

# Seminar Energieinformatik

## Mechanismen für die AC-Leistungsallokation

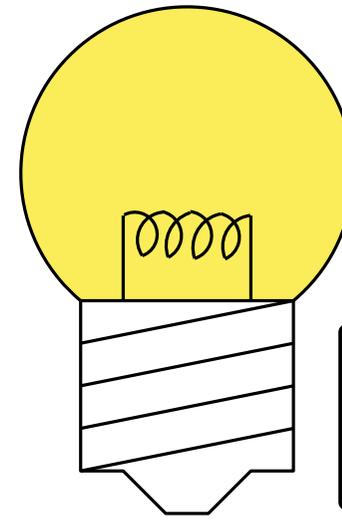
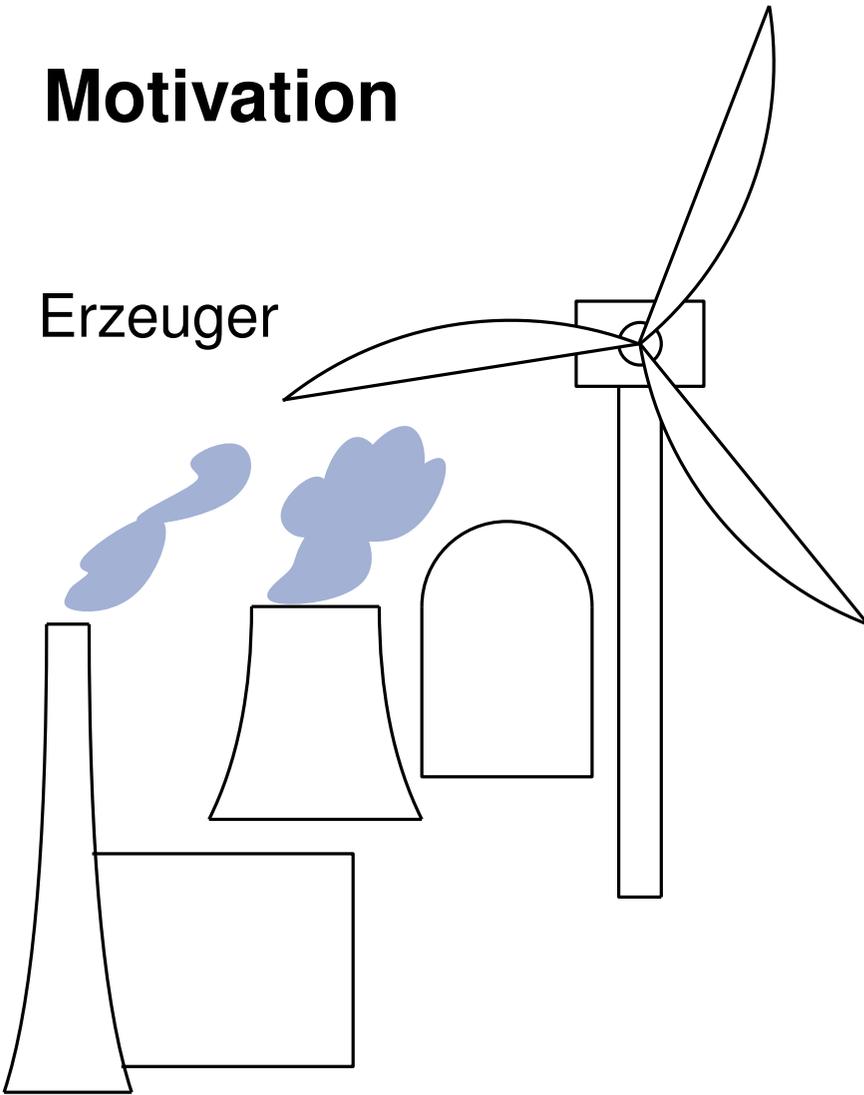
7. Februar 2017  
Anselm Erdmann

INSTITUT FÜR THEORETISCHE INFORMATIK · LEHRSTUHL ALGORITHMIK

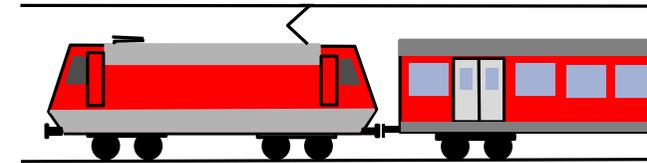
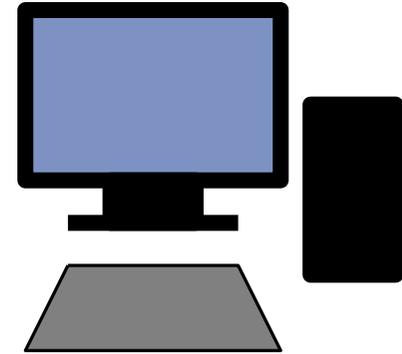


# Motivation

Erzeuger

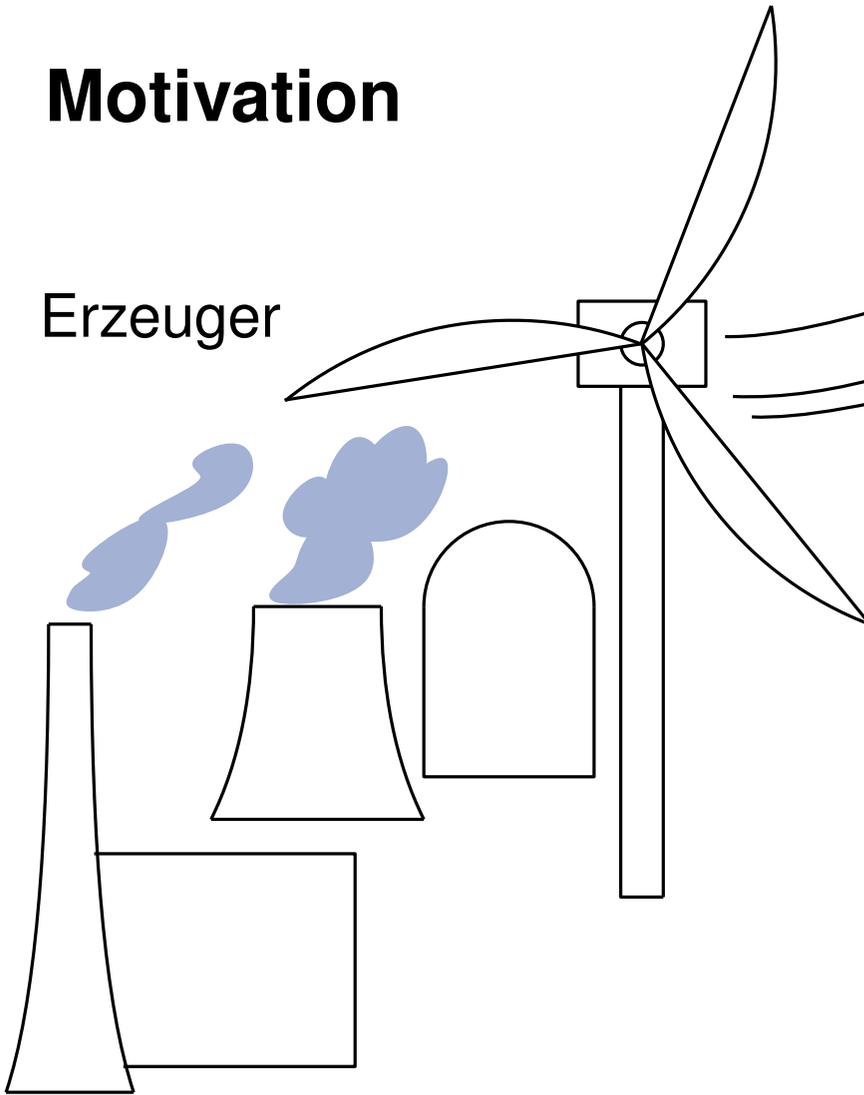


Verbraucher

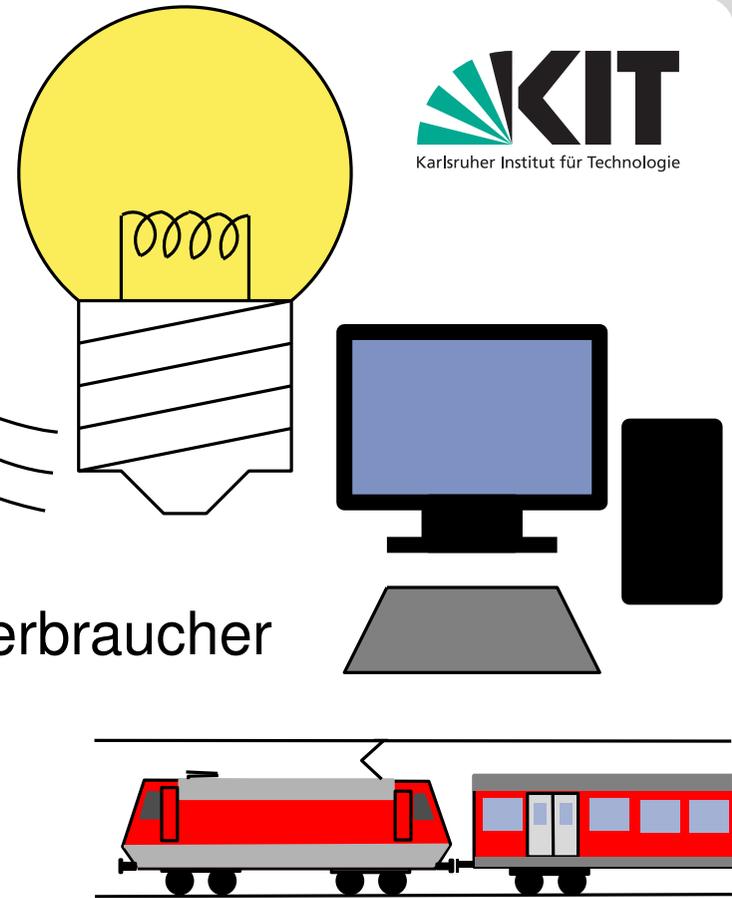


# Motivation

Erzeuger

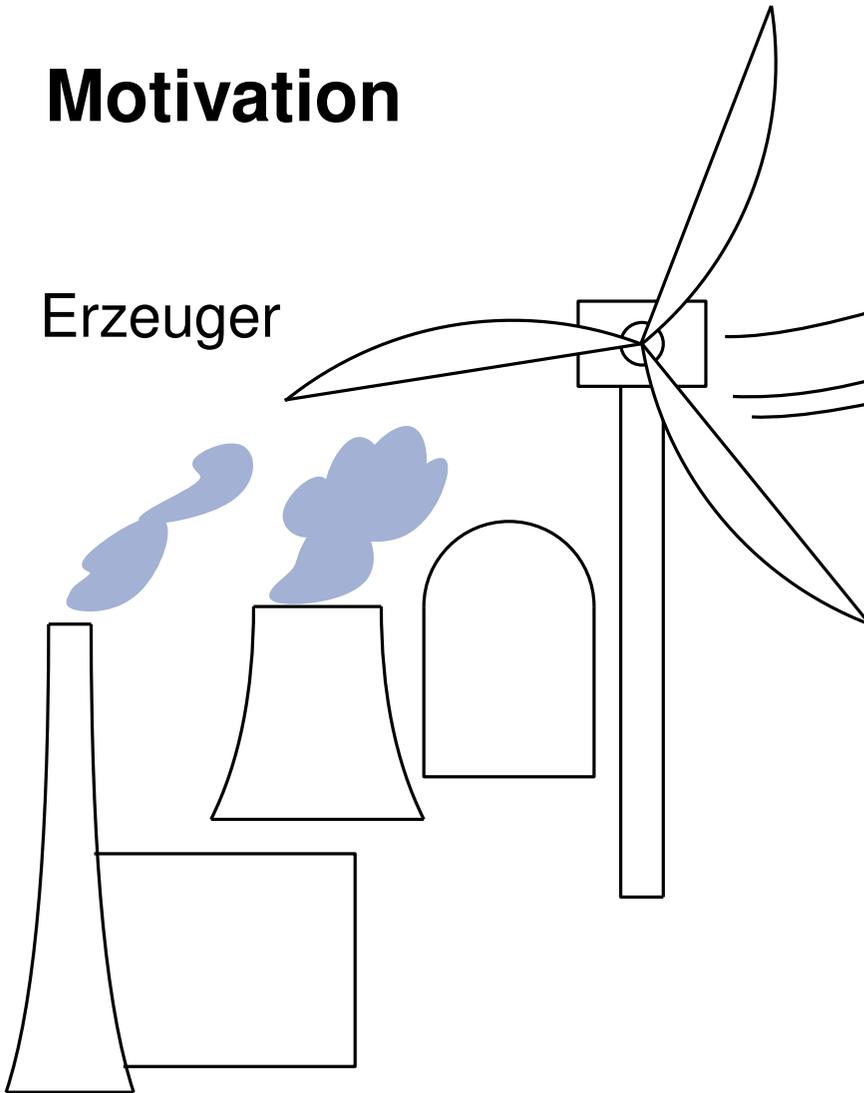


Verbraucher

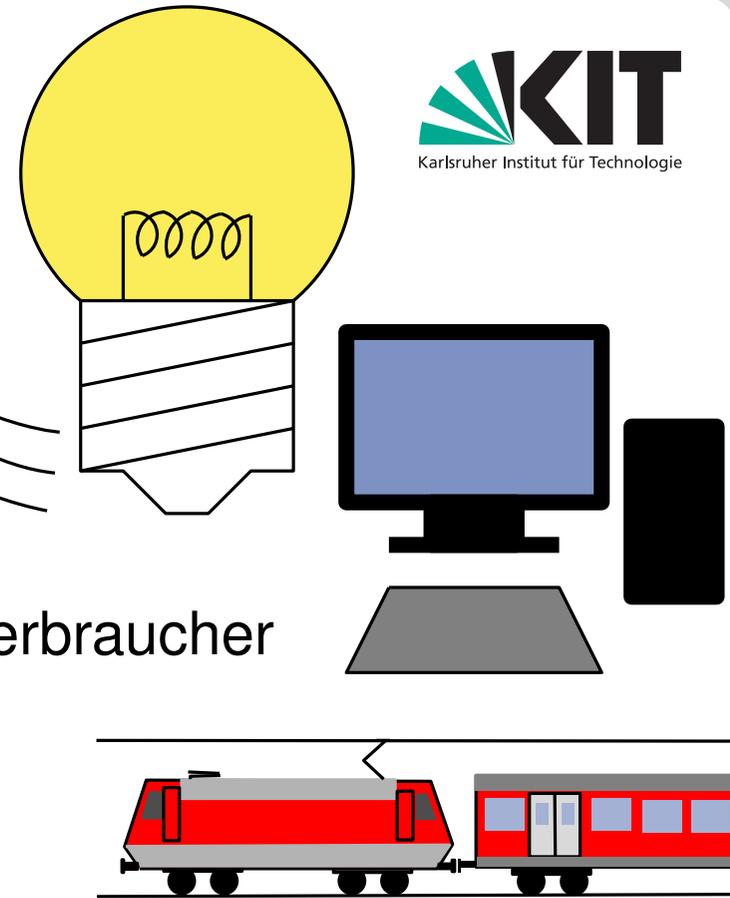


# Motivation

Erzeuger



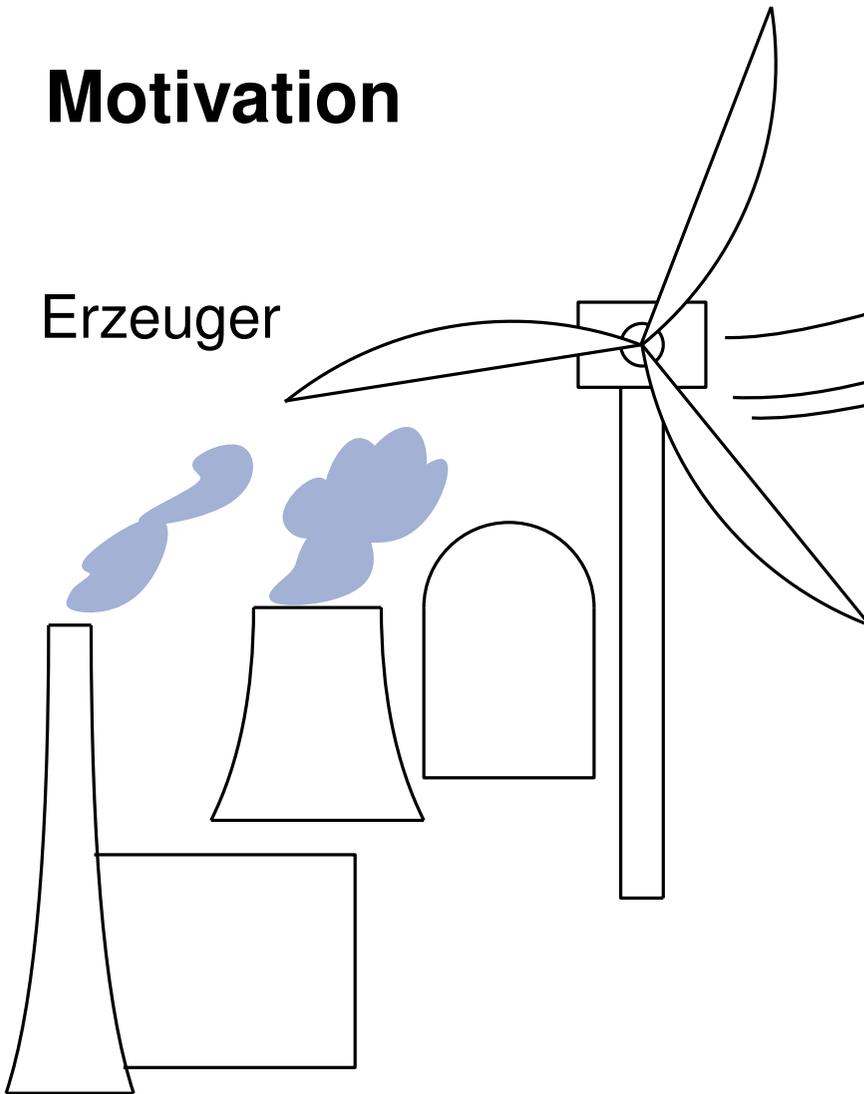
Verbraucher



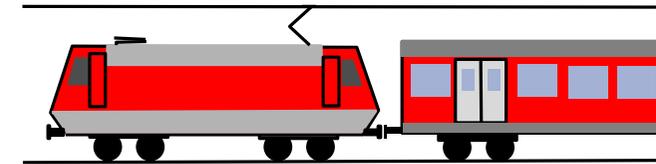
Bereitgestellte Leistung = Abgerufene Leistung

# Motivation

Erzeuger



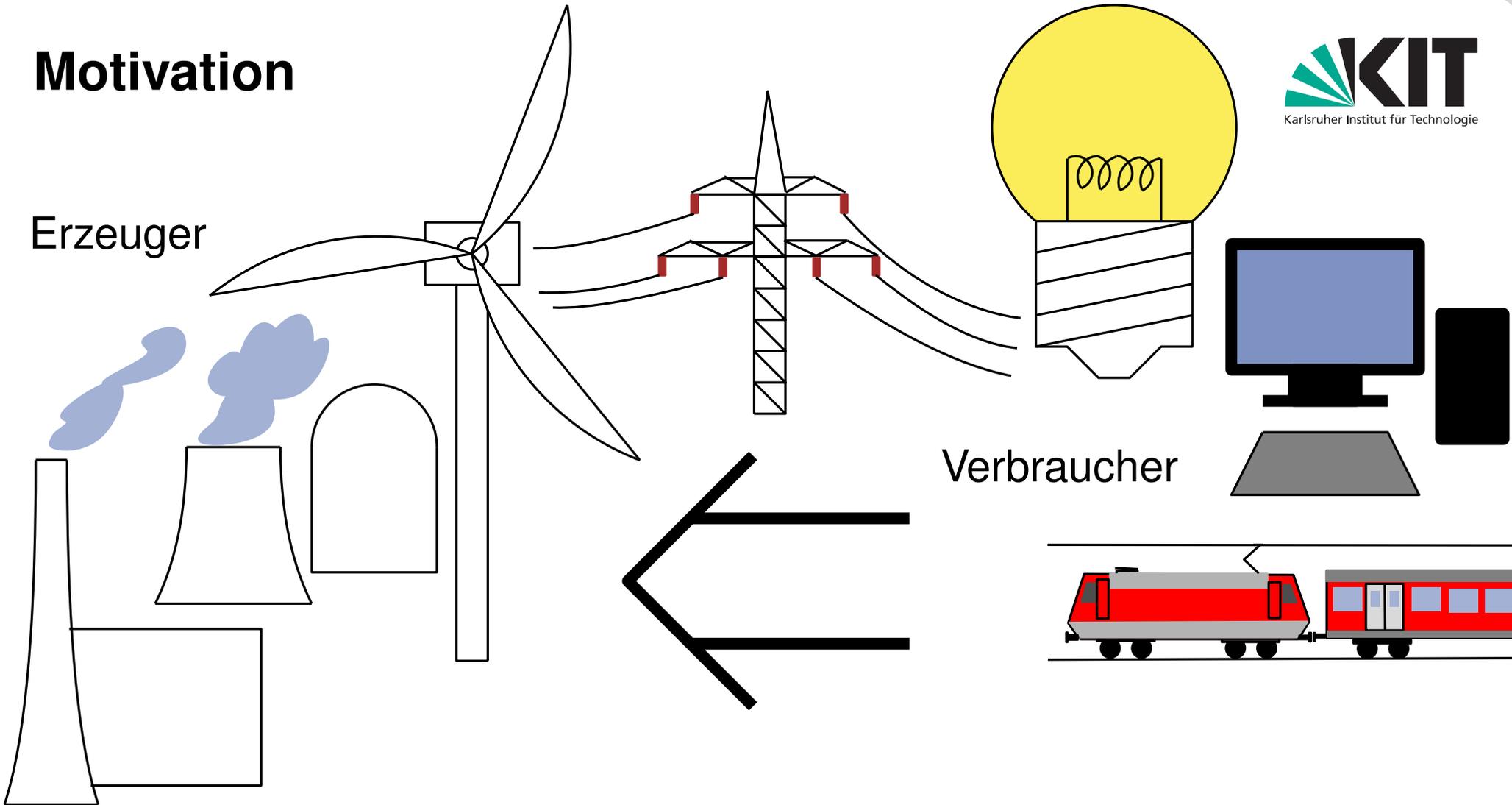
Verbraucher



Bereitgestellte Leistung  $\neq$  Abgerufene Leistung

# Motivation

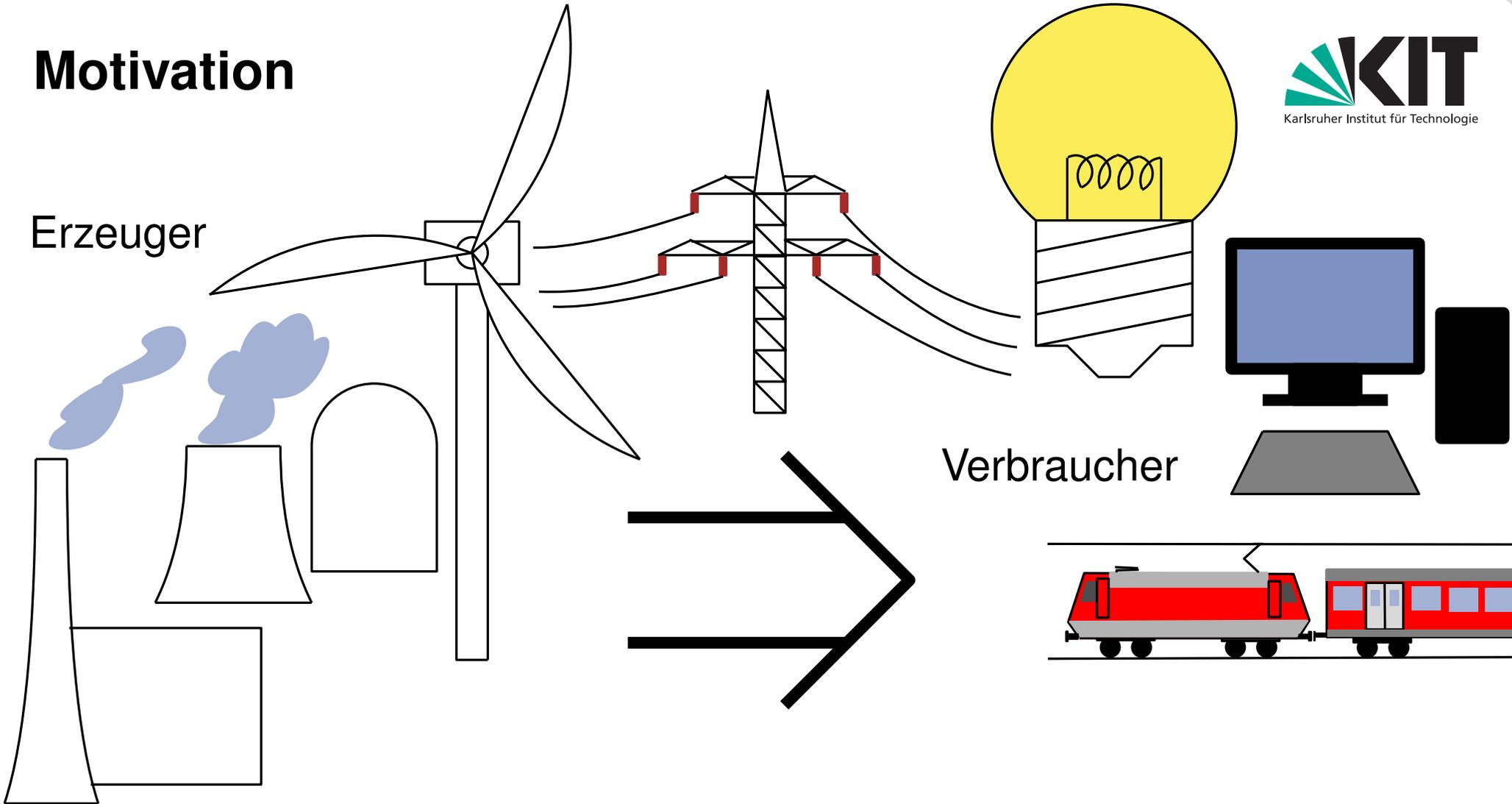
Erzeuger



Bereitgestellte Leistung  $\neq$  Abgerufene Leistung

# Motivation

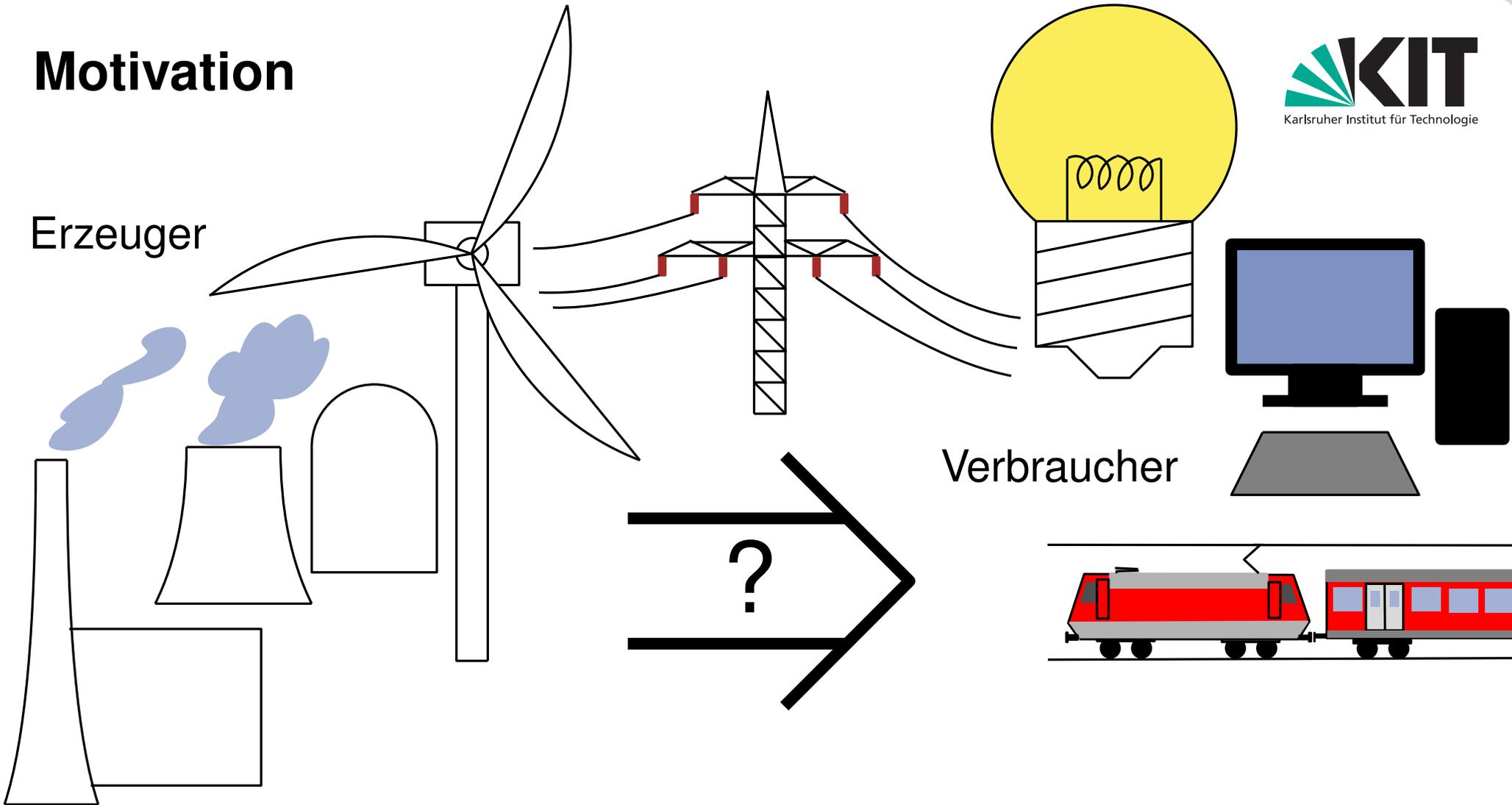
Erzeuger



Bereitgestellte Leistung  $\neq$  Abgerufene Leistung

# Motivation

Erzeuger



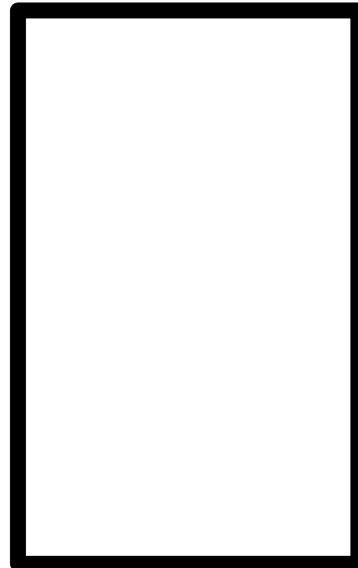
Bereitgestellte Leistung  $\neq$  Abgerufene Leistung

