

Nichtelastische Lasten in AC Netzwerken und ihre Nichtapproximierbarkeit

Lars Gottesbüren

December 5, 2016

Motivation: Optimal Power Flow Problem

Gewinnmaximierende Allokation von Strom für Endkunden



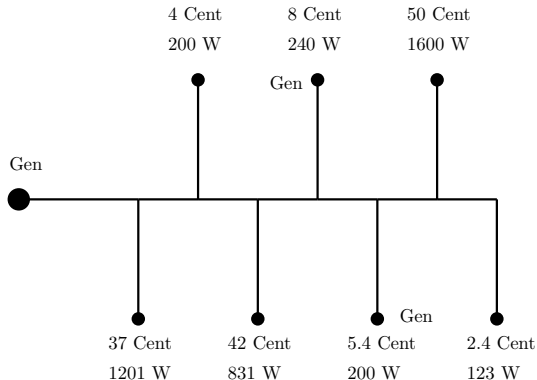
0

Abruf 8.11.2016

http://www.infineon.com/export/sites/default/media/press/Image/press_photo/Smart_Grid.jpg

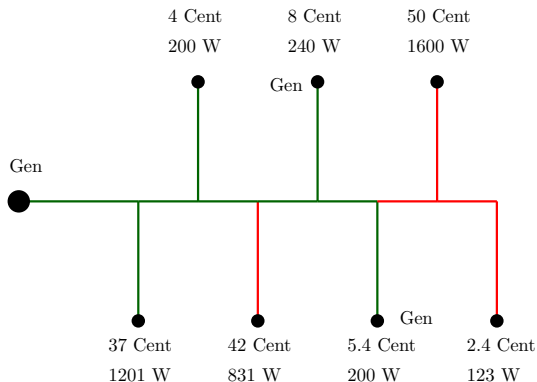
Motivation: Optimal Power Flow Problem

- Verbraucherinformation: Last und den Preis den sie dafür zahlen
- Netzbetreiber entscheidet welche Lasten erfüllt werden
- Netzbetreiber möchte seinen Gewinn maximieren



Motivation: Optimal Power Flow Problem

- Verbraucherinformation: Last und den Preis den sie dafür zahlen
- Netzbetreiber entscheidet welche Lasten erfüllt werden
- Netzbetreiber möchte seinen Gewinn maximieren



Problemdefinition **MaxOPF** in allgemeiner Form

Für Verbraucher k : x_k Entscheidungsvariable, u_k Preis den k für Verbrauch bietet

- Maximiere Nutzen $\sum_{k \in \mathcal{N}} x_k \cdot u_k$ unter den Bedingungen

- Kantenkapazität: $|S_e| := \left| \sum_{k: e \in P_k} x_k \cdot d_k \right| \leq C_e$

- Spannungskapazität an Knoten: $v_{\min} \leq |V_i| \leq v_{\max}$

- Ohm'sche Gesetze und Kirchhoff's Regel

Verschiedene Ausführungen des Problems: **MaxOPF**, **MaxOPF_C**, **sMaxOPF**

Nichtelastisch : $x_k \in \{0, 1\}$



1

Elastisch: $x_k \in [0, 1]$



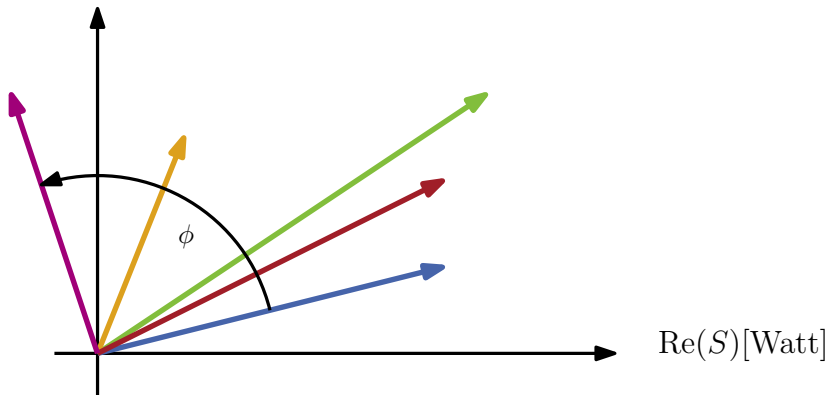
2

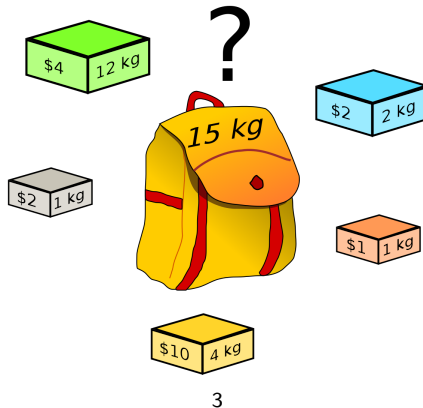
¹ http://www.vs-elektro.net/out/fck_pictures/Jura/E60.jpg

² <https://www.tesla.com/charging/high-power-wall-connector>

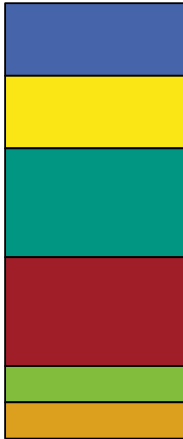
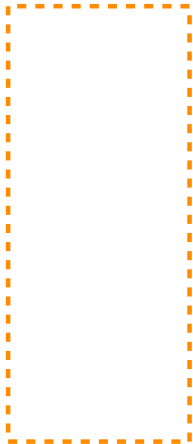
Phasenwinkel $\phi := \max$ Winkel zwischen Lasten

