Das Stable-Matching-Problem

Algorithmen und Datenstrukturen
VU 186.866, 5.5h, 8 ECTS, 2023S
Letzte Änderung: 1. März 2024
Vorlesungsfolien

Stable-Matching-Problem

Gegeben seien n Kinder und n Gastfamilien, die an einem Austauschprogramm für Schülerinnen und Schülern teilnehmen.

Ziel: Finde eine passende Zuordnung von Kindern zu Gastfamilien.

- Jedes Kind hat eine Präferenzliste von Gastfamilien.
- Jede Gastfamilie hat eine Präferenzliste von Kindern.

Kinder: Xaver, Yvonne, Zola.

Gastfamilien: Abel, Boole, Church.

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Xaver	Abel	Boole	Church
Yvonne	Boole	Abel	Church
Zola	Abel	Boole	Church

Präferenzlisten der Kinder

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Abel	Yvonne	Xaver	Zola
Boole	Xaver	Yvonne	Zola
Church	Xaver	Yvonne	Zola

Präferenzlisten der Familien

Stable-Matching-Problem

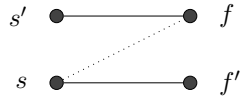
Perfektes Matching: Jedem Kind wird genau eine Familie zugewiesen.

- Jedes Kind bekommt genau eine Gastfamilie.
- Jede Gastfamilie bekommt genau ein Kind.

Instabiles Paar:

- In einem Matching M ist ein nicht zugewiesenes Paar s-f instabil, wenn ein Kind s und eine Familie f sich gegenseitig gegenüber ihren aktuellen Partnern bevorzugen.
- Das instabile Paar s-f könnte seine Situation durch Verlassen der aktuellen Partner verbessern.

s für *student*, f für *family*



Stable-Matching-Problem

Stable Matching: Perfektes Matching ohne instabile Paare. Es besteht daher für kein Paar der Anreiz, durch gemeinsames Handeln die Zuteilung zu unterlaufen.

Stable-Matching-Problem: Ausgehend von den Präferenzlisten von n Kindern und n Familien, finde ein Stable Matching, wenn eines existiert.

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Xaver	Abel	Boole	Church
Yvonne	Boole	Abel	Church
Zola	Abel	Boole	Church

Präferenzlisten der Kinder

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Abel	Yvonne	Xaver	Zola
Boole	Xaver	Yvonne	Zola
Church	Xaver	Yvonne	Zola

Präferenzlisten der Familien

Stable-Matching-Problem

Frage: Ist die Zuordnung X-C, Y-B, Z-A stabil?

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Xaver	Abel	Boole	Church
Yvonne	Boole	Abel	Church
Zola	Abel	Boole	Church

Präferenzlisten der Kinder

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Abel	Yvonne	Xaver	Zola
Boole	Xaver	Yvonne	Zola
Church	Xaver	Yvonne	Zola

Präferenzlisten der Familien

Stable-Matching-Problem

Frage: Ist die Zuordnung X-C, Y-B, Z-A stabil?
Antwort: Nein. Boole und Xaver können ihre Situation verbessern (Xaver-Boole ist ein instabiles Paar).

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Xaver	Abel	Boole	Church
Yvonne	Boole	Abel	Church
Zola	Abel	Boole	Church

Präferenzlisten der Kinder

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Abel	Yvonne	Xaver	Zola
Boole	Xaver	Yvonne	Zola
Church	Xaver	Yvonne	Zola

Präferenzlisten der Familien

Stable-Matching-Problem

Frage: Ist die Zuordnung X-A, Y-B, Z-C stabil?
Antwort: Ja. Es gibt kein instabiles Paar.

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Xaver	Abel	Boole	Church
Yvonne	Boole	Abel	Church
Zola	Abel	Boole	Church

Präferenzlisten der Kinder

höchste Präferenz ↓ niedrigste Präferenz ↓

	1.	2.	3.
Abel	Yvonne	Xaver	Zola
Boole	Xaver	Yvonne	Zola
Church	Xaver	Yvonne	Zola

Präferenzlisten der Familien

Stable-Matching-Problem: Fragen

Frage: Gibt es immer ein Stable Matching?
Hinweis: Das ist nicht von vornherein klar!