- Beteiligte
  - Leistungserzeuger: Bereitstellen von Leistung
  - Verbraucher: Anmelden von Bedarf und ggf. Nutzung der Leistung
- Problemmodellierung
  - Im Prinzip ein Knapsack-Problem

# Leistungsallokation

- Beteiligte
  - Leistungserzeuger: Bereitstellen von Leistung
  - Verbraucher: Anmelden von Bedarf und ggf. Nutzung der Leistung
- Problemmodellierung
  - Im Prinzip ein Knapsack-Problem

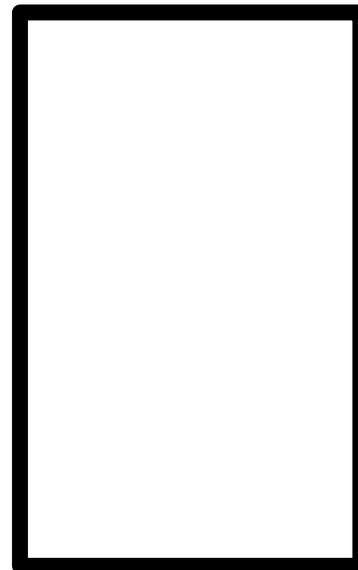
Leistungskapazität  
(von allen Erzeugern)



# Leistungsallokation

- Beteiligte
  - Leistungserzeuger: Bereitstellen von Leistung
  - Verbraucher: Anmelden von Bedarf und ggf. Nutzung der Leistung
- Problemmodellierung
  - Im Prinzip ein Knapsack-Problem

Leistungskapazität  
(von allen Erzeugern)



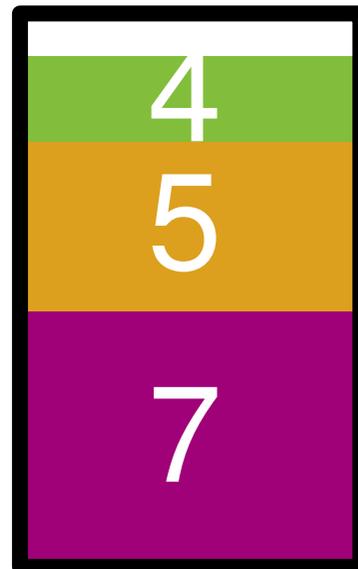
Bedarfe (der Verbraucher)



# Leistungsallokation

- Beteiligte
  - Leistungserzeuger: Bereitstellen von Leistung
  - Verbraucher: Anmelden von Bedarf und ggf. Nutzung der Leistung
- Problemmodellierung
  - Im Prinzip ein Knapsack-Problem

Leistungskapazität  
(von allen Erzeugern)

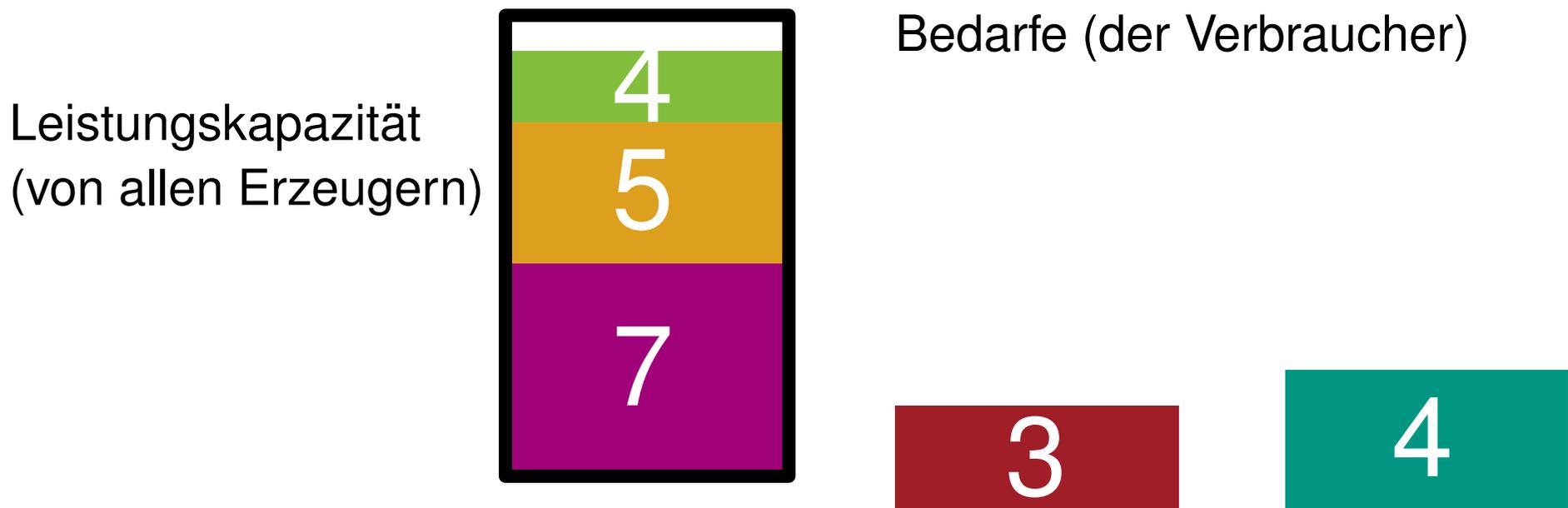


Bedarfe (der Verbraucher)



# Leistungsallokation

- Beteiligte
  - Leistungserzeuger: Bereitstellen von Leistung
  - Verbraucher: Anmelden von Bedarf und ggf. Nutzung der Leistung
- Problemmodellierung
  - Im Prinzip ein Knapsack-Problem



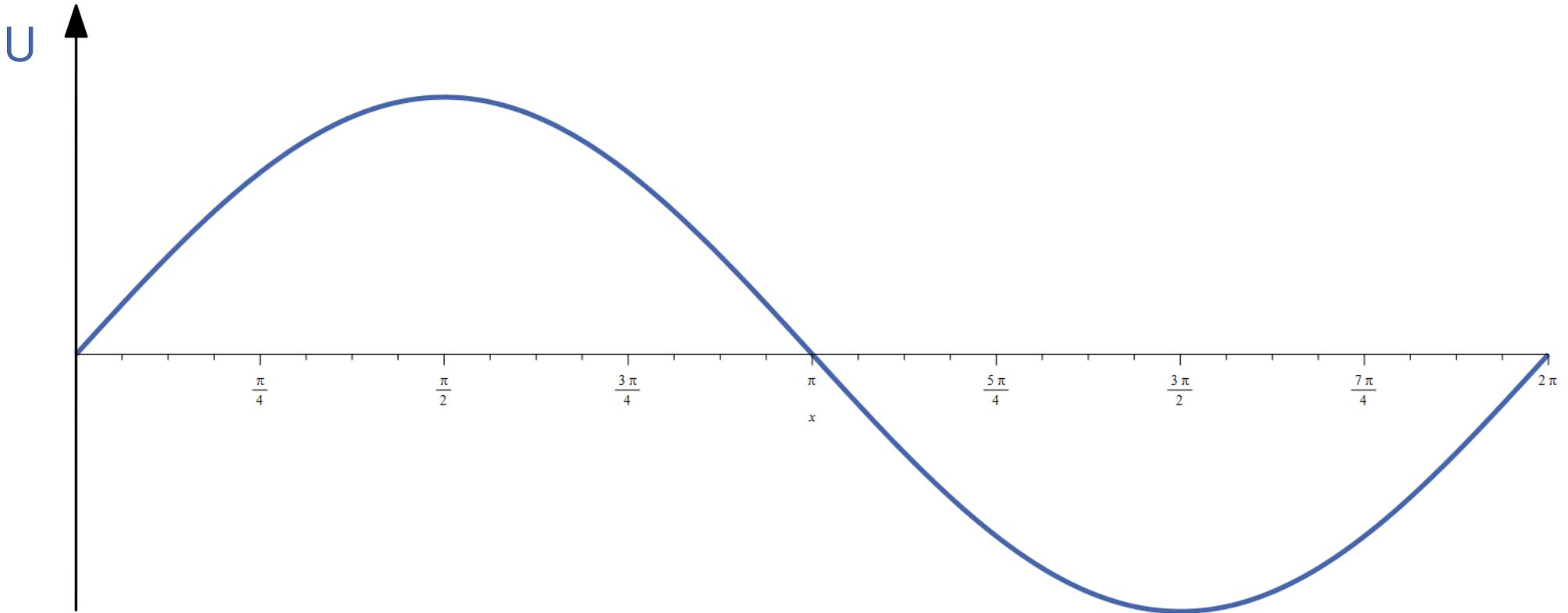
- In Wechselstromnetzen nicht ohne Weiteres anwendbar

# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger Spannungsverlauf
- Phasenlage des Stroms kann verschoben sein
- $P(t) = U(t) \cdot I(t)$

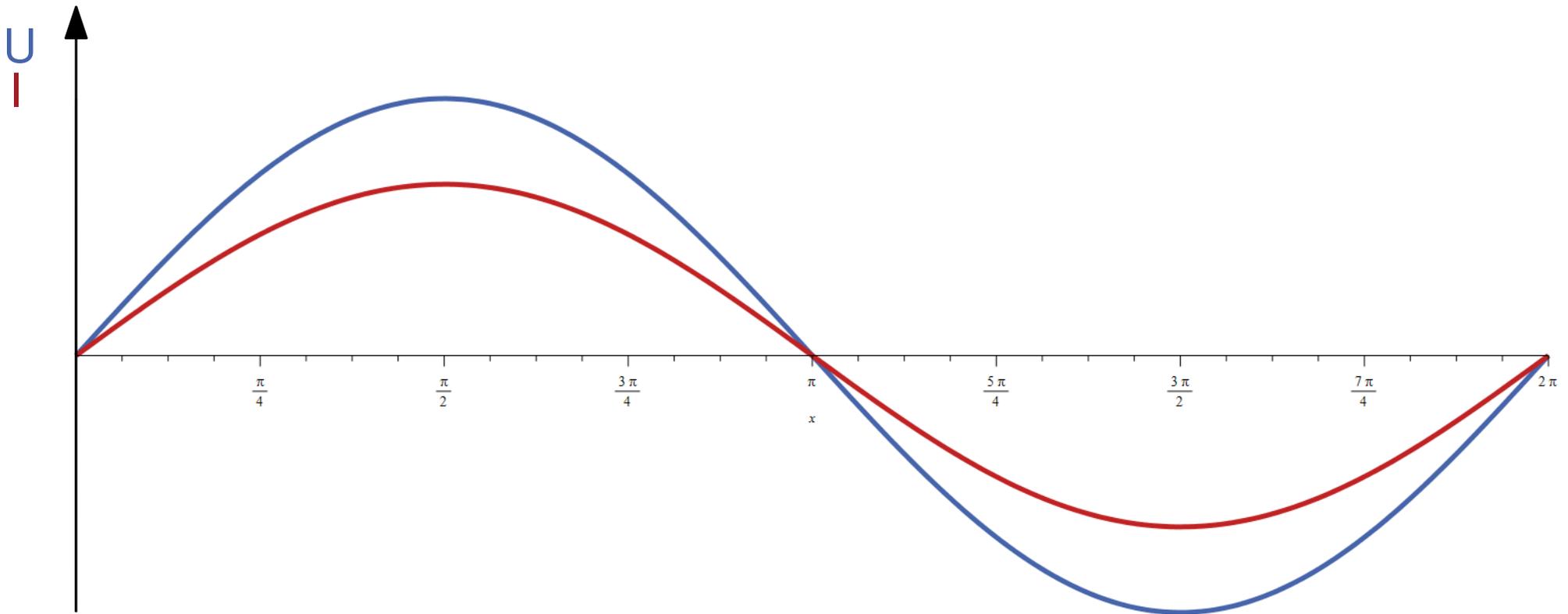
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des Stroms kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



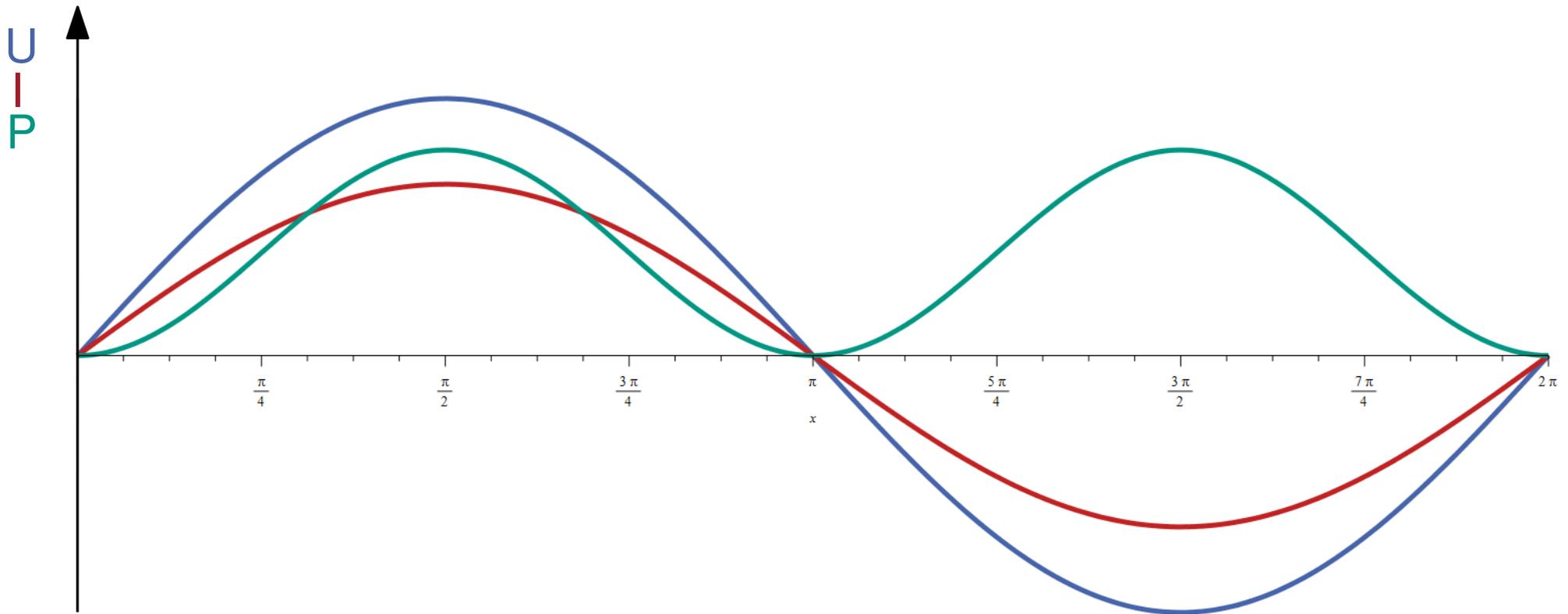
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



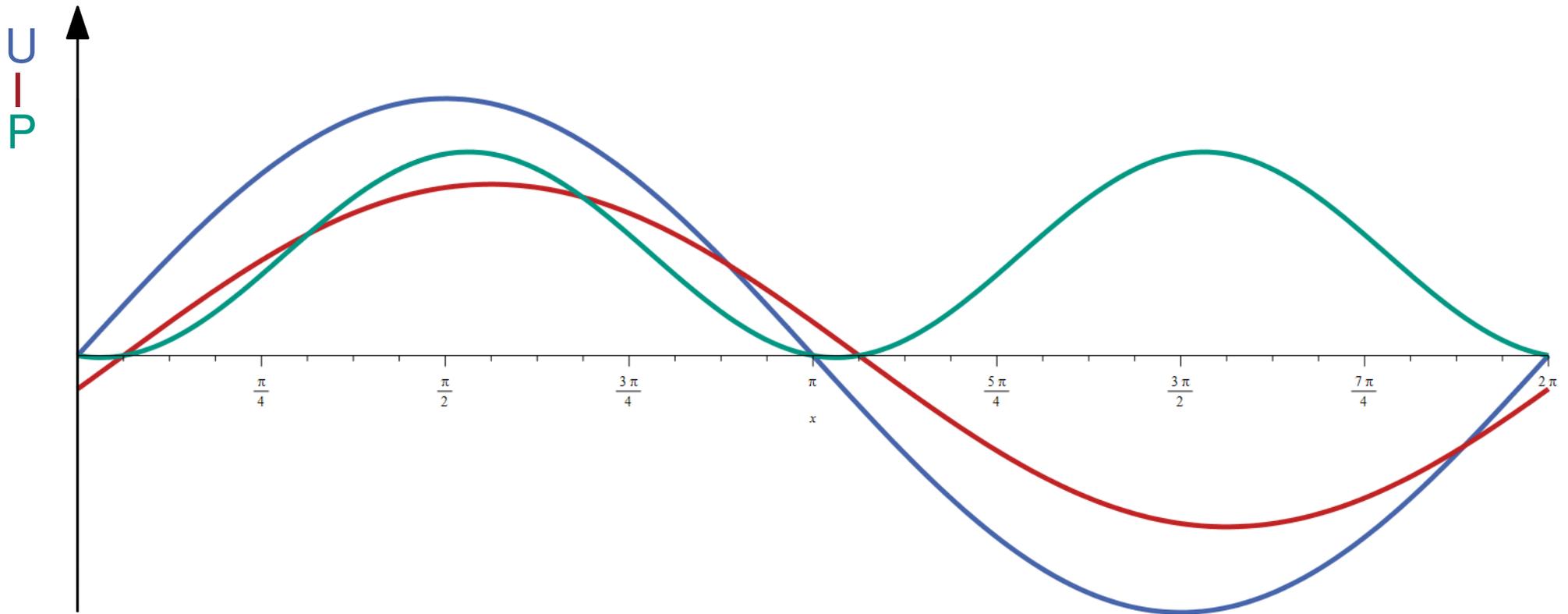
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



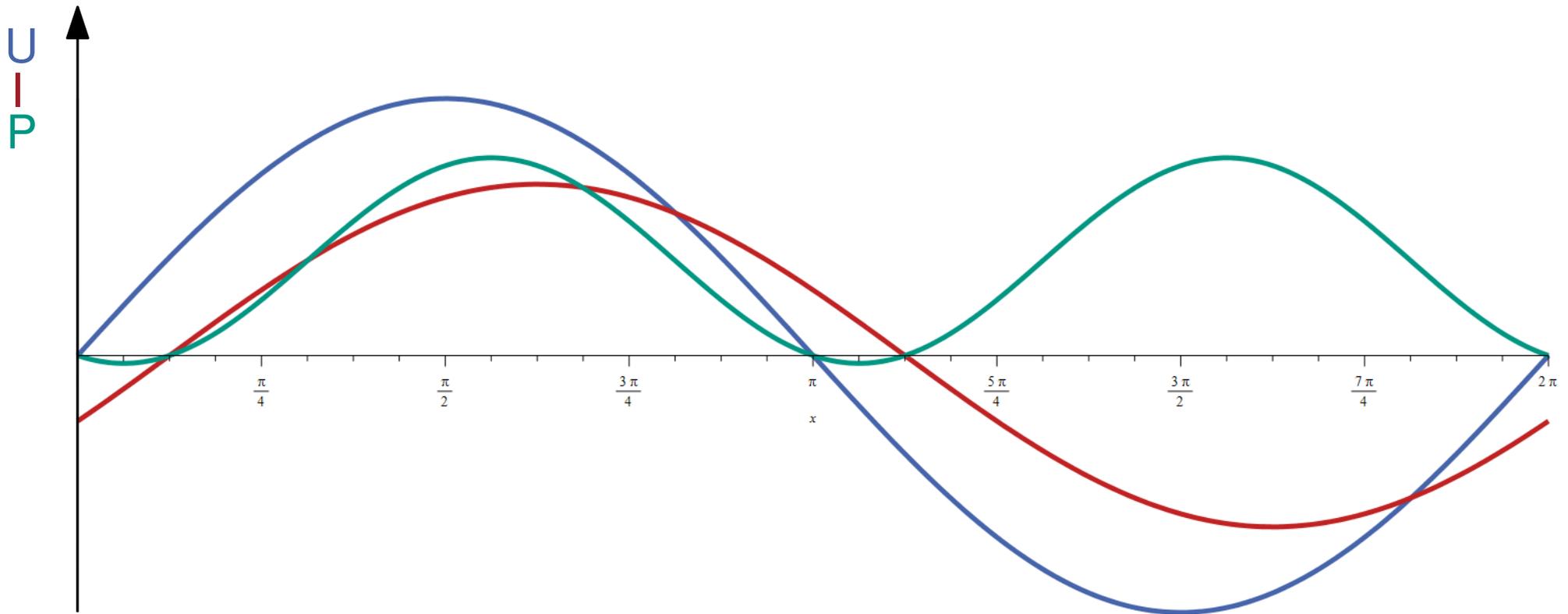
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



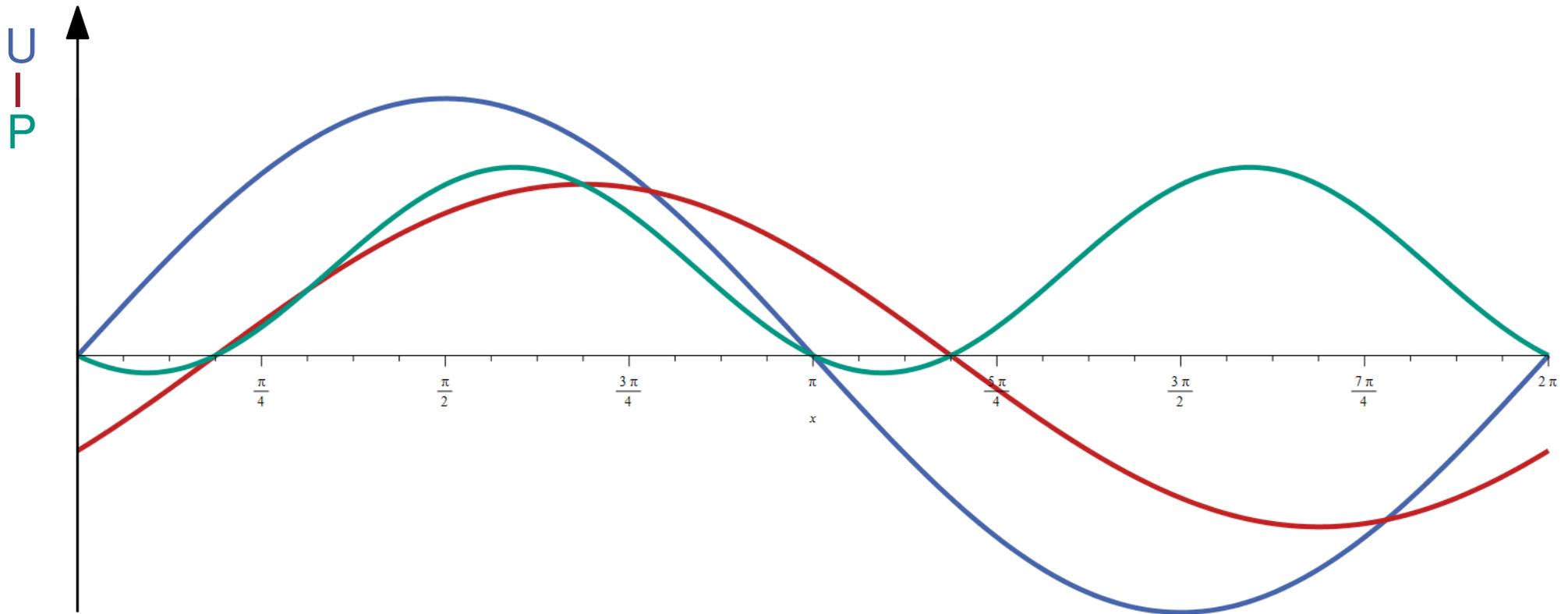
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



