



## 3.1 Abbildung logischer Ausdrücke

## 3.2 Entscheidungsabhängige Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von Zeitperioden

Disjunktive Nebenbedingungen

## 3.3 OPL: Kompakte Schreibweise

## 3.4 Stückweise definierte Funktionen

Treppenfunktionen

Stückweise lineare Funktionen

OPL: Der `piecewise`-Befehl

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

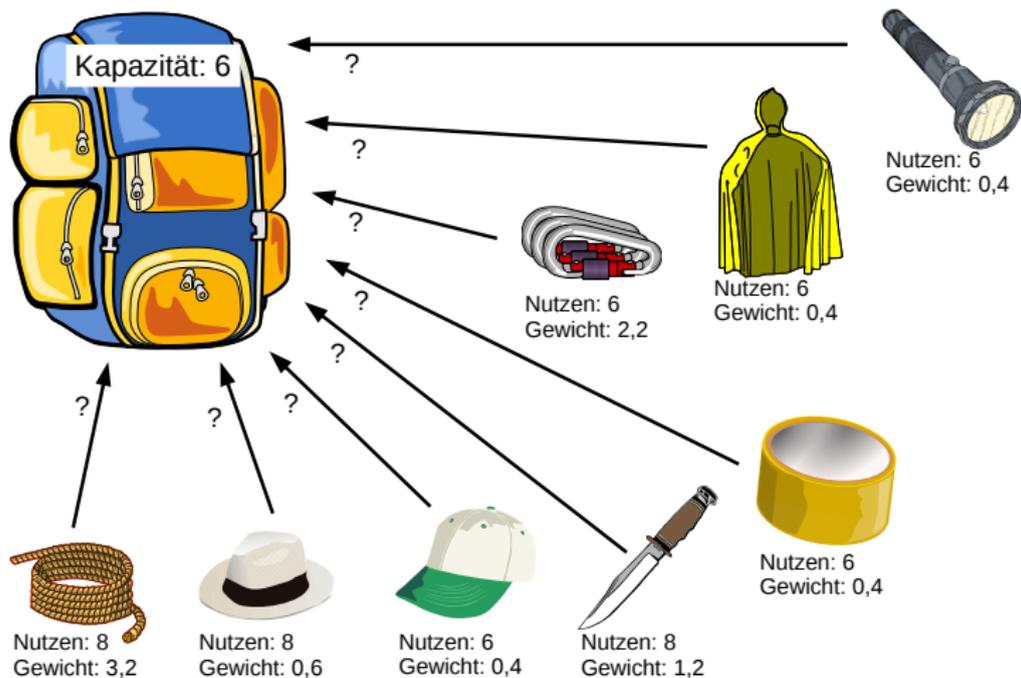
Treppenfunktionen

Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# 3.1 Abbildung logischer Ausdrücke

# Beispiel: Adventure Inc.



## 3 Techniken der binären Modellierung

CC-BY-SA  
A. Popp

### 3.1 Abbildung logischer Ausdrücke

### 3.2 Entscheidungsabhängige Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von Zeitperioden  
Disjunktive Nebenbedingungen

### 3.3 OPL: Kompakte Schreibweise

### 3.4 Stückweise definierte Funktionen

Treppenfunktionen  
Stückweise lineare Funktionen  
OPL: Der piecewise-Befehl



- ¬ Die logische **Negation**
- ∧ Das logische **Und**
- ∨ Das logische **Oder**
- ⊔ Das logische **exklusiv Oder** („Entweder-Oder“)
- ⇒ Die logische **Implikation**
- ⇔ Die logische **Äquivalenz**

## Wahrheitstafel in numerischer Beschreibung

A	B	¬A	¬B	A ∧ B	A ∨ B	A ⊔ B	A ⇒ B	A ⇔ B
1	1	0	0	1	1	0	1	1
1	0	0	1	0	1	1	0	0
0	1	1	0	0	1	1	1	0
0	0	1	1	0	0	0	1	1

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen

Stückweise lineare  
Funktionen

OPL: Der piecewise-Befehl

# Logische Verknüpfungen in binären Optimierungsmodellen

Beispiel: Seien  $x_1$  und  $x_2$  binäre Entscheidungsvariablen eines Rucksackproblems, die angeben, ob die Gegenstände  $l_1$  und  $l_2$  eingepackt werden.

$\neg l_1$ : Man möchte wissen ob  $l_1$  nicht eingepackt ist.

▶  $1 - x_1$

$l_1 \wedge l_2$ : Sowohl  $l_1$  als auch  $l_2$  müssen eingepackt werden.

▶  $x_1 + x_2 = 2$

$l_1 \vee l_2$ : Mindestens einer der Gegenstände muss eingepackt werden.

