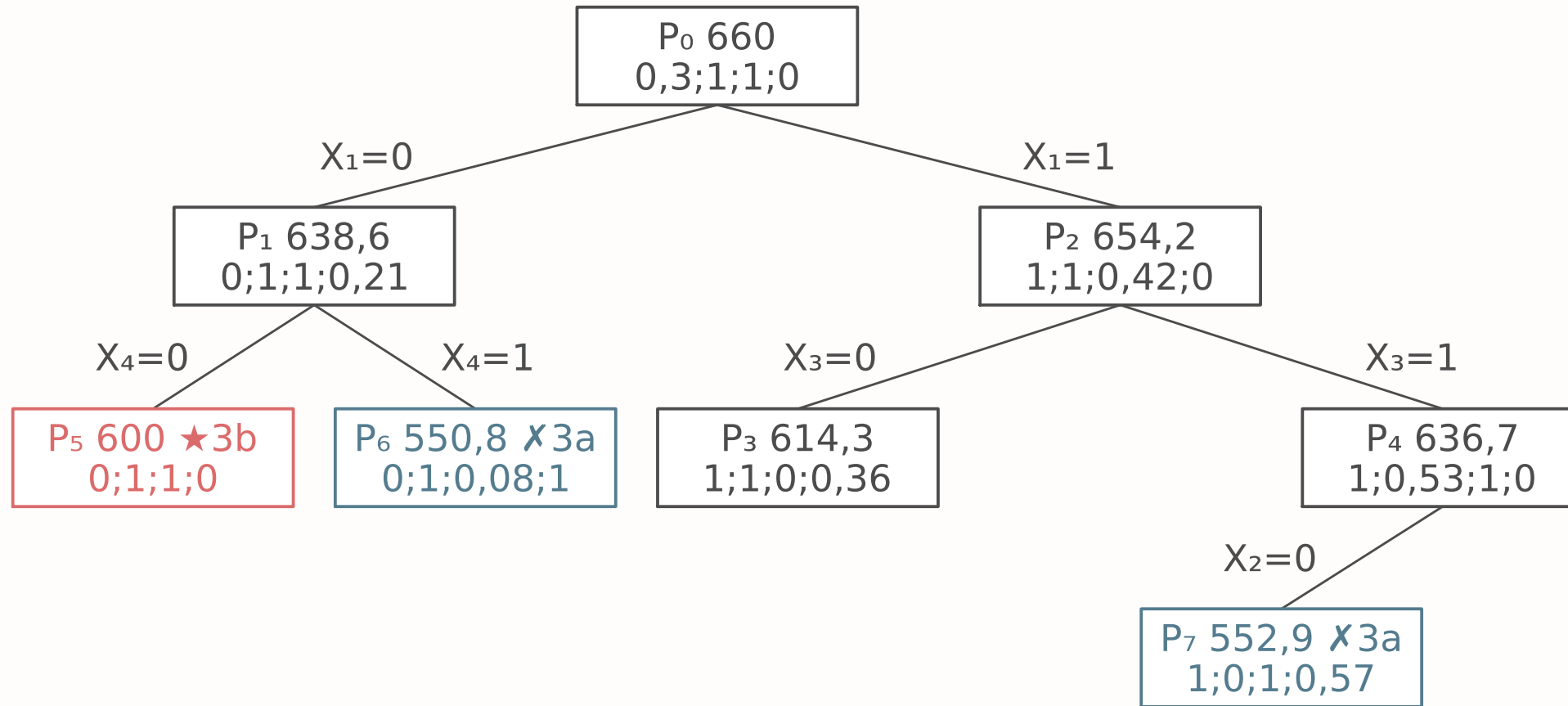
$P_5$  mit  $X_4 = 0$ :  $(0, 1, 1, 0)$  ganzzahlig,  $600 > \underline{F} = 0 \text{ (3b)} \Rightarrow \underline{F} = 600$ .

