technische universität
dortmund

Effiziente Algorithmen und
Komplexitätstheorie /

Informatik

Aufteilungs- und Zuweisungsalgorithmen

Proseminar
im Wintersemester 2017/2018

Anja Rey, 11. Oktober 2017

Organisatorisches

Proseminar (3 CP)

Organisatorisches

Proseminar (3 CP)

- schriftliche Ausarbeitung zu einem ausgewählten Thema
  - 5 bis 7 Seiten, PDF-Format (LaTeX)
  - in eigenen Worten (reine Übersetzung nicht ausreichend)
  - formal und anschaulich
  - vollständige Quellenangaben

Organisatorisches

Proseminar (3 CP)

- schriftliche Ausarbeitung zu einem ausgewählten Thema
  - 5 bis 7 Seiten, PDF-Format (LaTeX)
  - in eigenen Worten (reine Übersetzung nicht ausreichend)
  - formal und anschaulich
  - vollständige Quellenangaben
- Rückmeldung zu einer anderen Ausarbeitung in Form eines *Peer-Reviews*

Organisatorisches

Proseminar (3 CP)

- schriftliche Ausarbeitung zu einem ausgewählten Thema
  - 5 bis 7 Seiten, PDF-Format (LaTeX)
  - in eigenen Worten (reine Übersetzung nicht ausreichend)
  - formal und anschaulich
  - vollständige Quellenangaben
- Rückmeldung zu einer anderen Ausarbeitung in Form eines *Peer-Reviews*
- erfolgreich absolvierte Zwischengespräche
  1. Besprechung der Ausarbeitung vor erster Abgabe
  2. Rückmeldung der Seminarleiterin, Vorstellung eines schlüssigen Vortragskonzepts

Organisatorisches

Proseminar (3 CP)

- schriftliche Ausarbeitung zu einem ausgewählten Thema
  - 5 bis 7 Seiten, PDF-Format (LaTeX)
  - in eigenen Worten (reine Übersetzung nicht ausreichend)
  - formal und anschaulich
  - vollständige Quellenangaben
- Rückmeldung zu einer anderen Ausarbeitung in Form eines *Peer-Reviews*
- erfolgreich absolvierte Zwischengespräche
  1. Besprechung der Ausarbeitung vor erster Abgabe
  2. Rückmeldung der Seminarleiterin, Vorstellung eines schlüssigen Vortragskonzepts
- 30-minütiger Vortrag des Themas (+10 min für Fragen und Diskussion)

Organisatorisches

Proseminar (3 CP)

- schriftliche Ausarbeitung zu einem ausgewählten Thema
  - 5 bis 7 Seiten, PDF-Format (LaTeX)
  - in eigenen Worten (reine Übersetzung nicht ausreichend)
  - formal und anschaulich
  - vollständige Quellenangaben
- Rückmeldung zu einer anderen Ausarbeitung in Form eines *Peer-Reviews*
- erfolgreich absolvierte Zwischengespräche
  1. Besprechung der Ausarbeitung vor erster Abgabe
  2. Rückmeldung der Seminarleiterin, Vorstellung eines schlüssigen Vortragskonzepts
- 30-minütiger Vortrag des Themas (+10 min für Fragen und Diskussion)
- Frage zu mindestens einem Vortrag

Organisatorisches

Proseminar (3 CP)

- schriftliche Ausarbeitung zu einem ausgewählten Thema
  - 5 bis 7 Seiten, PDF-Format (LaTeX)
  - in eigenen Worten (reine Übersetzung nicht ausreichend)
  - formal und anschaulich
  - vollständige Quellenangaben
- Rückmeldung zu einer anderen Ausarbeitung in Form eines *Peer-Reviews*
- erfolgreich absolvierte Zwischengespräche
  1. Besprechung der Ausarbeitung vor erster Abgabe
  2. Rückmeldung der Seminarleiterin, Vorstellung eines schlüssigen Vortragskonzepts
- 30-minütiger Vortrag des Themas (+10 min für Fragen und Diskussion)
- Frage zu mindestens einem Vortrag

Präsentationskurs (1 CP)

begleitend: 4 Blöcke, Mi 12–16 Uhr oder Fr 14–18 Uhr, je in OH 14, 304

Zeitplan

Beginn

- Mi 11.10.17 Einleitung, Themen

Zeitplan

Beginn

- Mi 11.10.17 Einleitung, Themen

Themenvergabe

- Thema auswählen bis 17.10 unter moodle

Zeitplan

Beginn

- Mi 11.10.17 Einleitung, Themen

Themenvergabe

- Thema auswählen bis 17.10 unter moodle

Ausarbeitungsphase

- Mi 18.10.17 / Fr 20.10.17 Präsentationskurs 1
- Mi 25.10.17 Bearbeitung
- Mi 08.11.17 Besprechung & Bearbeitung
- 6. Woche Mo–Mi Zwischenbesprechung 1
- Do 16.11.17 erste Abgabe Ausarbeitung

— Rechtzeitig Vortragsplanung beginnen! —

Zeitplan

Beginn

- Mi 11.10.17 Einleitung, Themen

Themenvergabe

- Thema auswählen bis 17.10 unter moodle

Ausarbeitungsphase

- Mi 18.10.17 / Fr 20.10.17 Präsentationskurs 1
- Mi 25.10.17 Bearbeitung
- Mi 08.11.17 Besprechung & Bearbeitung
- 6. Woche Mo–Mi Zwischenbesprechung 1
- Do 16.11.17 erste Abgabe Ausarbeitung

— Rechtzeitig Vortragsplanung beginnen! —

Reviewphase

- Verteilung Reviews (per E-Mail)
- Mi 22.11.17 / Fr 24.11.17 Präsentationskurs 2
- Mi 29.11.17 Besprechung & Bearbeitung
- Mi 29.11.17 Abgabe Reviews
- Di 05.12.17 zweite Abgabe Ausarbeitung

Zeitplan

Beginn

- Mi 11.10.17 Einleitung, Themen

Themenvergabe

- Thema auswählen bis 17.10 unter moodle

Ausarbeitungsphase

- Mi 18.10.17 / Fr 20.10.17 Präsentationskurs 1
- Mi 25.10.17 Bearbeitung
- Mi 08.11.17 Besprechung & Bearbeitung
- 6. Woche Mo–Mi Zwischenbesprechung 1
- Do 16.11.17 erste Abgabe Ausarbeitung

— Rechtzeitig Vortragsplanung beginnen! —

Reviewphase

- Verteilung Reviews (per E-Mail)
- Mi 22.11.17 / Fr 24.11.17 Präsentationskurs 2
- Mi 29.11.17 Besprechung & Bearbeitung
- Mi 29.11.17 Abgabe Reviews
- Di 05.12.17 zweite Abgabe Ausarbeitung

Vortragsvorbereitung

- Mi 06.12.17 / Fr 08.12.17 Präsentationskurs 3
- 10. Woche Mo–Do Zwischenbesprechung 2
- Fr 15.12.17 / Mi 18.12.17 Präsentationskurs 4

Zeitplan

Beginn

- Mi 11.10.17 Einleitung, Themen

Themenvergabe

- Thema auswählen bis 17.10 unter moodle

Ausarbeitungsphase

- Mi 18.10.17 / Fr 20.10.17 Präsentationskurs 1
- Mi 25.10.17 Bearbeitung
- Mi 08.11.17 Besprechung & Bearbeitung
- 6. Woche Mo–Mi Zwischenbesprechung 1
- Do 16.11.17 erste Abgabe Ausarbeitung

— Rechtzeitig Vortragsplanung beginnen! —

Reviewphase

- Verteilung Reviews (per E-Mail)
- Mi 22.11.17 / Fr 24.11.17 Präsentationskurs 2
- Mi 29.11.17 Besprechung & Bearbeitung
- Mi 29.11.17 Abgabe Reviews
- Di 05.12.17 zweite Abgabe Ausarbeitung

Vortragsvorbereitung

- Mi 06.12.17 / Fr 08.12.17 Präsentationskurs 3
- 10. Woche Mo–Do Zwischenbesprechung 2
- Fr 15.12.17 / Mi 18.12.17 Präsentationskurs 4

Vortragsphase und Ende

- 12.–15. Woche: Vorträge
- Feedback am Ende

Fragen?