▶  $x_1 + x_2 \geq 1$

$\neg(l_1 \wedge l_2)$ : Höchstens einer der Gegenstände darf eingepackt werden.

▶  $x_1 + x_2 \leq 1$

## 3.1 Abbildung logischer Ausdrücke

## 3.2 Entscheidungsabhängige Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von Zeitperioden  
Disjunktive Nebenbedingungen

## 3.3 OPL: Kompakte Schreibweise

## 3.4 Stückweise definierte Funktionen

Treppenfunktionen  
Stückweise lineare Funktionen  
OPL: Der `piecewise`-Befehl

# Logische Verknüpfungen in binären Optimierungsmodellen

Beispiel: Seien  $x_1$  und  $x_2$  binäre Entscheidungsvariablen eines Rucksackproblems, die angeben, ob die Gegenstände  $I_1$  und  $I_2$  eingepackt werden.

$\neg(I_1 \vee I_2)$ : Keiner der Gegenstände darf eingepackt werden.

▶  $x_1 + x_2 = 0$

$I_1 \underline{\vee} I_2$ : Genau einer der Gegenstände muss eingepackt werden.

▶  $x_1 + x_2 = 1$

$I_1 \Rightarrow I_2$ : Wenn  $I_1$  eingepackt ist, muss auch  $I_2$  eingepackt werden.

▶  $x_1 \leq x_2$

$I_1 \Leftrightarrow I_2$ : Die Entscheidung für beide Gegenstände ist identisch.

▶  $x_1 = x_2$

3.1 Abbildung logischer Ausdrücke

3.2 Entscheidungsabhängige Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von Zeitperioden

Disjunktive Nebenbedingungen

3.3 OPL: Kompakte Schreibweise

3.4 Stückweise definierte Funktionen

Treppenfunktionen

Stückweise lineare Funktionen

OPL: Der piecewise-Befehl

## 3.2 Entscheidungsabhängige Nebenbedingungen

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

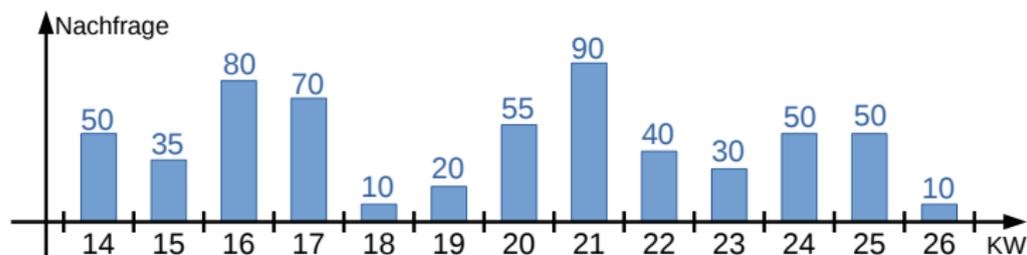
3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen

Stückweise lineare  
Funktionen

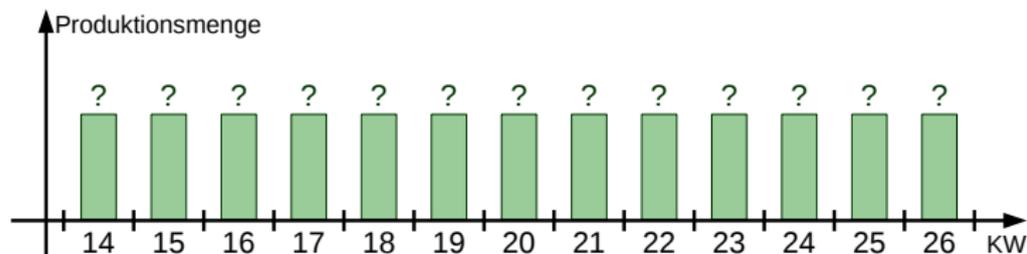
OPL: Der `piecewise`-Befehl

# Beispiel: Lewig Wakuxi



Rüstkosten pro Periode: 25.000€

Lagerkosten von einer Periode auf die nächste pro Einheit: 100€



3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen  
Stückweise lineare  
Funktionen  
OPL: Der piecewise-Befehl



Sei  $\bar{x}$  der Vektor der Entscheidungsvariablen und  $f$  eine lineare Funktion. Die Nebenbedingung

$$f(\bar{x}) \leq b \quad \text{bzw.} \quad f(\bar{x}) \geq b$$

soll nur zwingend sein, wenn eine Entscheidung getroffen wurde, die dadurch repräsentiert wird, dass die Binärvariable  $y$  den Wert 0 annimmt.

