

Verfeinertes Schema:

$M := \emptyset$

while \exists augmentierender Pfad bzgl. M **do**

$l :=$ Länge eines kürzesten augmentierenden Pfades bzgl. M
bestimme eine bzgl. „ \subseteq “ maximale Menge $\{Q_1, \dots, Q_k\}$
augmentierender Pfade bzgl. M , die alle Länge l haben und
knotendisjunkt sind

$M := M \oplus Q_1 \oplus \dots \oplus Q_k$

od

Korollar 144

Die obige *while*-Schleife wird höchstens $\mathcal{O}\left(|V|^{\frac{1}{2}}\right)$ mal durchlaufen.

3. Maximum Matchings in bipartiten Graphen

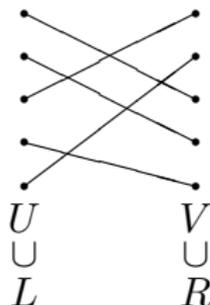
Sei $G = (U, V, E)$ ein bipartiter Graph, M ein Matching in G .

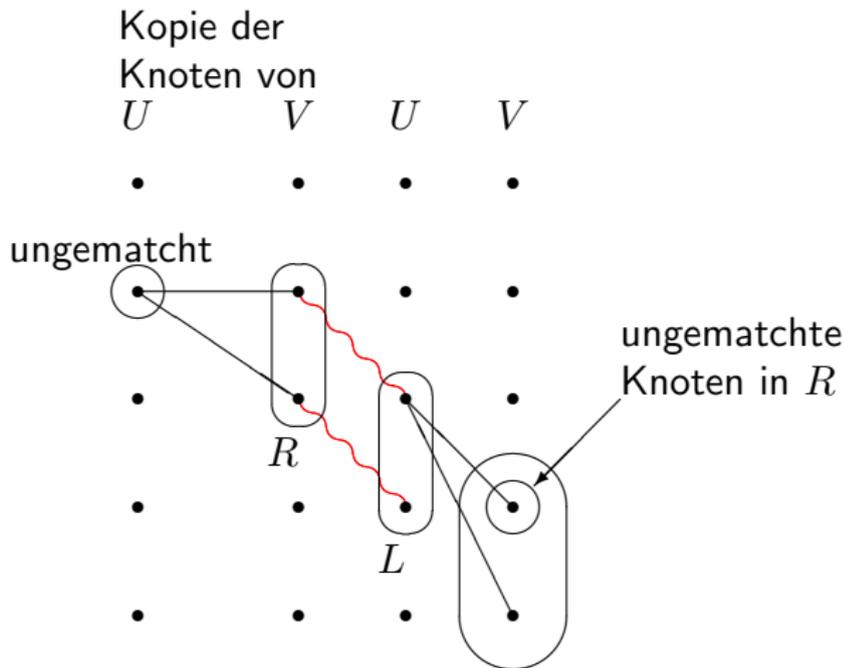
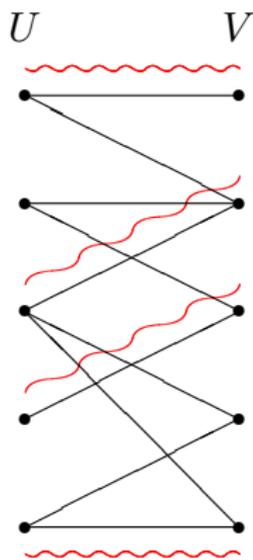
Zur Bestimmung der Länge eines kürzesten augmentierten Pfades bzgl. M führen wir eine **simultane alternierende BFS** durch, die von allen in M ungematchten Knoten in U aus startet.

```

for alle  $v \in U \cup V$  do  $label[v] := 0$  od
 $R := \emptyset$ ;  $l := 1$ 
for alle ungematchten Knoten  $v \in U$  do
    for alle  $\{v, w\} \in E$  do  $label[w] := 1$ ;  $R := R \cup \{w\}$  od
od
while  $R \neq \emptyset$  and  $R$  enthält keinen ungematchten Knoten do
     $L := \emptyset$ ;  $l ++$ 
    for  $w \in R, \{v, w\} \in M$  do  $L := L \cup \{v\}$ ;  $label[v] := l$  od
     $R := \emptyset$ ;  $l ++$ 
    for alle  $v \in L, \{v, w\} \in E \setminus M$  do
        if  $label[w] = 0$  then
             $R := R \cup \{w\}$ ;  $label[w] := l$ 
        fi
    od
od
 $R :=$  Menge der ungematchten Knoten in  $R$ 