- Informationen unter

  `http://ls2-www.cs.uni-dortmund.de/~rey/wise1718proseminar`

- Themen und weitere Materialien unter

  `https://moodle.tu-dortmund.de/course/view.php?id=8368`

- Abgaben und weitere Fragen an anja.rey@tu-dortmund.de
- Zwischengespräche und weitere Fragen in OH 14, 313
- weitere Fragen in Seminarterminen und Sprechstunden

  Mo 13:30 Uhr bis 14:30 Uhr

Inhalte

Bewertungskriterien

1. Lesbarkeit
   (Struktur, Länge, Sprache, Verständlichkeit, Verknüpfungen)

Inhalte

Bewertungskriterien

1. Lesbarkeit
   (Struktur, Länge, Sprache, Verständlichkeit, Verknüpfungen)
2. technische Qualität
   (Definitionen, Anschaulichkeit, Korrektheit, Genauigkeit)

Inhalte

Bewertungskriterien

1. Lesbarkeit
   (Struktur, Länge, Sprache, Verständlichkeit, Verknüpfungen)
2. technische Qualität
   (Definitionen, Anschaulichkeit, Korrektheit, Genauigkeit)
3. Referenzen und Eigenanteil
   (Quellenangaben, Literaturverzeichnis, eigene Formulierungen)

Inhalte

Bewertungskriterien

1. Lesbarkeit
(Struktur, Länge, Sprache, Verständlichkeit, Verknüpfungen)
2. technische Qualität
(Definitionen, Anschaulichkeit, Korrektheit, Genauigkeit)
3. Referenzen und Eigenanteil
(Quellenangaben, Literaturverzeichnis, eigene Formulierungen)

Themenübersicht

- faire Aufteilung teilbarer und unteilbarer Güter
- das Häuser-Zuordnungs-Problem
- das *Hospitals-Residents*-Problem
- das Mitbewohner-Problem, Koalitionsbildung in hedonischen Spielen

Aufteilung und Zuweisung mit Präferenzen

Eine Instanz besteht aus

- Agenten (häufig zwei disjunkte Mengen)
- ordinalen Präferenzen über Teilmengen der Agenten
- anderen Bedingungen: Kapazitäten etc.

Aufteilung und Zuweisung mit Präferenzen

Eine Instanz besteht aus

- Agenten (häufig zwei disjunkte Mengen)
- ordinalen Präferenzen über Teilmengen der Agenten
- anderen Bedingungen: Kapazitäten etc.

Ziel: faire oder optimale Aufteilung (*Allocation*) oder Zuweisung (*Matching*) finden

Aufteilung und Zuweisung mit Präferenzen

Eine Instanz besteht aus

- Agenten (häufig zwei disjunkte Mengen)
- ordinalen Präferenzen über Teilmengen der Agenten
- anderen Bedingungen: Kapazitäten etc.

Ziel: faire oder optimale Aufteilung (*Allocation*) oder Zuweisung (*Matching*) finden

Wir unterscheiden zwischen

- bipartiten Zuweisungen mit beidseitigen Präferenzen
- bipartiten Aufteilungen mit einseitigen Präferenzen
  - Güter: teilbar (sharable, divisible) vs. nicht teilbar
- nicht-bipartiten Zuweisungen mit Präferenzen

Aufteilung und Zuweisung mit Präferenzen

Eine Instanz besteht aus

- Agenten (häufig zwei disjunkte Mengen)
- ordinalen Präferenzen über Teilmengen der Agenten
- anderen Bedingungen: Kapazitäten etc.

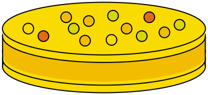
Ziel: faire oder optimale Aufteilung (*Allocation*) oder Zuweisung (*Matching*) finden

Wir unterscheiden zwischen

- bipartiten Zuweisungen mit beidseitigen Präferenzen
- bipartiten Aufteilungen mit einseitigen Präferenzen
  - Güter: teilbar (sharable, divisible) vs. nicht teilbar
- nicht-bipartiten Zuweisungen mit Präferenzen

Hier: Design und Analyse effizienter Algorithmen (oder ggf. Aussagen über Nicht-Existenz oder Härte solcher Algorithmen)

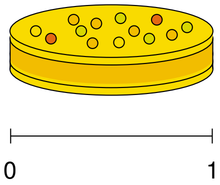
Aufteilung teilbarer Güter



Situation: inhomogener *Kuchen*

Ziel: jeder bekommt eine *faire* Portion

Aufteilung teilbarer Güter



Situation: inhomogener *Kuchen*

Ziel: jeder bekommt eine *faire* Portion

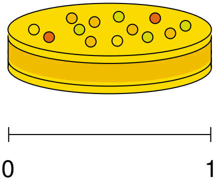
Definition 1

Eine Cake-Cutting-Instanz besteht aus einem Kuchen $[0, 1]$,

Aufteilung teilbarer Güter

Situation: inhomogener *Kuchen*

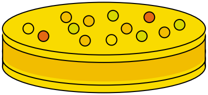
Ziel: jeder bekommt eine *faire* Portion



Definition 1

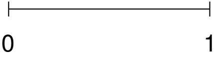
Eine Cake-Cutting-Instanz besteht aus einem Kuchen $[0, 1]$, Spielern $N = \{1, \ldots, n\}$

Aufteilung teilbarer Güter



Situation: inhomogener *Kuchen*

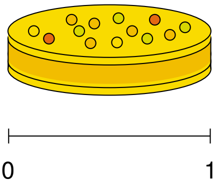
Ziel: jeder bekommt eine *faire* Portion



Definition 1

Eine Cake-Cutting-Instanz besteht aus einem Kuchen $[0,1]$, Spielern $N = \{1, \ldots, n\}$ und einer Präferenzfunktion $v_i : \{X \mid X \subseteq [0,1]\} \to \mathbb{R}_{\geq 0}$ für alle $i \in N$, sodass

Aufteilung teilbarer Güter



Situation: inhomogener *Kuchen*

Ziel: jeder bekommt eine *faire* Portion

Definition 1

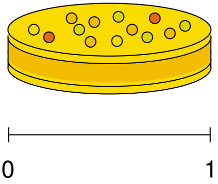
Eine Cake-Cutting-Instanz besteht aus einem Kuchen $[0, 1]$, Spielern $N = \{1, \dots, n\}$ und einer Präferenzfunktion $v_i : \{X \mid X \subseteq [0, 1]\} \to \mathbb{R}_{\geq 0}$ für alle $i \in N$, sodass

- $v_i([a, a]) = 0 \; \forall a \in [0, 1]$ (Nicht-Atomarität);
- $v_i([0, 1]) = 1$ (Normalisierung);
- für jedes Teilintervall $X \subseteq [0, 1]$: $v_i(X) \geq 0$ (Nicht-Negativität);

Aufteilung teilbarer Güter

Situation: inhomogener *Kuchen*

Ziel: jeder bekommt eine *faire* Portion



Definition 1

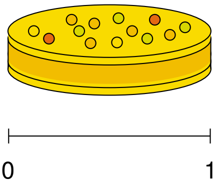
Eine Cake-Cutting-Instanz besteht aus einem Kuchen $[0, 1]$, Spielern $N = \{1, \dots, n\}$ und einer Präferenzfunktion $v_i : \{X \mid X \subseteq [0, 1]\} \to \mathbb{R}_{\geq 0}$ für alle $i \in N$, sodass

- $v_i([a, a]) = 0 \; \forall a \in [0, 1]$ (Nicht-Atomarität);
- $v_i([0, 1]) = 1$ (Normalisierung);
- für jedes Teilintervall $X \subseteq [0, 1]$: $v_i(X) \geq 0$ (Nicht-Negativität);
- für jedes Teilintervall $[a, b] \subseteq [0, 1]$ und λ: $0 \leq \lambda \leq 1$, existiert ein Punkt $c \in [a, b]$ mit $v_i([a, c]) = \lambda v_i([a, b])$ (Teilbarkeit);