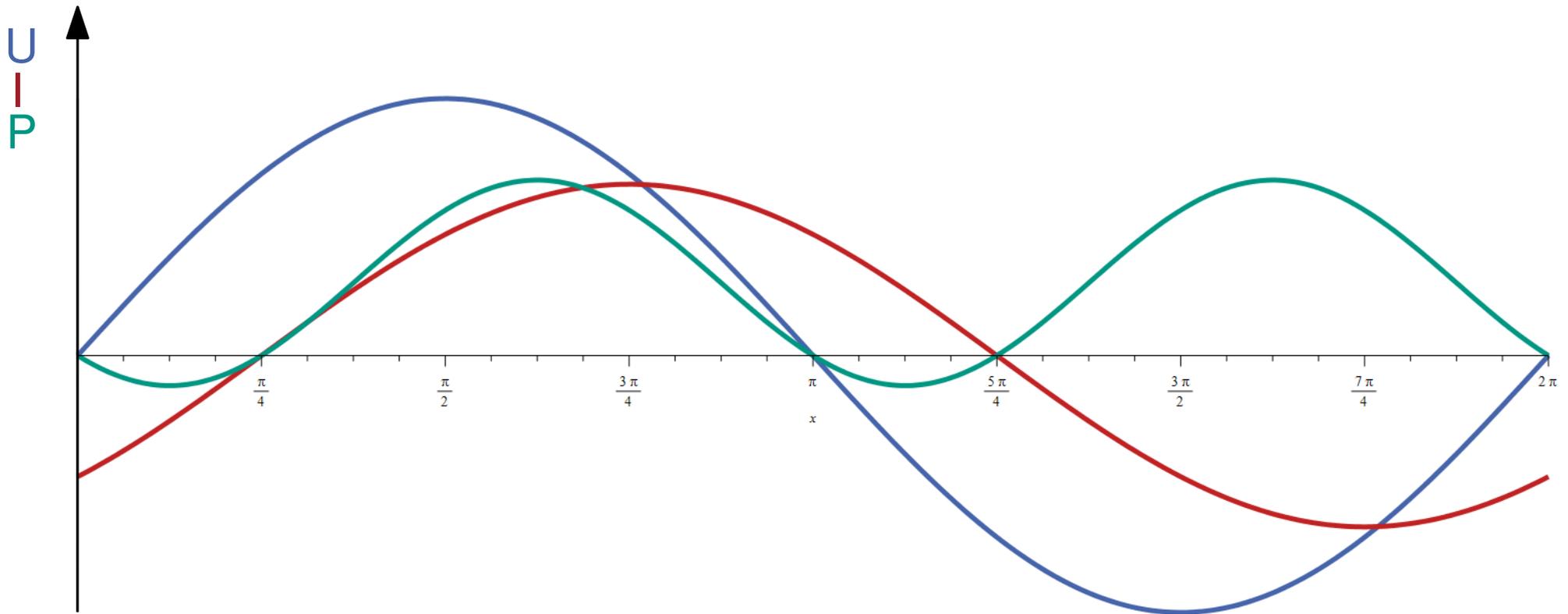
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



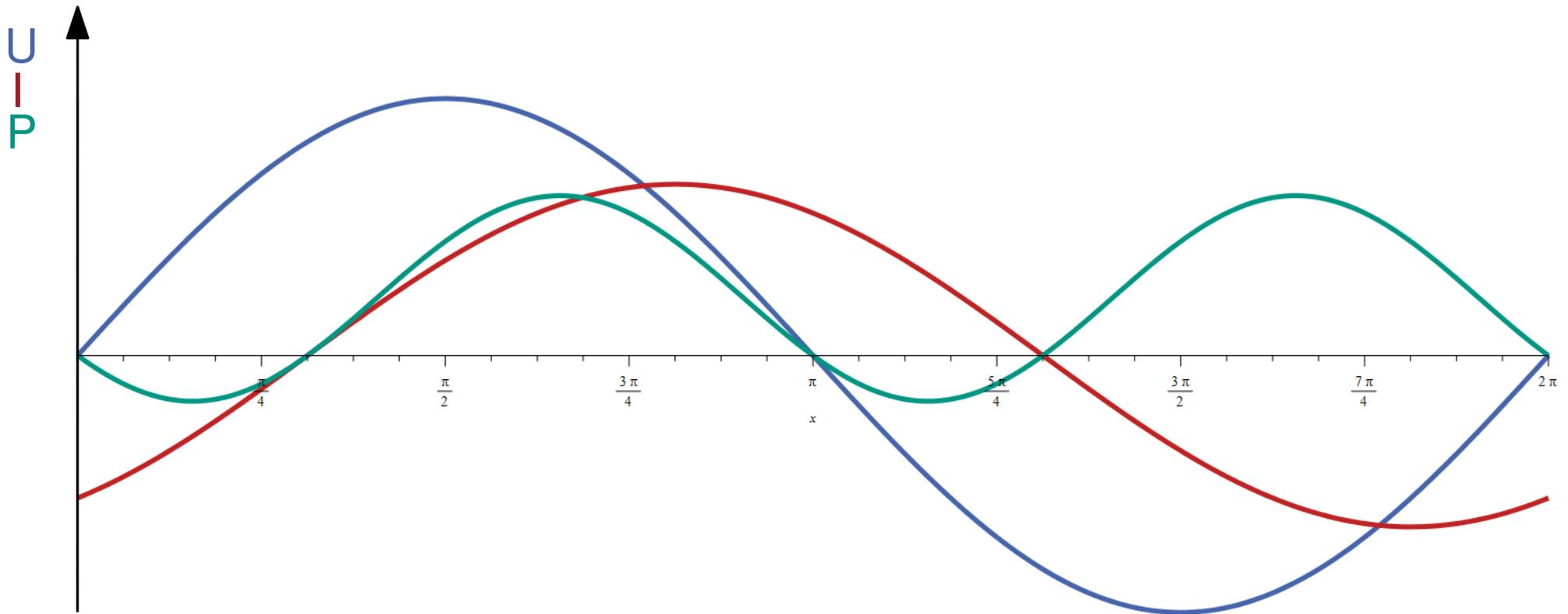
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



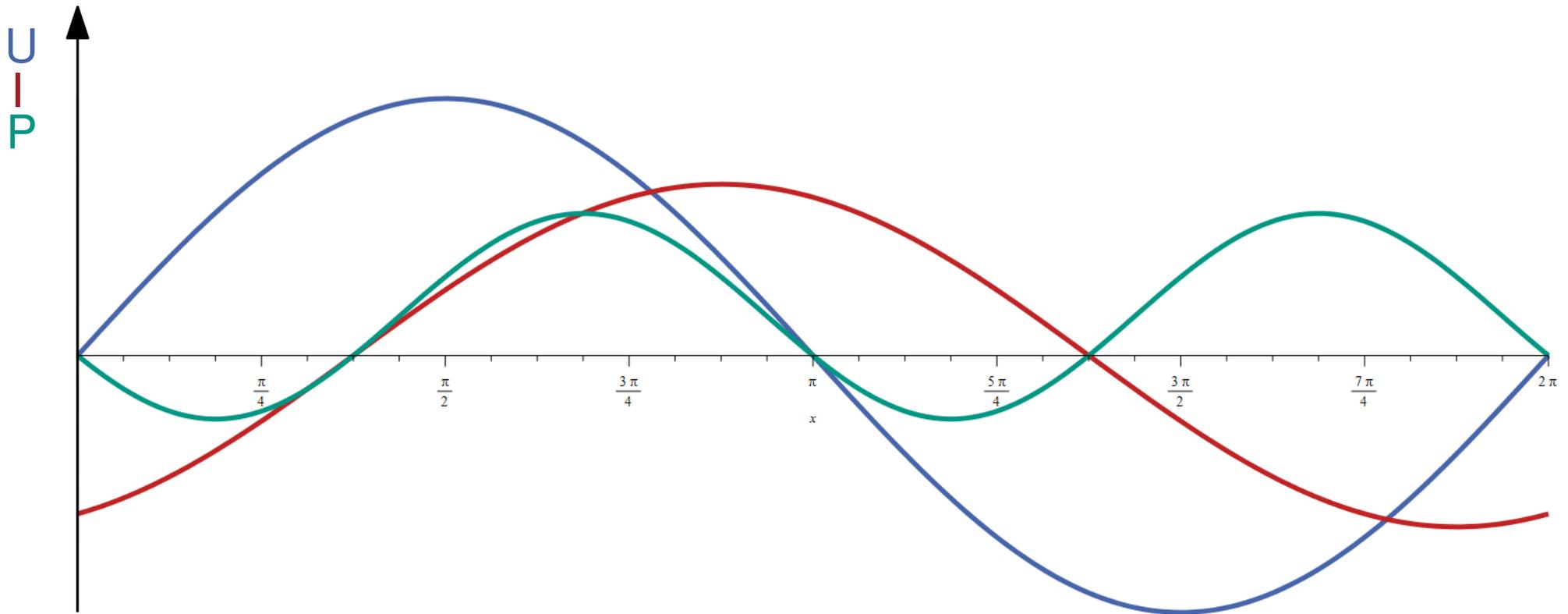
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



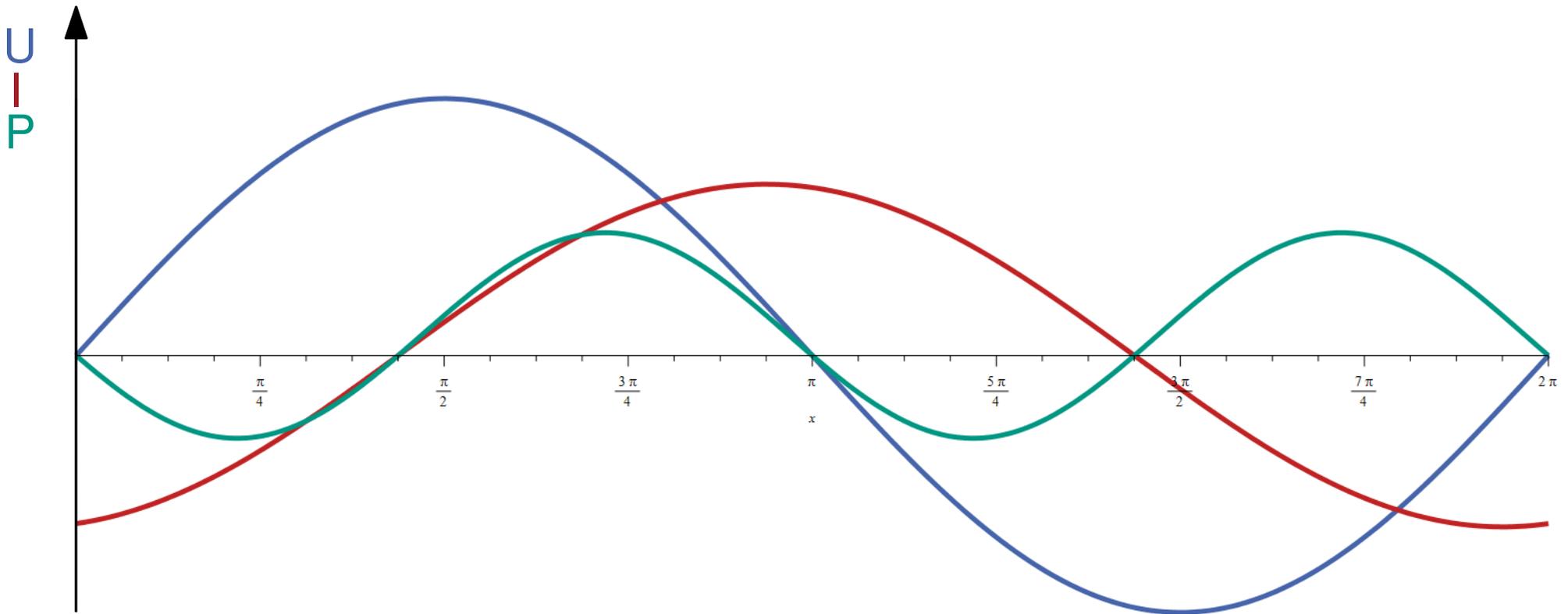
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



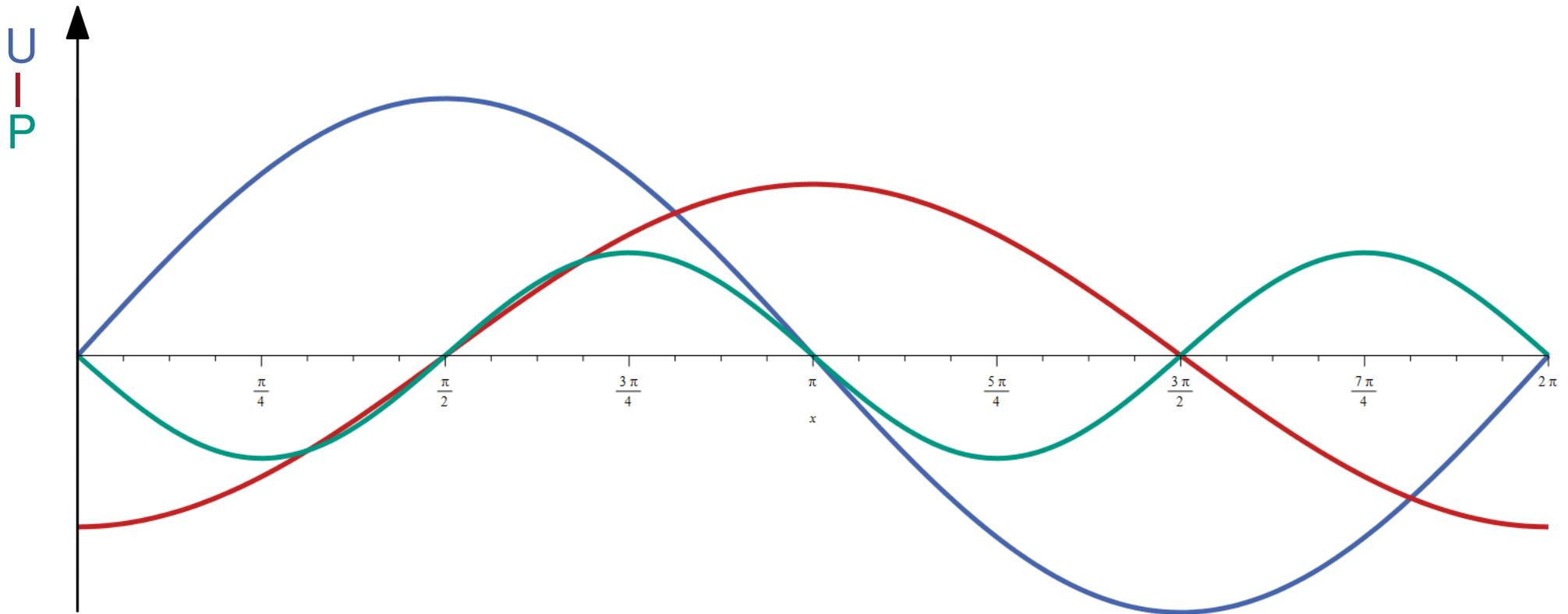
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



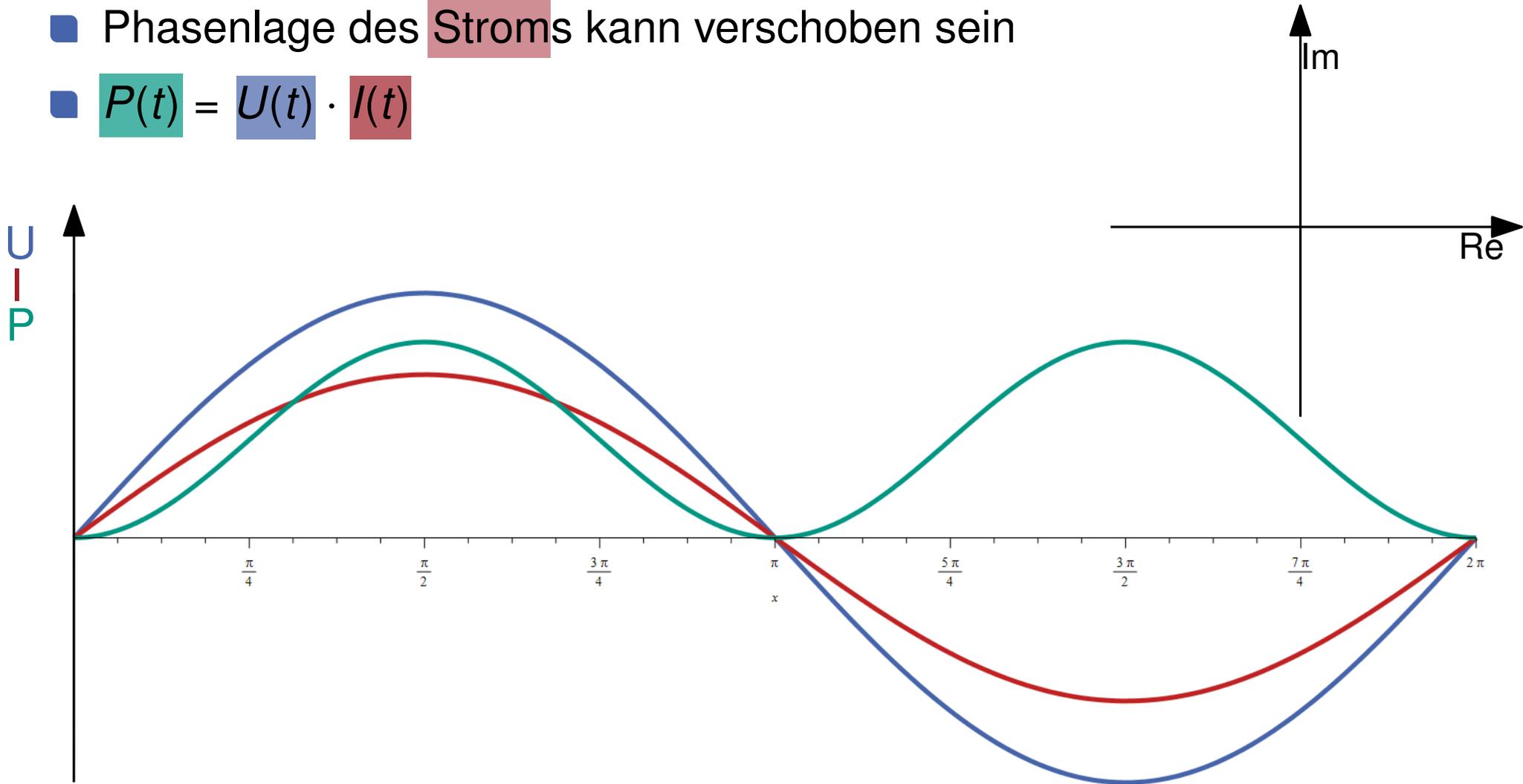
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



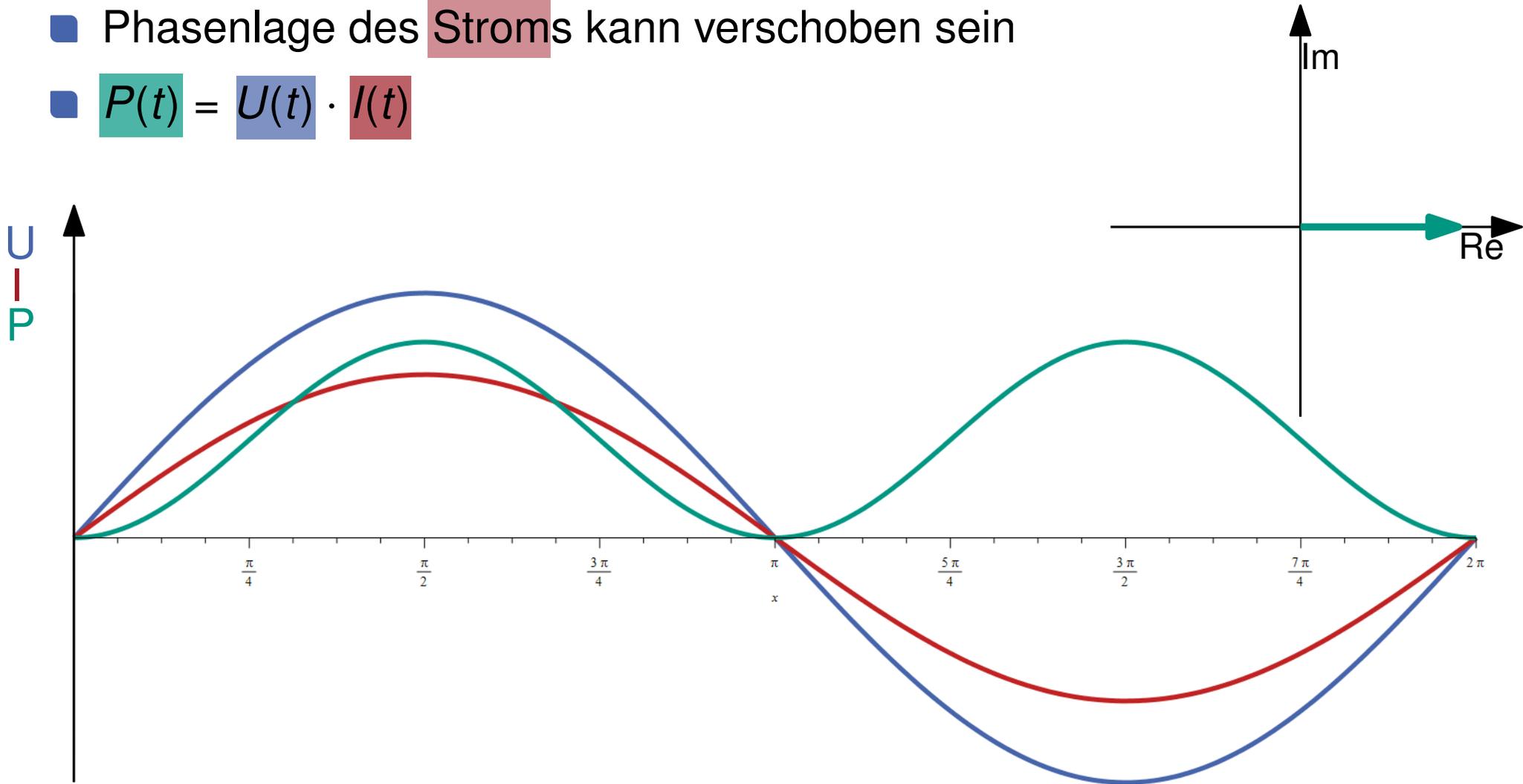
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



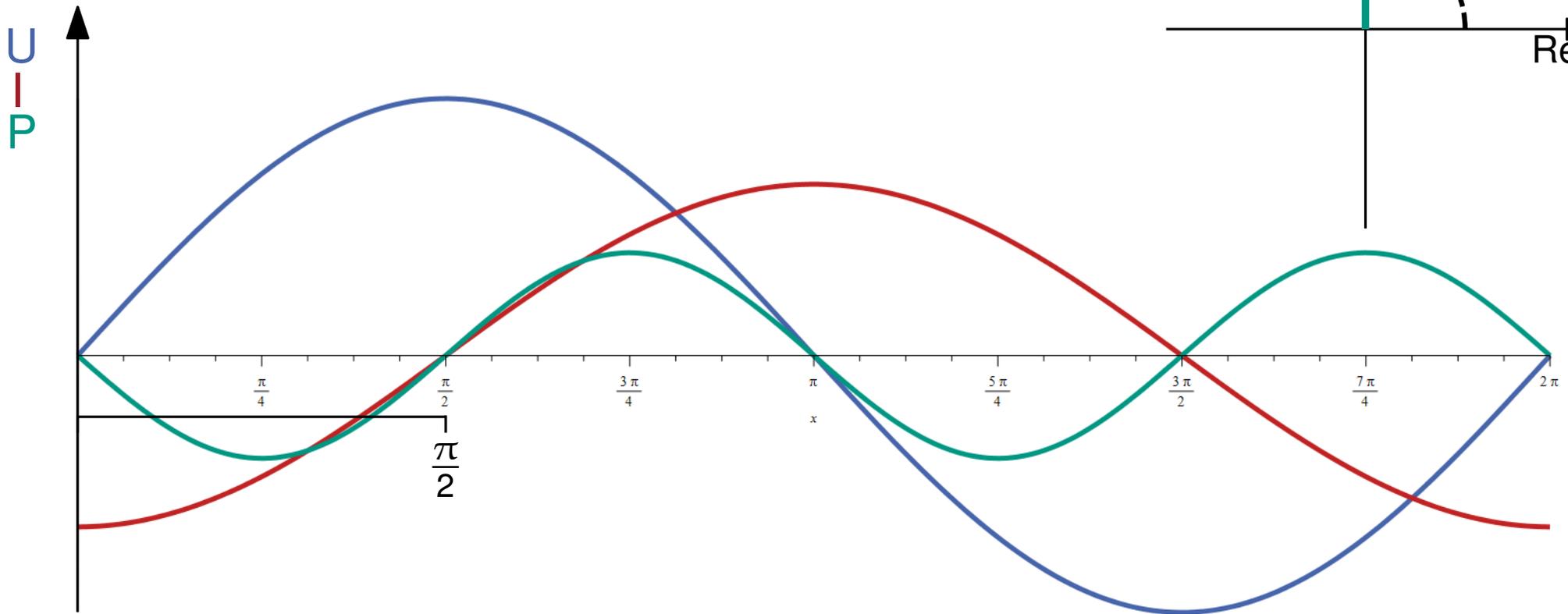
# Spannung, Strom und Leistung in AC-Netzen

- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$

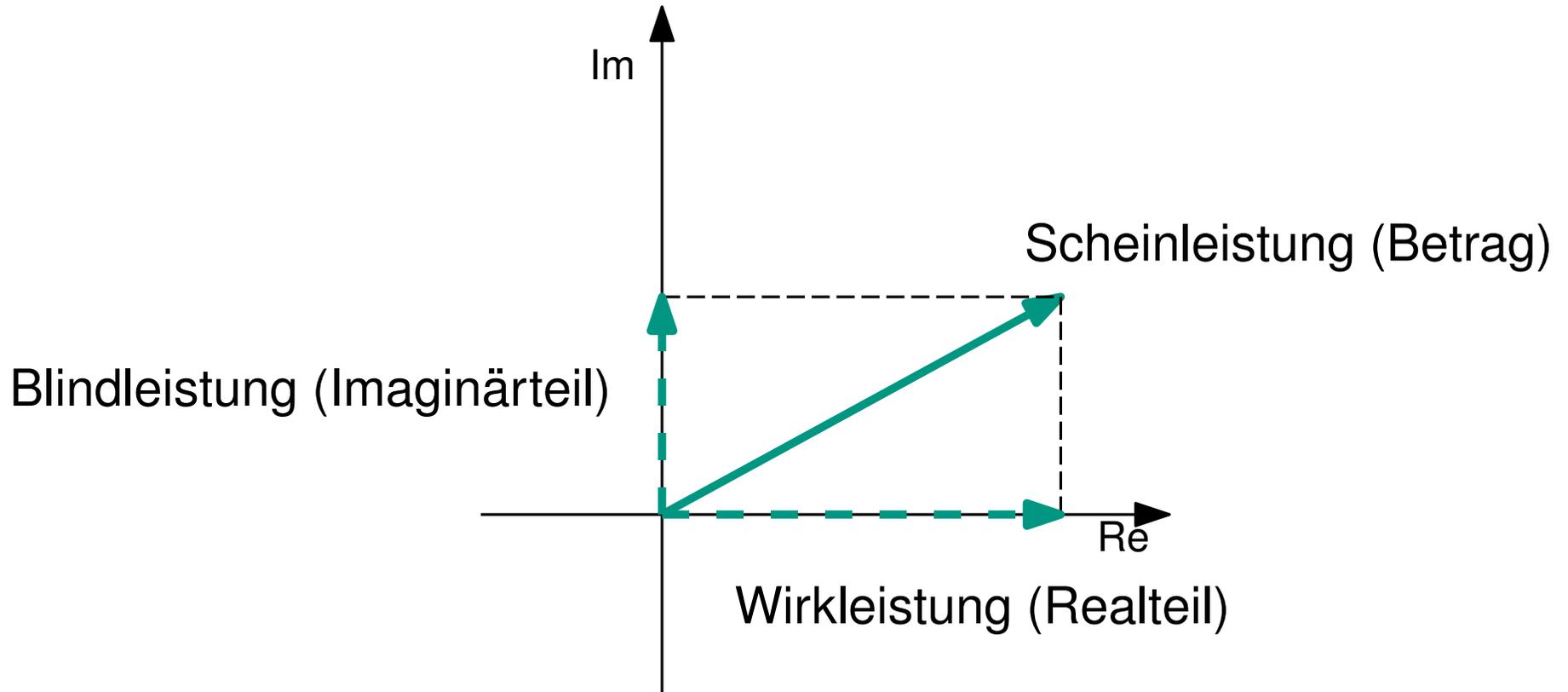


# Spannung, Strom und Leistung in AC-Netzen

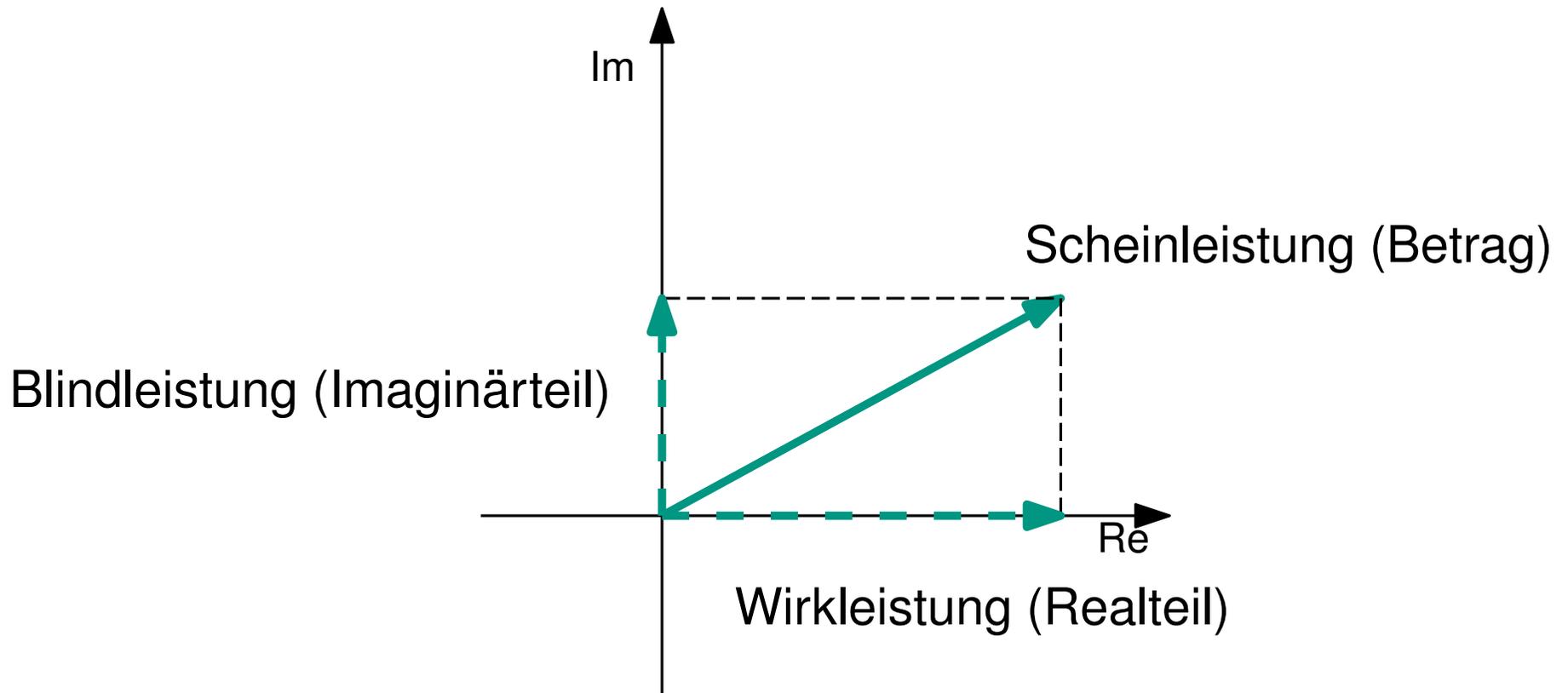
- Sinusförmiger **Spannungsverlauf**
- Phasenlage des **Stroms** kann verschoben sein
- $P(t) = U(t) \cdot I(t)$



# Komplexe Leistungsdarstellung



# Komplexe Leistungsdarstellung



Leistungserzeugung (Realteil  $< 0$ )

Leistungsverbrauch (Realteil  $> 0$ )

Chau, Elbassioni, Khonji 2016:

## **Truthful Mechanisms for Combinatorial Allocation of Electric Power in Alternating Current Electric Systems for Smart Grid**

- Lan You an Chi-Kin Chau:  
Complex-Demand Knapsack Problems and Incentives in AC Power Systems. 2013.  
*(siehe Vortrag vom 13. 12. 2016)*
- Carleton Coffrin, Pascal van Hentenryck:  
A Linear-Programming Approximation of AC Power Flows. 2014  
*Betrachtung des Leistungsflusses im Wechselstromnetz.  
Berücksichtigung tiefergehender physikalischer Randbedingungen.*
- Shahar Dobzinski, Noam Nisan:  
Mechanisms for Multi-Unit Auctions. 2010. In: Journal of Artificial Intelligence und Research 37, S. 85-98.  
*Ähnliches Problem, ohne Energienetzkontext.*

# Modellbedingungen

- Ausschließliche Betrachtung der Leistungssumme aller Erzeuger
- Leistungserzeuger stellen eine feste Scheinleistung zur Verfügung
- Die bereitstehende Scheinleistung kann beliebig auf Wirk- und Blindleistung aufgeteilt werden kann
- Keine Betrachtung von Verlusten
- Keine Betrachtung des Netzes
- Verbraucher geben ihre komplexen Bedarfe bekannt

# Komplexes Knapsackproblem CKP

- Bedarf
  - Größe  $d_k \in \mathbb{C}$
  - Wert  $v_k \in \mathbb{R}^+$
  - Gewichtung  $x_k \in \{0, 1\}$
- Erzeugungskapazität  $C \in \mathbb{R}^+$

# Komplexes Knapsackproblem CKP

- Bedarf
  - Größe  $d_k \in \mathbb{C}$
  - Wert  $v_k \in \mathbb{R}^+$
  - Gewichtung  $x_k \in \{0, 1\}$
- Erzeugungskapazität  $C \in \mathbb{R}^+$

$$\max \sum_{k=1}^n x_k v_k \quad \text{mit} \quad \left| \sum_{k=1}^n x_k d_k \right| \leq C$$

# Komplexes Knapsackproblem CKP

- Bedarf
  - Größe  $d_k \in \mathbb{C}$
  - Wert  $v_k \in \mathbb{R}^+$
  - Gewichtung  $x_k \in \{0, 1\}$
- Erzeugungskapazität  $C \in \mathbb{R}^+$

$$\max \sum_{k=1}^n x_k v_k \text{ mit } \left| \sum_{k=1}^n x_k d_k \right| \leq C$$

- NP-vollständig

# Komplexes Knapsackproblem CKP

- Bedarf

- Größe  $d_k \in \mathbb{C}$

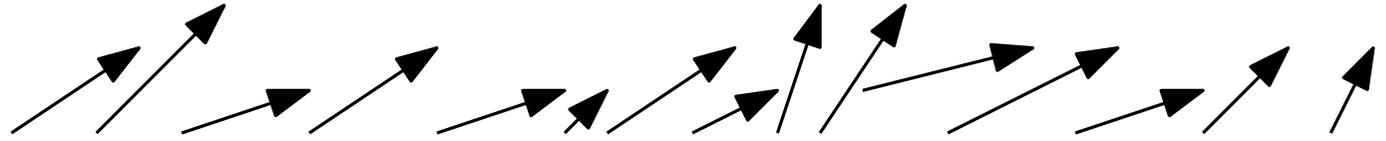
- Wert  $v_k \in \mathbb{R}^+$

- Gewichtung  $x_k \in \{0, 1\}$

- Erzeugungskapazität  $C \in \mathbb{R}^+$

$$\max \sum_{k=1}^n x_k v_k \text{ mit } \left| \sum_{k=1}^n x_k d_k \right| \leq C$$

- NP-vollständig



# Komplexes Knapsackproblem CKP

- Bedarf

- Größe  $d_k \in \mathbb{C}$

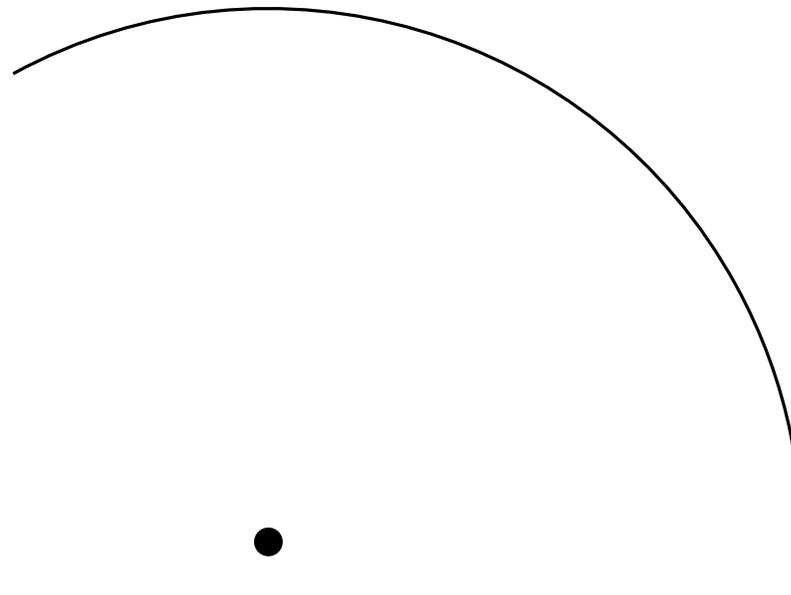
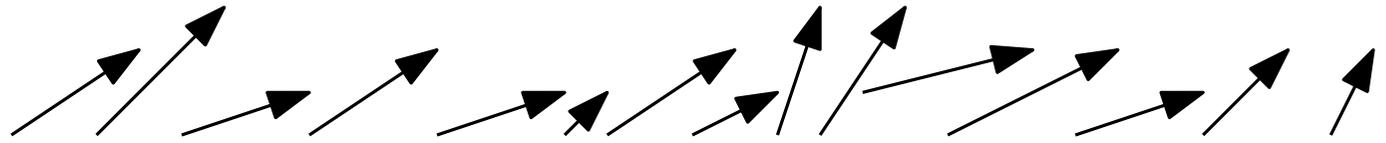
- Wert  $v_k \in \mathbb{R}^+$

- Gewichtung  $x_k \in \{0, 1\}$

- Erzeugungskapazität  $C \in \mathbb{R}^+$

$$\max \sum_{k=1}^n x_k v_k \text{ mit } \left| \sum_{k=1}^n x_k d_k \right| \leq C$$

- NP-vollständig



# Komplexes Knapsackproblem CKP

- Bedarf

- Größe  $d_k \in \mathbb{C}$

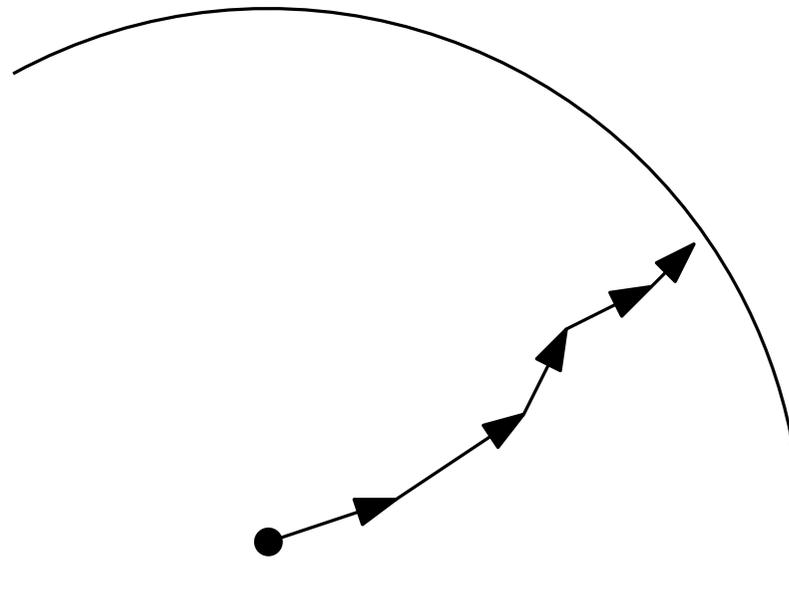
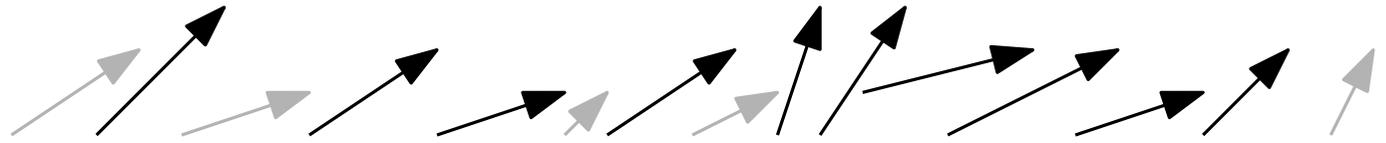
- Wert  $v_k \in \mathbb{R}^+$

- Gewichtung  $x_k \in \{0, 1\}$

- Erzeugungskapazität  $C \in \mathbb{R}^+$

$$\max \sum_{k=1}^n x_k v_k \text{ mit } \left| \sum_{k=1}^n x_k d_k \right| \leq C$$

- NP-vollständig



# Non-single-minded CKP

- Menge von  $n$  Nutzern  $\mathcal{N}$
- Menge der Bedarfe  $D \subseteq \mathbb{C}$
- Jeder Nutzer  $k$  hat eine monotone Bewertungsfunktion  $v_k : D \rightarrow \mathbb{R}^+$

# Non-single-minded CKP

- Menge von  $n$  Nutzern  $\mathcal{N}$
- Menge der Bedarfe  $D \subseteq \mathbb{C}$
- Jeder Nutzer  $k$  hat eine monotone Bewertungsfunktion  $v_k : D \rightarrow \mathbb{R}^+$

$$\max \sum_{k=1}^n v_k(d_k) \text{ mit } \left| \sum_{k=1}^n d_k \right| \leq C$$

# Non-single-minded CKP

- Menge von  $n$  Nutzern  $\mathcal{N}$
- Menge der Bedarfe  $D \subseteq \mathbb{C}$
- Jeder Nutzer  $k$  hat eine monotone Bewertungsfunktion  $v_k : D \rightarrow \mathbb{R}^+$

$$\max \sum_{k=1}^n v_k(d_k) \text{ mit } \left| \sum_{k=1}^n d_k \right| \leq C$$

Zusätzlich gilt für MULTICKP

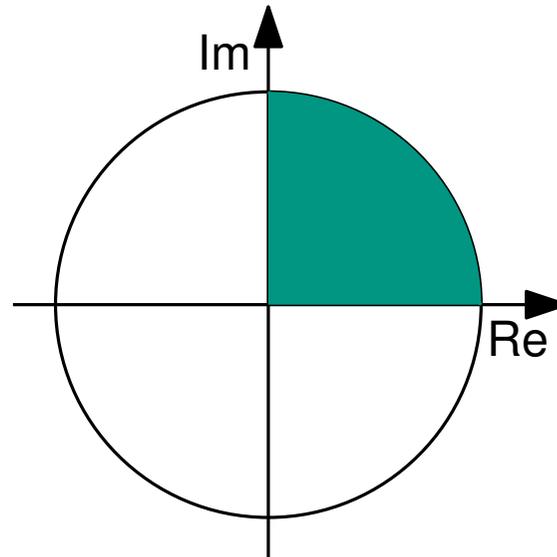
- Nutzer bekundet sein Interesse an einer Teilmenge  $D_k \subseteq D$
- $|D_k| \in \text{poly}(n)$

# PTAS für MULTICKP $[0, \frac{\pi}{2}]$

- $(1 - 3\epsilon, 1)$ -Approximation
  - Wert(Approximation)  $\geq (1 - 3\epsilon) \cdot$  Wert(Optimallösung)
  - Kapazitätsbedingung wird nicht verletzt
- Genauigkeitsparameter  $\epsilon$ 
  - $\epsilon < \frac{1}{4}$
  - $\frac{1}{\epsilon} \in \mathbb{Z}^+$
- Alle Bedarfe müssen im ersten Quadranten  $[0, \frac{\pi}{2}]$  liegen

# PTAS für MULTICKP $[0, \frac{\pi}{2}]$

- $(1 - 3\epsilon, 1)$ -Approximation
  - Wert(Approximation)  $\geq (1 - 3\epsilon) \cdot$  Wert(Optimallösung)
  - Kapazitätsbedingung wird nicht verletzt
- Genauigkeitsparameter  $\epsilon$ 
  - $\epsilon < \frac{1}{4}$
  - $\frac{1}{\epsilon} \in \mathbb{Z}^+$
- Alle Bedarfe müssen im ersten Quadranten  $[0, \frac{\pi}{2}]$  liegen

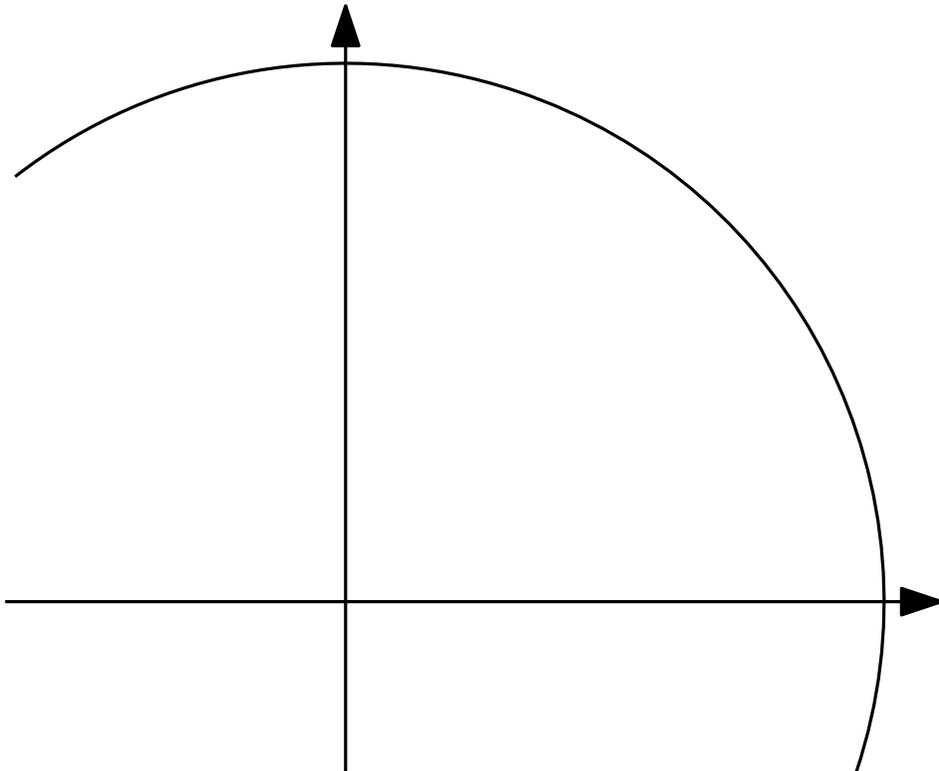


# PTAS für MULTICKP $[0, \frac{\pi}{2}]$

- Ansatz
  - Bestimmung eines ausreichend genau approximierenden Polygons
  - Problemdefinition im Polygon
  - Abbildung auf ein äquivalentes Problem

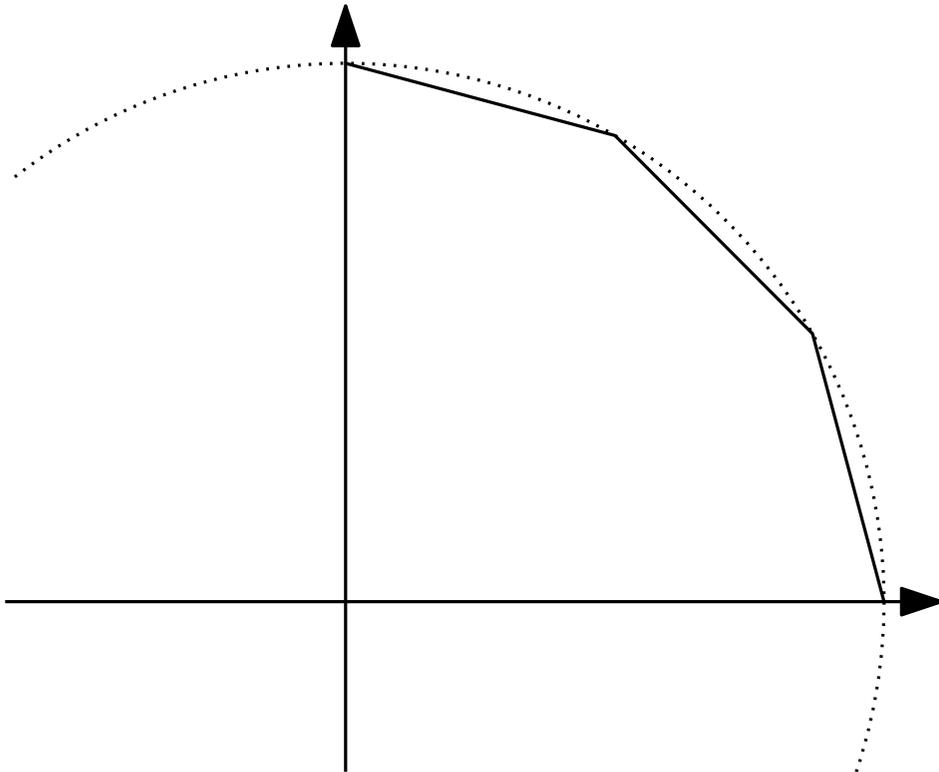
# PTAS für MULTICKP $[0, \frac{\pi}{2}]$

- Ansatz
  - Bestimmung eines ausreichend genau approximierenden Polygons
  - Problemdefinition im Polygon
  - Abbildung auf ein äquivalentes Problem



# PTAS für MULTICKP $[0, \frac{\pi}{2}]$

- Ansatz
  - Bestimmung eines ausreichend genau approximierenden Polygons
  - Problemdefinition im Polygon
  - Abbildung auf ein äquivalentes Problem



**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0);$   
**for** jede Teilmenge  $N \subseteq \mathcal{N}$  und  
jede Teilmenge der Bedarfe  $T$  die Nutzer in  $N$  haben,  
mit maximal  $\frac{1}{\epsilon}$  Elementen und  $|\sum_{d \in T} d| \leq C$  **do**  
     $d_T \leftarrow \sum_{d \in T} d;$   
    definiere die polygonisierte Region  
     $(d_1, \dots, d_n) \leftarrow \text{MULTI-mDKP-PTAS}[\text{MULTI-mDKP}\{\sigma_T^n\}]$   
    **if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$  **then**  
         $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n);$   
    **end**  
**end**  
return  $(\hat{d}_1, \dots, \hat{d}_n)$

**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$

$(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0);$

**for** jede Teilmenge  $N \subseteq \mathcal{N}$  und

jede Teilmenge der Bedarfe  $T$  die Nutzer in  $N$  haben,

mit maximal  $\frac{1}{\epsilon}$  Elementen und  $|\sum_{d \in T} d| \leq C$  **do**

$d_T \leftarrow \sum_{d \in T} d;$

definiere die polygonisierte Region

$(d_1, \dots, d_n) \leftarrow \text{MULTI-mDKP-PTAS}[\text{MULTI-mDKP}\{\sigma_T^n\}]$

**if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$  **then**

$(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n);$

**end**

**end**

**return**  $(\hat{d}_1, \dots, \hat{d}_n)$

**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0);$

**for** jede Teilmenge  $N \subseteq \mathcal{N}$  und  
jede Teilmenge der Bedarfe  $T$  die Nutzer in  $N$  haben,  
mit maximal  $\frac{1}{\epsilon}$  Elementen und  $|\sum_{d \in T} d| \leq C$  **do**

$d_T \leftarrow \sum_{d \in T} d;$

definiere die polygonisierte Region

$(d_1, \dots, d_n) \leftarrow \text{MULTI-mDKP-PTAS}[\text{MULTI-mDKP}\{\sigma_T^n\}]$

**if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$  **then**

$(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n);$

**end**

**end**

return  $(\hat{d}_1, \dots, \hat{d}_n)$

**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$ ;  
**for** jede Teilmenge  $N \subseteq \mathcal{N}$  und  
jede Teilmenge der Bedarfe  $T$  die Nutzer in  $N$  haben,  
mit maximal  $\frac{1}{\epsilon}$  Elementen und  $|\sum_{d \in T} d| \leq C$  **do**  
     $d_T \leftarrow \sum_{d \in T} d$ ;  
    definiere die polygonisierte Region  
     $(d_1, \dots, d_n) \leftarrow \text{MULTI-mDKP-PTAS}[\text{MULTI-mDKP}\{\sigma_T^n\}]$   
    **if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$  **then**  
         $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$ ;  
    **end**  
**end**  
return  $(\hat{d}_1, \dots, \hat{d}_n)$

# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$ ;  
**for** jede Teilmenge  $N \subseteq \mathcal{N}$   
jede Teilmenge der Beda  
mit maximal  $\frac{1}{\epsilon}$  Elemente

$$d_T \leftarrow \sum_{d \in T} d;$$

definiere die polygoni

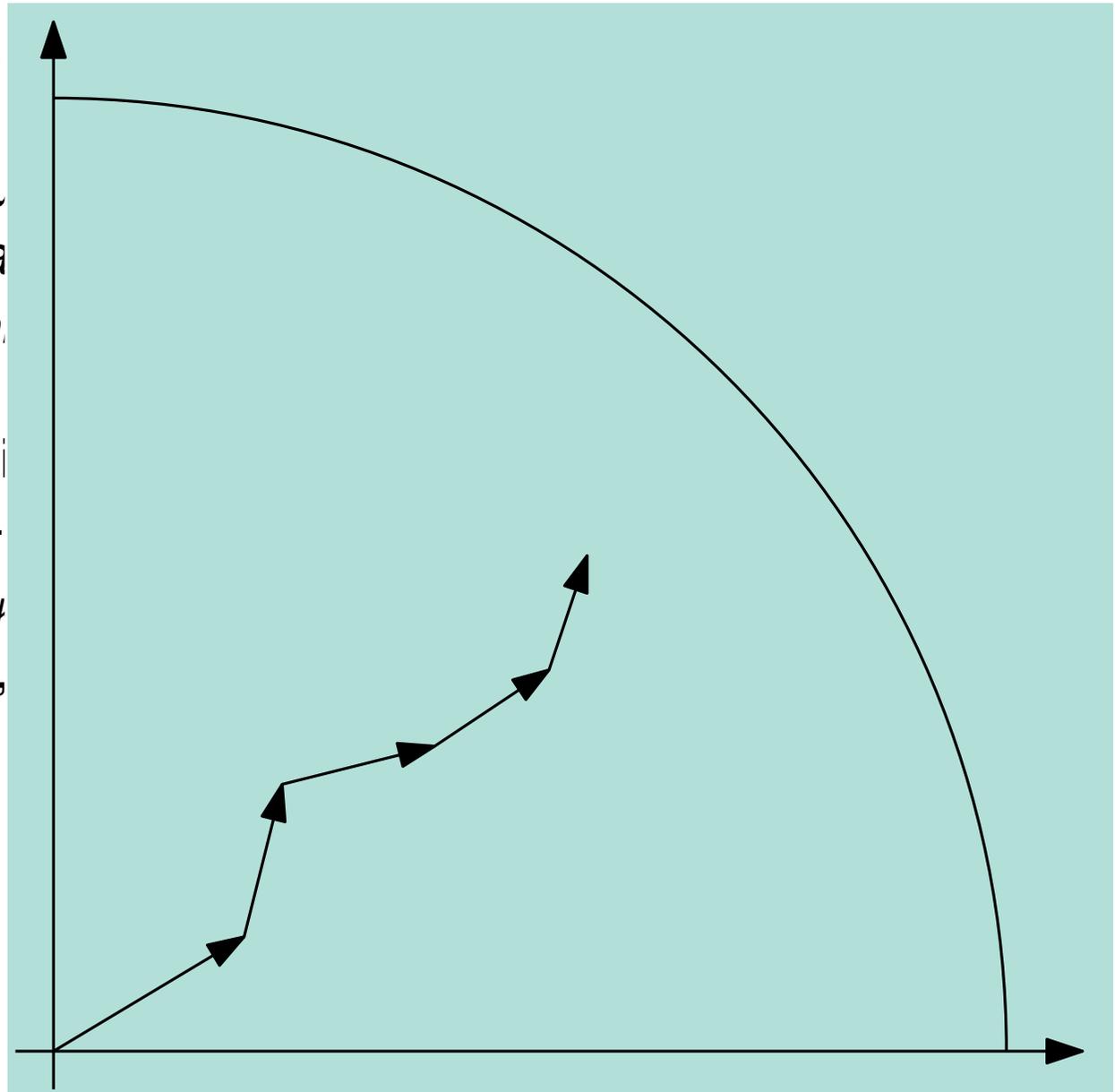
$$(d_1, \dots, d_n) \leftarrow \text{MULTI-}$$