## Entscheidungsabhängige Nebenbedingungen

Sei  $M$  eine ausreichend große Zahl.

$$f(\bar{x}) \leq b \quad \rightarrow \quad f(\bar{x}) \leq b + M \cdot y$$

$$f(\bar{x}) \geq b \quad \rightarrow \quad f(\bar{x}) \geq b - M \cdot y$$

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

### Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen

Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# Problem bei der Implementierung von Zeitperioden

## Nebenbedingung aus Beispiel „Lewig Wakuxi“

$$i_t = i_{t-1} + x_t - d_t \quad \forall t \in T \quad (I)$$

## Implementierungsversuch 1

```
{string} T = {"KW14", "KW15", "KW16", "KW17"};  
dvar float+ i[T];  
forall(t in T) i[t] == i[t-1] + x[t] - d[t];
```

❌ Operator für string - int nicht verfügbar.

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

**OPL: Modellieren von  
Zeitperioden**

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen

Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# Problem bei der Implementierung von Zeitperioden

## Nebenbedingung aus Beispiel „Lewig Wakuxi“

$$i_t = i_{t-1} + x_t - d_t \quad \forall t \in T \quad (I)$$

## Implementierungsversuch 2

```
{int} T = {14, 15, 16, 17};  
dvar float+ i[T];  
forall(t in T) i[t] == i[t-1] + x[t] - d[t];
```

❌ Der Index für den Array "i" liegt außerhalb des gültigen Bereichs: 13.

3.1 Abbildung logischer Ausdrücke

3.2 Entscheidungsabhängige Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von Zeitperioden

Disjunktive Nebenbedingungen

3.3 OPL: Kompakte Schreibweise

3.4 Stückweise definierte Funktionen

Treppenfunktionen

Stückweise lineare Funktionen

OPL: Der `piecewise`-Befehl

# Problem bei der Implementierung von Zeitperioden

## Nebenbedingung aus Beispiel „Lewig Wakuxi“

$$i_t = i_{t-1} + x_t - d_t \quad \forall t \in T \quad (I)$$

## Implementierungsversuch 3

```
{int} T = {14, 15, 16, 17};  
{int} T0 = {13, 14, 15, 16, 17};  
dvar float+ i[T0];  
forall(t in T) i[t] == i[t-1] + x[t] - d[t];
```

3.1 Abbildung logischer Ausdrücke

3.2 Entscheidungsabhängige Nebenbedingungen

Die Big-M-Methode

**OPL: Modellieren von Zeitperioden**

Disjunktive Nebenbedingungen

3.3 OPL: Kompakte Schreibweise

3.4 Stückweise definierte Funktionen

Treppenfunktionen

Stückweise lineare Funktionen

OPL: Der `piecewise`-Befehl

# Problem bei der Implementierung von Zeitperioden

## Nebenbedingung aus Beispiel „Lewig Wakuxi“

$$i_t = i_{t-1} + x_t - d_t \quad \forall t \in T \quad (I)$$

## Implementierungsversuch 4

```
int Tmin = 14;  
int Tmax = 17;  
range T = Tmin..Tmax;  
dvar float+ i[Tmin-1..Tmax];  
forall(t in T) i[t] == i[t-1] + x[t] - d[t];
```

3.1 Abbildung logischer Ausdrücke

3.2 Entscheidungsabhängige Nebenbedingungen

Die Big-M-Methode

**OPL: Modellieren von Zeitperioden**

Disjunktive Nebenbedingungen

3.3 OPL: Kompakte Schreibweise

3.4 Stückweise definierte Funktionen

Treppenfunktionen

Stückweise lineare Funktionen

OPL: Der `piecewise`-Befehl



# Disjunktive Nebenbedingungen II

Ein Modell habe die folgende Nebenbedingungen:

$$g(\bar{x}) \leq d$$

Diese muss nur erfüllt werden, wenn gilt:

$$f(\bar{x}) > b$$

## Disjunktive Nebenbedingungen

Sei  $M$  eine ausreichend große Zahl und  $y$  eine binäre Hilfsvariable.

$$\begin{aligned} f(\bar{x}) &\leq b + M \cdot y \\ g(\bar{x}) &\leq d + M \cdot (1 - y) \end{aligned}$$

$\geq$ -Nebenbedingungen analog

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen  
Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

## 3.3 OPL: Kompakte Schreibweise

Zielfunktion des Wagner-Whitin-Problems:

```
// Zielfunktion  
minimize sum(t in T)(s[t]*y[t] + h[t]* i[t]);
```

Zerlegen in Entscheidungsausdrücke:

```
// Entscheidungsausdrücke  
dexpr float setupCost = sum(t in T)(s[t]*y[t]);  
dexpr float inventoryCost = sum(t in T)(h[t]*i[t]);  
  
// Zielfunktion  
minimize setupCost + inventoryCost;
```

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen

Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# Arrays von Entscheidungsausdrücken