```





Nachdem wir die Länge l eines **kürzesten augmentierenden Pfades** bzgl. M ermittelt haben, führen wir **nacheinander** von jedem ungematchten Knoten in U aus eine (zwischen ungematchten und gematchten Kanten) **alternierende** DFS bis zur Tiefe l aus, wobei wir

- 1 wenn wir einen ungematchten Knoten (in Tiefe l) erreichen, einen kürzesten augmentierenden Pfad Q_i gefunden haben; für den weiteren Verlauf der DFSs markieren wir Q_i als **gelöscht**;
- 2 jede Kante, über die die DFS zurücksetzt, ebenfalls als **gelöscht** markieren.

Der Zeitaufwand für diese DFSs beträgt $\mathcal{O}(n + m)$, da wir jede Kante höchstens zweimal (einmal in der DFS vorwärts, einmal rückwärts) besuchen.

Lemma 145

Gegeben die Länge eines kürzesten augmentierenden Pfades, kann eine bzgl. „ \subseteq “ maximale Menge kürzester augmentierender Pfade in Zeit $\mathcal{O}(n + m)$ gefunden werden.

Satz 146

In bipartiten Graphen kann ein Matching maximaler Kardinalität in Zeit

$$\mathcal{O}\left(n^{\frac{1}{2}}(n + m)\right)$$

gefunden werden.

Beweis:

Gemäß Korollar 144 genügen $\mathcal{O}(n^{\frac{1}{2}})$ Phasen, in denen jeweils mittels einer simultanen BFS und einer sequentiellen DFS (beide in Zeit $\mathcal{O}(n + m)$) eine maximale Menge kürzester augmentierender Pfade bestimmt wird. □



John Hopcroft, Richard Karp:

An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs

SIAM J. Comput. **2**(4), pp. 225–231 (1973)

4. Maximum Matchings in allgemeinen Graphen

In einem allgemeinen (nicht unbedingt bipartiten) Graphen können wir in Zeit $\mathcal{O}(n + m)$ jeweils einen kürzesten augmentierenden Pfad finden und erhalten damit

Satz 147

In einem Graph $G = (V, E)$ kann ein Matching maximaler Kardinalität in Zeit

$$\mathcal{O}(n \cdot m)$$

gefunden werden.

Für Maximum-Matching-Algorithmen in allgemeinen Graphen mit Laufzeit $\mathcal{O}(n^{\frac{1}{2}}(n + m))$ verweisen wir auf die Literatur:



Silvio Micali, Vijay V. Vazirani:

An $O(\sqrt{|V|} \cdot |E|)$ algorithm for finding maximum matching in general graphs

Proceedings of the 21st Annual IEEE Symposium on Foundations of Computer Science, FOCS'80 (Syracuse, NY, October 13–15, 1980), pp. 17–27 (1980)



Vijay V. Vazirani:

A theory of alternating paths and blossoms for proving correctness of the $O(\sqrt{V}E)$ general graph maximum matching algorithm

Combinatorica **14**(1), pp. 71–109 (1994)



Norbert Blum:

A new approach to maximum matching in general graphs

Proceedings of the 17th International Colloquium on Automata, Languages and Programming, ICALP'90 (Warwick University, England, July 16–20, 1990), LNCS **443**, pp. 586–597 (1990)

5. Matchings in gewichteten bipartiten Graphen

5.1 Zerlegung doppelt stochastischer Matrizen

Sei M eine $n \times n$ -Matrix mit reellen Einträgen $m_{ij} \geq 0$, so dass alle Zeilen- und alle Spaltensummen $= r > 0$ sind. Wir assoziieren mit M den bipartiten Graphen $G = G_M = (U, V, E)$, wobei $U = \{u_1, \dots, u_n\}$ die n Zeilen und $V = \{v_1, \dots, v_n\}$ die n Spalten von M repräsentiert. G enthält die Kante $\{u_i, v_j\}$ gdw $m_{ij} > 0$.