Frage: Kann ein Stable Matching effizient gefunden werden?
Hinweis: Brute-Force-Ansatz (alle möglichen Zuordnungen ausprobieren) betrachtet $n!$ viele mögliche Lösungen, was extrem ineffizient ist.

Gale-Shapley-Algorithmus: Wir stellen einen Algorithmus vor, mit dem wir beide Fragen mit „Ja“ beantworten können.

Gale–Shapley-Algorithmus (GS-Algorithmus)

Gale-Shapley-Algorithmus:

- 1962 gaben David Gale und Lloyd Shapley einen Algorithmus zum Auffinden von Stable Matchings an.
- Shapley bekam für seine Arbeiten (einschließlich Stable Matching) den Wirtschaftsnobelpreis 2012.

  D. Gale and L. S. Shapley: College Admissions and the Stability of Marriage, American Mathematical Monthly, Vol. 69, 1962, Seite 9–15

Anwendung: Das Stable-Matching-Problem hat viele Anwendungen, z.B. bei der Zuteilung von Medizinstudenten an das erste Krankenhaus, in dem sie ihren Turnus ableisten.

Gale and Shapley 1962

COLLEGE ADMISSIONS AND THE STABILITY OF MARRIAGE

D. GALE* AND L. S. SHAPLEY, Brown University and the RAND Corporation

1. Introduction. The problem with which we shall be concerned relates to the following typical situation: A college is considering a set of n applicants of which it can admit a quota of only q. Having evaluated their qualifications, the admissions office must decide which ones to admit. The procedure of offering admission only to the q best-qualified applicants will not generally be satisfactory, for it cannot be assumed that all who are offered admission will accept. Accordingly, in order for a college to receive q acceptances, it will generally have to offer to admit more than q applicants. The problem of determining how many and which ones to admit requires some rather involved guesswork. It may not be known (a) whether a given applicant has also applied elsewhere; if this is known it may not be known (b) how he ranks the colleges to which he has applied; even if this is known it will not be known (c) which of the other colleges will offer to admit him. A result of all this uncertainty is that colleges can expect only that the entering class will come reasonably close in numbers to the desired quota, and be reasonably close to the attainable optimum in quality.

Gale–Shapley-Algorithmus (GS-Algorithmus)

Gale-Shapley-Algorithmus:

```
Kennzeichne jede Familie/jedes Kind als frei
while ein Kind ist frei und kann noch eine Familie wählen
    Wähle solch ein Kind s aus
    f ist erste Familie in der Präferenzliste von s,
    die von s noch nicht gewählt wurde
    if f ist frei
        Kennzeichne s und f als einander zugeordnet
    elseif f bevorzugt s gegenüber ihrem aktuellen Partner s'
        Kennzeichne s und f als einander zugeordnet und s' als frei
    else
        f weist s zurück
```

Beispiel für Ablauf

Ausgangssituation:

- 4 Kinder (W-Z) mit Präferenzlisten (links).
- 4 Familien (A-D) mit Präferenzlisten (rechts).

	1.	2.	3.	4.
W	B	A	C	D
X	C	B	A	D
Y	B	C	A	D
Z	B	A	D	C

	1.	2.	3.	4.
A	X	W	Y	Z
B	W	Y	X	Z
C	Z	Y	W	X
D	X	W	Y	Z

Beispiel für Ablauf

Erste Zuordnung:

- Wähle erstes freies Kind aus (z.B. W).
- Dieses wählt die erste Familie in seiner Präferenzliste aus (in diesem Fall B).
- Da B frei ist, werden die beiden einstweilig einander zugeordnet.
- Aktuelle Zuordnungen: W-B.