Aufteilung teilbarer Güter

Situation: inhomogener *Kuchen*

Ziel: jeder bekommt eine *faire* Portion



Definition 1

Eine Cake-Cutting-Instanz besteht aus einem Kuchen $[0, 1]$, Spielern $N = \{1, \ldots, n\}$ und einer Präferenzfunktion $v_i : \{X \mid X \subseteq [0, 1]\} \to \mathbb{R}_{\geq 0}$ für alle $i \in N$, sodass

- $v_i([a, a]) = 0 \; \forall a \in [0, 1]$ (Nicht-Atomarität);
- $v_i([0, 1]) = 1$ (Normalisierung);
- für jedes Teilintervall $X \subseteq [0, 1]$: $v_i(X) \geq 0$ (Nicht-Negativität);
- für jedes Teilintervall $[a, b] \subseteq [0, 1]$ und λ: $0 \leq \lambda \leq 1$, existiert ein Punkt $c \in [a, b]$ mit $v_i([a, c]) = \lambda v_i([a, b])$ (Teilbarkeit);
- für zwei Teilintervalle $X, Y \subseteq [0, 1]$: $v_i(X) + v_i(Y) = v_i(X \cup Y)$ (Additivität).

Aufteilung teilbarer Güter

Beispiel: Cut & Choose

- $n = 2$

Aufteilung teilbarer Güter

Beispiel: Cut & Choose

- $n = 2$
- Spieler 1 schneidet den Kuchen in zwei disjunkte Stücke X_1 und X_2, sodass $v_1(X_1) = v_1(X_2)$ und $X_1 \cup X_2 = [0, 1]$.

Aufteilung teilbarer Güter

Beispiel: Cut & Choose

- $n = 2$
- Spieler 1 schneidet den Kuchen in zwei disjunkte Stücke X_1 und X_2, sodass $v_1(X_1) = v_1(X_2)$ und $X_1 \cup X_2 = [0, 1]$.
- Spieler 2 wählt eine der beiden Stücke X_i mit $v_2(X_i) \geq v_2(X, j)$, $j \neq i$.

Aufteilung teilbarer Güter

Beispiel: Cut & Choose

- $n = 2$
- Spieler 1 schneidet den Kuchen in zwei disjunkte Stücke X_1 und X_2, sodass $v_1(X_1) = v_1(X_2)$ und $X_1 \cup X_2 = [0, 1]$.
- Spieler 2 wählt eine der beiden Stücke X_i mit $v_2(X_i) \geq v_2(X, j)$, $j \neq i$.
- Spieler 1 bekommt das andere Stück.

Aufteilung teilbarer Güter

Beispiel: Cut & Choose

- $n = 2$
- Spieler 1 schneidet den Kuchen in zwei disjunkte Stücke X_1 und X_2, sodass $v_1(X_1) = v_1(X_2)$ und $X_1 \cup X_2 = [0, 1]$.
- Spieler 2 wählt eine der beiden Stücke X_i mit $v_2(X_i) \geq v_2(X, j)$, $j \neq i$.
- Spieler 1 bekommt das andere Stück.

weitere Protokolle, z. B. Dubins–Spanier, Evan–Paz, Selfridge–Conway

Aufteilung teilbarer Güter

Beispiel: Cut & Choose

- $n = 2$
- Spieler 1 schneidet den Kuchen in zwei disjunkte Stücke X_1 und X_2, sodass $v_1(X_1) = v_1(X_2)$ und $X_1 \cup X_2 = [0, 1]$.
- Spieler 2 wählt eine der beiden Stücke X_i mit $v_2(X_i) \geq v_2(X, j)$, $j \neq i$.
- Spieler 1 bekommt das andere Stück.

weitere Protokolle, z. B. Dubins–Spanier, Evan–Paz, Selfridge–Conway

Fairness-Kriterien

- Proportionalität
- Neidfreiheit

Aufteilung teilbarer Güter

Beispiel: Cut & Choose

- $n = 2$
- Spieler 1 schneidet den Kuchen in zwei disjunkte Stücke X_1 und X_2, sodass $v_1(X_1) = v_1(X_2)$ und $X_1 \cup X_2 = [0, 1]$.
- Spieler 2 wählt eine der beiden Stücke X_i mit $v_2(X_i) \geq v_2(X, j)$, $j \neq i$.
- Spieler 1 bekommt das andere Stück.

weitere Protokolle, z. B. Dubins–Spanier, Evan–Paz, Selfridge–Conway

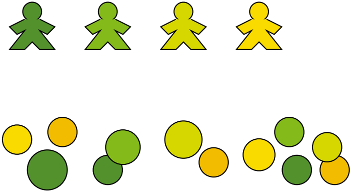
Fairness-Kriterien

- Proportionalität
- Neidfreiheit

Thema 1

Thema 2

Aufteilung unteilbarer Güter

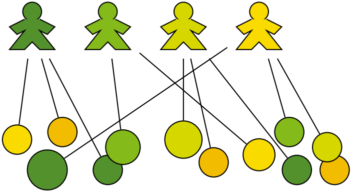


Situation: Objekte/Ressourcen mit unterschiedlichem Wert für Agenten

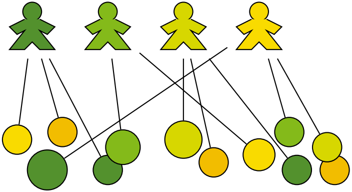
Aufteilung unteilbarer Güter

Situation: Objekte/Ressourcen mit unterschiedlichem Wert für Agenten

Ziel: jeder bekommt einen *optimalen* Anteil (1 : *many*)



Aufteilung unteilbarer Güter



Situation: Objekte/Ressourcen mit unterschiedlichem Wert für Agenten

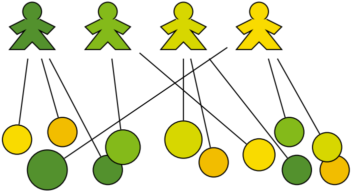
Ziel: jeder bekommt einen *optimalen* Anteil (1 : *many*)

Definition 2 (*MultiAgent Resource Allocation* (MARA))

Eine MARA-Instanz besteht aus

- Agenten $N = \{1, \ldots, n\}$
- Objekten $\mathcal{O} = \{o_1, \ldots, o_p\}$
- Präferenzen der Agenten in N

Aufteilung unteilbarer Güter



Situation: Objekte/Ressourcen mit unterschiedlichem Wert für Agenten

Ziel: jeder bekommt einen *optimalen* Anteil (1 : *many*)

Definition 2 (*MultiAgent Resource Allocation* (MARA))

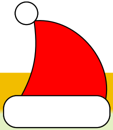
Eine MARA-Instanz besteht aus

- Agenten $N = \{1, \dots, n\}$
- Objekten $\mathcal{O} = \{o_1, \dots, o_p\}$
- Präferenzen der Agenten in N

unterschiedliche Darstellungen der Präferenzen:

- kardinale Präferenzen (Nutzenfunktion $u : \mathcal{O} \to \mathbb{R}$)
- ordinale Präferenzen (z. B. Ranking über Bündel von Objekten)

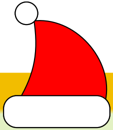
Aufteilung unteilbarer Güter



Beispiel: das *Santa-Claus*-Problem

- MARA-Instanz mit kardinalen Präferenzen
- Nutzenfunktion u ist modular: für alle $S, T \subseteq \mathcal{O}$:
  $u(S \cup T) = u(S) + u(T) - u(S \cap T)$.
- Ziel: Nutzen für das unglücklichste Kind maximieren.

Aufteilung unteilbarer Güter



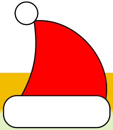
Beispiel: das *Santa-Claus*-Problem

- MARA-Instanz mit kardinalen Präferenzen
- Nutzenfunktion u ist modular: für alle $S, T \subseteq \mathcal{O}$:
  $u(S \cup T) = u(S) + u(T) - u(S \cap T)$.
- Ziel: Nutzen für das unglücklichste Kind maximieren.

