- Erfordert viel weniger Arbeit als das Aufzählen sämtlicher Wege. Das Potential p gibt ein effizientes Zertifikat für die Optimalität des Weges P.

<u>Bew.:</u> (von Satz 6)

<u>max $\leq$ min:</u> Sei $P = (v_0, a_1, v_1, \ldots, a_m, v_m)$, $s = v_0$, $t = v_m$ eine s-t-Kantenfolge, und sei $p: V \to \mathbb{R}$ ein Potential.

Dann gilt

$$\ell(P) = \sum_{i=1}^{m} \ell(a_i) \geq \sum_{i=1}^{m} \left(p(v_i) - p(v_{i-1})\right) = p(t) - p(s).$$

D.h. die Länge jedes s-t-Weges ist immer wenigstens so groß wie die Differenz $p(t) - p(s)$.

<u>max $\geq$ min:</u> Für $v \in V$ definiere $p(v) = \mathrm{dist}(s, v)$.

Dann ist $p(t) - p(s) = \mathrm{dist}(s,t)$, weil $\mathrm{dist}(s,s) = 0$.

Überprüfe jetzt (genauso wie im Beweis von Satz 5 „$\Leftarrow$"), dass p ein Potential ist. $\boxtimes$

<u>Bem.:</u> Ein optimales Potential ist nie eindeutig bestimmt: Falls p optimal, dann ist auch $p + c$ für $c \in \mathbb{R}$ optimal.

§ 3 Berechnung kürzester Wege

Algorithmus von Bellman (1958) und Ford (1956)

<u>Eingabe</u>
- $D = (V, A)$ gerichteter Graph mit $n = |V|$
- $s \in V$ Startknoten
- $\ell : A \to \mathbb{R}$ Kantenlängenfunktion

<u>Ausgabe</u>
- $d_0, d_1, \dots, d_n : V \to \mathbb{R} \cup \{\infty\}$
- $g : V \to V$

in Pseudocode (Einrückung wie in der Programmiersprache „python")

```
d_0(s) = 0,   d_0(v) = ∞   ∀ v ∈ V \ {s}
for k = 0 to n-1:
  d_{k+1}(v) = d_k(v)   ∀ v ∈ V
  for (u,v) ∈ A:
    if d_{k+1}(v) > d_k(u) + ℓ((u,v)):
      d_{k+1}(v) = d_k(u) + ℓ((u,v))
      g(v) = u
if d_n ≠ d_{n-1}:
  output „∃ ger. negative Kreis, der von s erreichbar ist."
```

Satz 1 Nach Ablauf der k-ten Iteration der for-Schleife gilt

$$d_k(v) = \min\{\ell(P) : P \text{ ist } s\text{-}v\text{-Kantenfolge, die } \leq k \text{ Kanten enthält}\}.$$

Bew.: Induktion nach k (trivial).

Satz 2 Nach Ablauf des Algorithmus gilt $d_n = d_{n-1}$ genau dann, wenn alle von s aus erreichbaren gerichteten Kreise eine nicht-negative Länge besitzen.

Bew.: $\rightarrow$ Aufgabe

Bem. 3 a) Laufzeit: proportional zu $|V| \cdot |A| \leq |V|^3$

b) Falls D keine gerichteten Kreise negativer Länge enthält, die von s aus erreichbar sind, dann ist

$$d_{n-1}(v) = \operatorname{dist}(s, v)$$ (vergleiche auch mit Satz 2.6),

und $v, g(v), g(g(v)), \ldots, s$ ist Umkehrung eines kürzesten s-v-Weges.

Frage: Warum betrachtet man negative Kantenlängen?