$\text{Im}(S)[\text{VAR}]$



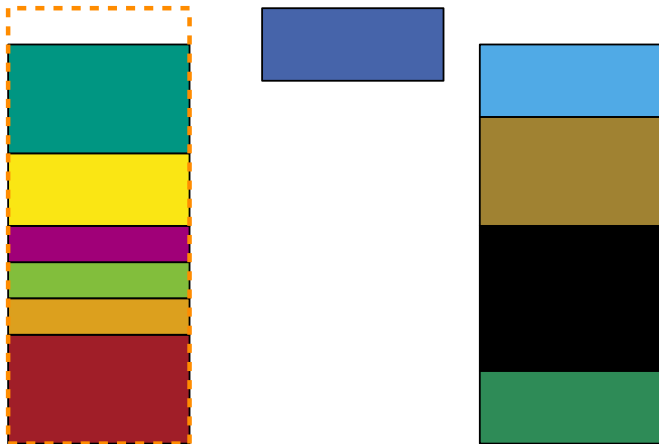


³<https://commons.wikimedia.org/wiki/File:Knapsack.svg>

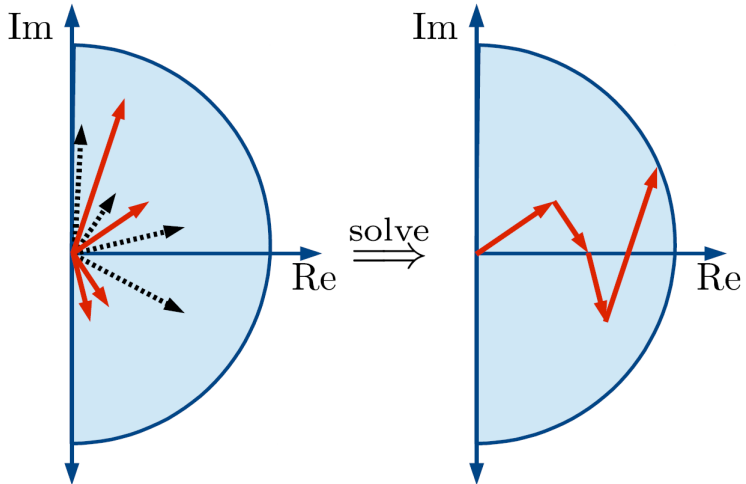


3





3



3

Complex Demand Knapsack Problem **CKP**

- $\phi \in \left[0, \frac{\pi}{2}\right]$: **CKP** PTAS [Chau et al., 2014]
- $\phi \in \left[0, \frac{\pi}{2}\right]$ kein FPTAS [Yu and Chau, 2013] falls **P** \neq **NP**
- $\phi \in \left(\frac{\pi}{2}, \pi - \delta\right]$: keine const-factor-Approximation ohne constraint-Verletzung, falls **P** \neq **NP** [Khonji et al., 2014]. Für δ in Polylänge kodiert
- Für beliebig kleine δ keine bikriterielle Approximation, i.e. mit Constraint-Verletzung

- Konvexe Optimierung für **MaxOPF** mit elastischen Lasten [Farivar and Low, 2013]
- **MaxOPF** mit Particle Swarm Optimization [Abido, 2002]
- **MaxOPF** gemischte Lasten genetische/evolutionäre Algorithmen [Bakirtzis et al., 2002], [Yuryevich and Wong, 1999]
- Newton-Verfahren [Dommel and Tinney, 1968]

Nichtapproximierbarkeit um konstanten Faktor

$\text{MaxOPF}_c \notin \text{APX}$ [Khonji et al., 2014]

Approximierbarkeit um nichtkonstanten Faktor in Bäumen

$\Omega\left(\frac{1}{\eta \log n}\right)$ Approximation für sMaxOPF in
Baumtopologien [Khonji et al., 2016]

- 1 Netzwerkmodell
- 2 Nichtapproximierbarkeit: Beweisausschnitt
- 3 Algorithmen für Baumtopologien

Optimierungsproblem

- Nutzen $u_k \in \mathbb{R}$
- Last $d_k \in \mathbb{C}$
- Entscheidungsvariable $x_k \in \{0, 1\}$

Netzwerk

- Stromstärke $I_{i,j} \in \mathbb{C}$
- Leistung $S_{i,j} \in \mathbb{C}$
- Leistung an Knoten s_j
- Impedanz $Z_{i,j} \in \mathbb{C}$
- Spannung $V_i \in \mathbb{R}$
- Leistungs- und Spannungskapazitäten $C_{i,j}, v_{\max}$ in \mathbb{R}

Maximiere Nutzen $\sum_{k \in \mathcal{N}} x_k \cdot u_k$ unter den Bedingungen

- Kantenkapazität: $|S_e| \leq C_e$
- Ohm'sche Gesetze und Kirchhoff's Regel (stay tuned)
- keine Spannungskapazität

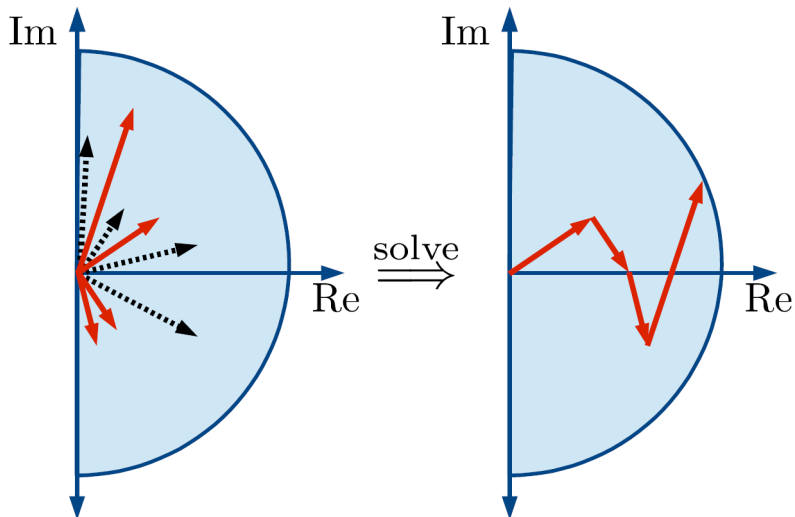


Figure: [Khonji et al., 2014]

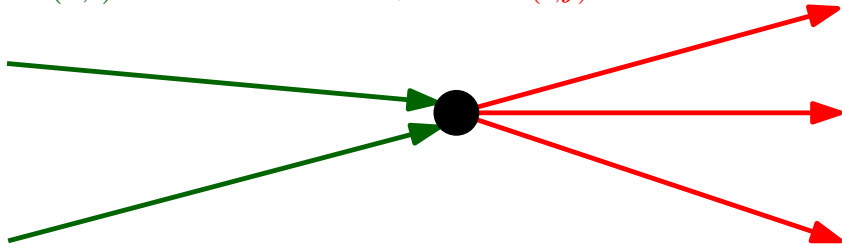
- $S_{i,j} = V_i \cdot \overline{I_{i,j}}$