# Breitensuche



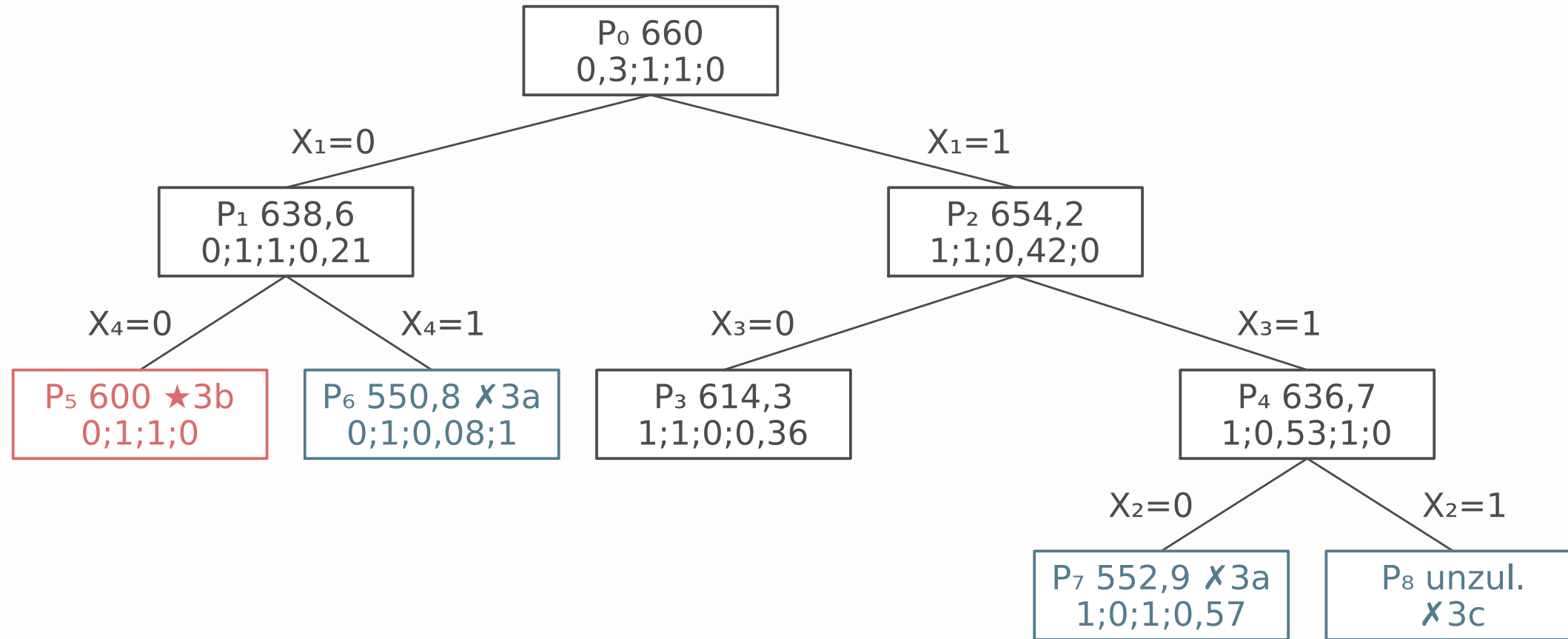
$P_6$  mit  $X_4 = 1$ :  $550,8 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a).  $P_3, P_4$  bleiben offen, weiter mit  $P_4$ .