**if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
|  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$

**end**

**end**

**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$ ;  
**for** jede Teilmenge  $N \subseteq \mathcal{N}$   
jede Teilmenge der Beda  
mit maximal  $\frac{1}{\epsilon}$  Elemente

$$d_T \leftarrow \sum_{d \in T} d;$$

definiere die polygoni

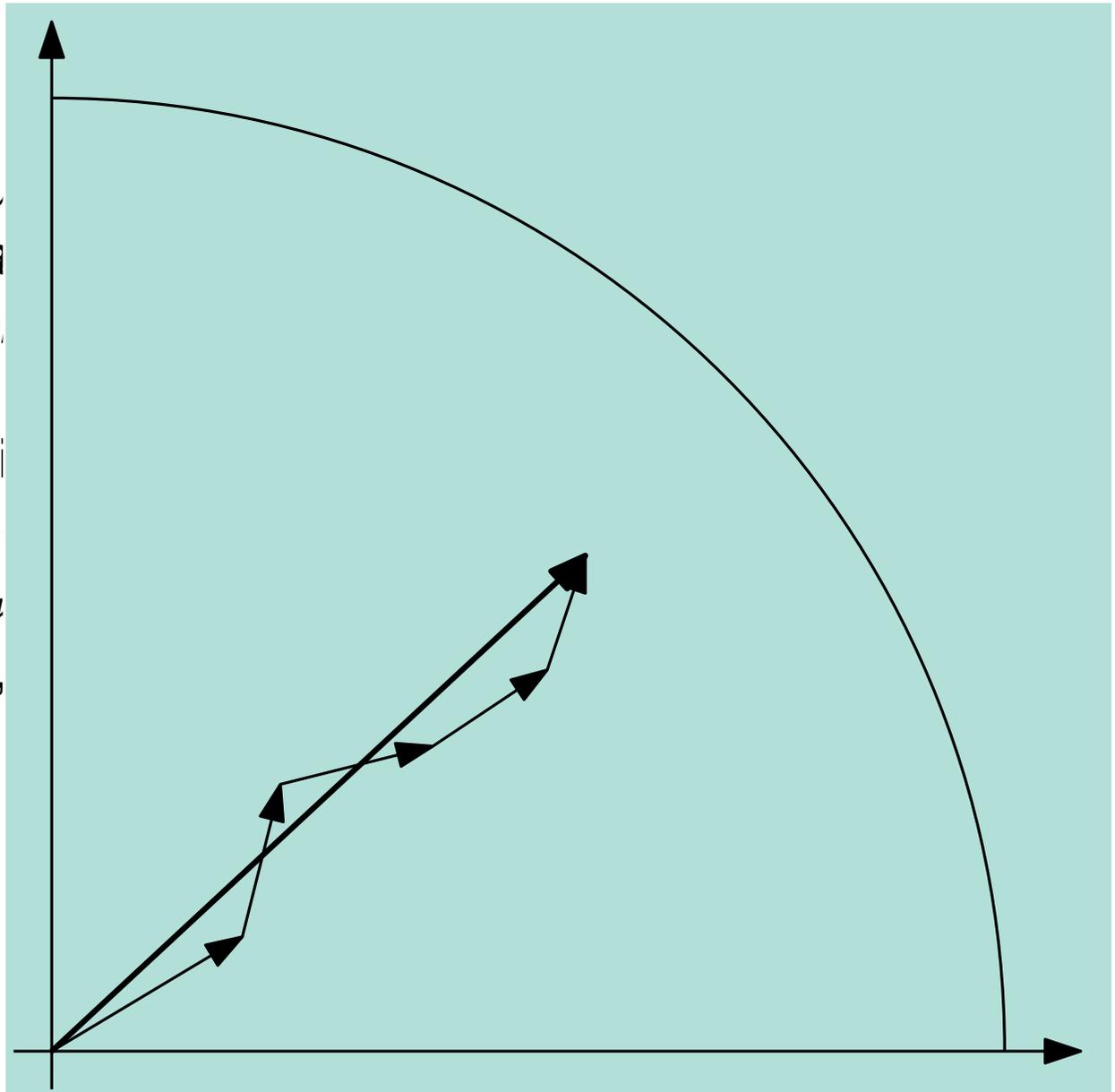
$$(d_1, \dots, d_n) \leftarrow \text{MULTI-}$$

**if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
|  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$

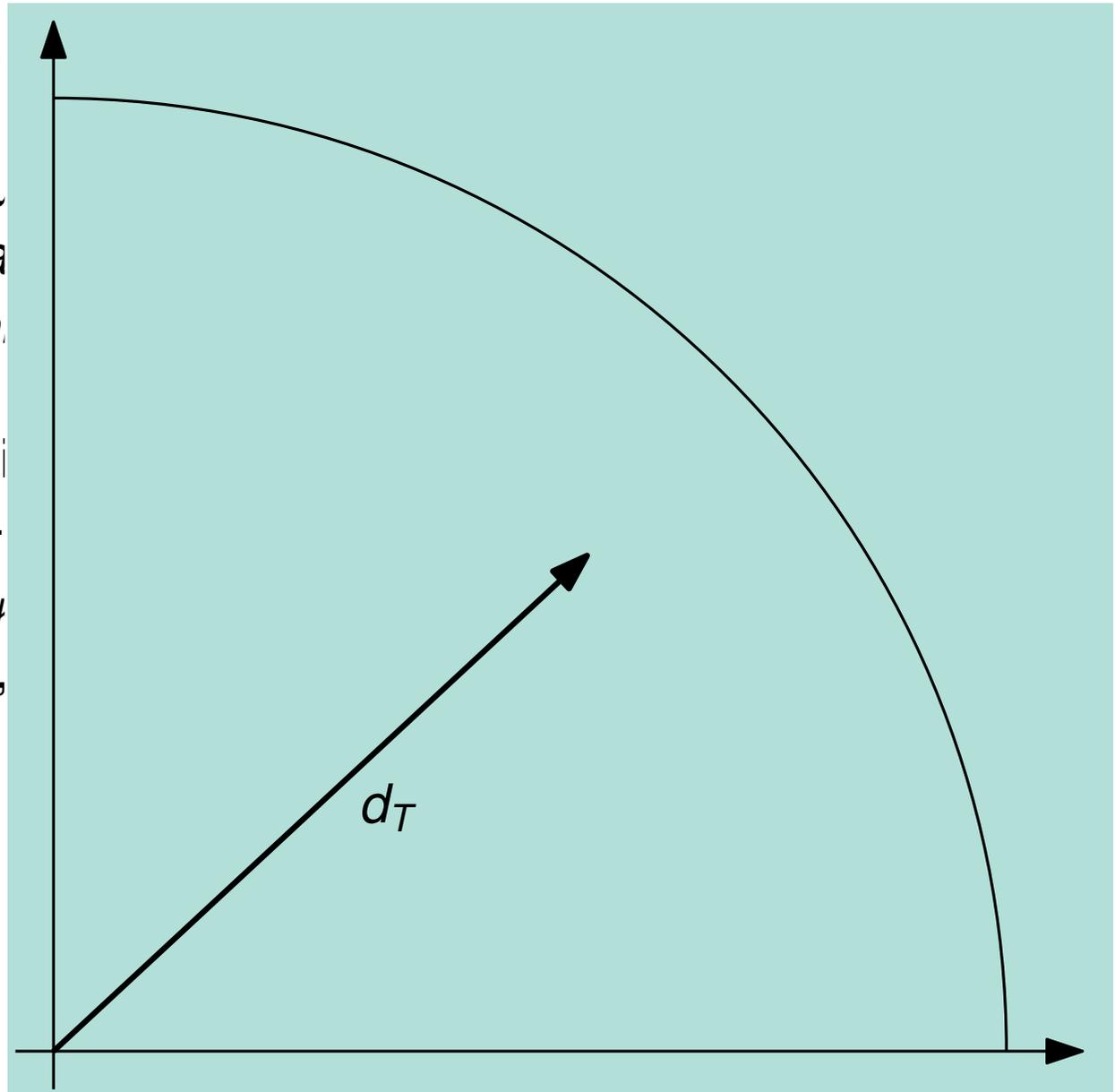
**end**

**end**

**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



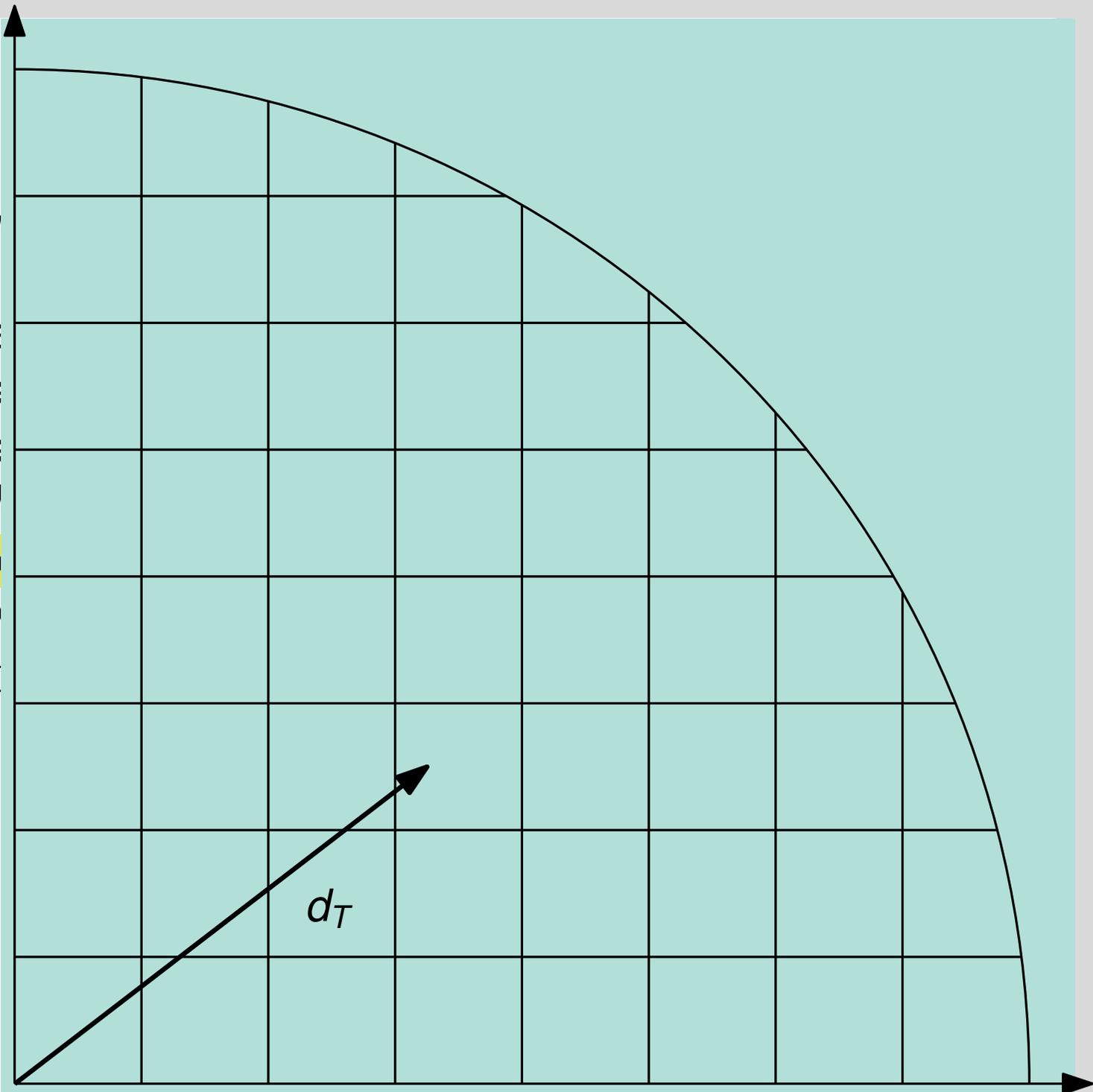
**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$ ;  
**for** jede Teilmenge  $N \subseteq \mathcal{N}$   
jede Teilmenge der Beda  
mit maximal  $\frac{1}{\epsilon}$  Elemente  
     $d_T \leftarrow \sum_{d \in T} d$ ;  
    definiere die polygoni  
     $(d_1, \dots, d_n) \leftarrow \text{MULTI-}$   
    **if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
    |  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
    **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0);$   
**for** jede Teilmenge  $N \subseteq \mathcal{N}$  und  
jede Teilmenge der Bedarfe  $T$  die Nutzer in  $N$  haben,  
mit maximal  $\frac{1}{\epsilon}$  Elementen und  $|\sum_{d \in T} d| \leq C$  **do**  
     $d_T \leftarrow \sum_{d \in T} d;$   
    **definiere die polygonisierte Region**  
     $(d_1, \dots, d_n) \leftarrow \text{MULTI-mDKP-PTAS}[\text{MULTI-mDKP}\{\sigma_T^n\}]$   
    **if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$  **then**  
         $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n);$   
    **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$

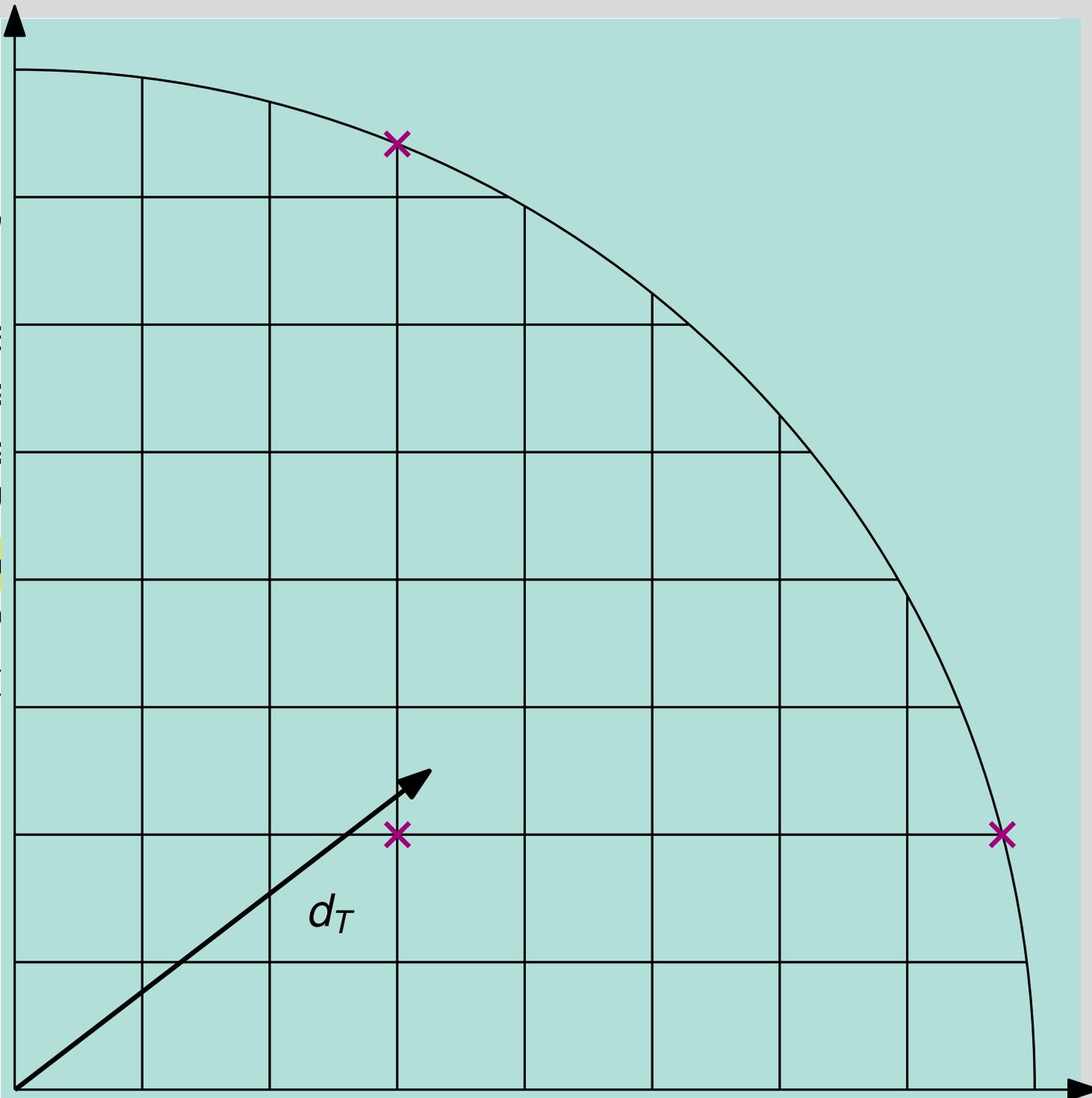
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T \subseteq J$   
  jede Teilmenge  $d \in T$   
  mit maximal  $\frac{1}{\epsilon}$  Elementen  
     $d_T \leftarrow \sum_{d \in T} d$   
    **definiere die**  $p$   
     $(d_1, \dots, d_n) \leftarrow M$   
    **if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
       $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
    **end**  
  **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



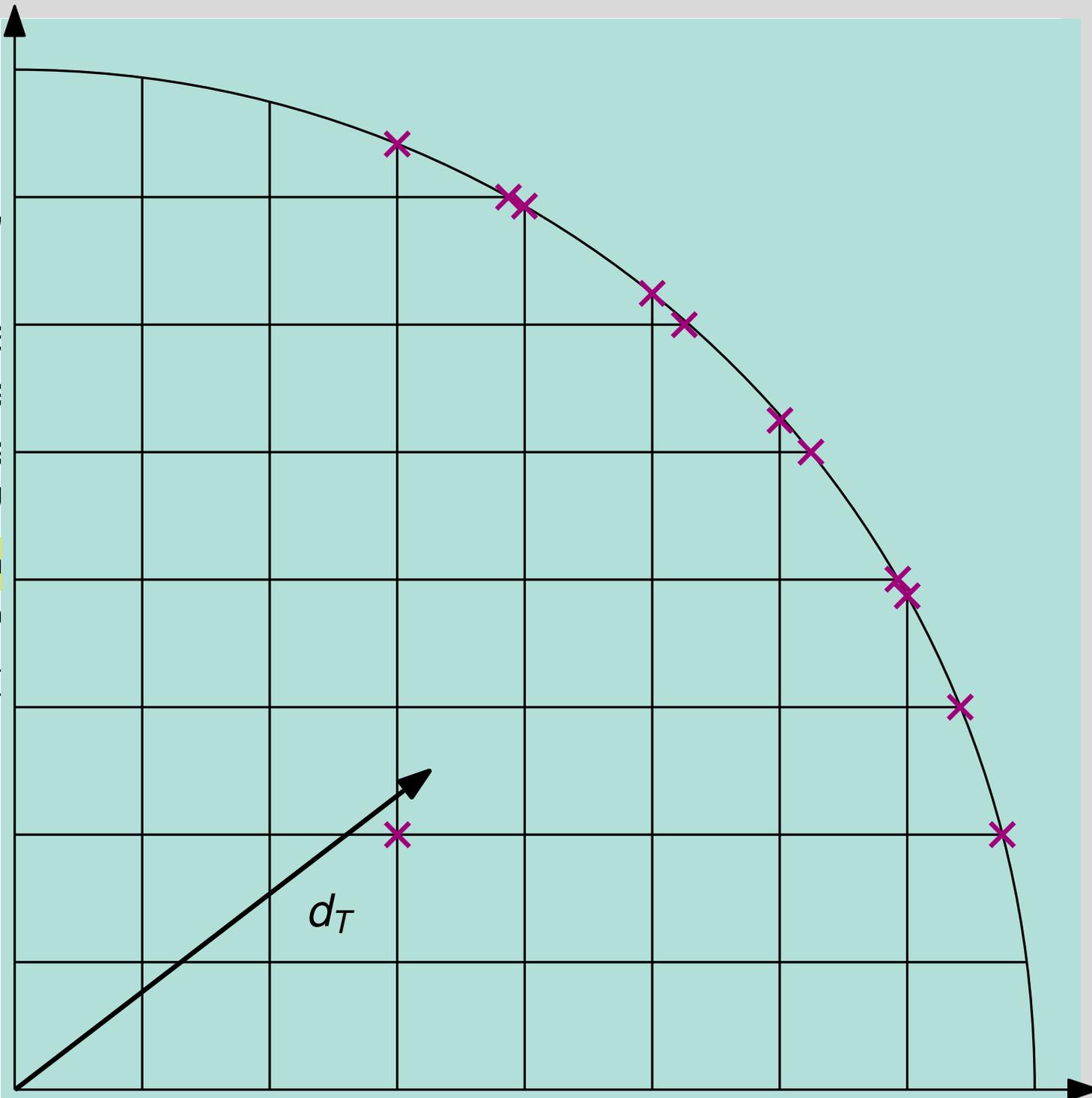
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T$  der  $J$   
jede Teilmenge  $d \in T$   
mit maximal  $\frac{1}{\epsilon}$  Elementen  
|  $d_T \leftarrow \sum_{d \in T} d$   
| **definiere die**  $p$   
|  $(d_1, \dots, d_n) \leftarrow \mathcal{M}(d_T)$   
| **if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
| |  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
| **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



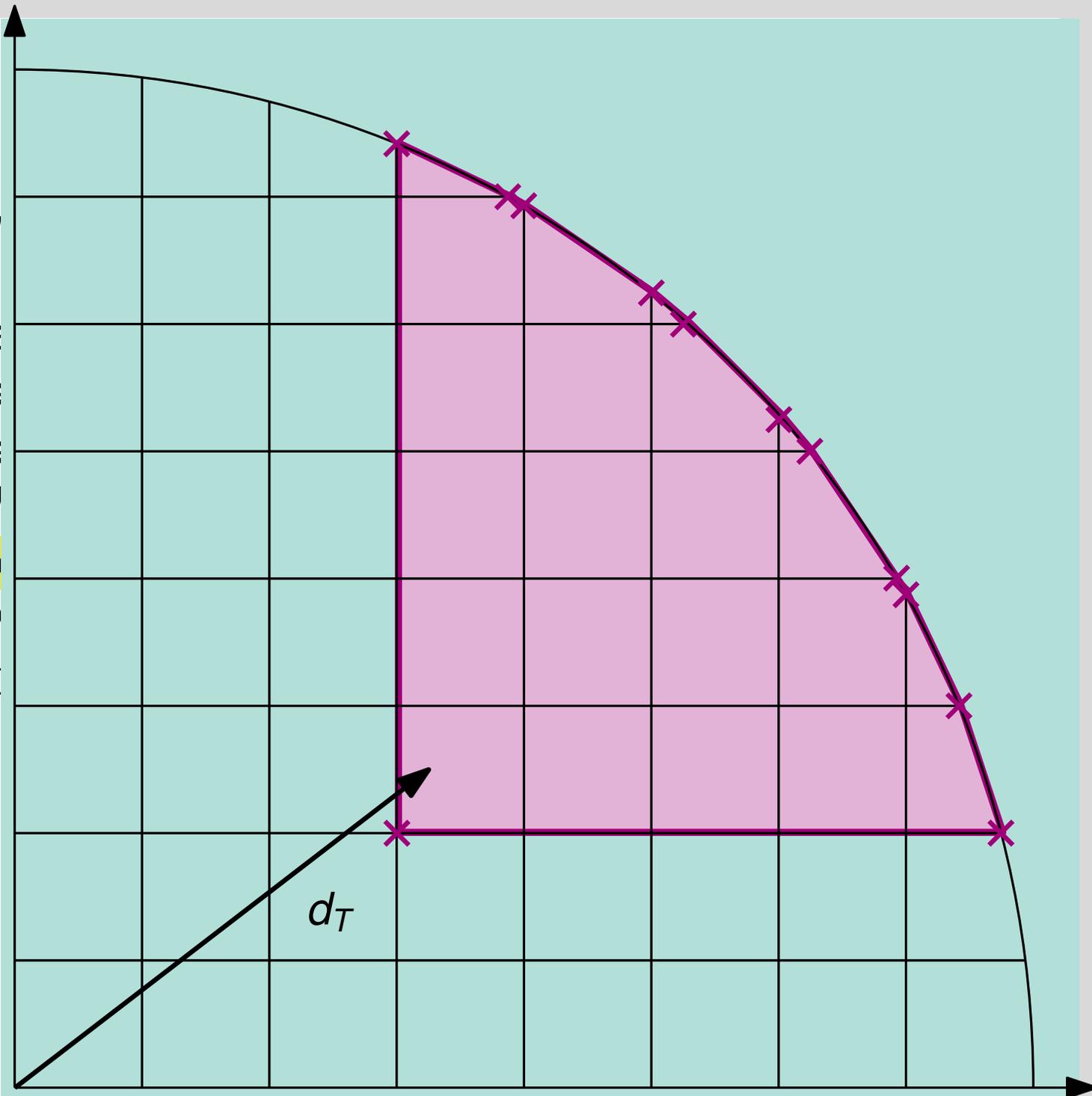
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T \subseteq J$   
 jede Teilmenge  $d \in T$   
 mit maximal  $\frac{1}{\epsilon}$  Elementen  
 $d_T \leftarrow \sum_{d \in T} d$   
 definiere die p...  
 $(d_1, \dots, d_n) \leftarrow \dots$   
**if**  $\sum_k v_k(d_k) < \dots$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow \dots$   
**end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