Zielfunktion des Wagner-Whitin-Problems:

```
// Zielfunktion  
minimize sum(t in T)(s[t]*y[t] + h[t]* i[t]);
```

Zerlegen in Entscheidungsausdrücke:

```
//Entscheidungsausdrücke  
dexpr float periodCost[t in T]  
= s[t]*y[t] + h[t]*i[t];  
  
// Zielfunktion  
minimize sum (t in T)(periodCost[t]);
```

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen

Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# 3.4 Stückweise definierte Funktionen

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

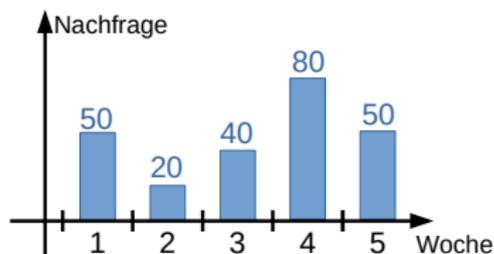
**3.4 Stückweise  
definierte  
Funktionen**

Treppenfunktionen

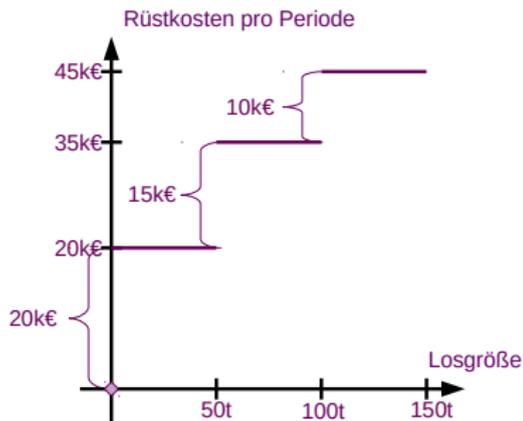
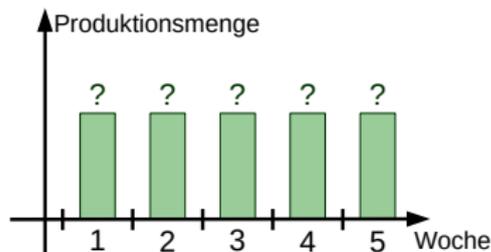
Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# Beispiel: Lewig Xanxi



Rüstkosten  
Lagerkosten: 100€



3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

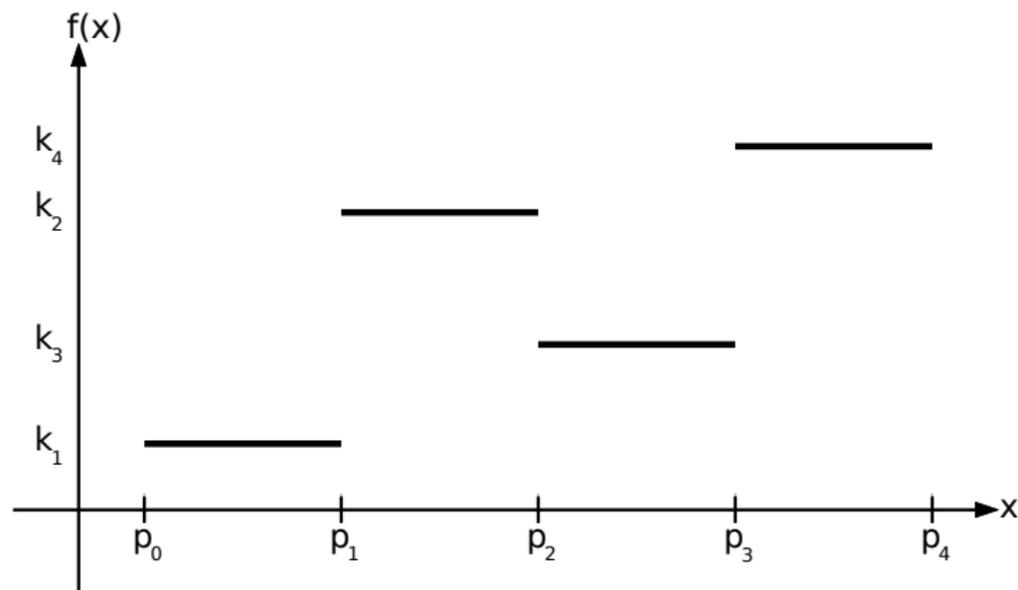
**Treppenfunktionen**

Stückweise lineare  
Funktionen

OPL: Der piecewise-Befehl

# Treppenfunktionen

Sei  $x$  eine kontinuierliche Entscheidungsvariable:



3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden  
Disjunktive  
Nebenbedingungen