# Breitensuche



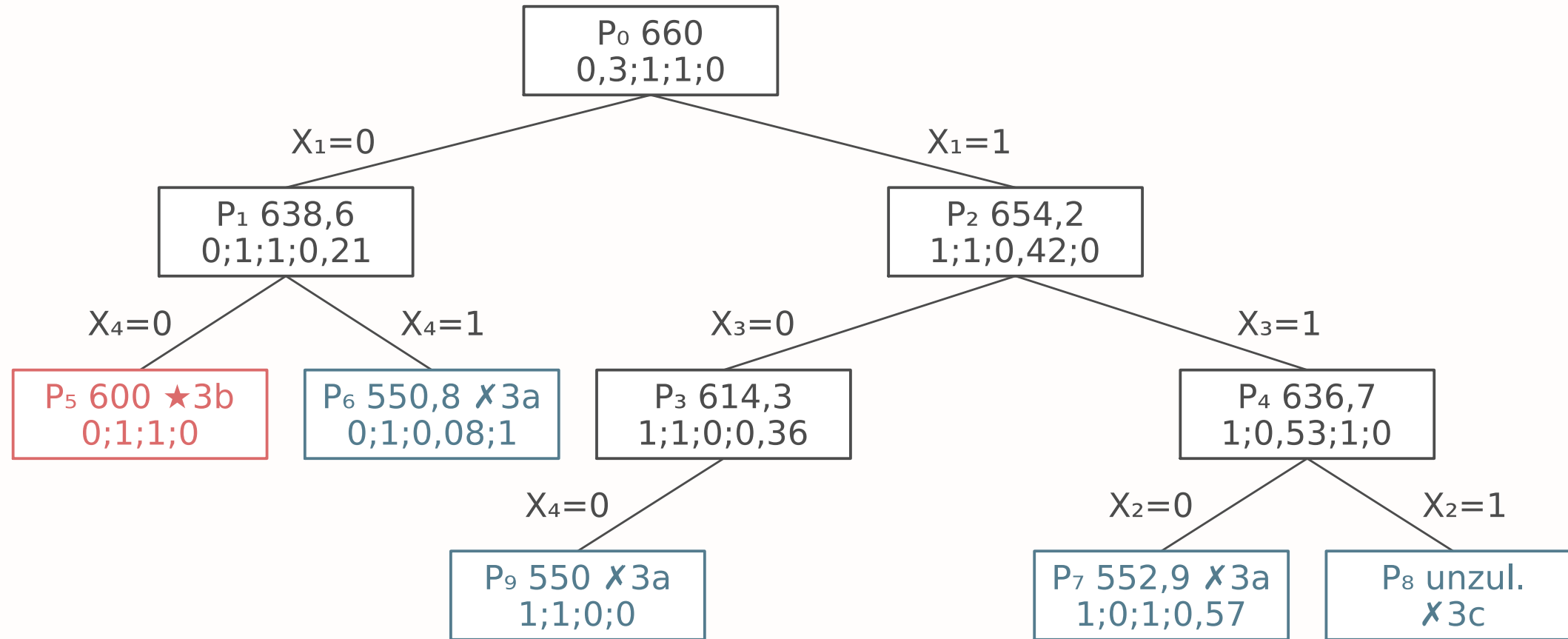
$P_7$  mit  $X_2 = 0$ :  $552,9 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a).

# Breitensuche



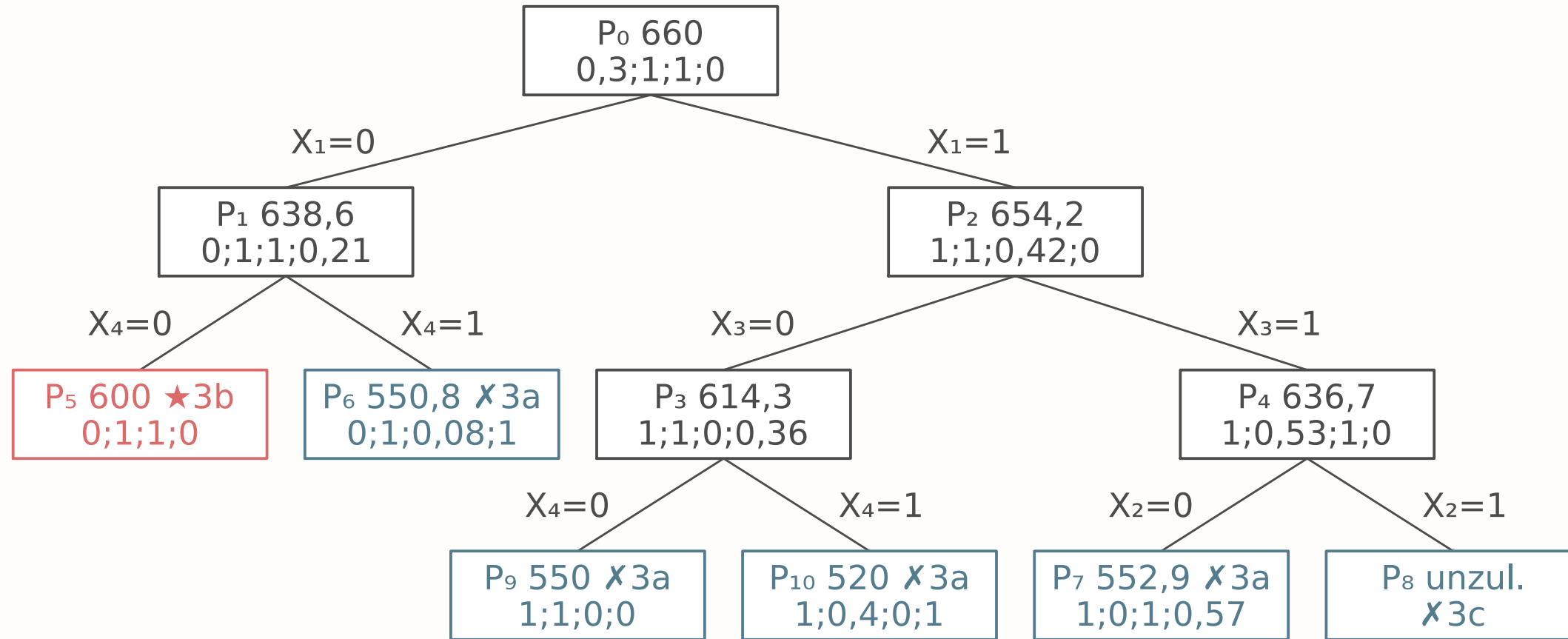
$P_8$  mit  $X_2 = 1$ : unzulässig (3c), Gewicht  $37 > 30$ . Offen bleibt nur  $P_3$ , verzweige nach  $X_4$ .