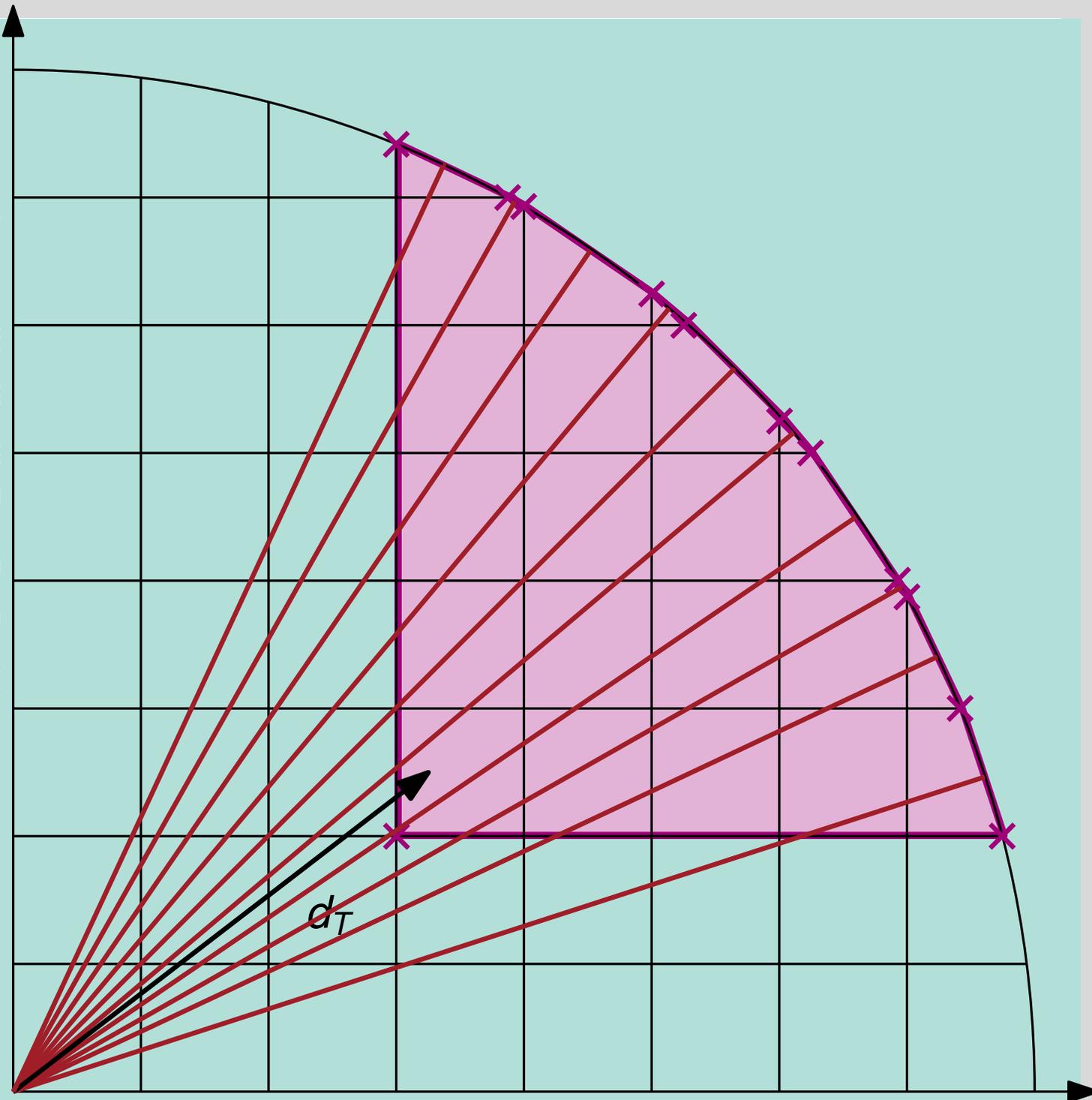
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T$  der  $J$   
jede Teilmenge  $d \in T$   
mit maximal  $\frac{1}{\epsilon}$  Elementen  
|  $d_T \leftarrow \sum_{d \in T} d$   
| **definiere die p**  
|  $(d_1, \dots, d_n) \leftarrow \mathbf{1}$   
| **if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
| |  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
| **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



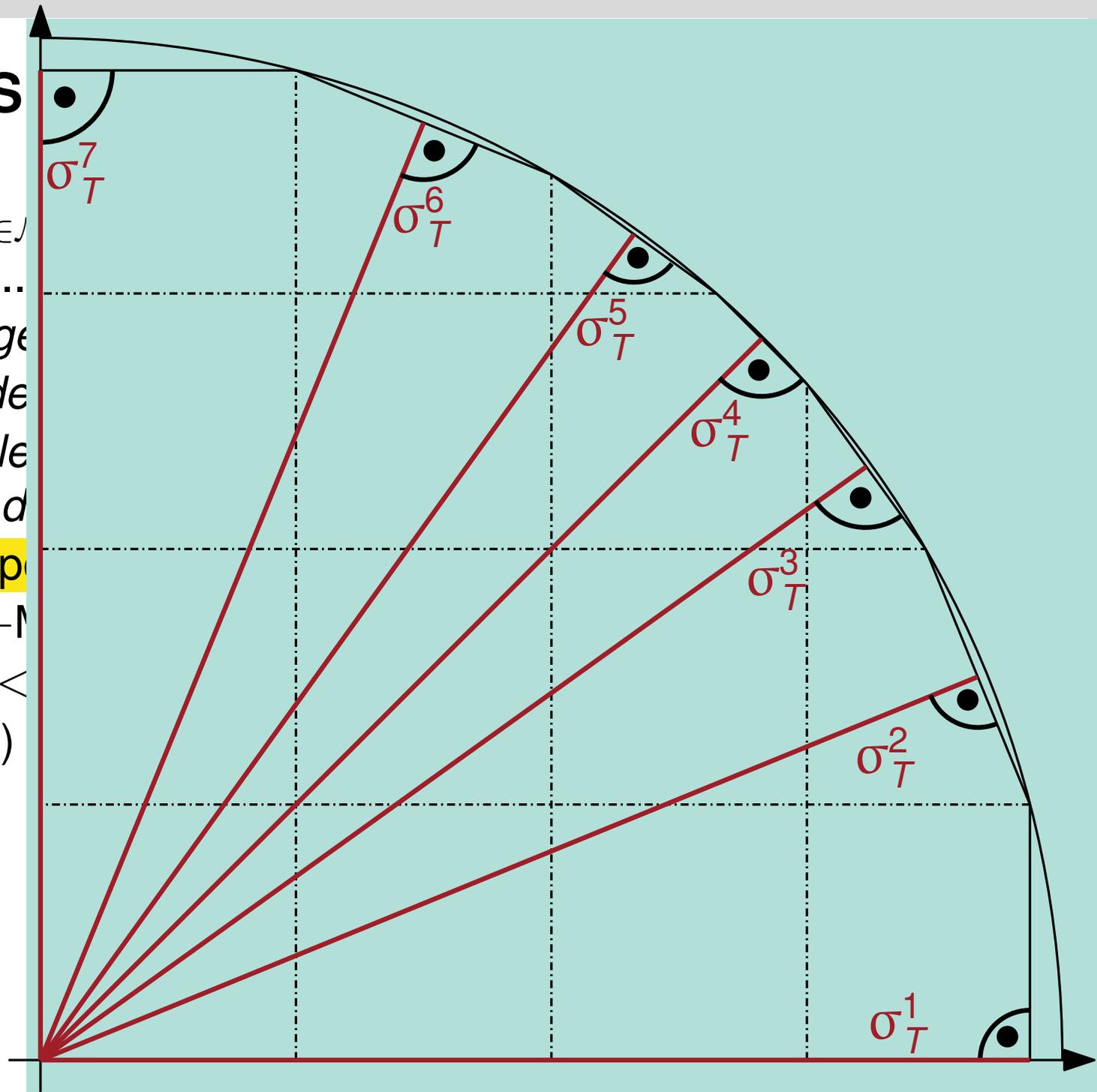
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T$  der  $J$   
 jede Teilmenge  $d \in T$   
 mit maximal  $\frac{1}{\epsilon}$  Elementen  
 |  $d_T \leftarrow \sum_{d \in T} d$   
 | **definiere die p**  
 |  $(d_1, \dots, d_n) \leftarrow \mathbf{1}$   
 | **if**  $\sum_k v_k(d_k) < v(d_T)$   
 | |  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
 | **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



# MULTICKP-PTAS

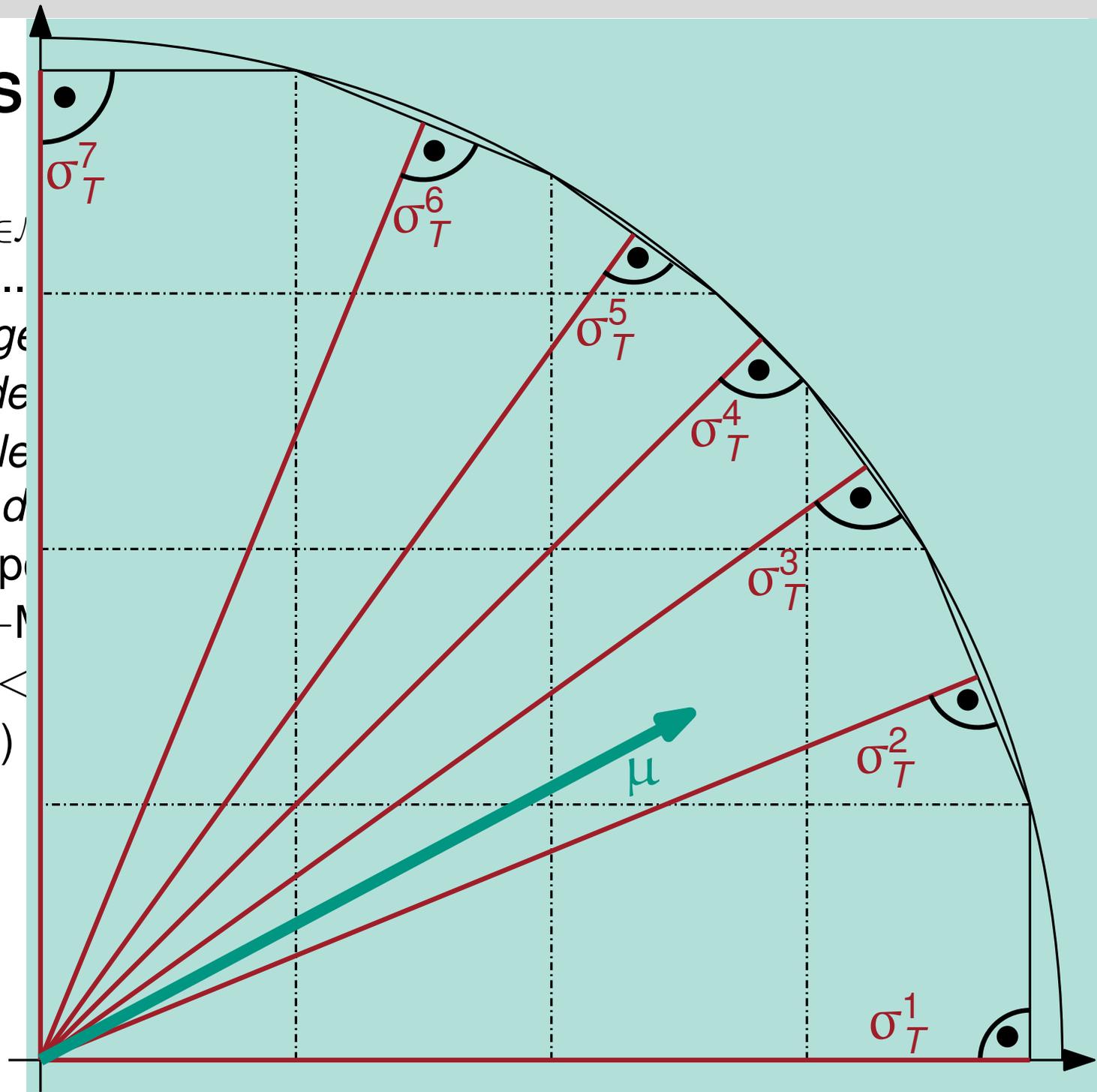
**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T$  von  $J$   
 jede Teilmenge  $d \in T$  mit maximal  $\frac{1}{\epsilon}$  Elementen  
      $d_T \leftarrow \sum_{d \in T} d$   
     **definiere die**  $p$   
      $(d_1, \dots, d_n) \leftarrow \mathcal{M}(d_T)$   
     **if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
          $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
     **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



**Input:**  $\{v_k, D_k\}_{k \in \mathcal{N}}, C, \epsilon$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$ ;  
**for** jede Teilmenge  $N \subseteq \mathcal{N}$  und  
jede Teilmenge der Bedarfe  $T$  die Nutzer in  $N$  haben,  
mit maximal  $\frac{1}{\epsilon}$  Elementen und  $|\sum_{d \in T} d| \leq C$  **do**  
     $d_T \leftarrow \sum_{d \in T} d$ ;  
    definiere die polygonisierte Region  
     $(d_1, \dots, d_n) \leftarrow \text{MULTI-mDKP-PTAS}[\text{MULTI-mDKP}\{\sigma_T^n\}]$   
    **if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$  **then**  
         $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$ ;  
    **end**  
**end**  
return  $(\hat{d}_1, \dots, \hat{d}_n)$

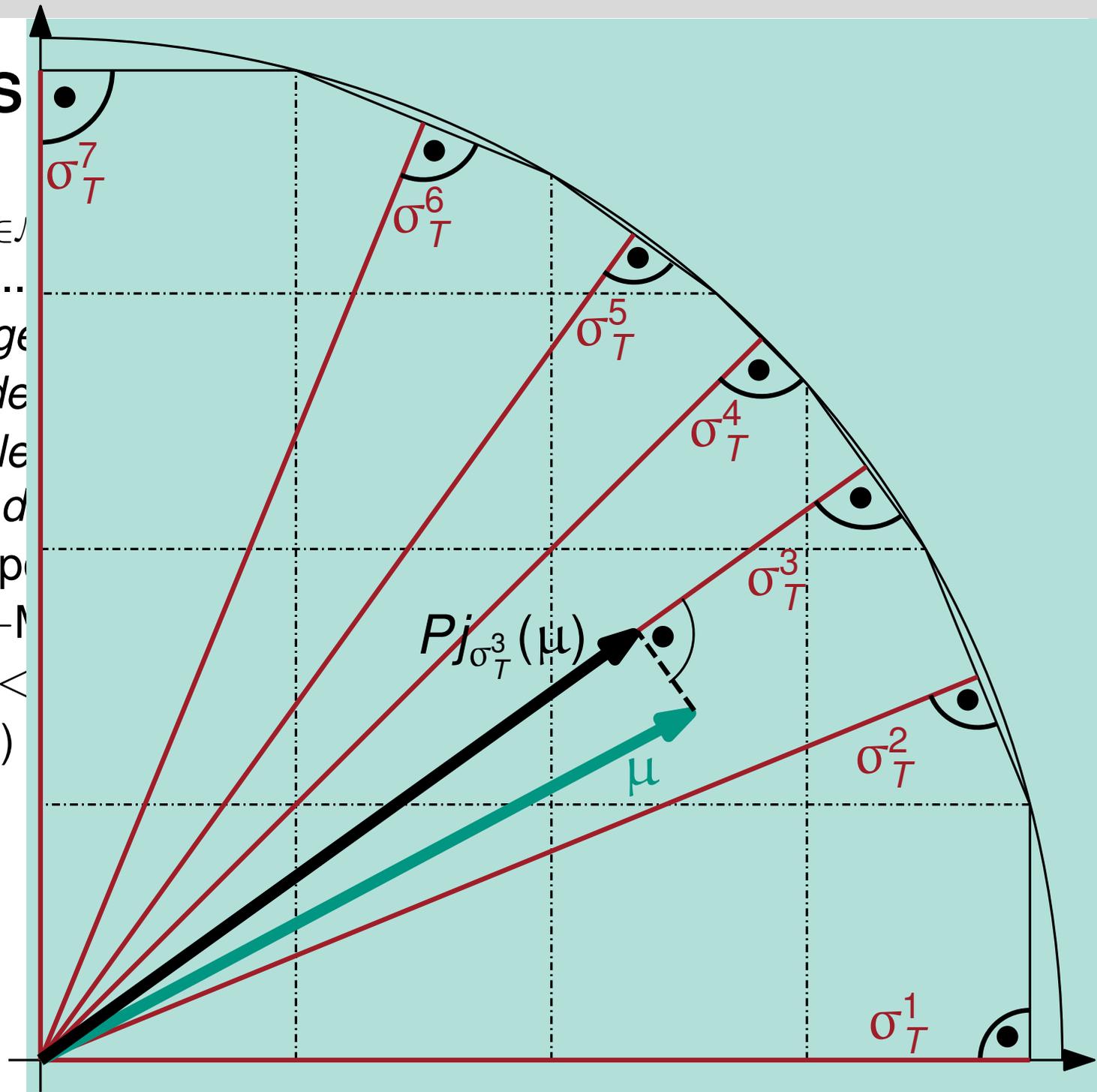
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T \subseteq J$   
 jede Teilmenge  $d \in T$   
 mit maximal  $\frac{1}{\epsilon}$  Elementen  
 |  $d_T \leftarrow \sum_{d \in T} d$   
 | definiere die partielle Instanz  
 |  $(d_1, \dots, d_n) \leftarrow \mathcal{M}(d_T)$   
 | **if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$   
 | |  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
 | **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



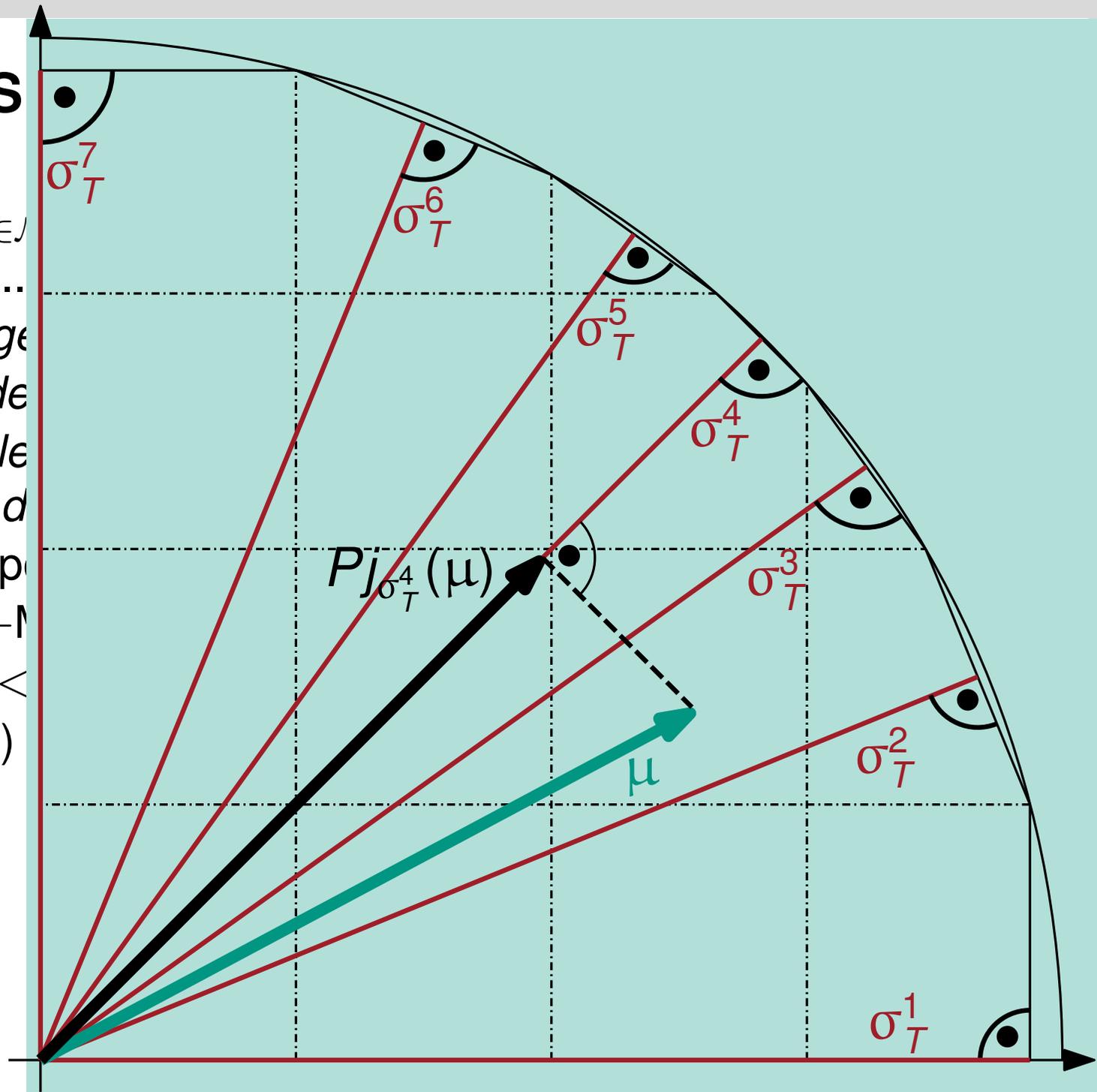
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T \subseteq J$   
 jede Teilmenge  $d \in T$   
 mit maximal  $\frac{1}{\epsilon}$  Elementen  
 |  $d_T \leftarrow \sum_{d \in T} d$   
 | definiere die projektive Projektion  
 |  $(d_1, \dots, d_n) \leftarrow P_{j_{\sigma_T^3}(\mu)}$   
 | **if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
 | |  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
 | **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



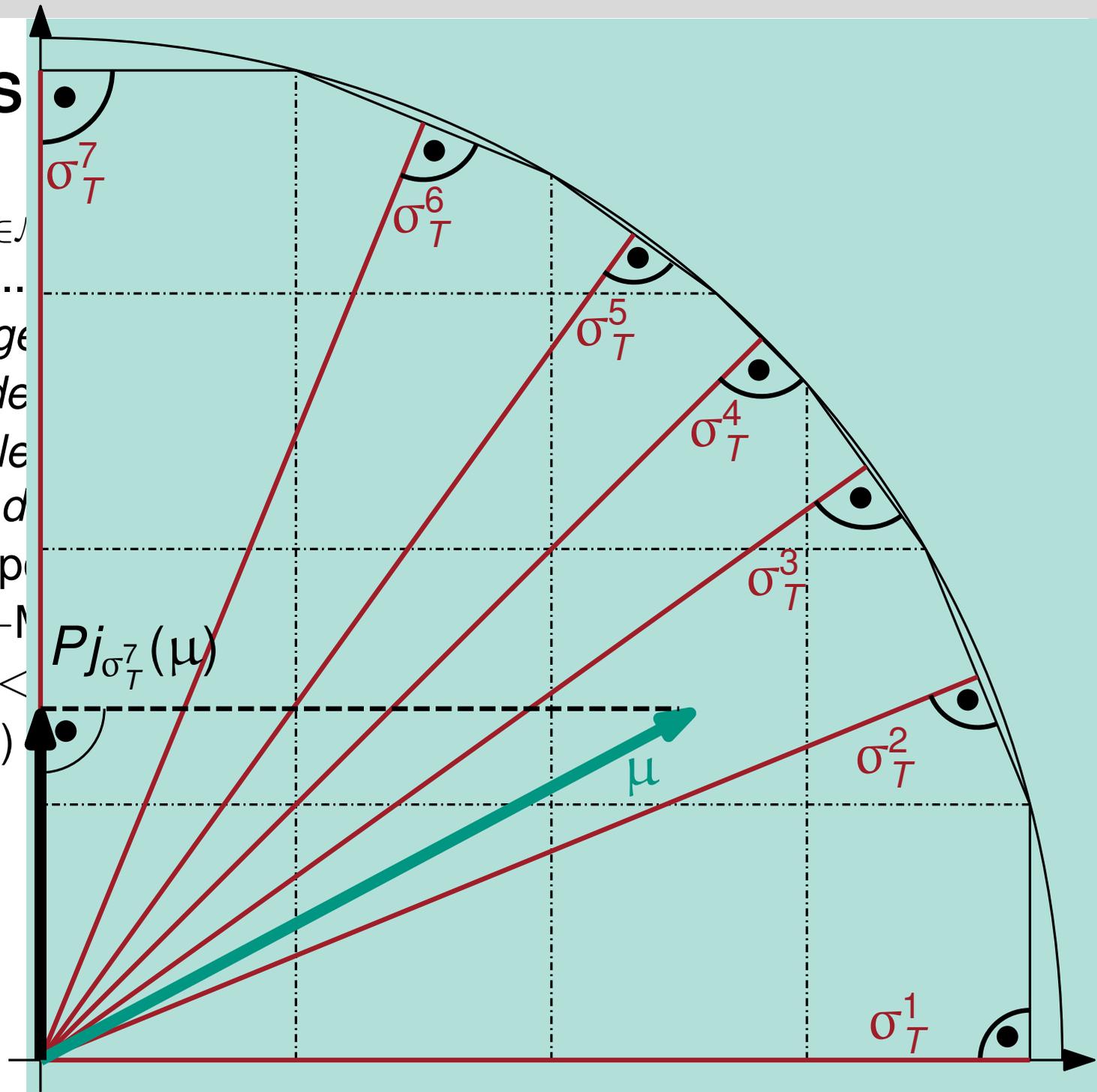
# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T \subseteq J$   
 jede Teilmenge  $d \in T$   
 mit maximal  $\frac{1}{\epsilon}$  Elementen  
 |  $d_T \leftarrow \sum_{d \in T} d$   
 | definiere die projektive Abbildung  
 |  $(d_1, \dots, d_n) \leftarrow P_{J \setminus T}(d_T)$   
 | **if**  $\sum_k v_k(d_k) < \sum_k v_k(\hat{d}_k)$   
 | |  $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
 | **end**  
**end**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



# MULTICKP-PTAS

**Input:**  $\{v_k, D_k\}_{k \in J}$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (0, \dots, 0)$   
**for** jede Teilmenge  $T$  der  $J$   
**do** jede Teilmenge  $d \in T$  mit maximal  $\frac{1}{\epsilon}$  Elementen  
 $d_T \leftarrow \sum_{d \in T} d$   
 definiere die partielle Funktion  $P_{j_{\sigma_T^7}}(\mu)$   
 $(d_1, \dots, d_n) \leftarrow \text{argmax}_{(d_1, \dots, d_n) \in \mathcal{D}} P_{j_{\sigma_T^7}}(\mu)$   
**if**  $\sum_k v_k(\hat{d}_k) < \sum_k v_k(d_k)$   
 $(\hat{d}_1, \dots, \hat{d}_n) \leftarrow (d_1, \dots, d_n)$   
**end if**  
**end do**  
**end for**  
**return**  $(\hat{d}_1, \dots, \hat{d}_n)$



# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$

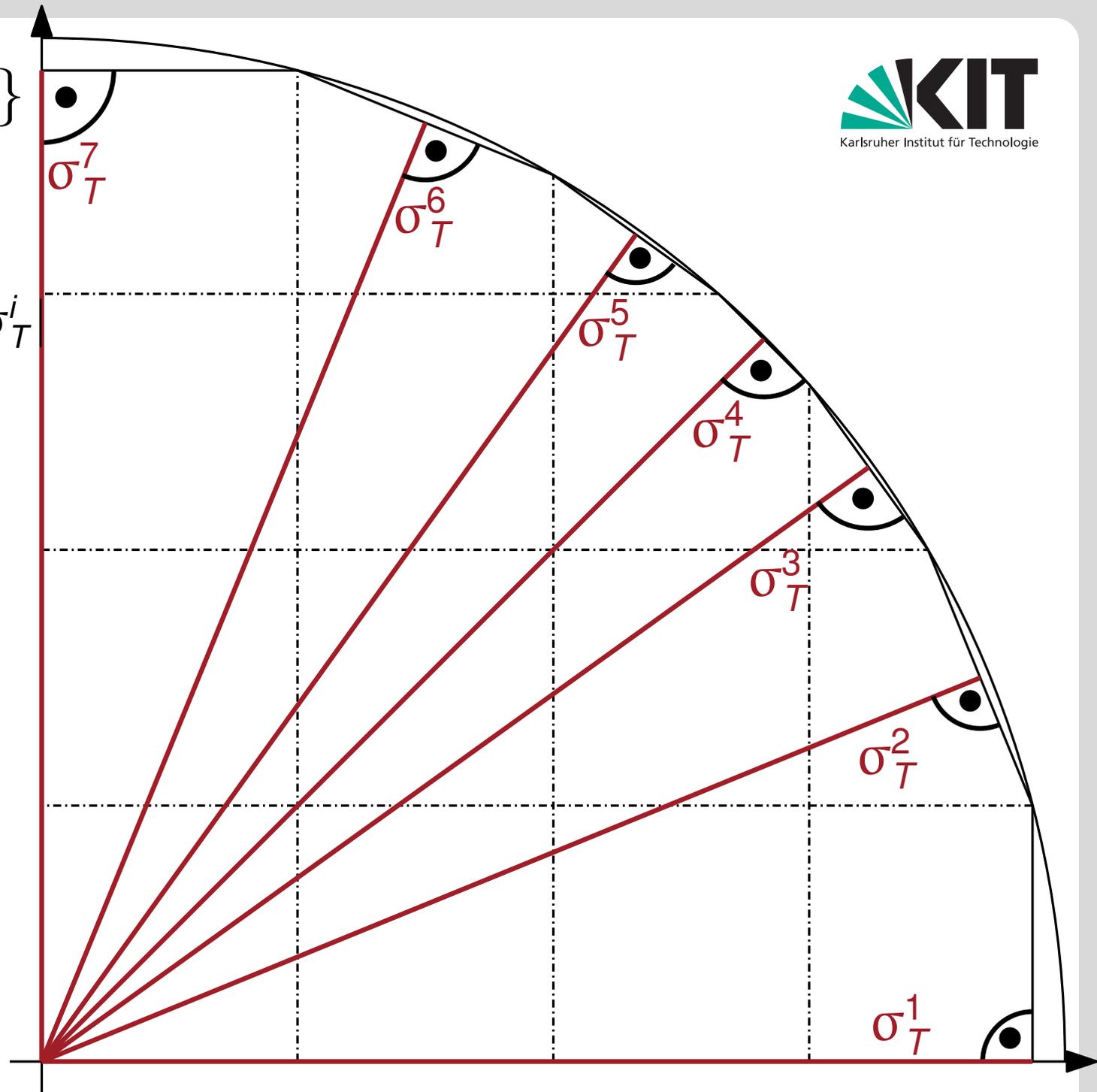
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



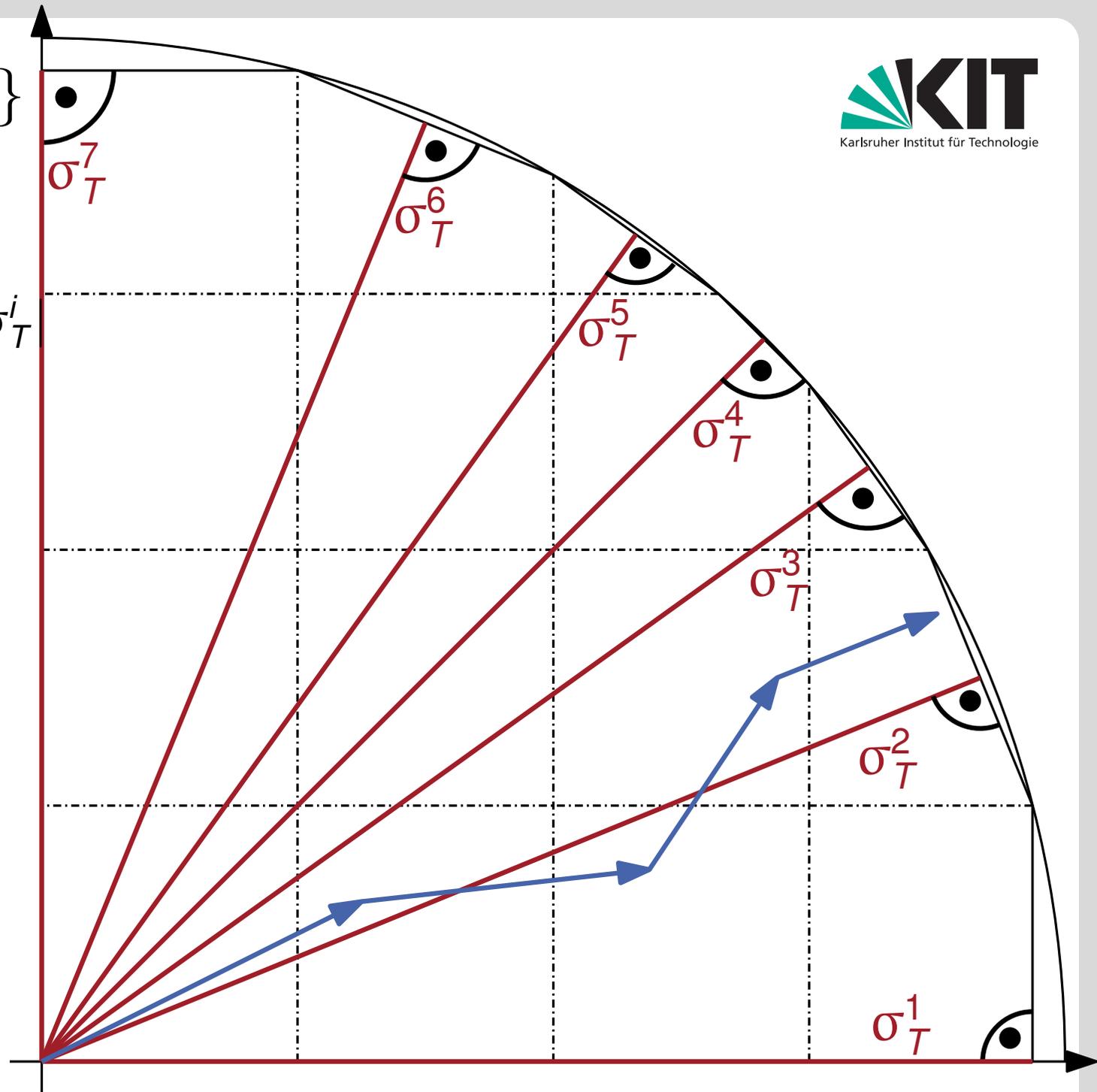
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



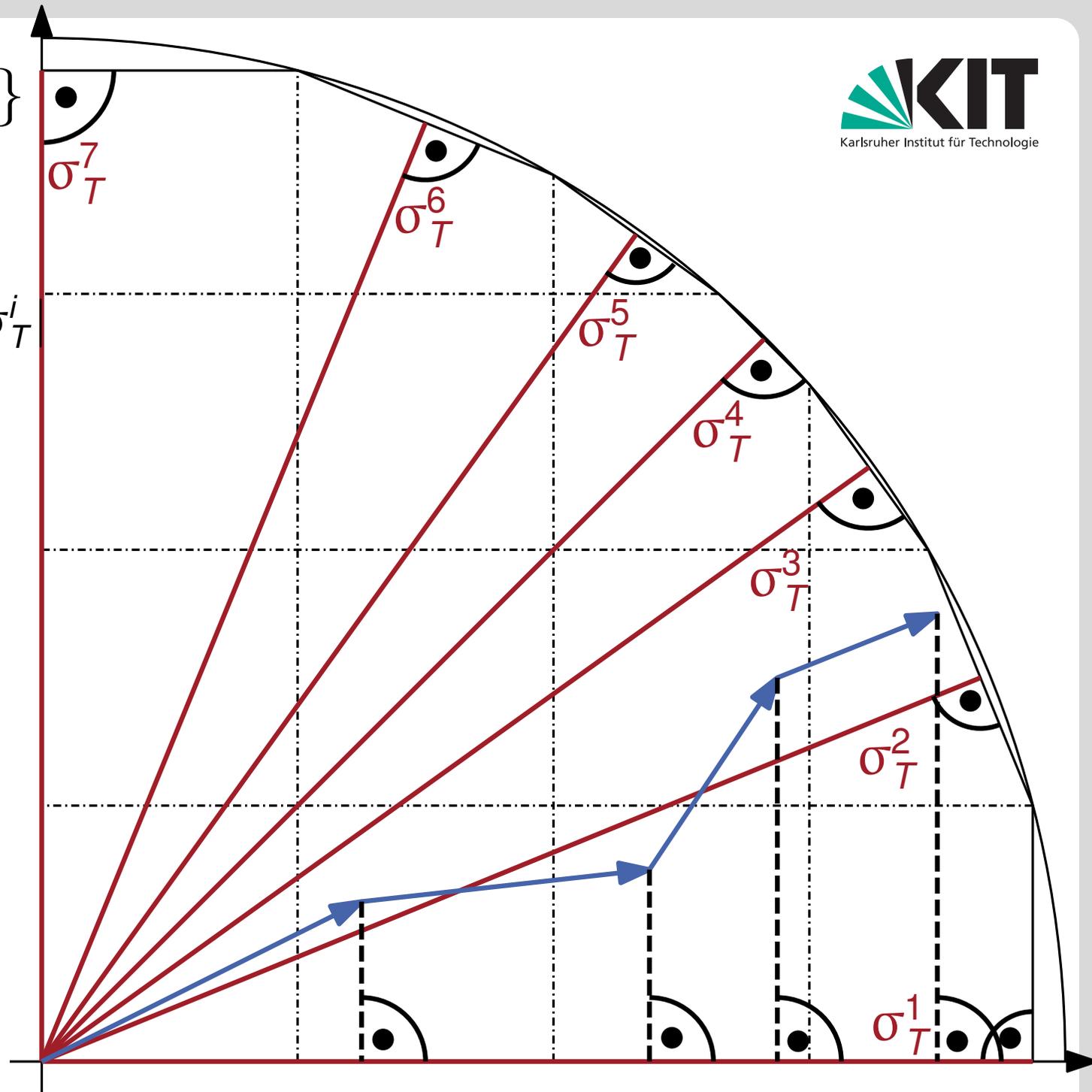
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



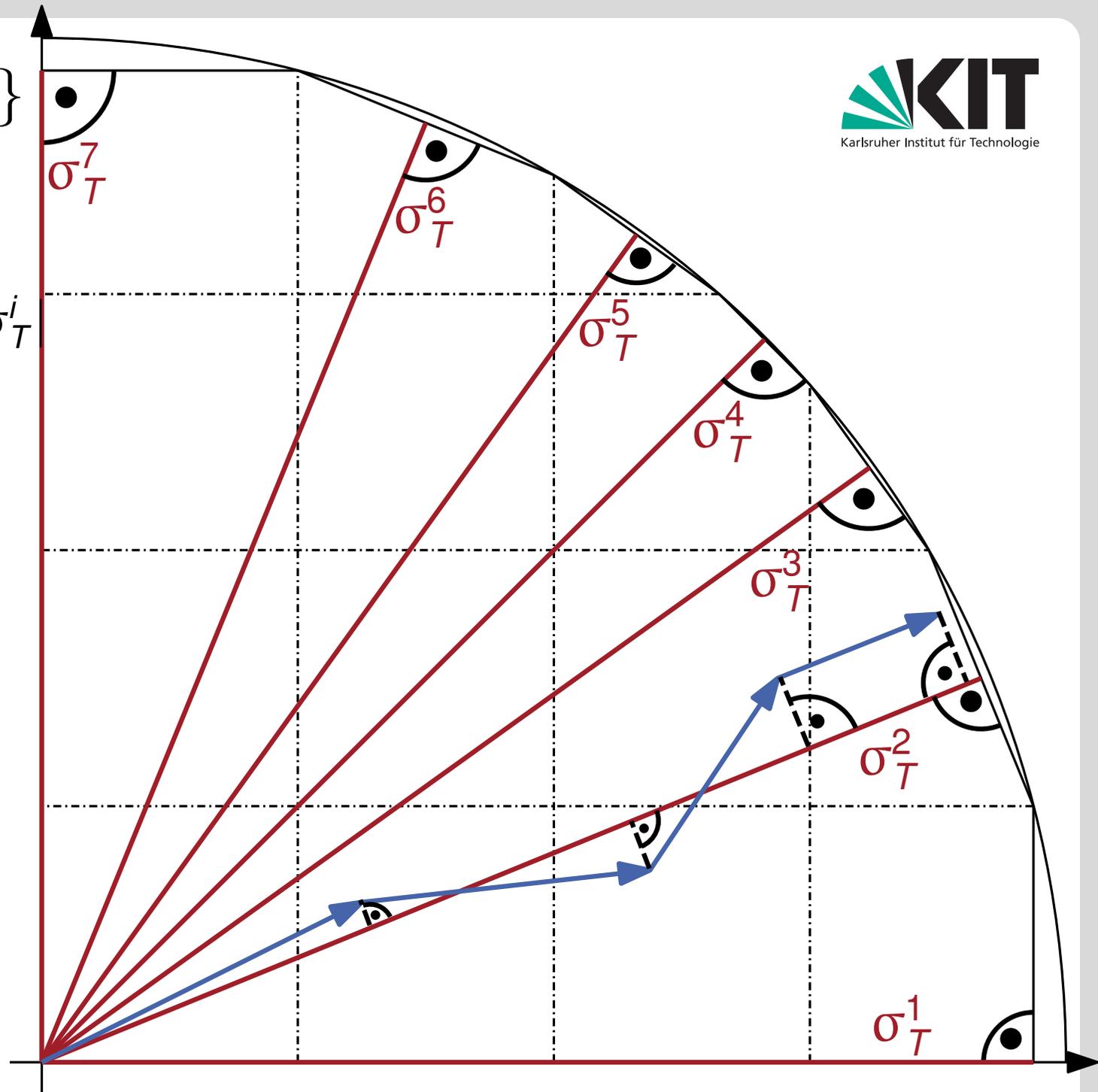
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



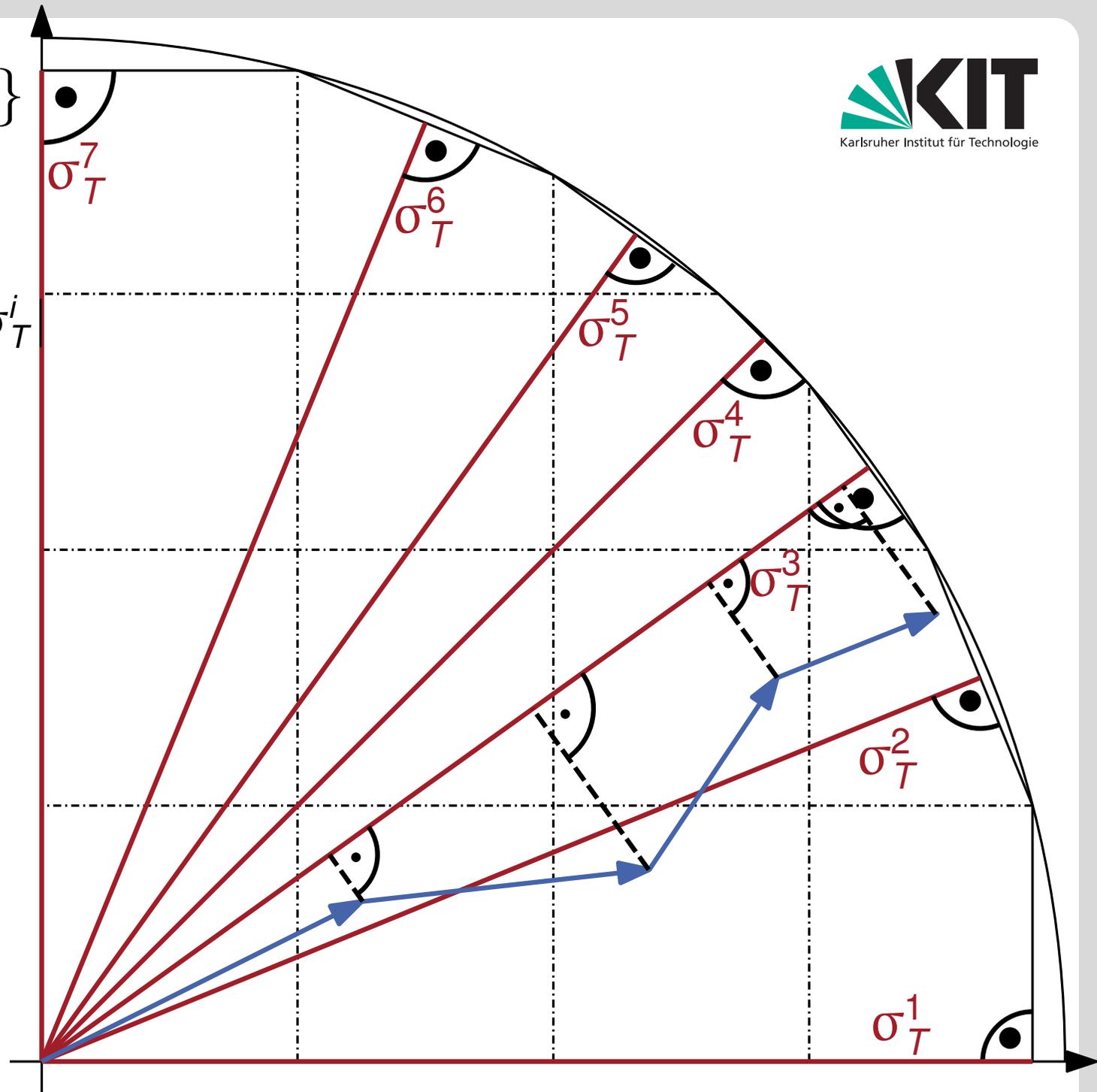
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



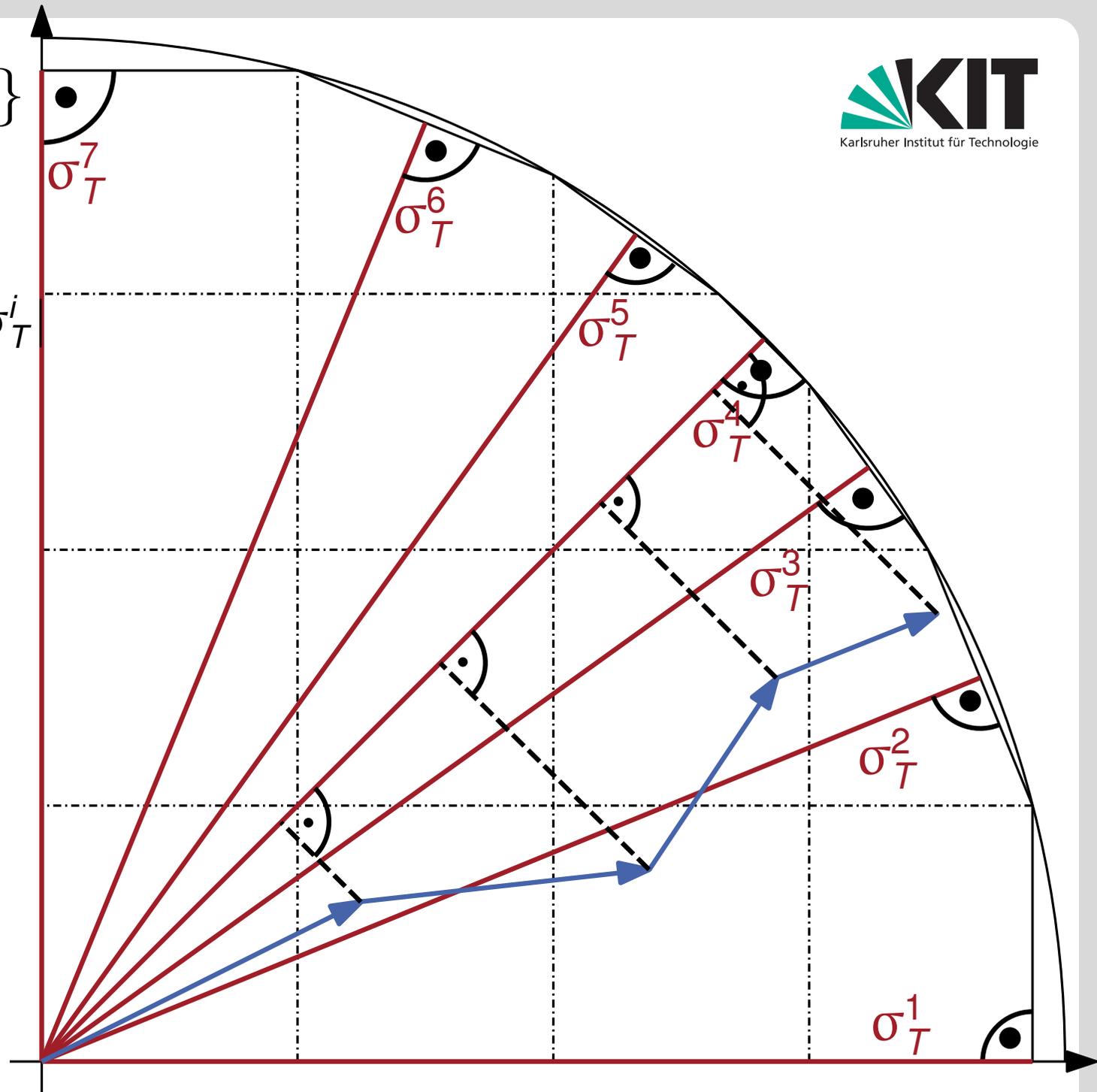
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



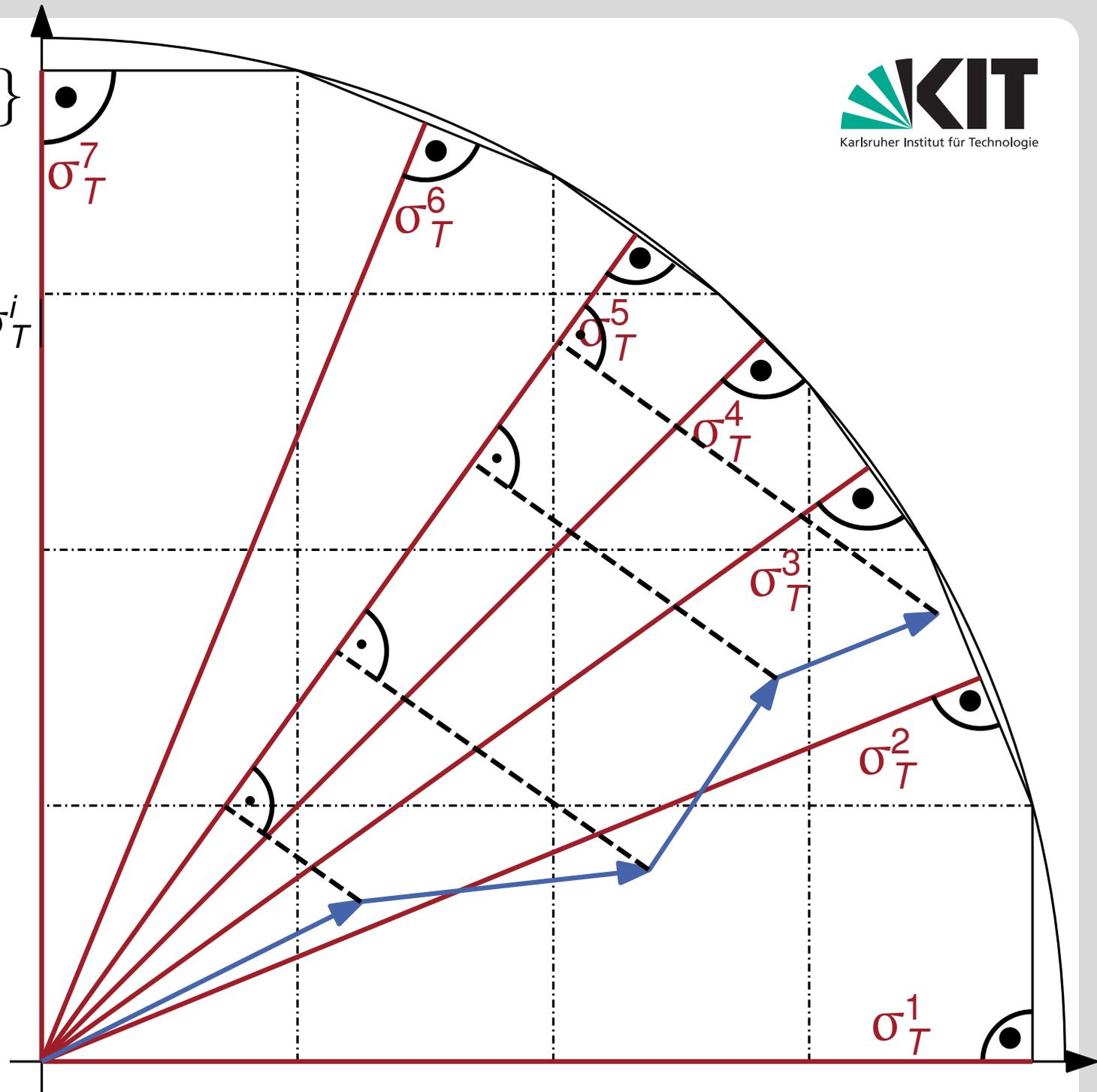
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



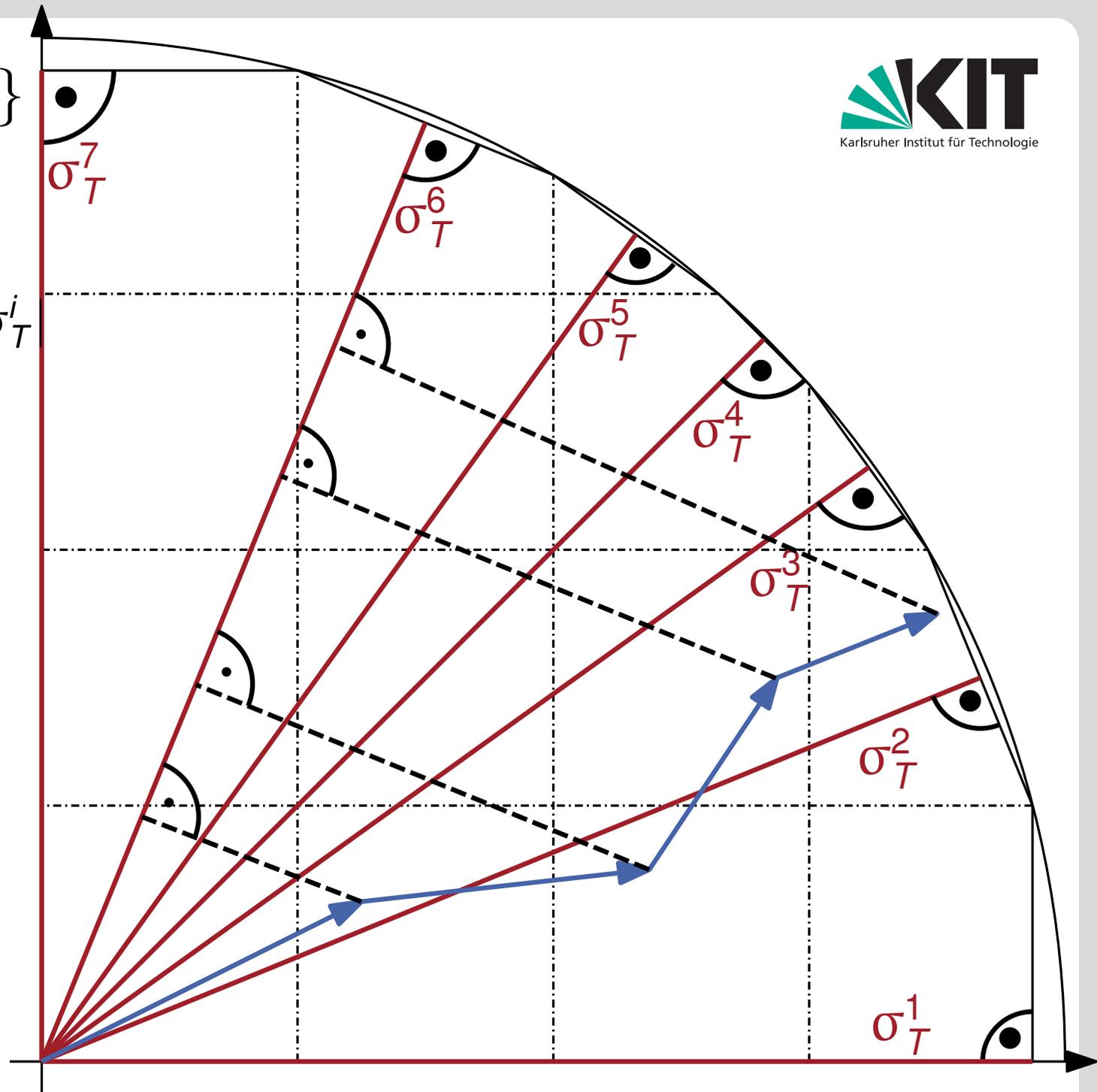
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