Ein Matching in G entspricht einer Menge von Einträgen $m_{ij} > 0$, die alle in **verschiedenen** Zeilen und Spalten vorkommen. Wir nennen eine solche Menge von Positionen eine **Diagonale** der Matrix M .

Ein **Träger** von M ist eine Menge von Zeilen und Spalten, die zusammen alle Matrixeinträge > 0 enthalten.

Beispiel 148

2	0	1	3	0
0	4	0	0	2
3	1	2	0	0
1	1	3	0	1
0	0	0	3	3

Die markierten Elemente bilden eine Diagonale der Größe $n = 5$.

Annahme: M hat keine Diagonale der Größe n . Dann gibt es nach Satz 136 z Zeilen und s Spalten von M mit $z + s < n$, die alle Einträge > 0 von M bedecken. Damit wäre aber

$$r \cdot n = \sum m_{ij} \leq r \cdot (z + s) < r \cdot n$$

Widerspruch.

Also existiert eine Diagonale der Größe n und ein entsprechendes perfektes Matching M_1 von G . Die Adjazenzmatrix P_1 von $G_1 = (V, M_1)$ enthält in jeder Zeile und Spalte genau eine 1, ist also eine so genannte **Permutationsmatrix** (alle anderen Einträge sind 0). Sei nun

$$m_1 := \min\{m_{ij}; \{u_i, v_j\} \in M_1\}.$$

Die Matrix $M - m_1 M_1$ hat wiederum konstante Zeilen- und Spaltensummen (nämlich $r - m_1$) und strikt mehr Einträge $= 0$ als M .

Durch Iteration ergibt sich damit

Satz 149 (Birkhoff, von Neumann)

Sei M eine doppelt-stochastische Matrix (d.h. alle Zeilen- und Spaltensummen sind $= 1$, alle Einträge sind reell und ≥ 0), dann gibt es eine Darstellung

$$M = \sum_{i=1}^k m_i P_i,$$

wobei die P_i Permutationsmatrizen und die $m_i \in \mathbb{R}$, mit $m_i > 0$ und $\sum_{i=1}^k m_i = 1$.

Bemerkung: Jede doppelt-stochastische Matrix ist also eine Konvexkombination von endlich vielen Permutationsmatrizen.

5.2 Matchings in knotengewichteten bipartiten Graphen

Sei $G = (U, V, E)$ ein bipartiter Graph mit einer Gewichtsfunktion $w : U \rightarrow \mathbb{R}^+$. Wir suchen ein Matching M in G , so dass die Summe der Gewichte der gematchten Knoten in U maximiert wird. Eine Teilmenge $T \subseteq U$, die durch ein Matching in G gematcht werden kann, heißt auch **Transversale**.

Sei \mathcal{T} die Menge der Transversalen von G (beachte: $\emptyset \in \mathcal{T}$).

Satz 150

Die Menge \mathcal{T} der Transversalen von G bildet ein Matroid.

Beweis:

Es ist klar, dass $\emptyset \in \mathcal{T}$ und dass \mathcal{T} unter Teilmengenbildung abgeschlossen ist. Seien T und T' Transversalen $\in \mathcal{T}$ mit $|T'| = |T| + 1$, und seien M bzw. M' zugehörige Matchings. Dann gibt es bzgl. M einen augmentierenden Pfad P . Ein Endknoten von P liegt in U . Augmentieren wir M mittels P , erhalten wir also eine Transversale der Kardinalität $|T| + 1$. \square

Das **greedy**-Paradigma lässt sich also anwenden, um in Zeit $\mathcal{O}(n \cdot m)$ eine Transversale maximalen Gewichts zu konstruieren.