Fairness vs. Effizienz:

- Maxmin-Aufteilung
- Neidfreiheit
- Pareto-Effizienz

Aufteilung unteilbarer Güter



Beispiel: das *Santa-Claus*-Problem

- MARA-Instanz mit kardinalen Präferenzen
- Nutzenfunktion u ist modular: für alle $S, T \subseteq \mathcal{O}$:
  $u(S \cup T) = u(S) + u(T) - u(S \cap T)$.
- Ziel: Nutzen für das unglücklichste Kind maximieren.

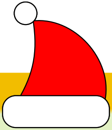
Fairness vs. Effizienz:

- Maxmin-Aufteilung
- Neidfreiheit
- Pareto-Effizienz

Protokolle:

- *Adjusted-Winner*-Verfahren
- Proportionales Aufteilungsverfahren
- ...

Aufteilung unteilbarer Güter



Beispiel: das *Santa-Claus*-Problem

- MARA-Instanz mit kardinalen Präferenzen
- Nutzenfunktion u ist modular: für alle $S, T \subseteq \mathcal{O}$:
  $u(S \cup T) = u(S) + u(T) - u(S \cap T)$.
- Ziel: Nutzen für das unglücklichste Kind maximieren.

Fairness vs. Effizienz:

- Maxmin-Aufteilung
- Neidfreiheit
- Pareto-Effizienz

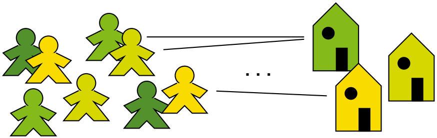
Protokolle:

- *Adjusted-Winner*-Verfahren (Thema 3)
- Proportionales Aufteilungsverfahren (Thema 4)
- ...

Zuweisung von Ärzten auf Krankenhäuser

Situation: beidseitige Präferenzen

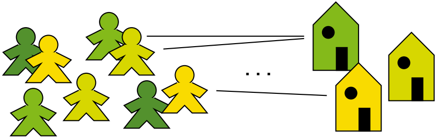
Ziel: (*many* : 1)-Zuweisung



Zuweisung von Ärzten auf Krankenhäuser

Situation: beidseitige Präferenzen

Ziel: (*many* : 1)-Zuweisung



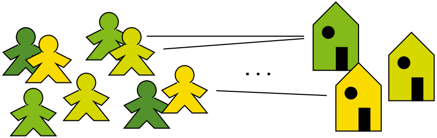
Definition 3 (Das *Hospital-Residents*-Problem (HR))

Eine HR-Instanz besteht aus

Zuweisung von Ärzten auf Krankenhäuser

Situation: beidseitige Präferenzen

Ziel: (*many* : 1)-Zuweisung



Definition 3 (Das *Hospital-Residents*-Problem (HR))

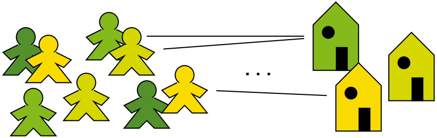
Eine HR-Instanz besteht aus

- den disjunkten Agentenmengen $R = \{r_1, \dots, r_{n_1}\}$ und $H = \{h_1, \dots, h_{n_2}\}$

Zuweisung von Ärzten auf Krankenhäuser

Situation: beidseitige Präferenzen

Ziel: (*many* : 1)-Zuweisung



Definition 3 (Das *Hospital-Residents*-Problem (HR))

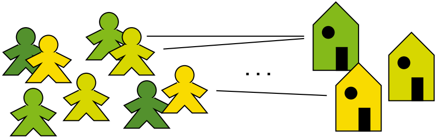
Eine HR-Instanz besteht aus

- den disjunkten Agentenmengen $R = \{r_1, \ldots, r_{n_1}\}$ und $H = \{h_1, \ldots, h_{n_2}\}$
- $\forall h_j \in H$: Kapazität c_j

Zuweisung von Ärzten auf Krankenhäuser

Situation: beidseitige Präferenzen

Ziel: (*many* : 1)-Zuweisung



Definition 3 (Das *Hospital-Residents*-Problem (HR))

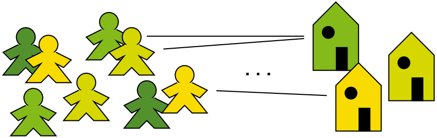
Eine HR-Instanz besteht aus

- den disjunkten Agentenmengen $R = \{r_1, \ldots, r_{n_1}\}$ und $H = \{h_1, \ldots, h_{n_2}\}$
- $\forall h_j \in H$: Kapazität c_j
- $E \subseteq R \times H$ akzeptable Paare, $m = \|E\|$
- $\forall r_i \in R$: $A(r_i) = \{h_j \in H \mid (r_i, h_j) \in E\}$, $\quad \forall h_j \in R$: $A(h_j) = \{r_i \in R \mid (r_i, h_j) \in E\}$

Zuweisung von Ärzten auf Krankenhäuser

Situation: beidseitige Präferenzen

Ziel: (*many* : 1)-Zuweisung



Definition 3 (Das *Hospital-Residents*-Problem (HR))

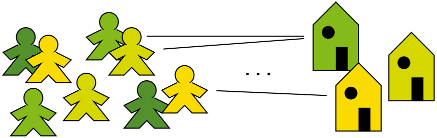
Eine HR-Instanz besteht aus

- den disjunkten Agentenmengen $R = \{r_1, \ldots, r_{n_1}\}$ und $H = \{h_1, \ldots, h_{n_2}\}$
- $\forall h_j \in H$: Kapazität c_j
- $E \subseteq R \times H$ akzeptable Paare, $m = \|E\|$
- $\forall r_i \in R$: $A(r_i) = \{h_j \in H \mid (r_i, h_j) \in E\}$, $\quad \forall h_j \in R$: $A(h_j) = \{r_i \in R \mid (r_i, h_j) \in E\}$
- $\forall a_k \in R \cup H$: strikte Präferenzliste über $A(k)$

Zuweisung von Ärzten auf Krankenhäuser

Situation: beidseitige Präferenzen

Ziel: (*many* : 1)-Zuweisung



Definition 3 (Das *Hospital-Residents*-Problem (HR))

Eine HR-Instanz besteht aus

- den disjunkten Agentenmengen $R = \{r_1, \ldots, r_{n_1}\}$ und $H = \{h_1, \ldots, h_{n_2}\}$
- $\forall h_j \in H$: Kapazität c_j
- $E \subseteq R \times H$ akzeptable Paare, $m = \|E\|$
- $\forall r_i \in R$: $A(r_i) = \{h_j \in H \mid (r_i, h_j) \in E\}$, $\quad \forall h_j \in R$: $A(h_j) = \{r_i \in R \mid (r_i, h_j) \in E\}$
- $\forall a_k \in R \cup H$: strikte Präferenzliste über $A(k)$

M ist eine Zuweisung (*Matching*), wenn $|M(r_i)| \leq 1 \; \forall r_i \in R$ und $|M(h_j)| \leq c_j \; \forall h_j \in H$

Zuweisung von Ärzten auf Krankenhäuser

Seien $\mathcal{I}$ eine HR-Instanz und M eine Zuweisung in $\mathcal{I}$.

$(r_i, h_j) \in E \setminus M$ blockiert M, falls

Zuweisung von Ärzten auf Krankenhäuser

Seien $\mathcal{I}$ eine HR-Instanz und M eine Zuweisung in $\mathcal{I}$.

$(r_i, h_j) \in E \setminus M$ blockiert M, falls

- Arzt r_i ist nicht zugewiesen oder bevorzugt h_j vor $M(r_i)$;

Zuweisung von Ärzten auf Krankenhäuser

Seien $\mathcal{I}$ eine HR-Instanz und M eine Zuweisung in $\mathcal{I}$.

$(r_i, h_j) \in E \setminus M$ blockiert M, falls

- Arzt r_i ist nicht zugewiesen oder bevorzugt h_j vor $M(r_i)$;
- Krankenhaus h_j ist unterbesetzt oder bevorzugt r_i vor mindestens einem Arzt in $M(h_j)$.