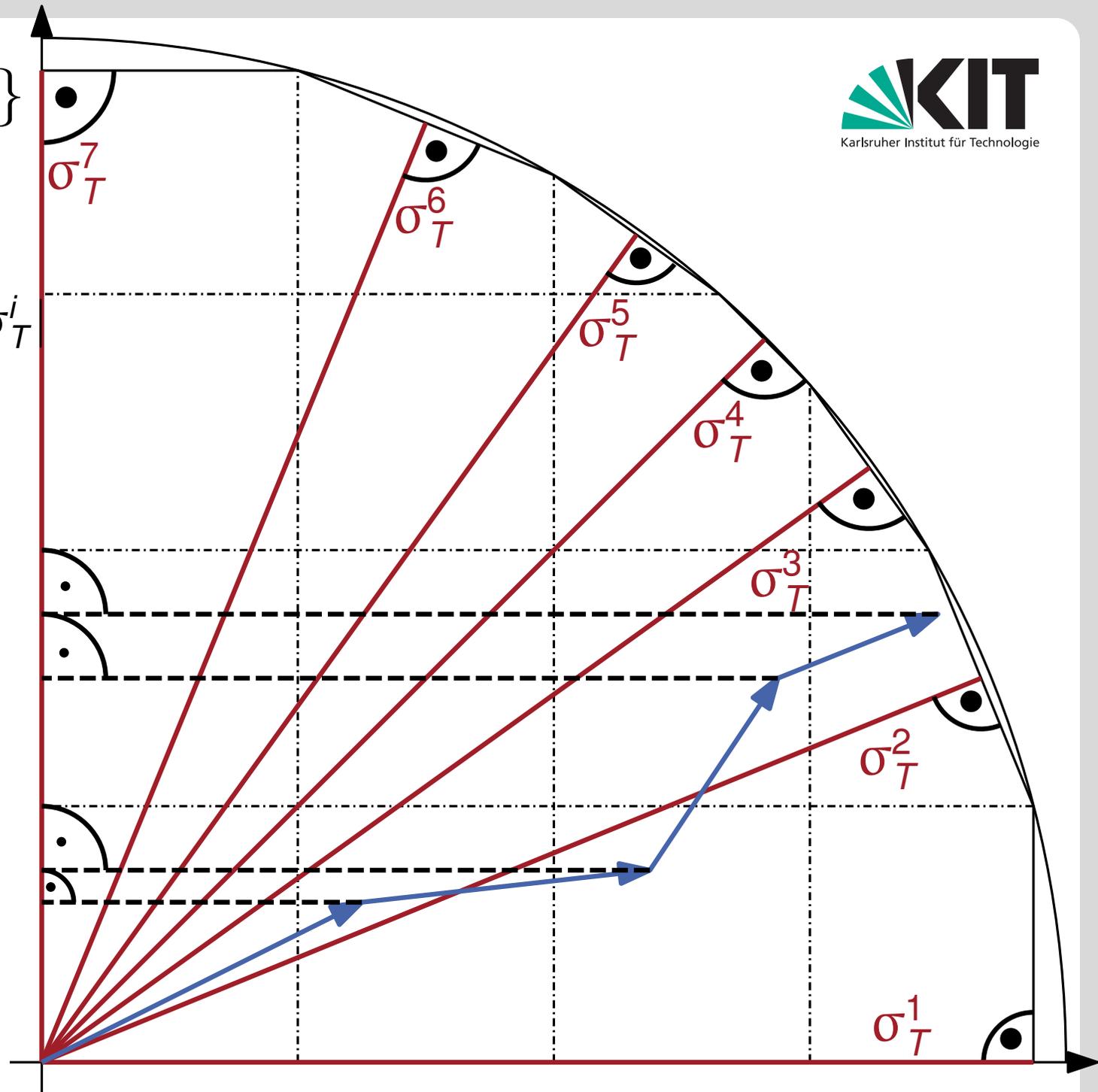
# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

$\forall$  Orthogonalen  $\sigma_T^i$



# MULTI-mDKP $\{\sigma_T^i\}$

$$\max \sum_{k \in \mathcal{N}} v_k(d_k)$$

mit

$$\sum_{k \in \mathcal{N}} |P_{j_{\sigma_T^i}}(d_k)| \leq |\sigma_T^i|$$

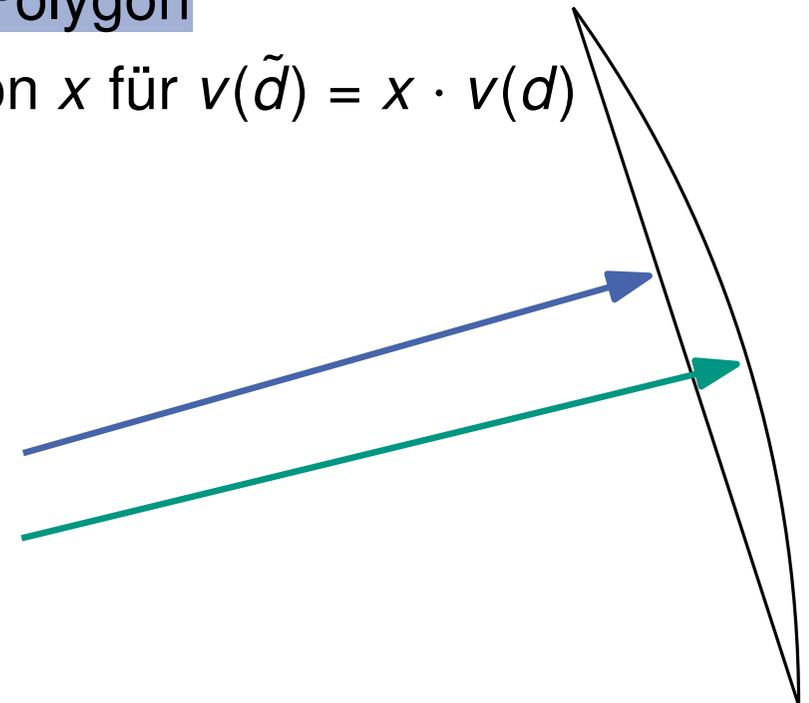
$\forall$  Orthogonalen  $\sigma_T^i$

- Mehrere eindimensionale Knapsackprobleme
- Äquivalentes Problem existiert bereits in [Frieze, Clarke 1984]
  - $(1 - \epsilon, 1)$ -Approximation dazu in [Frieze, Clarke 1984]

- MULTIm-DKP-PTAS liefert eine  $(1 - \epsilon, 1)$ -Approximation  
 $\Rightarrow (1 - \epsilon, 1)$ -Approximation innerhalb des Polygons
- Genauigkeitsverlust zwischen Kreis und Polygon muss ermittelt werden
- Vorgehen
  - Gegeben: Gültige Allokation  $d = (d_1, \dots, d_n)$  für MULTICKP
  - Finden einer Approximation  $\tilde{d}$  zu  $d$  im Polygon
  - Bestimmung des schlechtesten Falls von  $x$  für  $v(\tilde{d}) = x \cdot v(d)$

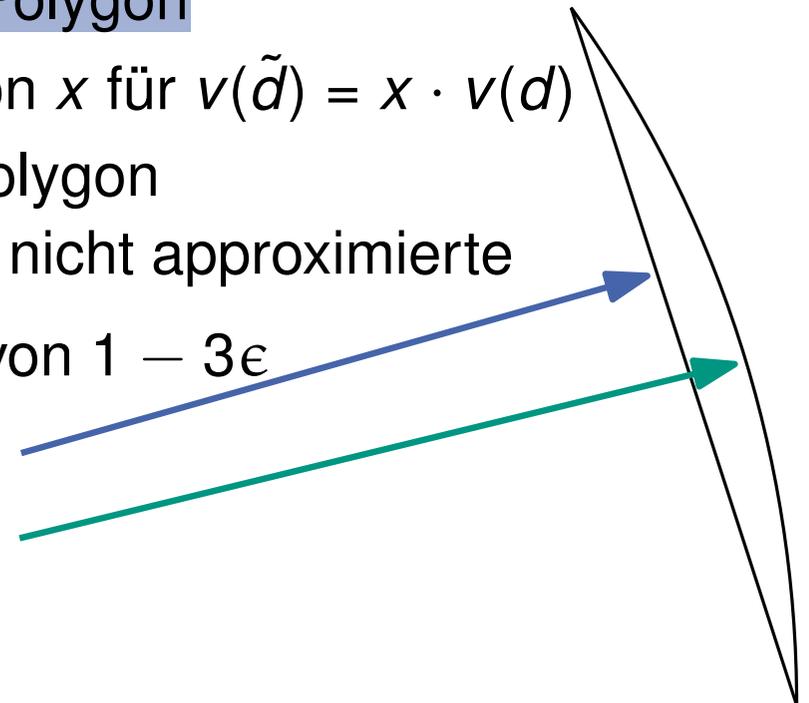
# Genauigkeitsbestimmung

- MULTIm-DKP-PTAS liefert eine  $(1 - \epsilon, 1)$ -Approximation  
=>  $(1 - \epsilon, 1)$ -Approximation innerhalb des Polygons
- Genauigkeitsverlust zwischen Kreis und Polygon muss ermittelt werden
- Vorgehen
  - Gegeben: Gültige Allokation  $d = (d_1, \dots, d_n)$  für MULTICKP
  - Finden einer Approximation  $\tilde{d}$  zu  $d$  im Polygon
  - Bestimmung des schlechtesten Falls von  $x$  für  $v(\tilde{d}) = x \cdot v(d)$



# Genauigkeitsbestimmung

- MULTIm-DKP-PTAS liefert eine  $(1 - \epsilon, 1)$ -Approximation  
 $\Rightarrow (1 - \epsilon, 1)$ -Approximation innerhalb des Polygons
- Genauigkeitsverlust zwischen Kreis und Polygon muss ermittelt werden
- Vorgehen
  - Gegeben: Gültige Allokation  $d = (d_1, \dots, d_n)$  für MULTICKP
  - Finden einer Approximation  $\tilde{d}$  zu  $d$  im Polygon
  - Bestimmung des schlechtesten Falls von  $x$  für  $v(\tilde{d}) = x \cdot v(d)$
- Ergebnis: Die approximierte Lösung im Polygon ist mindestens  $1 - 2\epsilon$  mal so gut, wie die nicht approximierte
- Insgesamt mindestens eine Genauigkeit von  $1 - 3\epsilon$

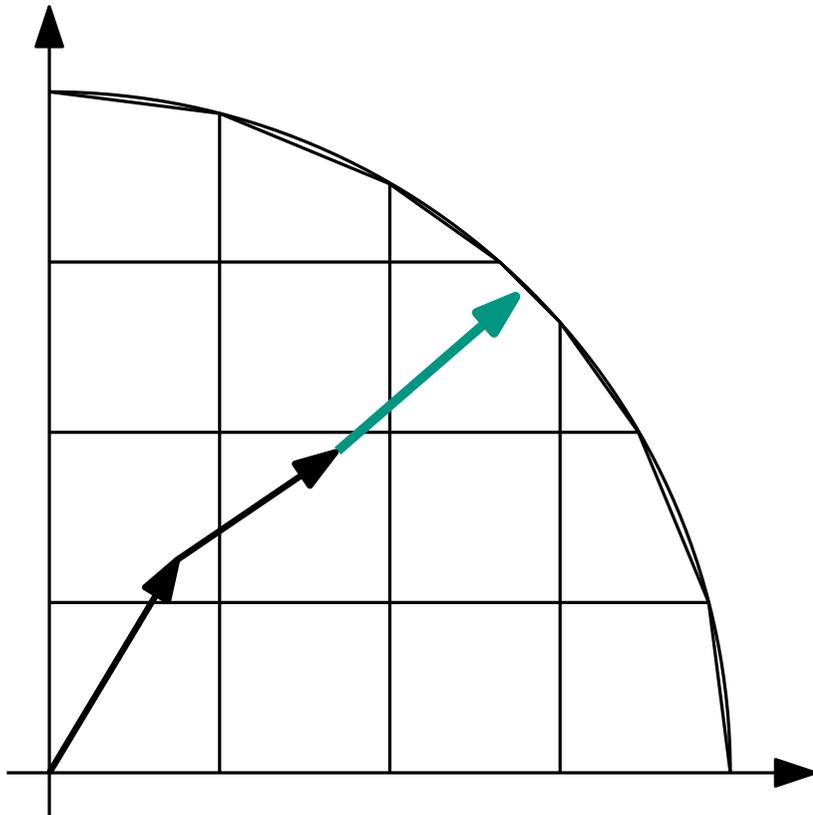


# Truthfulness

- Problem: Nutzer kann falschen Bedarf angeben
  - Approximationsgenauigkeit ändert sich
  - Allokation ändert sich

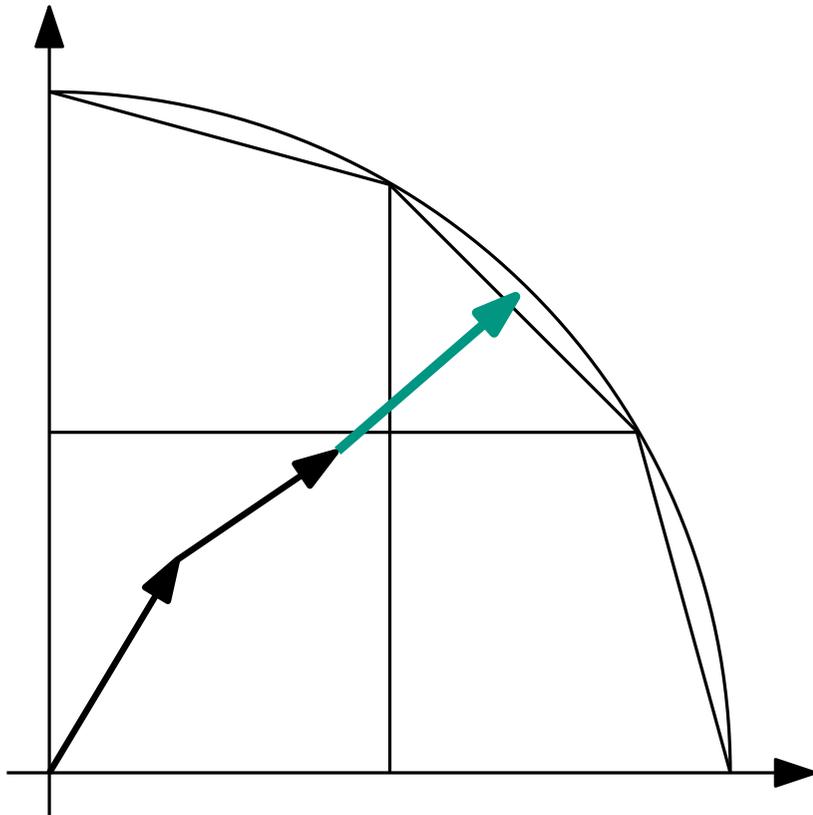
# Truthfulness

- Problem: Nutzer kann falschen Bedarf angeben
  - Approximationsgenauigkeit ändert sich
  - Allokation ändert sich



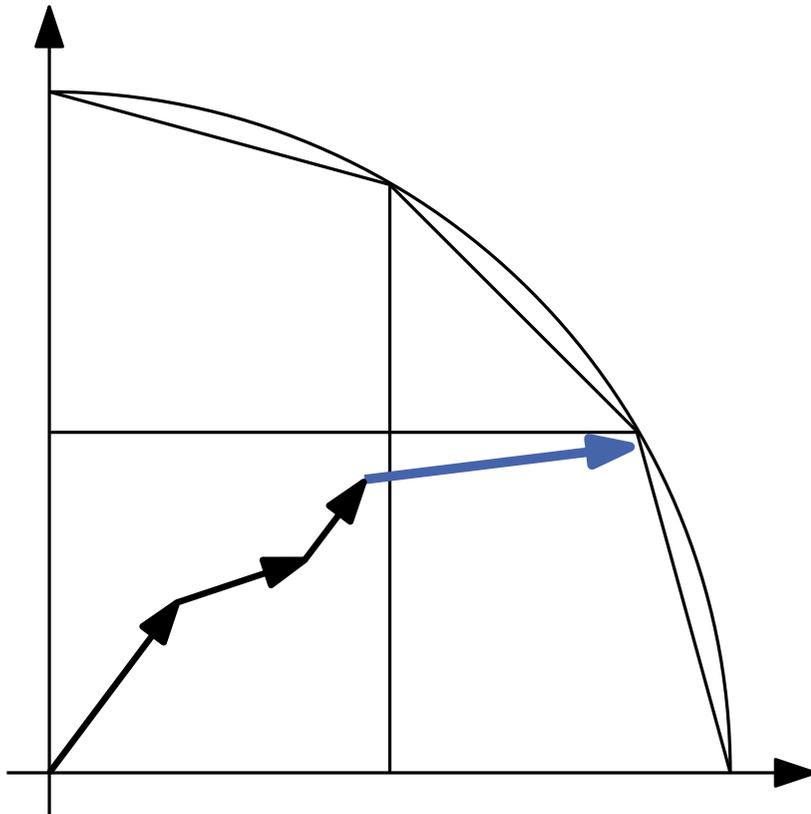
# Truthfulness

- Problem: Nutzer kann falschen Bedarf angeben
  - Approximationsgenauigkeit ändert sich
  - Allokation ändert sich



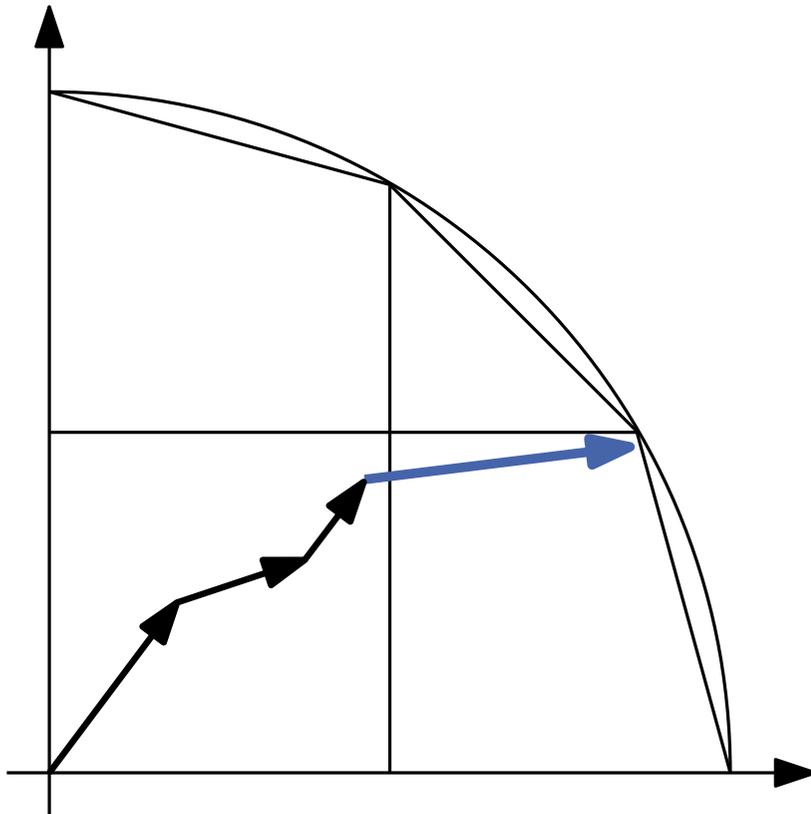
# Truthfulness

- Problem: Nutzer kann falschen Bedarf angeben
  - Approximationsgenauigkeit ändert sich
  - Allokation ändert sich



# Truthfulness

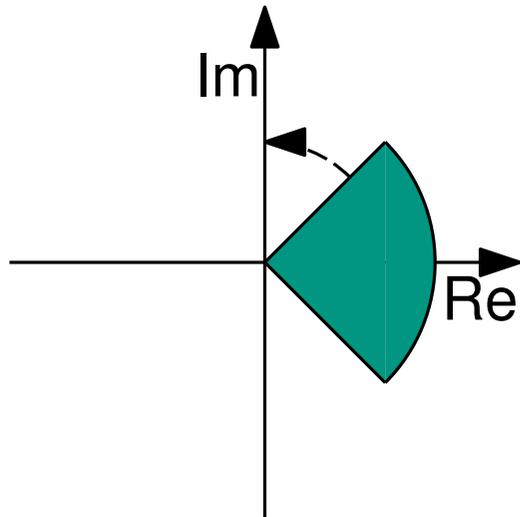
- Problem: Nutzer kann falschen Bedarf angeben
  - Approximationsgenauigkeit ändert sich
  - Allokation ändert sich



- Truthfulness verhindert solche Manipulationen

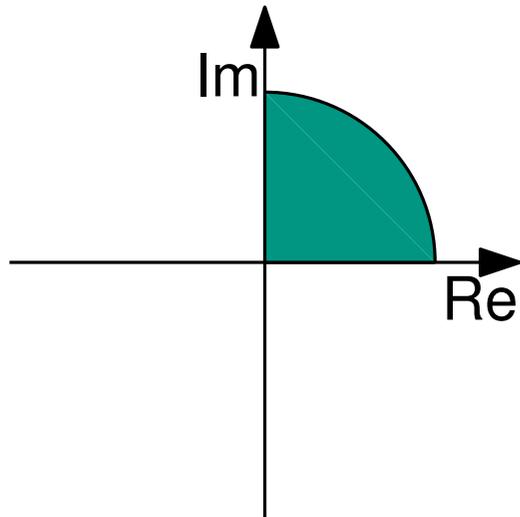
# Bemerkungen

- Drehung ermöglicht Approximation außerhalb von  $[0, \frac{\pi}{2}]$



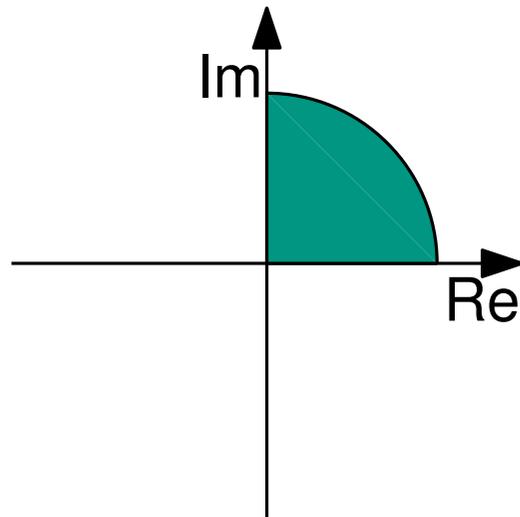
# Bemerkungen

- Drehung ermöglicht Approximation außerhalb von  $[0, \frac{\pi}{2}]$



# Bemerkungen

- Drehung ermöglicht Approximation außerhalb von  $[0, \frac{\pi}{2}]$



- $(1, 1 + \epsilon)$ -FPTAS [Chau, Elbassioni, Khonji 2016]
  - für  $\text{MULTICKP}[0, \pi - \delta]$
  - Verletzt Kapazitätsbedingung!
- Zeitliche Aspekte wurden bisher nicht berücksichtigt

Bundesverband der Energie- und Wasserwirtschaft:

„Energienetzwerk, welches das Verbrauchs- und Einspeiseverhalten aller Marktteilnehmer, die mit ihm verbunden sind, integriert.“ [BDEW-Roadmap]

- Netzgekoppelte Wechselstromgeneratoren
  - Generatorfrequenz hängt von der Netzfrequenz ab
  - Blindleistungseinpeisung gemäß dem verwendeten Modell möglich
  - Z. B. bei Wasserkraftwerken oder Biogasanlagen [Basse 2008]

- Netzgekoppelte Wechselstromgeneratoren
  - Generatorfrequenz hängt von der Netzfrequenz ab
  - Blindleistungseinpeisung gemäß dem verwendeten Modell möglich
  - Z. B. bei Wasserkraftwerken oder Biogasanlagen [Basse 2008]
- Gleichstrom- oder netzentkoppelte Wechselstromgeneratoren
  - Gleichstromabschnitt
  - Keine Blindleistung im Gleichstrombereich
  - Blindleistung muss durch einen Wechselrichter bereitgestellt werden
  - Z. B. bei Photovoltaikanlagen und teilweise bei Windkraftanlagen

- Bisher durch konventionelle Kraftwerke
- Gemäß §49 [EnWG] ist die Norm [VDE-AR-N 4105] verpflichtend
  - Technische Anforderungen für Erzeugungsanlagen am Niederspannungsnetz
  - Phasenverschiebungen (je nach Leistungsklasse) bis etwa  $\pm 18^\circ$  oder  $\pm 25^\circ$  müssen möglich sein
  - Steuerung durch eine Kennlinienvorgabe vom Netzbetreiber abhängig von der Wirkleistung
  - Keine externe Steuerung der Blindleistungseinspeisung vorgesehen

- Die Modellierung von Wechselstromnetzen ist komplex
- Es existiert ein  $(1 - 3\epsilon, 1)$ PTAS zur Allokation von Leistung in AC-Netzen
- Stark vereinfachte Modellierung der Erzeuger
- Exakte Netzmodellierung benötigt weitere Methoden
- Benötigte Einflussmöglichkeiten auf die Erzeuger sind heute noch nicht im Einsatz

[Basse 2008] Hauke Basse: Spannungshaltung in Verteilnetzen bei Stützung durch dezentrale Erzeugungsanlagen mit lokaler Blindleistungskompensation. 2008.

[BDEW-Roadmap] Bundesverband der Energie- und Wasserwirtschaft e. V.: BDEW-Roadmap. Realistische Schritte zur Umsetzung von Smart Grids in Deutschland. 2013.

[Chau, Elbassioni, Khonji 2016] Chi-Kin Chau, Khaled Elbassioni, Majid Konji: Truthful Mechanisms for Combinatorial Allocation of Electric Power in Alternating Current Electric Systems for Smart Grid. 2016.

[EnWG] Gesetz über die Elektrizitäts- und Gasversorgung. 2005.

[Frieze und Clarke 1984] A. M. Frieze und M. R. B. Clarke: Approximation algorithms for the  $m$ -dimensional 0-1 knapsack problem: Worst-case and probabilistic analyses. 1984.

[VDE-AR-N 4105] Erzeugungsanlagen am Niederspannungsnetz – Technische Mindestanforderungen für Anschluss und Parallelbetrieb von Erzeugungsanlagen am Niederspannungsnetz. 2011.

**Ende**

**Vielen Dank für Ihre Aufmerksamkeit!**