- $Z_{i,j} \cdot I_{i,j} = V_i - V_j$

- Stromerhaltung:

- $s_j = \sum_{(i,j) \in E} (S_{i,j} - Z_{i,j} I_{i,j}^2) - \sum_{(j,i) \in E} S_{j,i}$

$$\sum_{(x,i) \in E} (S_{x,i} - Z_{x,i} I_{x,i}^2) = \sum_{(i,j) \in E} S_{i,j} + s_i$$



- 1 Netzwerkmodell
- 2 Nichtapproximierbarkeit: Beweisausschnitt
- 3 Algorithmen für Baumtopologien

(α, β) -Approximation

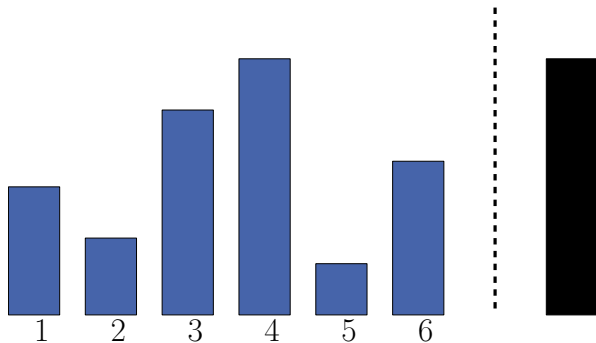
- $\alpha \in (0, 1), \beta \in (1, \infty)$
- Lösungsqualität $\geq \alpha \cdot \text{OPT}$
- Constraints max um Faktor β verletzt: i.e. $|S_e| \leq \beta C_e$

Theorem

Falls $\mathbf{P} \neq \mathbf{NP}$ gibt es keinen (α, β) -Approximationsalgorithmus in Polynomialzeit für MaxOPF_C

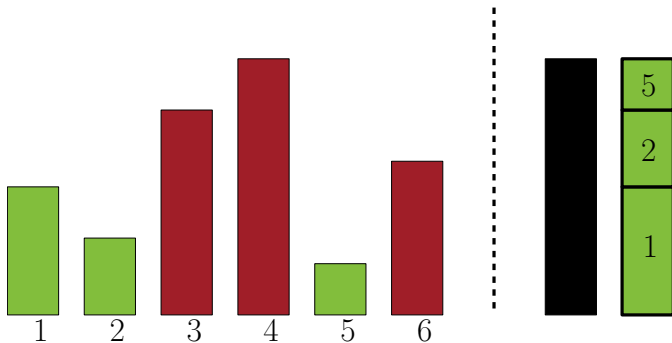
SUBSETSUM

- Gegeben: $a_1, \dots, a_n \in \mathbb{N}, B \in \mathbb{N}$
- Entscheidungsfrage: $\exists I \subset \{1, \dots, n\} : \sum_{i \in I} a_i = B ?$



SUBSETSUM

- Gegeben: $a_1, \dots, a_n \in \mathbb{N}, B \in \mathbb{N}$
- Entscheidungsfrage: $\exists I \subset \{1, \dots, n\} : \sum_{i \in I} a_i = B$?



(α, β) -Approximation

- $\alpha \in (0, 1), \beta \in (1, \infty)$
- Lösungsqualität $\geq \alpha \cdot \text{OPT}$
- Constraints max um Faktor β verletzt: i.e. $|S_e| \leq \beta C_e$

Beweisstrategie

SUBSETSUM Instanz / feasible \iff

konstruierte **MaxOPF_C** Instanz hat Nutzen $\sum_{k=1}^{n+1} x_k u_k \geq \alpha$

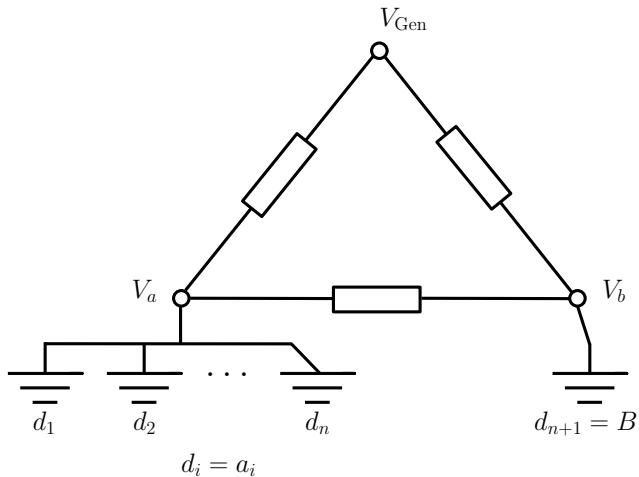


Figure: [Khonji et al., 2016]

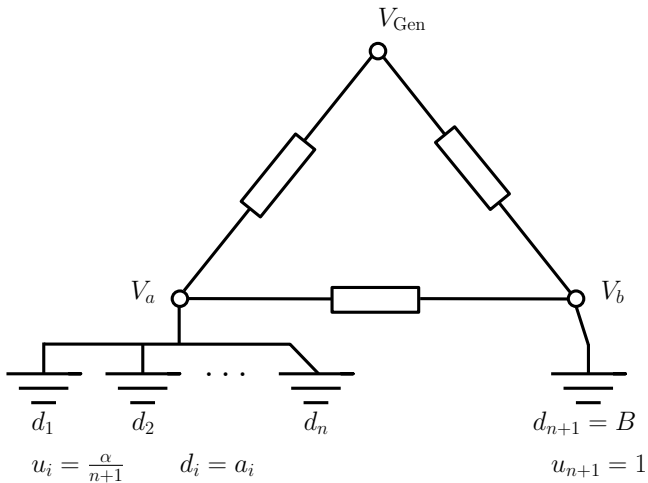


Figure: [Khonji et al., 2016]