Zuweisung von Ärzten auf Krankenhäuser

Seien $\mathcal{I}$ eine HR-Instanz und M eine Zuweisung in $\mathcal{I}$.

$(r_i, h_j) \in E \setminus M$ blockiert M, falls

- Arzt r_i ist nicht zugewiesen oder bevorzugt h_j vor $M(r_i)$;
- Krankenhaus h_j ist unterbesetzt oder bevorzugt r_i vor mindestens einem Arzt in $M(h_j)$.

Eine Zuweisung M heißt stabil, falls kein Paar M blockiert.

Zuweisung von Ärzten auf Krankenhäuser

Seien $\mathcal{I}$ eine HR-Instanz und M eine Zuweisung in $\mathcal{I}$.

$(r_i, h_j) \in E \setminus M$ blockiert M, falls

- Arzt r_i ist nicht zugewiesen oder bevorzugt h_j vor $M(r_i)$;
- Krankenhaus h_j ist unterbesetzt oder bevorzugt r_i vor mindestens einem Arzt in $M(h_j)$.

Eine Zuweisung M heißt stabil, falls kein Paar M blockiert.

Satz 4

Es gibt immer eine stabile Zuweisung.

Zuweisung von Ärzten auf Krankenhäuser

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Eingabe: HR-Instanz
Ausgabe: stabile Zuweisung M (optimal für Ärzte)

Zuweisung von Ärzten auf Krankenhäuser

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Nobelpreis an L. Shapley und A. Roth, 2012

Eingabe: HR-Instanz

Ausgabe: stabile Zuweisung M (optimal für Ärzte)

Zuweisung von Ärzten auf Krankenhäuser

Nobelpreis an L. Shapley und A. Roth, 2012

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Eingabe: HR-Instanz

Ausgabe: stabile Zuweisung M (optimal für Ärzte)

1. alle Ärzte frei
2. alle Krankenhäuser komplett unbesetzt

Zuweisung von Ärzten auf Krankenhäuser

Nobelpreis an L. Shapley und A. Roth, 2012

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Eingabe: HR-Instanz
Ausgabe: stabile Zuweisung M (optimal für Ärzte)

1. alle Ärzte frei
2. alle Krankenhäuser komplett unbesetzt
3. Solange (ein $r \in R$ frei) und (Präferenzliste von r nicht-leer):
4. $\quad h \leftarrow$ erstes Krankenhaus auf Liste von r

Zuweisung von Ärzten auf Krankenhäuser

Nobelpreis an L. Shapley und A. Roth, 2012

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Eingabe: HR-Instanz

Ausgabe: stabile Zuweisung M (optimal für Ärzte)

```
alle Ärzte frei
alle Krankenhäuser komplett unbesetzt
Solange (ein r ∈ R frei) und (Präferenzliste von r nicht-leer):
    h ← erstes Krankenhaus auf Liste von r
    Falls h vollständig besetzt:
        r' ← schlechtester Arzt, der bisher h zugewiesen ist
        r' frei
```

Zuweisung von Ärzten auf Krankenhäuser

Nobelpreis an L. Shapley und A. Roth, 2012

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Eingabe: HR-Instanz

Ausgabe: stabile Zuweisung M (optimal für Ärzte)

```
alle Ärzte frei
alle Krankenhäuser komplett unbesetzt
Solange (ein r ∈ R frei) und (Präferenzliste von r nicht-leer):
    h ← erstes Krankenhaus auf Liste von r
    Falls h vollständig besetzt:
        r′ ← schlechtester Arzt, der bisher h zugewiesen ist
        r′ frei
    Weise r zu h zu
```

Zuweisung von Ärzten auf Krankenhäuser

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Nobelpreis an L. Shapley und A. Roth, 2012

Eingabe: HR-Instanz
Ausgabe: stabile Zuweisung M (optimal für Ärzte)

```
alle Ärzte frei
alle Krankenhäuser komplett unbesetzt
Solange (ein r ∈ R frei) und (Präferenzliste von r nicht-leer):
    h ← erstes Krankenhaus auf Liste von r
    Falls h vollständig besetzt:
        r′ ← schlechtester Arzt, der bisher h zugewiesen ist
        r′ frei
    Weise r zu h zu
    Falls h vollständig besetzt:
        s ← schlechtester Arzt, der bisher h zugewiesen ist
```

Zuweisung von Ärzten auf Krankenhäuser

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Nobelpreis an L. Shapley und A. Roth, 2012

Eingabe: HR-Instanz
Ausgabe: stabile Zuweisung M (optimal für Ärzte)

alle Ärzte frei
alle Krankenhäuser komplett unbesetzt
Solange (ein $r \in R$ frei) und (Präferenzliste von r nicht-leer):
    $h \leftarrow$ erstes Krankenhaus auf Liste von r
    Falls h vollständig besetzt:
        $r' \leftarrow$ schlechtester Arzt, der bisher h zugewiesen ist
        r' frei
    Weise r zu h zu
    Falls h vollständig besetzt:
        $s \leftarrow$ schlechtester Arzt, der bisher h zugewiesen ist
        Für jeden Nachfolger s' von s auf Liste von h
            Entferne s' und h aus deren jeweiligen Listen

Zuweisung von Ärzten auf Krankenhäuser

Gale–Shapley-Algorithmus (*resident oriented*, 1962)

Nobelpreis an L. Shapley und A. Roth, 2012

Eingabe: HR-Instanz

Ausgabe: stabile Zuweisung M (optimal für Ärzte)

alle Ärzte frei
alle Krankenhäuser komplett unbesetzt
Solange (ein $r \in R$ frei) und (Präferenzliste von r nicht-leer):
    $h \leftarrow$ erstes Krankenhaus auf Liste von r
    Falls h vollständig besetzt:
        $r' \leftarrow$ schlechtester Arzt, der bisher h zugewiesen ist
        r' frei
    Weise r zu h zu
    Falls h vollständig besetzt:
        $s \leftarrow$ schlechtester Arzt, der bisher h zugewiesen ist
        Für jeden Nachfolger s' von s auf Liste von h
            Entferne s' und h aus deren jeweiligen Listen
Gib die Menge der zugewiesenen Paare **zurück**.

Zuweisung von Ärzten auf Krankenhäuser

Der oben angegebene Algorithmus gibt die eindeutige stabile Zuweisung M_a aus, die aus Sicht der Ärzte optimal ist.

Zuweisung von Ärzten auf Krankenhäuser

Der oben angegebene Algorithmus gibt die eindeutige stabile Zuweisung M_a aus, die aus Sicht der Ärzte optimal ist.

Analog gibt es eine eindeutige stabile Zuweisung M_z, die aus Sicht der Krankenhäuser optimal ist.

Zuweisung von Ärzten auf Krankenhäuser

Der oben angegebene Algorithmus gibt die eindeutige stabile Zuweisung M_a aus, die aus Sicht der Ärzte optimal ist.

Analog gibt es eine eindeutige stabile Zuweisung M_z, die aus Sicht der Krankenhäuser optimal ist.

Dazwischen kann es noch weitere stabile Zuweisungen geben, die jedes Krankenhaus mindestens so gut findet wie M_a und jeder Arzt mindestens so gut wie M_z.

Zuweisung von Ärzten auf Krankenhäuser

Der oben angegebene Algorithmus gibt die eindeutige stabile Zuweisung M_a aus, die aus Sicht der Ärzte optimal ist.

Analog gibt es eine eindeutige stabile Zuweisung M_z, die aus Sicht der Krankenhäuser optimal ist.

Dazwischen kann es noch weitere stabile Zuweisungen geben, die jedes Krankenhaus mindestens so gut findet wie M_a und jeder Arzt mindestens so gut wie M_z.

Satz 5 (*Rural Hospitals*)

Für eine gegebene HR*-Instanz gelten:*

Zuweisung von Ärzten auf Krankenhäuser

Der oben angegebene Algorithmus gibt die eindeutige stabile Zuweisung M_a aus, die aus Sicht der Ärzte optimal ist.