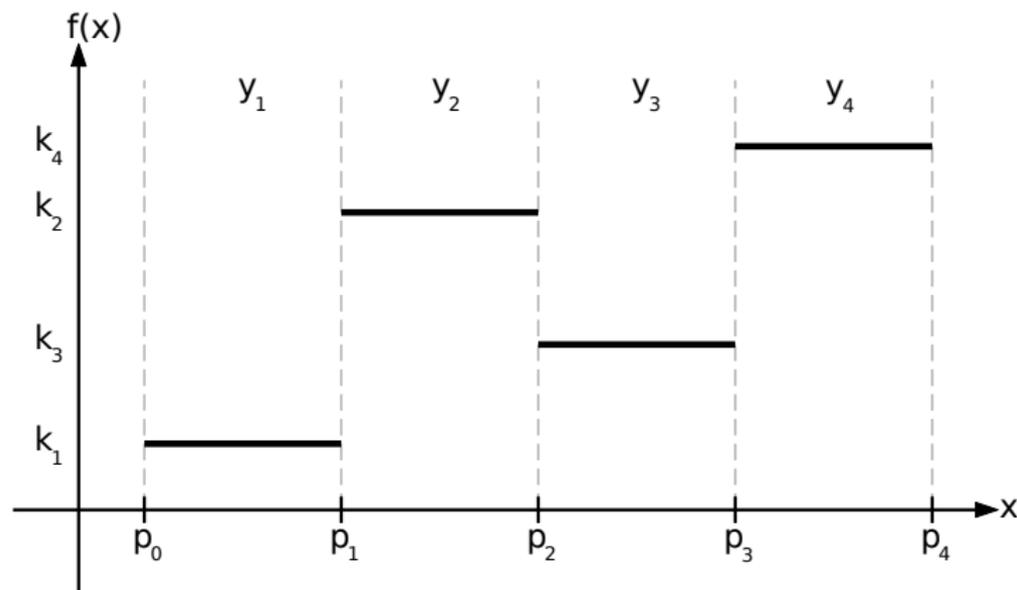
3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

**Treppenfunktionen**  
Stückweise lineare  
Funktionen  
OPL: Der `piecewise`-Befehl

# Treppenfunktionen

Sei  $x$  eine kontinuierliche Entscheidungsvariable:



3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden  
Disjunktive  
Nebenbedingungen

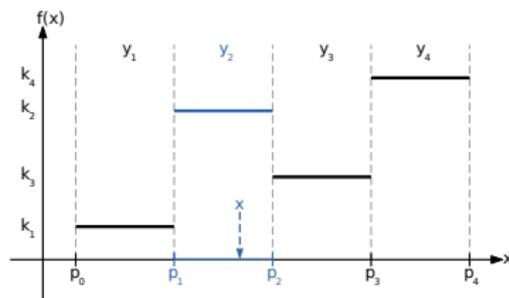
3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

**Treppenfunktionen**  
Stückweise lineare  
Funktionen  
OPL: Der `piecewise`-Befehl



# Entscheidungsvariablen als Konvexkombination der Stützstellen



$$x = \sum_{n=0}^N z_n \cdot p_n$$

$$\sum_{n=0}^N z_n = 1$$

$$0 \leq z_n \leq 1 \quad \forall n \in \{1, \dots, N\}$$

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeiträumen

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

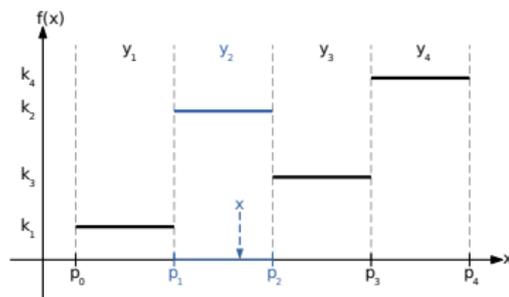
3.4 Stückweise  
definierte  
Funktionen

**Treppenfunktionen**

Stückweise lineare  
Funktionen

OPL: Der piecewise-Befehl

# Auswahl des korrekten Intervalls



$$\sum_{n=1}^N y_n = 1$$

$$z_0 \leq y_1$$

$$z_n \leq y_n + y_{n+1} \quad \forall n \in \{1, \dots, N-1\}$$

$$z_N \leq y_N$$

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode

OPL: Modellieren von  
Zeitperioden

Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

**Treppenfunktionen**

Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# Vollständige Modellierung

$$f(x) = \sum_{n=1}^N y_n \cdot k_n$$

$$x = \sum_{n=0}^N z_n \cdot p_n$$

$$\sum_{n=0}^N z_n = 1$$

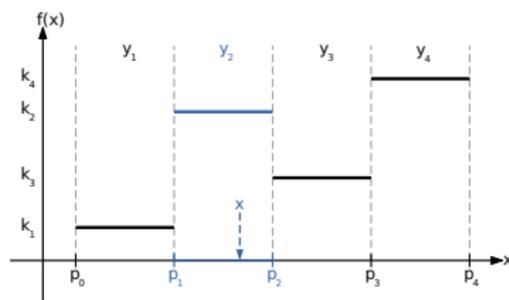
$$0 \leq z_n \leq 1 \quad \forall n \in \{1, \dots, N\}$$