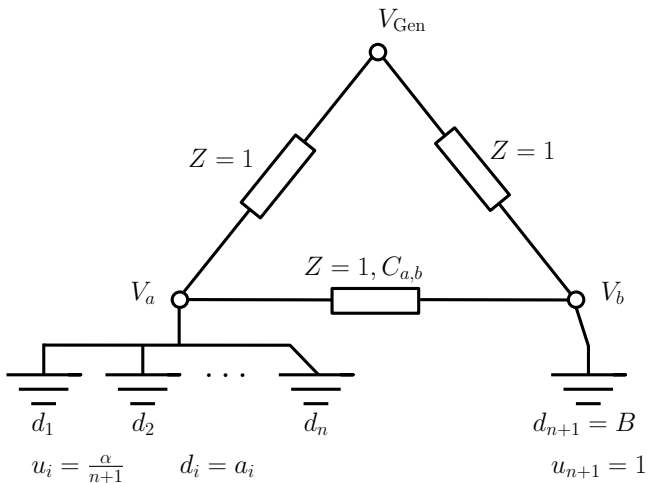


Figure: [Khonji et al., 2016]

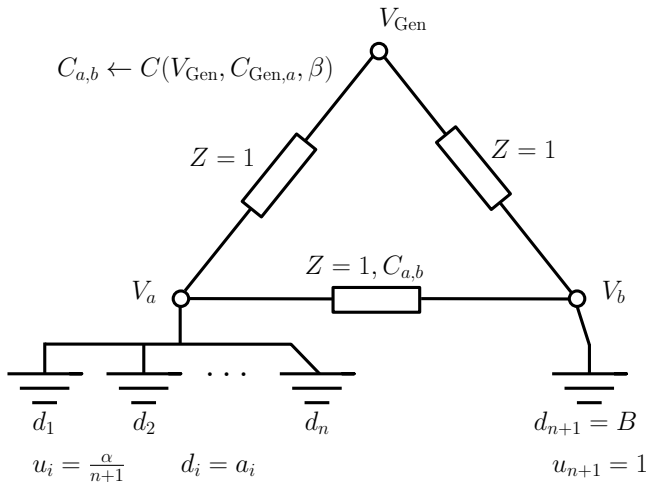


Figure: [Khonji et al., 2016]

- $s_a = \sum_{k=1}^n x_k d_k = \sum_{k=1}^n x_k a_k, s_b = x_{n+1} d_{n+1}$

- $s_a = \sum_{k=1}^n x_k d_k = \sum_{k=1}^n x_k a_k, s_b = x_{n+1} d_{n+1}$
- Zeige $|s_a - s_b| < 1$ über Umformungen mit Ohm's Gesetzen und Kirchhoff's Regel

- $s_a = \sum_{k=1}^n x_k d_k = \sum_{k=1}^n x_k a_k, s_b = x_{n+1} d_{n+1}$

- Zeige $|s_a - s_b| < 1$ über Umformungen mit Ohm's Gesetzen und Kirchhoff's Regel

- **MaxOPF_C** Instanz mit Nutzen $\sum_{k=1}^{n+1} x_k u_k \geq \alpha \Rightarrow x_{n+1} = 1$

- $s_a = \sum_{k=1}^n x_k d_k = \sum_{k=1}^n x_k a_k, s_b = x_{n+1} d_{n+1}$

- Zeige $|s_a - s_b| < 1$ über Umformungen mit Ohm's Gesetzen und Kirchhoff's Regel

- **MaxOPF_C** Instanz mit Nutzen $\sum_{k=1}^{n+1} x_k u_k \geq \alpha \Rightarrow x_{n+1} = 1$

$$\Rightarrow |s_a - B| < 1 \Rightarrow \sum_{k=1}^n x_k a_k = B$$

- $s_a = \sum_{k=1}^n x_k d_k = \sum_{k=1}^n x_k a_k, s_b = x_{n+1} d_{n+1}$

- Zeige $|s_a - s_b| < 1$ über Umformungen mit Ohm's Gesetzen und Kirchhoff's Regel

- feasible **SUBSETSUM** Instanz \Rightarrow Setze $x_{n+1} \leftarrow 1$

- 1 Netzwerkmodell
- 2 Nichtapproximierbarkeit: Beweisausschnitt
- 3 Algorithmen für Baumtopologien

- Erinnerung **MaxOPF**
- Branch Flow Modell: Reformulierung für Bäume
- Greedy Algorithmus
- Inelastic Demand Allocation: nutzt Greedy
- Mixed Demand Allocation
- Experimentelle Evaluation

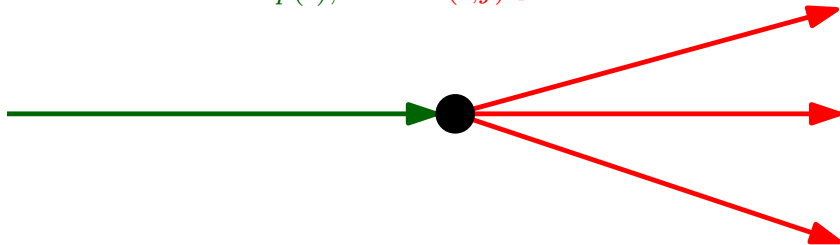
Maximiere Nutzen $\sum_{k \in \mathcal{N}} x_k \cdot u_k$ unter den Bedingungen

- Kantenkapazität: $|S_e| \leq C_e$
- Spannung an Knoten: $|V_i| \leq v_{\max}$
- Ohm'sche Gesetze und Kirchhoff's Regel (stay tuned)