# Breitensuche



$P_9$  mit  $X_4 = 0$ :  $(1, 1, 0, 0)$  ganzzahlig, aber  $550 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a).

# Breitensuche



$P_{10}$  mit  $X_4 = 1$ :  $520 \leq \underline{F} = 600 \Rightarrow$  ausgelotet (3a). Nichts mehr offen,  $P_5$  ist optimal.

# Optimum

! Beste Auftragsauswahl

Aufträge **2** und **3**:  $X^* = (0, 1, 1, 0)$ ,  $F^* = 600$ .

# Optimum

! Beste Auftragsauswahl

Aufträge 2 und 3:  $X^* = (0, 1, 1, 0)$ ,  $F^* = 600$ .

Von  $2^4 = 16$  möglichen Auftragskombinationen hat Branch & Bound nur einen Teil **explizit** geprüft. Der Rest war implizit ausgelotet: **implizite vollständige Enumeration**.

# Optimum

## ! Beste Auftragsauswahl

Aufträge **2 und 3**:  $X^* = (0, 1, 1, 0)$ ,  $F^* = 600$ .

Von  $2^4 = 16$  möglichen Auftragskombinationen hat Branch & Bound nur einen Teil **explizit** geprüft. Der Rest war implizit ausgelotet: **implizite vollständige Enumeration**.

## 💡 Gleich ausprobieren

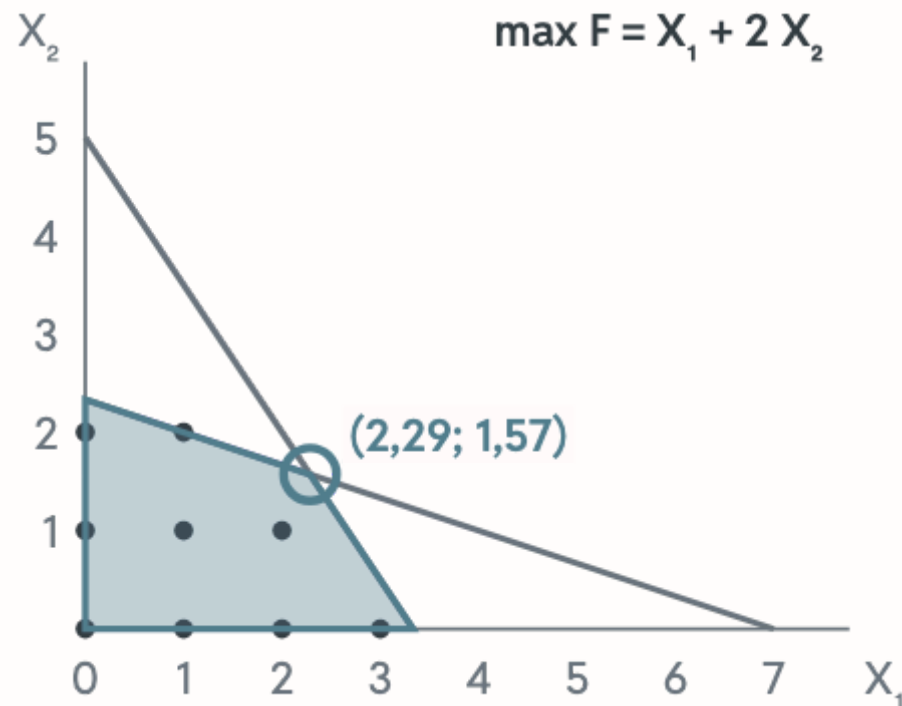
Auf der nächsten Folie führen Sie Branch & Bound selbst aus, mit allen drei Suchstrategien von vorhin.