Analog gibt es eine eindeutige stabile Zuweisung M_z, die aus Sicht der Krankenhäuser optimal ist.

Dazwischen kann es noch weitere stabile Zuweisungen geben, die jedes Krankenhaus mindestens so gut findet wie M_a und jeder Arzt mindestens so gut wie M_z.

Satz 5 (*Rural Hospitals*)

Für eine gegebene HR*-Instanz gelten:*

- *In allen stabilen Zuweisungen bekommen die gleichen Ärzte ein Krankenhaus zugewiesen.*

Zuweisung von Ärzten auf Krankenhäuser

Der oben angegebene Algorithmus gibt die eindeutige stabile Zuweisung M_a aus, die aus Sicht der Ärzte optimal ist.

Analog gibt es eine eindeutige stabile Zuweisung M_z, die aus Sicht der Krankenhäuser optimal ist.

Dazwischen kann es noch weitere stabile Zuweisungen geben, die jedes Krankenhaus mindestens so gut findet wie M_a und jeder Arzt mindestens so gut wie M_z.

Satz 5 (*Rural Hospitals*)

Für eine gegebene HR*-Instanz gelten:*

- *In allen stabilen Zuweisungen bekommen die gleichen Ärzte ein Krankenhaus zugewiesen.*
- *Jedes Krankenhaus bekommt in allen stabilen Zuweisungen die gleiche Anzahl an Ärzten.*

Zuweisung von Ärzten auf Krankenhäuser

Der oben angegebene Algorithmus gibt die eindeutige stabile Zuweisung M_a aus, die aus Sicht der Ärzte optimal ist.

Analog gibt es eine eindeutige stabile Zuweisung M_z, die aus Sicht der Krankenhäuser optimal ist.

Dazwischen kann es noch weitere stabile Zuweisungen geben, die jedes Krankenhaus mindestens so gut findet wie M_a und jeder Arzt mindestens so gut wie M_z.

Satz 5 (*Rural Hospitals*)

Für eine gegebene HR*-Instanz gelten:*

- *In allen stabilen Zuweisungen bekommen die gleichen Ärzte ein Krankenhaus zugewiesen.*
- *Jedes Krankenhaus bekommt in allen stabilen Zuweisungen die gleiche Anzahl an Ärzten.*
- *Jedes Krankenhaus, das in einer stabilen Zuweisung unterbesetzt ist, bekommt in jeder stabilen Zuweisung die gleichen Ärzten zugewiesen.*

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

- stabiles Heiratsproblem (1:1)

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

- stabiles Heiratsproblem (1:1)
- Indifferenzen in Präferenzen: schwache, starke und Super-Stabilität

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

Thema 12

Thema 13

- stabiles Heiratsproblem (1:1)
- Indifferenzen in Präferenzen: schwache, starke und Super-Stabilität

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

Thema 12

Thema 13

- stabiles Heiratsproblem (1:1)
- Indifferenzen in Präferenzen: schwache, starke und Super-Stabilität
- Berücksichtigung von Paaren

Thema 14

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

- stabiles Heiratsproblem (1:1)
- Indifferenzen in Präferenzen: schwache, starke und Super-Stabilität
- Berücksichtigung von Paaren
- Quoten: Mindestanzahlen und gemeinsame Kapazitäten

Thema 12

Thema 13

Thema 14

Thema 15

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

- stabiles Heiratsproblem (1:1)
- Indifferenzen in Präferenzen: schwache, starke und Super-Stabilität
- Berücksichtigung von Paaren
- Quoten: Mindestanzahlen und gemeinsame Kapazitäten
- (*many:many*)-Varianten: Zuweisung von Arbeitern und Firmen

Thema 12

Thema 13

Thema 14

Thema 15

Thema 17

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

- stabiles Heiratsproblem (1:1)
- Indifferenzen in Präferenzen: schwache, starke und Super-Stabilität
- Berücksichtigung von Paaren
- Quoten: Mindestanzahlen und gemeinsame Kapazitäten
- (*many:many*)-Varianten: Zuweisung von Arbeitern und Firmen
- das Projekt-Zuweisungs-Problem
  - Kapzitäten von Projekten und Dozenten
  - Studierende und Dozenten haben Präferenzen über Teilmengen (Studierende)
  - Angebote von Dozenten

Thema 12

Thema 13

Thema 14

Thema 15

Thema 16

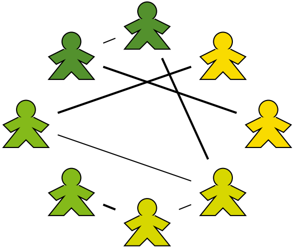
Thema 17

Zuweisung von Ärzten auf Krankenhäuser

Varianten:

- stabiles Heiratsproblem (1:1)
- Indifferenzen in Präferenzen: schwache, starke und Super-Stabilität (Thema 12, Thema 13)
- Berücksichtigung von Paaren (Thema 14)
- Quoten: Mindestanzahlen und gemeinsame Kapazitäten (Thema 15)
- (*many:many*)-Varianten: Zuweisung von Arbeitern und Firmen (Thema 17)
- das Projekt-Zuweisungs-Problem (Thema 16)
  - Kapzitäten von Projekten und Dozenten
  - Studierende und Dozenten haben Präferenzen über Teilmengen (Studierende)
  - Angebote von Dozenten
- tripartite Zuweisung
- ...

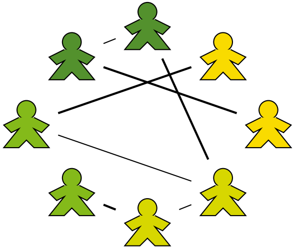
Stabiles Mitbewohnerproblem



Situation: nicht-bipartite, möglicherweise unvollständige Präferenzen über alle möglichen Mitbewohner

Ziel: paarweise Zuweisung

Stabiles Mitbewohnerproblem



Situation: nicht-bipartite, möglicherweise unvollständige Präferenzen über alle möglichen Mitbewohner

Ziel: paarweise Zuweisung

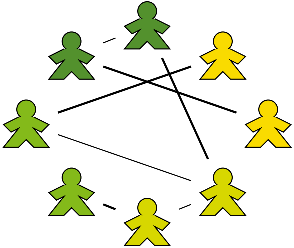
Definition 6 (Das *Stable-Roommates*-Problem (SR))

Eine SR-Instanz besteht aus

Stabiles Mitbewohnerproblem

Situation: nicht-bipartite, möglicherweise unvollständige Präferenzen über alle möglichen Mitbewohner

Ziel: paarweise Zuweisung



Definition 6 (Das *Stable-Roommates*-Problem (SR))

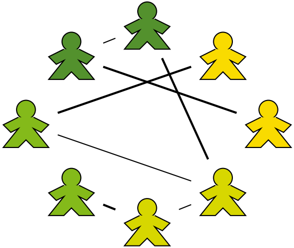
Eine SR-Instanz besteht aus

- Agenten $A = \{a_1, \ldots, a_n\}$

Stabiles Mitbewohnerproblem

Situation: nicht-bipartite, möglicherweise unvollständige Präferenzen über alle möglichen Mitbewohner

Ziel: paarweise Zuweisung

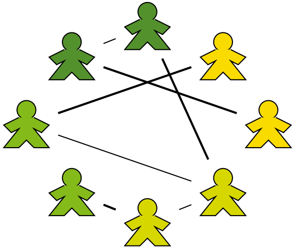


Definition 6 (Das *Stable-Roommates*-Problem (SR))

Eine SR-Instanz besteht aus

- Agenten $A = \{a_1, \ldots, a_n\}$
- $E = \{\{a_i, a_j\} \mid a_i, a_j \in A, i \neq j, \{a_i, a_j\}$ akzeptable Paare$\}$, $m = \|E\|$
- $\forall a_i \in A$: $A(a_i) = \{a_j \in E \mid \{a_i, a_j\} \in E\}$

Stabiles Mitbewohnerproblem



Situation: nicht-bipartite, möglicherweise unvollständige Präferenzen über alle möglichen Mitbewohner

Ziel: paarweise Zuweisung

Definition 6 (Das *Stable-Roommates*-Problem (SR))

Eine SR-Instanz besteht aus

- Agenten $A = \{a_1, \ldots, a_n\}$
- $E = \{\{a_i, a_j\} \mid a_i, a_j \in A, i \neq j, \{a_i, a_j\} \text{ akzeptable Paare}\}, m = \|E\|$
- $\forall a_i \in A$: $A(a_i) = \{a_j \in E \mid \{a_i, a_j\} \in E\}$
- $\forall a_i \in A$: strikte Präferenzliste über $A(a_i)$

Stabiles Mitbewohnerproblem

Sei $\mathcal{I}$ eine SR-Instanz. $M \subseteq E$ ist eine Zuweisung in $\mathcal{I}$, falls kein Agent zu mehr als einem anderen Agenten zugeordnet wird.