Ohm und Kirchhoff für **MaxOPF** in allgemeinen Topologien

- $S_{i,j} = V_i \cdot \overline{I_{i,j}}$
- $Z_{i,j} \cdot I_{i,j} = V_i - V_j$
- Stromerhaltung:
 - $s_j = \sum_{(i,j) \in E} (S_{i,j} - Z_{i,j} I_{i,j}^2) - \sum_{(j,i) \in E} S_{j,i}$

$$S_{p(i),i} - Z_{p(i),i} I_{p(i),i}^2 = \sum_{(i,j) \in E} S_{i,j} + s_i$$

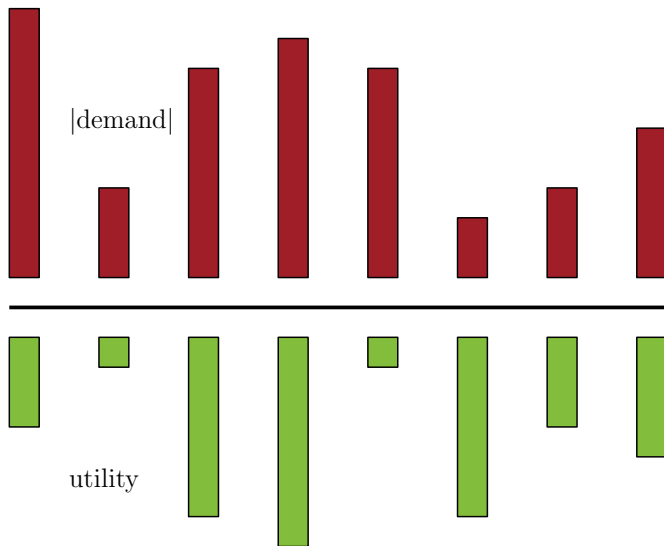


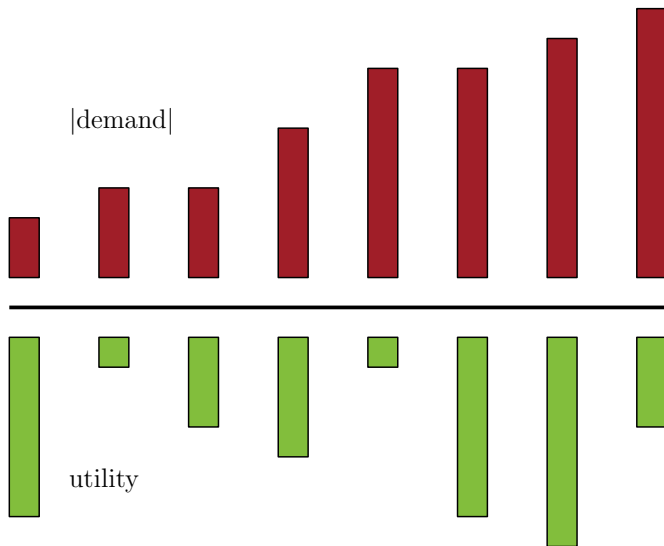
- $s_j = S_{i,j} - Z_{i,j}I_{i,j}^2 - \sum_{(j,l) \in E} S_{j,l}$, wobei $i = \text{parent}(j)$
- $s_{\text{Gen}} = - \sum_{(\text{Gen},j)} S_{\text{Gen},j}$
- $I_{i,j}^2 = \frac{S_{i,j}^2}{V_i^2}$
- $V_j^2 = V_i^2 + Z_{i,j}^2 I_{i,j}^2 - 2\text{Re}(\overline{Z_{i,j}} S_{i,j})$, wobei $i = \text{parent}(j)$

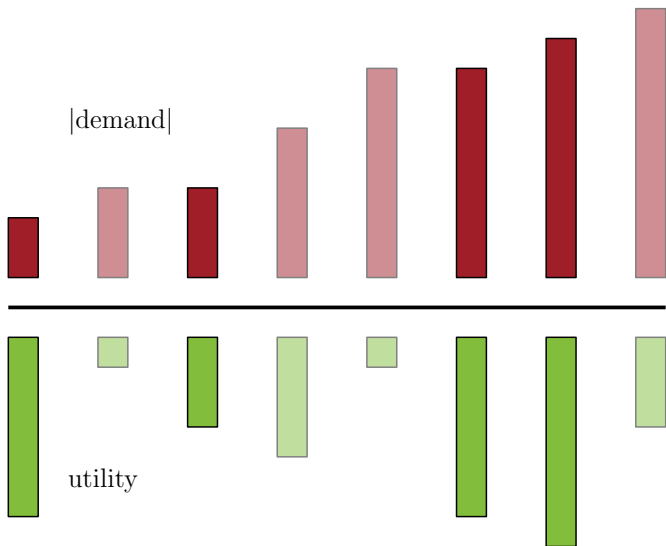
- Bei Bäumen keine zyklischen Abhängigkeiten