$$\sum_{n=1}^N y_n = 1$$

$$z_0 \leq y_1$$

$$z_n \leq y_n + y_{n+1} \quad \forall n \in \{1, \dots, N-1\}$$

$$z_N \leq y_N$$



3 Techniken der  
binären  
Modellierung

CC-BY-SA  
A. Popp

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

**Treppenfunktionen**

Stückweise lineare  
Funktionen

OPL: Der piecewise-Befehl

# Beispiel: Lewig Tadelbach

Ausgangssituation ist ein Produktionsproblem.

## Kapazitäten

	Maschine A	Maschine B	Maschine C
$l_1$	2,2	1,6	2,8
$l_2$	1,2	1,9	2,3
	72	48	60

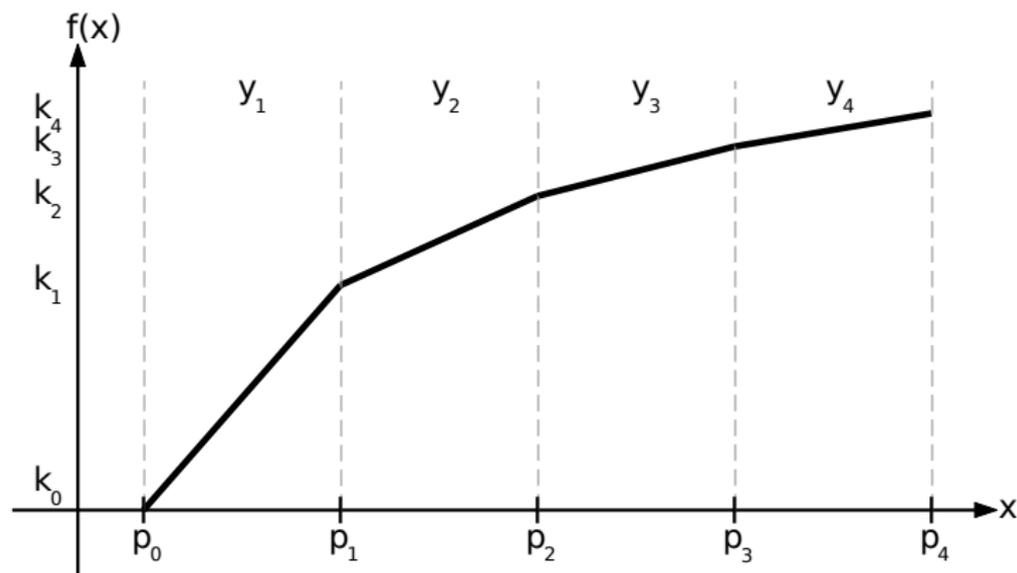
## Verkaufserlöse

$$f_1(x_1) = 2000 \cdot \sqrt{x_1}$$

$$f_2(x_2) = 1800 \cdot \sqrt{x_2}$$



# Stückweise lineare Funktionen



## 3.1 Abbildung logischer Ausdrücke

## 3.2 Entscheidungs- abhängige Nebenbedingungen

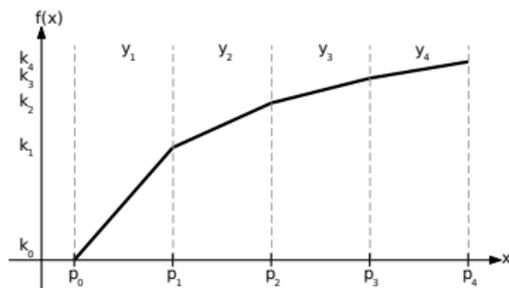
- Die Big-M-Methode
- OPL: Modellieren von  
Zeitperioden
- Disjunktive  
Nebenbedingungen

## 3.3 OPL: Kompakte Schreibweise

## 3.4 Stückweise definierte Funktionen

- Treppenfunktionen
- Stückweise lineare  
Funktionen
- OPL: Der `piecewise`-Befehl

# Funktionswerte als Konvexkombination



$$x = \sum_{n=0}^N z_n \cdot p_n$$

$$f(x) = \sum_{n=0}^N z_n \cdot f(p_n)$$

$$\sum_{n=0}^N z_n = 1$$

$$0 \leq z_n \leq 1 \quad \forall n \in \{1, \dots, N\}$$

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen  
Stückweise lineare  
Funktionen  
OPL: Der piecewise-Befehl