# Sie sind dran

Führen Sie Branch & Bound mit der angegebenen Suchstrategie aus und verzweigen Sie nach der Variablen, die einer ganzen Zahl am nächsten liegt. Neue Teilprobleme stufen Sie von links nach rechts ein:  
3a  $\bar{F} \leq \underline{F}$ , 3b ganzzahlig und besser, 3c unzulässig, sonst offen.

Tiefensuche + MUB ·  $\underline{F} = 0$  aus  $X = (0; 0)$

**$P_0$  5,43**  
offen



nach  $X_1$  nach  $X_2$

$P_0$ : LP-Optimum  $(2,29; 1,57)$ ,  $\bar{F} = 5,43 > \underline{F} = 0$ , offen.



Eigener Plan

0 Abweichungen

Noch offen:  $P_0$

Optimieren

Reset

Neu

# Kontrolle mit Julia

HiGHS geht genauso vor, nur mit geschickteren Regeln für Variable und Reihenfolge:

```
1 using JuMP, HiGHS
2 e = [200,350,250,180]; g = [10,15,12,14]; p = [12,30,16,10]
3 G, P, J = 30, 50, 1:4
4
5 modell = Model(HiGHS.Optimizer); set_silent(modell)
6 @variable(modell, X[J], Bin)
7 @objective(modell, Max, sum(e[j]*X[j] for j in J))
8 @constraint(modell, sum(g[j]*X[j] for j in J) <= G)
9 @constraint(modell, sum(p[j]*X[j] for j in J) <= P)
10
11 optimize!(modell)
12 println("F* = ", round{Int, objective_value(modell)},
13         ", angenommen: ", [j for j in J if value(X[j]) > 0.5])
```

# Kontrolle mit Julia

HiGHS geht genauso vor, nur mit geschickteren Regeln für Variable und Reihenfolge:

```
1 using JuMP, HiGHS
2 e = [200,350,250,180]; g = [10,15,12,14]; p = [12,30,16,10]
3 G, P, J = 30, 50, 1:4
4
5 modell = Model(HiGHS.Optimizer); set_silent(modell)
6 @variable(modell, X[J], Bin)
7 @objective(modell, Max, sum(e[j]*X[j] for j in J))
8 @constraint(modell, sum(g[j]*X[j] for j in J) <= G)
9 @constraint(modell, sum(p[j]*X[j] for j in J) <= P)
10
11 optimize!(modell)
12 println("F* = ", round{Int, objective_value(modell)},
13         ", angenommen: ", [j for j in J if value(X[j]) > 0.5])
```

F\* = 600, angenommen: [2, 3]

# Zusammenfassung

# Das Wichtigste auf einen Blick

# Das Wichtigste auf einen Blick

- Die **LP-Relaxation** lässt die Ganzzahligkeit weg und liefert eine obere Schranke.

# Das Wichtigste auf einen Blick

- Die **LP-Relaxation** lässt die Ganzzahligkeit weg und liefert eine obere Schranke.
- **Branch & Bound** verzweigt nach nicht ganzzahligen Variablen und **lotet** Teilprobleme über Schranken aus.

# Das Wichtigste auf einen Blick

- Die **LP-Relaxation** lässt die Ganzzahligkeit weg und liefert eine obere Schranke.
- **Branch & Bound** verzweigt nach nicht ganzzahligen Variablen und **lotet** Teilprobleme über Schranken aus.
- Ein Teilproblem  $P_i$  ist erledigt, wenn es ganzzahlig ist **oder** seine obere Schranke  $\bar{F}_i$  die untere Schranke  $\underline{F}$  nicht übertrifft.

# Das Wichtigste auf einen Blick

- Die **LP-Relaxation** lässt die Ganzzahligkeit weg und liefert eine obere Schranke.
- **Branch & Bound** verzweigt nach nicht ganzzahligen Variablen und **lotet** Teilprobleme über Schranken aus.
- Ein Teilproblem  $P_i$  ist erledigt, wenn es ganzzahlig ist **oder** seine obere Schranke  $\bar{F}_i$  die untere Schranke  $\underline{F}$  nicht übertrifft.
- **Runden** der LP-Lösung ist keine verlässliche Methode.

# 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>.