Antwort: Manchmal nützlich.

z.B. um längste s-t-Wege zu finden

(d.h. finde kürzesten, negativen Weg)

Bsp. 4 (Rucksackproblem)

Rucksack hat 8 l Volumen, 5 nützliche Gegenstände

Gegenstand i	Volumen a_i	Nutzen c_i
1	3	7
2	2	3
3	2	5
4	1	4
5	5	4

Aufgabe: Finde Auswahl von $1, 2, \ldots, 5$, so dass die Gegenstände in den Rucksack passen und ihr summierter Nutzen maximal ist:

$$\max \{ 7x_1 + 3x_2 + 5x_3 + 4x_4 + 4x_5 : x_1, \ldots, x_5 \in \{0,1\}, \; 3x_1 + 2x_2 + 2x_3 + x_4 + 5x_5 \leq 8 \}.$$

Formulierung als kürzeste Wege Problem

Definiere $D = (V, A)$ und $l : A \to \mathbb{R}$ wie folgt

Knoten V: $(i, x) \in V$ $\quad i = 0, 1, \dots, 6, \quad x = 0, 1, \dots, 8$

Kanten A:

$$\left.\begin{array}{l} - \ ((i-1, x), (i, x)) \text{ mit Länge } 0 \\ - \ ((i-1, x), (i, x+a_i)) \text{ mit Länge } -c_i \end{array}\right\} i = 1, \dots, 5$$

$-$ $((5, x), (6, 8))$ mit Länge 0 für alle x

Dann: Kürzeste $(0,0)-(6,8)$-Wege geben optimale Auswahl.

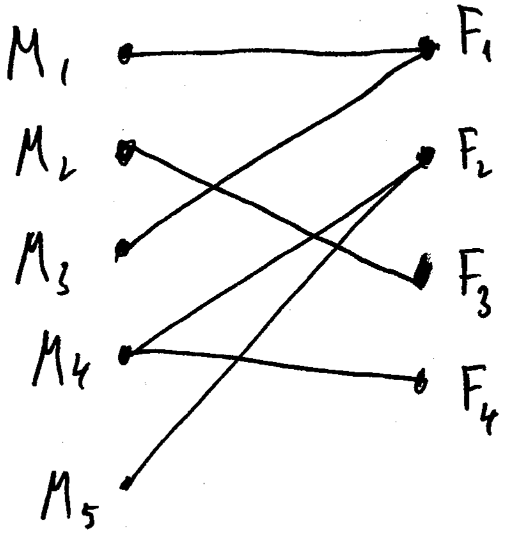
Kapitel II Matchings in bipartiten Graphen

Optimale Partnerfindung: m Männer $M_1, \dots, M_m$
f Frauen, $F_1, \dots, F_f$.

Jedes Paar (M_i, F_j) gibt an, ob sie prinzipiell heiraten würden.

Ziel: Finde möglichst viele Paare, die heiraten.

Modellierung in ungerichteten bipartiten Graphen



$\{M_1, F_1\}$

$\{M_2, F_3\}$

$\{M_4, F_4\}$

$\{M_5, F_2\}$

ist ein maximales Matching.

§ 1 Grundbegriffe

<u>Def. 1</u> Ein (ungerichteter) Graph $G = (V,E)$ ist ein Paar, bestehend aus einer endlichen Menge V (Menge der <u>Knoten</u>) und einer Menge $E \subseteq \{\{v,w\} : v,w \in V, v \neq w\}$ (Menge der (ungerichteten) <u>Kanten</u>). M.a.W. G beschreibt eine symmetrische irreflexive Relation auf $V \times V$.

Definiere <u>Kantenfolge</u>, <u>Weg</u>, <u>Kreis</u> wie bei gerichteten Graphen.

<u>Def. 2</u> Sei $G = (V,E)$ ein Graph mit Knoten $v, w \in V$.

a) v und w heißen <u>benachbart</u> (<u>adjazent</u>), falls $\{v,w\} \in E$.
v heißt dann ein <u>Nachbar</u> von w, und umgekehrt.

b) v und w heißen wegzusammenhängend, falls es einen v-w-Weg gibt.