→ Spannung, Stromstärke festgelegt durch Lasten und Impedanz

→ welche Lasten erfüllen? + feasibility check

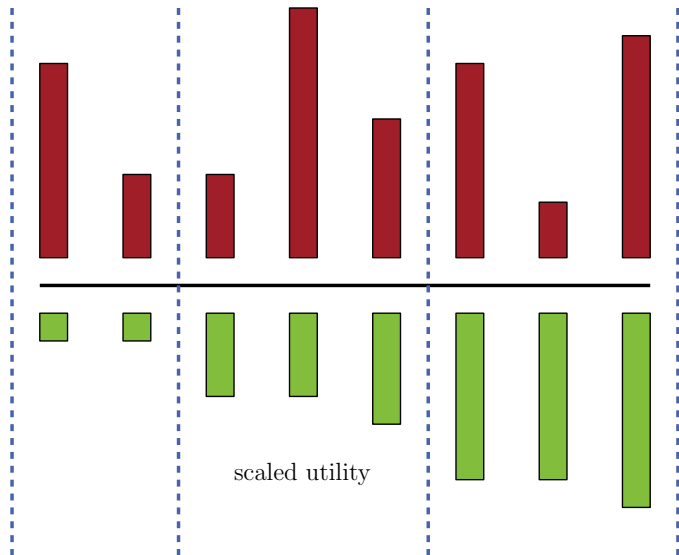




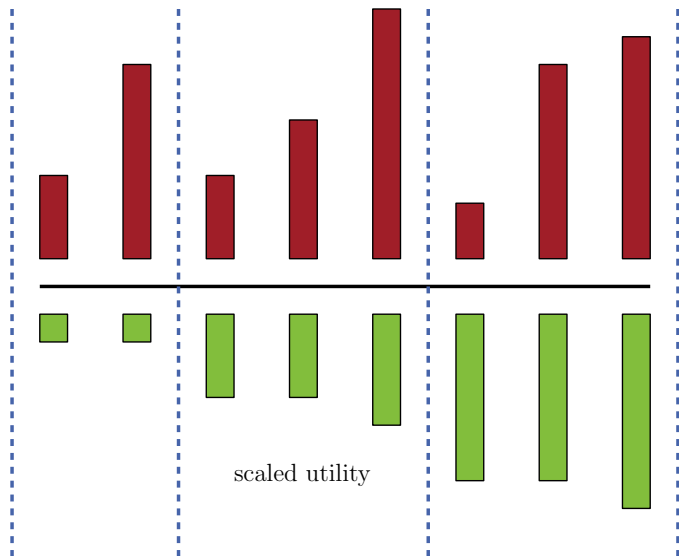


- $\bar{u} \leftarrow \frac{u \cdot n^2}{u_{\max}}$
- $G_i \leftarrow \{(\bar{u}_k, d_k) \mid \bar{u}_k \in (2^i, 2^{i+1}]\}$
- Greedy für jedes $G_i \rightarrow$ Maximum der Subresultate
- Nutzen $\geq \Omega\left(\frac{1}{\eta \log n}\right) \cdot \text{OPT}$ unter gewissen Annahmen
- ohne Beweis

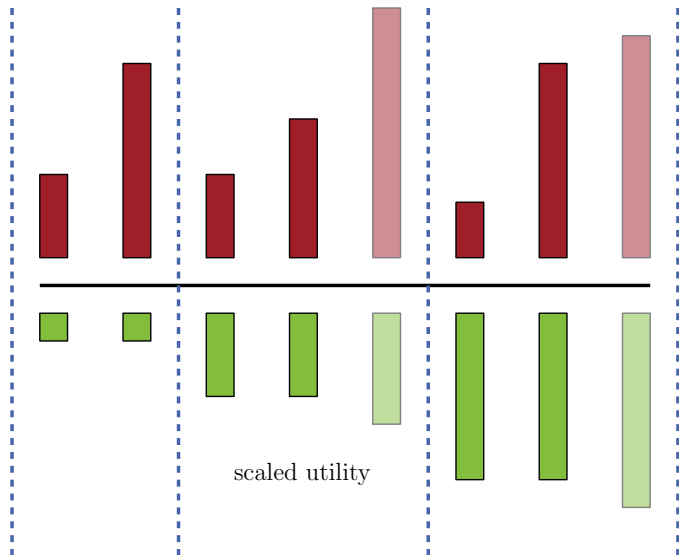
Inelastic Demand Allocation



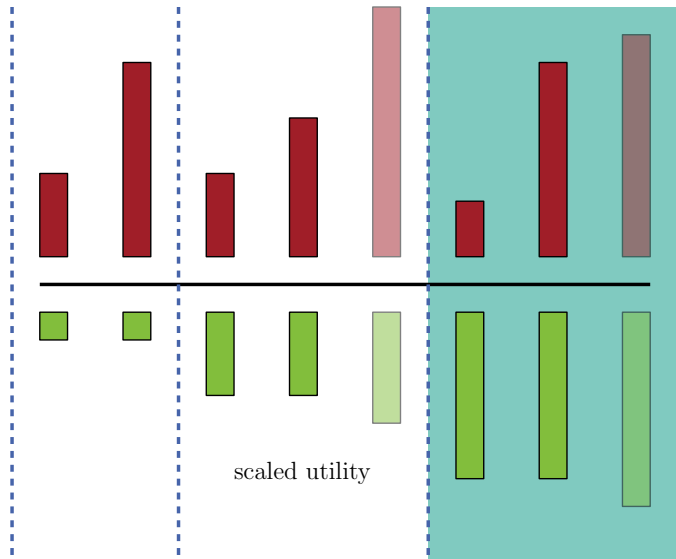
Inelastic Demand Allocation



Inelastic Demand Allocation



Inelastic Demand Allocation



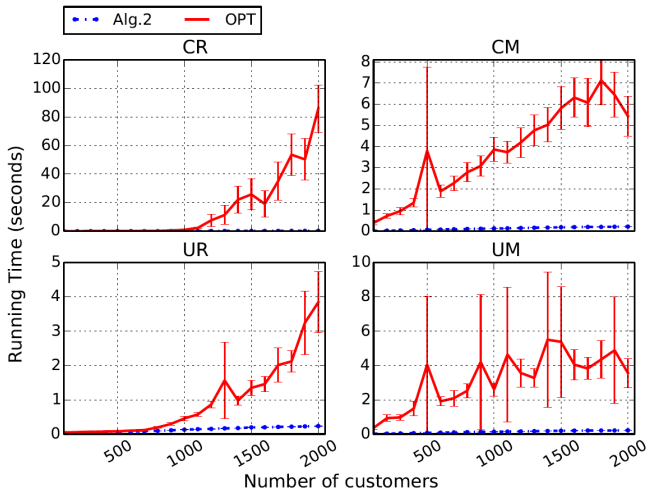


Figure: [Khonji et al., 2016]

C = correlated, U = uncorrelated, R = residential, M = residential + industrial

Alg.2 = Inelastic Demand Allocation, OPT = Gurobi

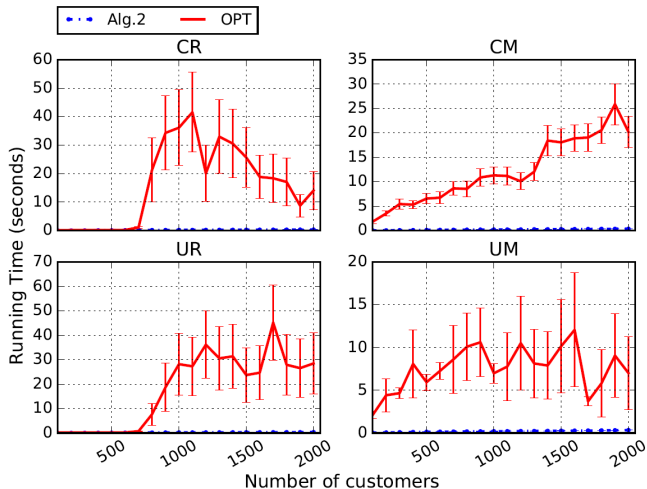


Figure: [Khonji et al., 2016]

C = correlated, U = uncorrelated, R = residential, M = residential + industrial

Alg.2 = Inelastic Demand Allocation, OPT = Gurobi

- IDA hat Laufzeit $O(\eta n \log n)$, dominiert quadratisches Programm

- IDA hat Laufzeit $O(\eta n \log n)$, dominiert quadratisches Programm
- experimentelle Qualität für IDA fehlt im Paper!