	1.	2.	3.	4.
W	B	A	C	D
X	C	B	A	D
Y	B	C	A	D
Z	B	A	D	C

	1.	2.	3.	4.
A	X	W	Y	Z
B	W	Y	X	Z
C	Z	Y	W	X
D	X	W	Y	Z

Beispiel für Ablauf

Zweite Zuordnung:

- Wähle nächstes freies Kind aus (z.B. X).
- Dieses wählt die erste Familie in seiner Präferenzliste aus (in diesem Fall C).
- Da C frei ist, werden die beiden einstweilig einander zugeordnet.
- Aktuelle Zuordnungen: W-B, X-C.

	1.	2.	3.	4.
W	B	A	C	D
X	C	B	A	D
Y	B	C	A	D
Z	B	A	D	C

	1.	2.	3.	4.
A	X	W	Y	Z
B	W	Y	X	Z
C	Z	Y	W	X
D	X	W	Y	Z

Beispiel für Ablauf

Dritte Zuordnung:

- Wähle nächstes freies Kind aus (z.B. Y).
- Dieses wählt die erste Familie in seiner Präferenzliste aus (in diesem Fall B).
- B ist aber W zugeordnet. In der Präferenzliste von B steht W vor Y, daher lässt B das Y abblitzen.
- Y wählt die nächste Familie in seiner Präferenzliste aus (in diesem Fall C).
- C bevorzugt Y vor ihrem Partner X, daher wird ihre Zuordnung zu X gelöst und statt dessen werden C und Y einander zugeordnet.
- Aktuelle Zuordnungen: W-B, Y-C.

	1.	2.	3.	4.
W	B	A	C	D
X	C	B	A	D
Y	B	C	A	D
Z	B	A	D	C

	1.	2.	3.	4.
A	X	W	Y	Z
B	W	Y	X	Z
C	Z	Y	W	X
D	X	W	Y	Z

Beispiel für Ablauf

Vierte Zuordnung:

- Wähle nächstes freies Kind aus, es ist X, das wieder frei geworden ist.
- X wählt die zweite Familie (B) auf seiner Präferenzliste aus.
- B bevorzugt W vor X.
- X wählt die dritte Familie (A) auf seiner Präferenzliste aus.
- A ist frei und die beiden werden einander zugeordnet.
- Aktuelle Zuordnungen: W-B, X-A, Y-C.

	1.	2.	3.	4.
W	B	A	C	D
X	C	B	A	D
Y	B	C	A	D
Z	B	A	D	C

	1.	2.	3.	4.
A	X	W	Y	Z
B	W	Y	X	Z
C	Z	Y	W	X
D	X	W	Y	Z

Beispiel für Ablauf

Fünfte Zuordnung:

- Wähle nächstes freies Kind aus (nur mehr Z übrig).
- Z wählt die erste Familie (B) auf seiner Präferenzliste aus. B bevorzugt aber W vor Z.
- Z wählt die zweite Familie (A) auf seiner Präferenzliste aus. A bevorzugt aber X vor Z.
- Z wählt die dritte Familie (D) auf seiner Präferenzliste aus.
- D ist frei, also werden Z und D einander zugeordnet.
- Aktuelle Zuordnungen: W-B, X-A, Y-C, Z-D.
- Es ist nun kein Kind mehr frei, und der Algorithmus terminiert. Wir haben ein Stable Matching gefunden.

	1.	2.	3.	4.
W	B	A	C	D
X	C	B	A	D
Y	B	C	A	D
Z	B	A	D	C

	1.	2.	3.	4.
A	X	W	Y	Z
B	W	Y	X	Z
C	Z	Y	W	X
D	X	W	Y	Z

Korrektheitsbeweis: Terminierung

Operation 1: Kinder wählen Familien in absteigender Reihenfolge aus.

Operation 2: Sobald eine Familie zugewiesen wurde, bleibt sie zugewiesen, die Zuteilung kann sich aber ändern.

Behauptung: Algorithmus terminiert nach höchstens n^2 Iterationen der while-Schleife.