# Vollständige Modellierung

$$f(x) = \sum_{n=0}^N z_n \cdot k_n$$

$$x = \sum_{n=0}^N z_n \cdot p_n$$

$$\sum_{n=0}^N z_n = 1$$

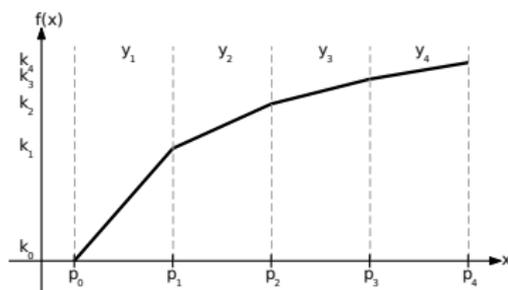
$$0 \leq z_n \leq 1 \quad \forall n \in \{1, \dots, N\}$$

$$\sum_{n=1}^N y_n = 1$$

$$z_0 \leq y_1$$

$$z_n \leq y_n + y_{n+1} \quad \forall n \in \{1, \dots, N-1\}$$

$$z_N \leq y_N$$



3 Techniken der  
binären  
Modellierung

CC-BY-SA  
A. Popp

3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeiträumen  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

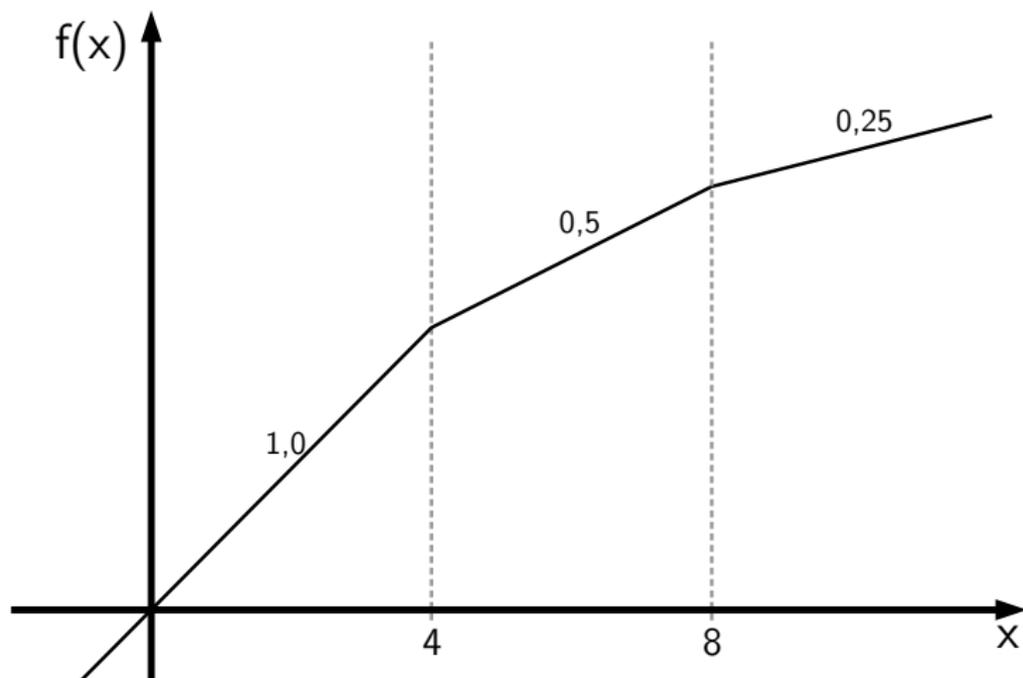
3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen

Stückweise lineare  
Funktionen

OPL: Der piecewise-Befehl

# Stückweise lineare Funktionen nach Steigung



3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeiträumen  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