- IDA hat Laufzeit $O(\eta n \log n)$, dominiert quadratisches Programm
- experimentelle Qualität für IDA fehlt im Paper!
- IDA restriktiv
 - nur ähnliche utilities \rightarrow nur ähnliche Geräte gleichzeitig, falls Nutzen korreliert mit Lasten

- IDA hat Laufzeit $O(\eta n \log n)$, dominiert quadratisches Programm
- experimentelle Qualität für IDA fehlt im Paper!
- IDA restriktiv
 - nur ähnliche utilities \rightarrow nur ähnliche Geräte gleichzeitig, falls Nutzen korreliert mit Lasten
 - evtl. freie Kapazität: fixiere optimale Gruppe, wiederhole mit Restkapazität

- Kantenkapazität reduzieren um Faktor δ

- Kantenkapazität reduzieren um Faktor δ
- relaxiertes, konvexifiziertes Problem (nur elastische Lasten),
SOC [Farivar and Low, 2013] Farivar, Low

- Kantenkapazität reduzieren um Faktor δ
- relaxiertes, konvexifiziertes Problem (nur elastische Lasten), SOC [Farivar and Low, 2013] Farivar, Low
- Relaxiere nur für SOC die Nichtelastizität. SOC auf beiden Lasttypen.

- Kantenkapazität reduzieren um Faktor δ
- relaxiertes, konvexifiziertes Problem (nur elastische Lasten), SOC [Farivar and Low, 2013] Farivar, Low
- Relaxiere nur für SOC die Nichtelastizität. SOC auf beiden Lasttypen.
- für elastische Lasten die Zuweisung übernehmen

- Kantenkapazität reduzieren um Faktor δ
- relaxiertes, konvexifiziertes Problem (nur elastische Lasten), SOC [Farivar and Low, 2013] Farivar, Low
- Relaxiere nur für SOC die Nichtelastizität. SOC auf beiden Lasttypen.
- für elastische Lasten die Zuweisung übernehmen
- Nichtelastische Lasten mit Inelastic Demand Allocation

- Kantenkapazität reduzieren um Faktor δ
- relaxiertes, konvexifiziertes Problem (nur elastische Lasten), SOC [Farivar and Low, 2013] Farivar, Low
- Relaxiere nur für SOC die Nichtelastizität. SOC auf beiden Lasttypen.
- für elastische Lasten die Zuweisung übernehmen
- Nichtelastische Lasten mit Inelastic Demand Allocation
- δ schrittweise erhöhen bis Feasibility erreicht

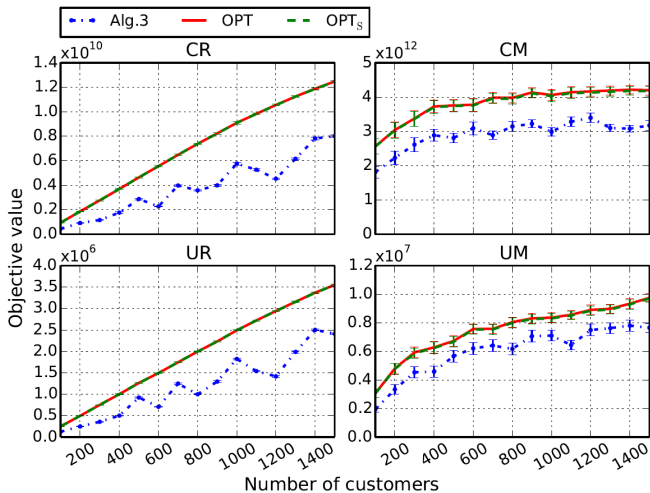


Figure: [Khonji et al., 2016]

C = correlated, U = uncorrelated, R = residential, M = residential + industrial

Alg.3 = Mixed Demand Allocation, OPT = Gurobi

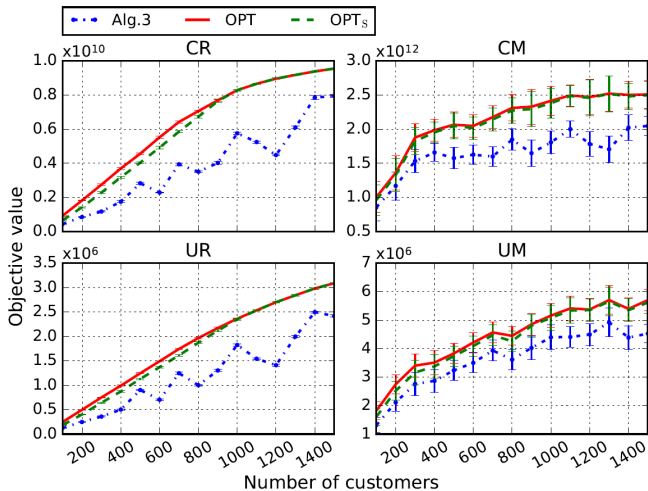


Figure: [Khonji et al., 2016]

C = correlated, U = uncorrelated, R = residential, M = residential + industrial

Alg.3 = Mixed Demand Allocation, OPT = Gurobi

- Qualität nahe an OPT aber keine Approximationsgüte

- Qualität nahe an OPT aber keine Approximationsgüte
- Laufzeit für MDA fehlt im Paper! **Vermutlich** langsam wegen SOC.