Stabiles Mitbewohnerproblem

Sei $\mathcal{I}$ eine SR-Instanz. $M \subseteq E$ ist eine Zuweisung in $\mathcal{I}$, falls kein Agent zu mehr als einem anderen Agenten zugeordnet wird.

Ein Paar $(a_i, a_j) \in E \setminus M$ blockiert M, falls

- Agent a_i nicht zugewiesen ist oder a_j vor $M(a_i)$ bevorzugt;
- Agent a_j nicht zugewiesen ist oder a_i vor $M(a_j)$ bevorzugt.

Stabiles Mitbewohnerproblem

Sei $\mathcal{I}$ eine SR-Instanz. $M \subseteq E$ ist eine Zuweisung in $\mathcal{I}$, falls kein Agent zu mehr als einem anderen Agenten zugeordnet wird.

Ein Paar $(a_i, a_j) \in E \setminus M$ blockiert M, falls

- Agent a_i nicht zugewiesen ist oder a_j vor $M(a_i)$ bevorzugt;
- Agent a_j nicht zugewiesen ist oder a_i vor $M(a_j)$ bevorzugt.

M heißt stabil, falls kein Paar M blockiert.

Stabiles Mitbewohnerproblem

Sei $\mathcal{I}$ eine SR-Instanz. $M \subseteq E$ ist eine Zuweisung in $\mathcal{I}$, falls kein Agent zu mehr als einem anderen Agenten zugeordnet wird.

Ein Paar $(a_i, a_j) \in E \setminus M$ blockiert M, falls

- Agent a_i nicht zugewiesen ist oder a_j vor $M(a_i)$ bevorzugt;
- Agent a_j nicht zugewiesen ist oder a_i vor $M(a_j)$ bevorzugt.

M heißt stabil, falls kein Paar M blockiert.

$\mathcal{I}$ heißt lösbar, falls es eine stabile Zuweisung in $\mathcal{I}$ gibt.

Stabiles Mitbewohnerproblem

Sei $\mathcal{I}$ eine SR-Instanz. $M \subseteq E$ ist eine Zuweisung in $\mathcal{I}$, falls kein Agent zu mehr als einem anderen Agenten zugeordnet wird.

Ein Paar $(a_i, a_j) \in E \setminus M$ blockiert M, falls

- Agent a_i nicht zugewiesen ist oder a_j vor $M(a_i)$ bevorzugt;
- Agent a_j nicht zugewiesen ist oder a_i vor $M(a_j)$ bevorzugt.

M heißt stabil, falls kein Paar M blockiert.

$\mathcal{I}$ heißt lösbar, falls es eine stabile Zuweisung in $\mathcal{I}$ gibt.

Konzepte

- SR-Instanzen sind nicht immer lösbar.
- Es gibt immer eine stabile Partitionierung.
- fast-stabile Zuweisungen berücksichtigen geringe Kompromisse.

Stabiles Mitbewohnerproblem

Sei $\mathcal{I}$ eine SR-Instanz. $M \subseteq E$ ist eine Zuweisung in $\mathcal{I}$, falls kein Agent zu mehr als einem anderen Agenten zugeordnet wird.

Ein Paar $(a_i, a_j) \in E \setminus M$ blockiert M, falls

- Agent a_i nicht zugewiesen ist oder a_j vor $M(a_i)$ bevorzugt;
- Agent a_j nicht zugewiesen ist oder a_i vor $M(a_j)$ bevorzugt.

M heißt stabil, falls kein Paar M blockiert.

$\mathcal{I}$ heißt lösbar, falls es eine stabile Zuweisung in $\mathcal{I}$ gibt.

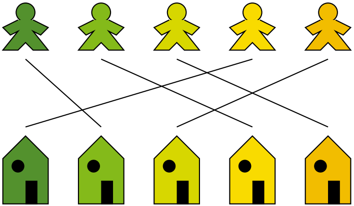
Konzepte

- SR-Instanzen sind nicht immer lösbar.
- Es gibt immer eine stabile Partitionierung.
- fast-stabile Zuweisungen berücksichtigen geringe Kompromisse.

Thema 18

Thema 19

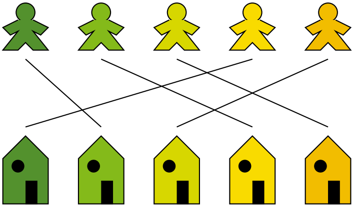
Zuweisung von Häusern



Situation: einseitige Präferenzen von Agenten über Häuser

Ziel: (1 : 1)-Zuweisung

Zuweisung von Häusern



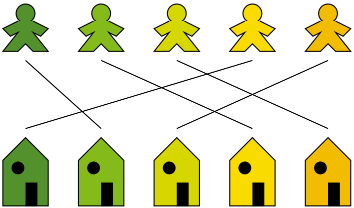
Situation: einseitige Präferenzen von Agenten über Häuser

Ziel: (1 : 1)-Zuweisung

Optimalitätskriterien

- Pareto-Optimalität
- Popularität
- profilbasierte Optimalität: rang-maximale, gierige & großzügige Zuweisungen

Zuweisung von Häusern



Situation: einseitige Präferenzen von Agenten über Häuser

Ziel: $(1:1)$-Zuweisung

Optimalitätskriterien

- Pareto-Optimalität
- Popularität
- profilbasierte Optimalität: rang-maximale, gierige & großzügige Zuweisungen

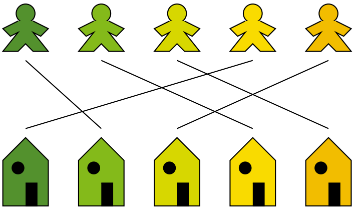
Varianten:

- *Housing Markets*: bestehende Zuweisung, individuelle Rationalität
- gewichtete Präferenzen
- (*many*:1)-Verallgemeinerung: WG mit Kapazitäten

Zuweisung von Häusern

Situation: einseitige Präferenzen von Agenten über Häuser

Ziel: (1 : 1)-Zuweisung



Optimalitätskriterien

- Pareto-Optimalität
- Popularität
- profilbasierte Optimalität: rang-maximale, gierige & großzügige Zuweisungen

Thema 6

Thema 7

Thema 8

Thema 11

Thema 9

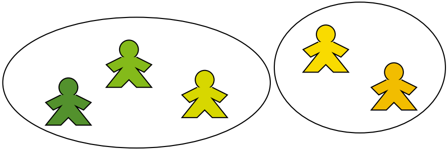
Thema 10

Var

- *Housing Markets*: bestehende Zuweisung, individuelle Rationalität
- gewichtete Präferenzen
- (*many*:1)-Verallgemeinerung: WG mit Kapazitäten

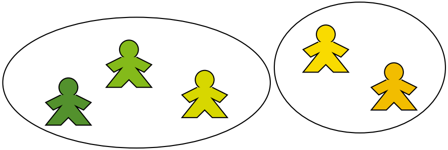
hedonische Spiele

Situation: Koalitionsbildung; Zufriedenheit der Spieler hängt nur von eigener Koalition ab.



hedonische Spiele

Situation: Koalitionsbildung; Zufriedenheit der Spieler hängt nur von eigener Koalition ab.