Beweis: In jeder Iteration der while-Schleife wählt ein Kind eine Familie aus. Es gibt nur n^2 Möglichkeiten dafür. □

	1.	2.	3.	4.	5.
Valentin	A	B	C	D	E
Werner	B	C	D	A	E
Xaver	C	D	A	B	E
Yvonne	D	A	B	C	E
Zola	A	B	C	D	E

	1.	2.	3.	4.	5.
Abel	W	X	Y	Z	V
Boole	X	Y	Z	V	W
Church	Y	Z	V	W	X
Dijkstra	Z	V	W	X	Y
Euler	V	W	X	Y	Z

$n(n-1)+1$ vorläufige Zuordnungen erfoderlich

Korrektheitsbeweis: Abschluss

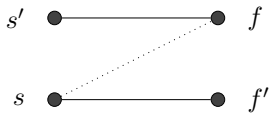
Behauptung: Alle Kinder und Familien werden zugewiesen.
Beweis: (durch Widerspruch)

- Angenommen, Kind s wurde nach Terminierung des Algorithmus nicht zugewiesen.
- Dann wurde auch eine Familie (z.B. f) nach Terminierung des Algorithmus nicht zugewiesen.
- Damit wurde f nie ausgewählt.
- Aber s hat jede Familie in der Liste ausgewählt, da es ja am Ende nicht zugewiesen wurde. $\square$

Korrektheitsbeweis: Stabilität

Behauptung: Nachdem der Algorithmus terminiert, existieren keine instabilen Paare.
Beweis: (durch Widerspruch)

- Angenommen, s-f ist ein instabiles Paar: s bevorzugt f gegenüber seinem aktuellen Partner f' und f bevorzugt s gegenüber seinem aktuellen Partner s' in einem Gale-Shapley-Matching.



Korrektheitsbeweis: Stabilität

Behauptung: Nachdem der Algorithmus terminiert, existieren keine instabilen Paare.
Beweis: (durch Widerspruch)

- Angenommen, s-f ist ein instabiles Paar: s bevorzugt f gegenüber seinem aktuellen Partner f' und f bevorzugt s gegenüber seinem aktuellen Partner s' in einem Gale-Shapley-Matching.
- Fall 1: s hat f nie ausgewählt.
  - $\Rightarrow$ s bevorzugt seinen GS-Partner f' gegenüber f.
  - $\Rightarrow$ s-f ist nicht instabil.
- Fall 2: s hat f ausgewählt.
  - $\Rightarrow$ f hat s zurückgewiesen (gleich oder später)
  - $\Rightarrow$ f bevorzugt s' gegenüber s.
  - $\Rightarrow$ s-f ist nicht instabil.
- In jedem Fall ist s-f nicht instabil, was ein Widerspruch ist. □

Kinder wählen Familien in absteigender Reihenfolge der Präferenzen aus.
Bei Familien kann sich die Situation nur verbessern

Zusammenfassung

Stable-Matching-Problem: Gegeben seien n Kinder und n Familien und ihre Präferenzen. Finde ein Stable Matching, wenn eines existiert.

Frage: Gibt es immer ein Stable Matching?

Frage: Kann ein Stable Matching effizient gefunden werden?

Gale-Shapley-Algorithmus: Findet garantiert ein Stable Matching für jede Problemeingabe (in Form von Präferenzlisten). Da der Algorithmus höchstens n^2 Iterationen benötigt, findet er ein Stable Matching auf effiziente Weise.

Effiziente Implementierung

Zeitaufwand:

- Stable Matching benötigt höchstens n^2 Iterationen.
- Einzelne Schritte in einer Iteration können naiv (z.B. lineare Suche in Listen) implementiert werden, das benötigt eine "Größenordnung" von n Schritten.
- Dann benötigt man insgesamt höchstens eine "Größenordnung" von n^3 Schritten.
- Das ist immer noch besser, als ein Brute-Force-Ansatz der alle möglichen Zuordnungen durchprobiert (es gibt $n!$ mögliche Zuordnungen).

Nächste Vorlesung:

- Mit asymptotischer Analyse können wir das exakt ausdrücken.
- Mit besseren Datenstrukturen können wir das auch schneller lösen.