- Qualität nahe an OPT aber keine Approximationsgüte
- Laufzeit für MDA fehlt im Paper! **Vermutlich** langsam wegen SOC.
- Evtl gar langsamer als OPT

- Qualität nahe an OPT aber keine Approximationsgüte
- Laufzeit für MDA fehlt im Paper! **Vermutlich** langsam wegen SOC.
- Evtl gar langsamer als OPT
- Bei höherem Anteil elastischer Lasten wird MDA besser

Anteil elastischer Lasten vs Qualität

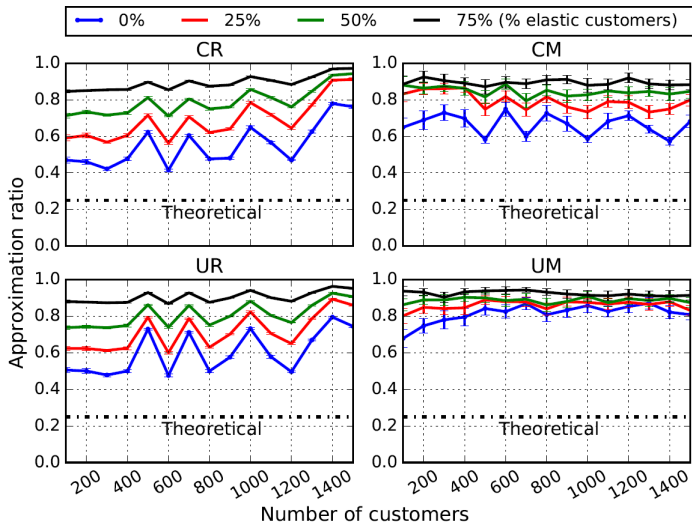


Figure: [Khonji et al., 2016]

- zeitliche Komponente nicht modelliert → Scheduling Modell

- zeitliche Komponente nicht modelliert → Scheduling Modell
- abgelehnte Lasten stören → Auktionssystem

- Approximationsbeweis verwendet viele Annahmen und eine relaxierte Problemvariante **sMaxOPF**

- Approximationsbeweis verwendet viele Annahmen und eine relaxierte Problemvariante **sMaxOPF**
- Approximationsalgorithmus für unrelaxiertes Problem **MaxOPF** finden

- Approximationsbeweis verwendet viele Annahmen und eine relaxierte Problemvariante **sMaxOPF**
- Approximationsalgorithmus für unrelaxiertes Problem **MaxOPF** finden
- **CKP** mit zusätzlichen Constraints auf $\text{Re}(d_k)$, $\text{Im}(d_k)$

- Approximationsbeweis verwendet viele Annahmen und eine relaxierte Problemvariante **sMaxOPF**
- Approximationsalgorithmus für unrelaxiertes Problem **MaxOPF** finden
- **CKP** mit zusätzlichen Constraints auf $\text{Re}(d_k), \text{Im}(d_k)$
- Heuristische Algorithmen mit sehr hoher Qualität auf echten Instanzen

- Approximationsbeweis verwendet viele Annahmen und eine relaxierte Problemvariante **sMaxOPF**
- Approximationsalgorithmus für unrelaxiertes Problem **MaxOPF** finden
- **CKP** mit zusätzlichen Constraints auf $\text{Re}(d_k), \text{Im}(d_k)$
- Heuristische Algorithmen mit sehr hoher Qualität auf echten Instanzen
- Benchmark Set zusammenstellen

- **MaxOPF** = Demand Allocation in Netzwerken

- **MaxOPF** = Demand Allocation in Netzwerken
- Netzwerkmodell verallgemeinert **CKP**

- **MaxOPF** = Demand Allocation in Netzwerken
- Netzwerkmodell verallgemeinert **CKP**
- erstmals Schwierigkeit von **nichtelastischen Lasten** im Netzwerkmodell bewiesen

- **MaxOPF** = Demand Allocation in Netzwerken
- Netzwerkmodell verallgemeinert **CKP**
- erstmals Schwierigkeit von **nichtelastischen Lasten** im Netzwerkmodell bewiesen
- Nicht um konstanten Faktor approximierbar: Reduktion von **SUBSETSUM**

- **MaxOPF** = Demand Allocation in Netzwerken
- Netzwerkmodell verallgemeinert **CKP**
- erstmals Schwierigkeit von **nichtelastischen Lasten** im Netzwerkmodell bewiesen
- Nicht um konstanten Faktor approximierbar: Reduktion von **SUBSETSUM**
- IDA sehr schnell, approximiert um $\Omega\left(\frac{1}{\eta \log n}\right)$ für Bäume **unter gewissen Annahmen**

- **MaxOPF** = Demand Allocation in Netzwerken
- Netzwerkmodell verallgemeinert **CKP**
- erstmals Schwierigkeit von **nichtelastischen Lasten** im Netzwerkmodell bewiesen
- Nicht um konstanten Faktor approximierbar: Reduktion von **SUBSETSUM**
- IDA sehr schnell, approximiert um $\Omega\left(\frac{1}{\eta \log n}\right)$ für Bäume **unter gewissen Annahmen**
- MDA für gemischte Lasten, hohe Qualität



Abido, M. (2002).

Optimal power flow using particle swarm optimization.

International Journal of Electrical Power & Energy Systems, 24(7):563–571.



Bakirtzis, A. G., Biskas, P. N., Zoumas, C. E., and Petridis, V. (2002).

Optimal power flow by enhanced genetic algorithm.

IEEE Transactions on power Systems, 17(2):229–236.



Chau, C.-K., Elbassioni, K., and Khonji, M. (2014).

Truthful mechanisms for combinatorial ac electric power allocation.

In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 1005–1012. International Foundation for Autonomous Agents and Multiagent Systems.



Dommel, H. W. and Tinney, W. F. (1968).

Optimal power flow solutions.

IEEE Transactions on power apparatus and systems, (10):1866–1876.



Farivar, M. and Low, S. H. (2013).

Branch flow model: Relaxations and convexification—part i.

IEEE Transactions on Power Systems, 28(3):2554–2564.



Khonji, M., Chau, C.-K., and Elbassioni, K. (2014).

Inapproximability of power allocation with inelastic demands in ac electric systems and networks.

In *2014 23rd International Conference on Computer Communication and Networks (ICCCN)*, pages 1–6. IEEE.



Khonji, M., Chau, C.-K., and Elbassioni, K. (2016).

Optimal power flow with inelastic demands for demand response in radial distribution networks.

arXiv preprint arXiv:1601.02323.



Yu, L. and Chau, C.-K. (2013).

Complex-demand knapsack problems and incentives in ac power systems.
In Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems, pages 973–980. International Foundation for Autonomous Agents and Multiagent Systems.



Yuryevich, J. and Wong, K. P. (1999).

Evolutionary programming based optimal power flow algorithm.
IEEE Transactions on Power Systems, 14(4):1245–1250.