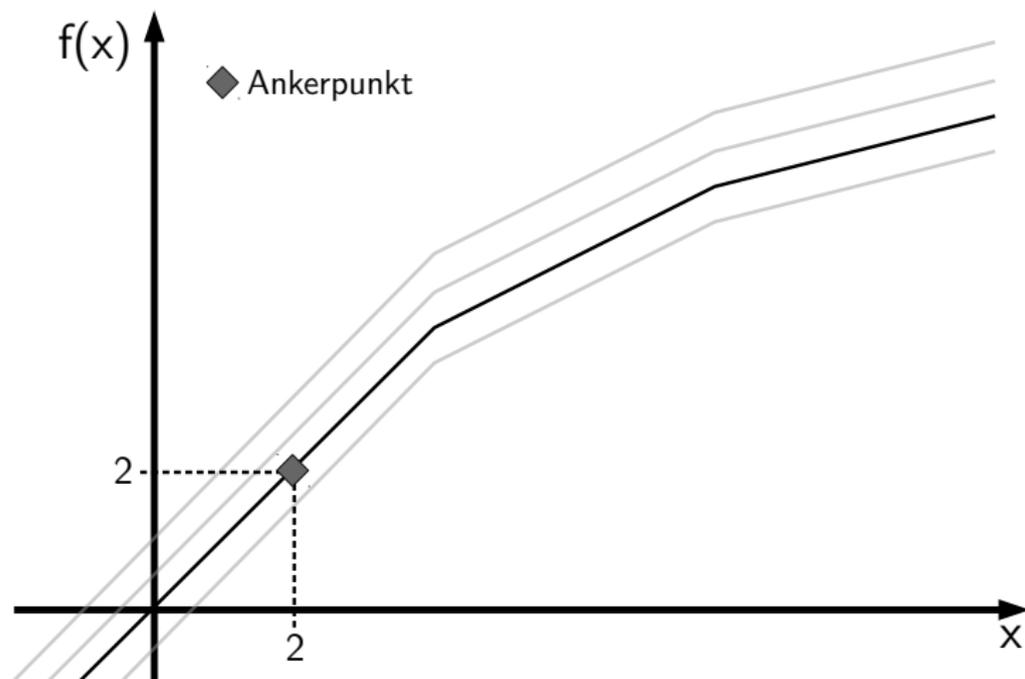
Treppenfunktionen  
Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

# Verankerung von stückweise linearen Funktionen

3 Techniken der  
binären  
Modellierung

CC-BY-SA  
A. Popp



3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeiträumen  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen  
Stückweise lineare  
Funktionen

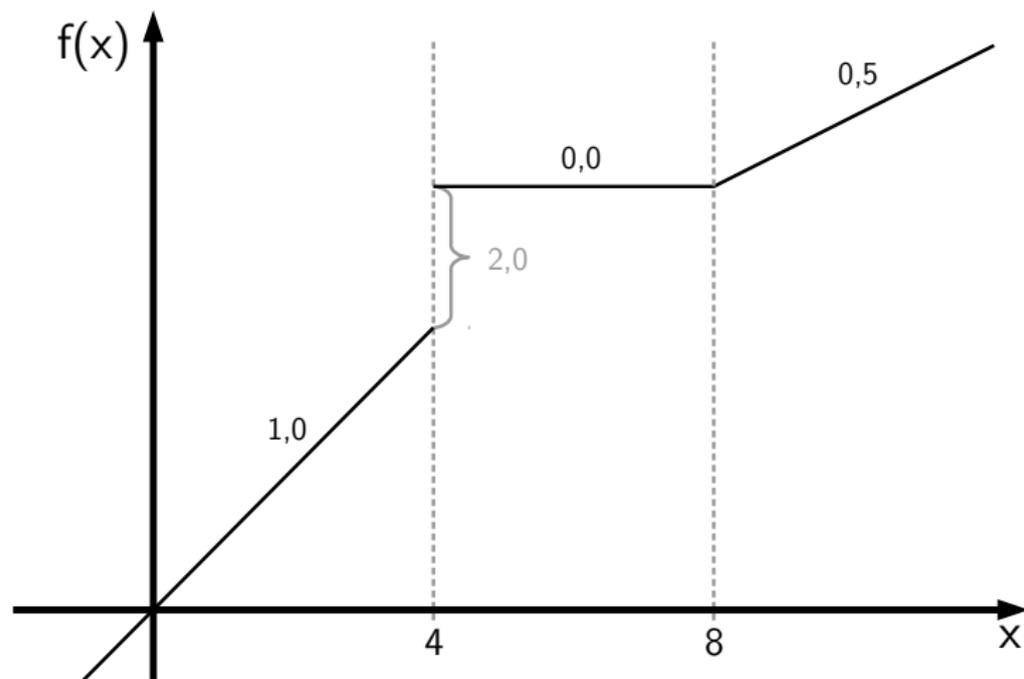
OPL: Der `piecewise`-Befehl



# Treppenfunktionen und allgemeine Unstetigkeiten

3 Techniken der  
binären  
Modellierung

CC-BY-SA  
A. Popp



3.1 Abbildung  
logischer  
Ausdrücke

3.2 Entscheidungs-  
abhängige  
Nebenbedingungen

Die Big-M-Methode  
OPL: Modellieren von  
Zeitperioden  
Disjunktive  
Nebenbedingungen

3.3 OPL:  
Kompakte  
Schreibweise

3.4 Stückweise  
definierte  
Funktionen

Treppenfunktionen  
Stückweise lineare  
Funktionen

OPL: Der `piecewise`-Befehl