Definition 7

Ein hedonisches Spiel $(N, \succeq)$ besteht aus:

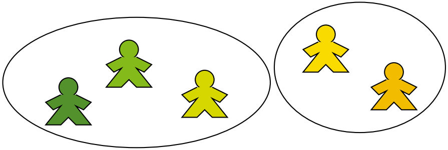
- endlicher Spielermenge $N = \{1, \ldots, n\}$,
- Präferenzprofil $\succeq = (\succeq_1, \ldots, \succeq_n)$
  mit $\succeq_i$ totaler Präferenzrelation über $\mathcal{N}_i = \{C \subseteq N \mid i \in C\}$ für $i \in N$.

Also ist $\succeq$ reflexiv, transitiv, nicht notwendigerweise antisymmetrisch und total.

Eine Teilmenge $C \subseteq N$ heißt Koalition.
Ein Partitionierung Γ von N heißt Koalitionsstruktur. $\Gamma(i) \in \Gamma$ mit $i \in \Gamma(i)$.

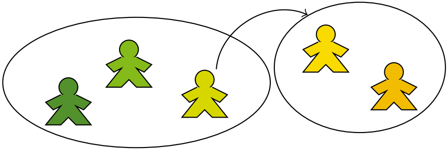
hedonische Spiele

Ziel: stabile Koalitionsstruktur finden



hedonische Spiele

Ziel: stabile Koalitionsstruktur finden



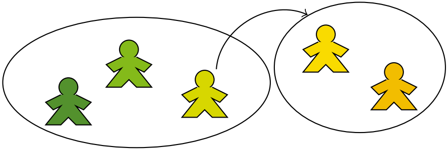
Definition 8

Sei ein hedonisches Spiel $(N, \succeq)$. Eine Koalitionsstruktur heißt

- Nash-stabil, wenn $\forall i \in N$, $\forall C \in \Gamma$: $\Gamma(i) \succeq C$.

hedonische Spiele

Ziel: stabile Koalitionsstruktur finden

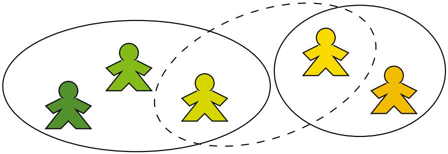


Definition 8

Sei ein hedonisches Spiel $(N, \succeq)$. Eine Koalitionsstruktur heißt

- Nash-stabil, wenn $\forall i \in N, \forall C \in \Gamma$: $\Gamma(i) \succeq C$.
- Varianten: individuell stabil und vertraglich individuell stabil

hedonische Spiele



Ziel: stabile Koalitionsstruktur finden

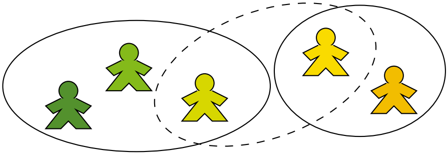
Definition 8

Sei ein hedonisches Spiel $(N, \succeq)$. Eine Koalitionsstruktur heißt

- kernstabil, wenn es keine blockierende Koalition gibt,
  d. h., $\forall C \subseteq N, \exists i \in C \colon \Gamma(i) \geq C$.

hedonische Spiele

Ziel: stabile Koalitionsstruktur finden



Definition 8

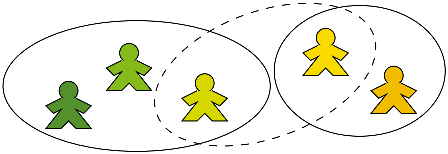
Sei ein hedonisches Spiel $(N, \succeq)$. Eine Koalitionsstruktur heißt

- kernstabil, wenn es keine blockierende Koalition gibt,
  d. h., $\forall C \subseteq N, \exists i \in C : \Gamma(i) \geq C$.

Problemstellungen:

- Darstellungen: individuell rational, additiv separabel, ...
- Existenz:
  - *Pareto-optimale* und *individuell rationale* Aufteilung existiert immer: *Preference-Refinement*-Algorithmus

hedonische Spiele



Ziel: stabile Koalitionsstruktur finden

Definition 8

Sei ein hedonisches Spiel $(N, \succeq)$. Eine Koalitionsstruktur heißt

- kernstabil, wenn es keine blockierende Koalition gibt,

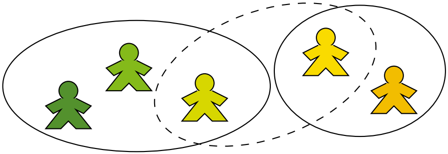
  d. h., $\forall C \subseteq N, \exists i \in C\colon \Gamma(i) \geq C$.

Problemstellungen:

- Darstellungen: individuell rational, additiv separabel, ...
- Existenz:
  - *Pareto-optimale* und *individuell rationale* Aufteilung existiert immer: *Preference-Refinement*-Algorithmus
  - kernstabile Aufteilung existiert nicht immer: *Top-Responsiveness*, *Top-Covering*-Algorithmus

hedonische Spiele

Ziel: stabile Koalitionsstruktur finden



Definition 8

Sei ein hedonisches Spiel $(N, \succeq)$. Eine Koalitionsstruktur heißt

- kernstabil, wenn es keine blockierende Koalition gibt,

  d. h., $\forall C \subseteq N, \exists i \in C \colon \Gamma(i) \geq C$.

Problemstellungen:

- Darstellungen: individuell rational, additiv separabel, . . .
- Existenz:
  - *Pareto-optimale* und *individuell rationale* Aufteilung existiert immer: *Preference-Refinement*-Algorithmus
  - kernstabile Aufteilung existiert nicht immer: *Top-Responsiveness*, *Top-Covering*-Algorithmus

Thema 21

Thema 20

Vortragstermine I

	Thema	Datum
1	Cake-Cutting – Proportionalität	10.01.2018
2	Cake-Cutting – Neidfreiheit	10.01.2018
3	Aufteilung unteilbarer Güter – Scheidungsformel	10.01.2018
4	Proportionale Aufteilung unteilbarer Güter	10.01.2018
5	Wege zur Stabilität	10.01.2018
6	HA – Pareto-Optimaliät	17.01.2018
7	HA mit Kapazitäten – Pareto-Optimalität	17.01.2018
8	HA – Popularität	17.01.2018
9	HA mit Gewichten – Popularität	17.01.2018
10	HA – rang-maximale Zuweisungen	17.01.2018
11	HA – gierige und großzügige Zuweisungen	17.01.2018

Vortragstermine II

	Thema	Datum
12	HR mit Indifferenzen – schwache Stabilität	24.01.2018
13	HR mit Indifferenzen – starke & Super-Stabilität	24.01.2018
14	HR mit Paaren	24.01.2018
15	HR mit Quoten	24.01.2018
16	Das Projekt-Zuweisungs-Problem	24.01.2018
17	Das Arbeiter-Firmen-Problem	24.01.2018
18	SR – stabile Partitionierungen	31.01.2018
19	SR – fast-stabile Zuweisungen	31.01.2018
20	Hedonische Spiele – Kernstabilität	31.01.2018
21	Hedonische Spiele – Pareto-Optimalität	31.01.2018

Literatur

F. Brandt, V. Conitzer, U. Endriss, J. Lang und A. Procaccia, Editoren.
Handbook of Computational Social Choice.
Cambridge University Press, 2016.

D. Gusfield und R. Irving
The Stable Marriage Problem: Structure and Algorithms.
MIT Press, 1989.

D. Knuth.
Stable Marriage and its Relation to Other Combinatorial Problems.
volume 10 of CRM Proceedings and Lecture Notes, American Mathematical Society, 1997. Original: Mariages Stables, Les Presses de L' Université de446 Montreal, 1976.

D. Manlove
Algorithmics Of Matching Under Preferences.
Volume 2 of Theoretical computer science. World Scientific Publishing, 2013.

...und jede Menge Referenzen darin.

Ausblick

nächste Schritte

1. Thema auswählen bis Dienstag,
2. zugehörige Aufgabenstellung per E-Mail erhalten,
3. bei Unklarheiten zeitnah nachfragen,
4. beginnen!