Rheinisch-Westfälische Technische Hochschule Aachen
Lehr- und Forschungsgebiet Theoretische Informatik
Prof. Dr. Peter Rossmanith

Systems of Distinct Representatives

Seminar: Extremal Combinatorics
SS 04

Peter Fritz
Matrikelnummer: 234707

Zusammenfassung

Ein System von verschiedenen Vertretern (engl. system of distinct representatives, SDR) besteht aus Elementen von Mengen einer Mengenfolge, so dass jeder Menge eines ihrer Elemente als eindeutigen Repräsentant zugeordnet wird. Das finden eines solchen Repräsentantensystems kann als Spezialfall des Maximum-Matching Problems auf bipartiten Graphen angesehen werden, mit den Mengen und Elementen als Knoten des Graphen und der Enthaltenseins-Relation als Kanten.

Inhaltsverzeichnis

1 Einleitung

Das *Heiratsproblem* ist der einfachste Spezialfall eines Matching Problems: Das Ziel ist es, m Frauen mit je einem von n Männern (üblicherweise mit $m \leq n$) zu verheiraten, unter der Bedingung, dass sie diesen Mann bereits vorher kannte. Die zugehörigen Matching-Graphen sind bipartit und nicht gewichtet, was die Matching-Suche erheblich erleichtert. Mit dem Satz von Hall wird eine notwendige und hinreichende Bedingung gegeben, mit der die Lösbarkeit des Heiratsproblems überprüft werden kann. Andere Anwendungsmöglichkeiten wären z.B. das Einteilen von Studenten in Übungsgruppen oder die Theorie der Lateinischen Quadrate, ein altes kombinatorisches Problem, welche man ähnlich wie magische Quadraten mit Zahlen auffüllen muss, ohne gewisse Regeln zu verletzen. Neben den kombinatorischen Sätzen liefert diese Arbeit auch einen algorithmischen Zugang zu Matchings und Repräsentantensystemen, da allein das Wissen über die Existenz solcher Systeme in der Praxis noch nicht ausreichend ist. Man möchte die optimale Zuordnungsvorschrift berechnen und nutzen. Die theoretischen Sätze der Ausarbeitung basieren hauptsächlich auf [1], während die Lösungsalgorithmen aus [3] stammen.

2 Der Heiratssatz von Hall

Definition 2.1 *Ein System von verschiedenen Vertretern (engl. system of distinct representatives, SDR) von einer Folge von Mengen $S_1, S_2, \ldots, S_n$ ist eine Folge von verschiedenen Elementen $x_1, x_2, \ldots, x_n$ dieser Mengen, so dass $x_i \in S_i$ für alle $1 \leq i \leq m$.*

Falls $S_1, S_2, \ldots, S_n$ ein Repräsentantensystem besitzen, ist es somit möglich, eine injektive Abbildung f zu definieren, die jeder Menge S_i ihren eindeutigen Repräsentanten zuordnet, also

$$f(S_i) \neq f(S_j) \text{ für } i \neq j.$$

Die Frage nach der Existenz solch einer Abbildung ist das Heiratsproblem. Die Mengen S_i stehen dabei für die Mädchen, die Elemente $x_{i_j} \in S_i$ für die potentiellen Ehemänner, die das i-te Mädchen kennt, sowie f für die „Heiratsabbildung“. Falls eine Gruppe von k Mädchen insgesamt weniger als k Jungen kennt, ist es klar, dass nicht alle Mädchen verheiratet werden können. Diese Bedingung ist auch als *Hall-Bedingung* bekannt:

$$\left| \bigcup_{i \in I} S_i \right| \geq |I| \text{ für alle } I \subseteq \{1, \ldots, m\} \tag{1}$$

Aus dieser Überlegung folgt der Heiratssatz von Hall, einem englischer Gruppentheoretiker, den er 1935 bewiesen hat und damit das Heiratsproblem löste:

Satz 2.1 *Die Mengen $S_1, S_2, \dots, S_m$ besitzen genau ein System von verschiedenen Vertretern, wenn (1) erfüllt ist.*

Beweis: Die Rückrichtung ist klar, da, falls $S_1, \dots, S_m$ ein SDR besitzen, die Vereinigung von k beliebigen S_i mindestens aus deren k Repräsentanten besteht.
Die Hinrichtung wird per Induktion über m gezeigt, wobei der Fall $m = 1$ trivial ist. Angenommen die Behauptung wäre für bis zu m Mengen der S_i richtig und $S_1, \dots, S_m, S_{m+1}$ erfüllten die Hall-Bedingung (1).

Fall 1: Falls jede beliebige Vereinigung von $k \leq m$ Mengen mindestens $k + 1$ Elemente besitzt, also

$$\left|\bigcup_{i \in I} S_i\right| > |I| \text{ für alle } I \subseteq \{1, \dots, m\}, \tag{2}$$

ordne irgendeiner Menge $S_{i_{m+1}}$ einen beliebigen Repräsentanten $x_i \in S_{i_{m+1}}$ zu und entferne ihn aus den restlichen Mengen. Da die restlichen m Mengen immer noch mindestens k Elemente besitzen und (1) erfüllen, haben sie nach Induktionsvorraussetzung ein SDR.

Fall 2: Für k der S_i mit $1 \leq k \leq m$ gilt

$$\left|\bigcup_{i \in J} S_i\right| = |J| = k \text{ für alle } J = \{j_1, \dots, j_k\} \subseteq \{1, \dots, m\}. \tag{3}$$

Wegen $k < m$ und der Induktionsvorraussetzung besitzen diese Mengen ein SDR. Entferne diese k Repräsentanten aus den restlichen $m+1-k$ Mengen. Jetzt bleibt nur noch zu zeigen, dass diese Mengen die Hall-Bedingung (1) erfüllen und somit nach Induktionsvorraussetzung ebenfalls ein Repräsentantensystem besitzen, welches mit den anderen k Repräsentanten ein gemeinsames SDR bildet. Angenommen die Vereinigung von s der restlichen $m + 1 - k$ Mengen würde weniger als s Elemente besitzen und die Hall-Bedingung verletzen, dann besäße die Vereinigung dieser s Mengen mit den ersten k Mengen weniger als $s + k$ Elemente. Dies wäre ein Widerspruch zur Induktionsvorraussetzung, und deshalb besitzen $S_1, \dots, S_m, S_{m+1}$ ein SDR. □

Bereits 1931 zeigten König und Egeváry, dass $S_1, \dots, S_m$ ein SDR besitzen, falls der zugehörige bipartite Graph $G = (\{S_1, \dots, S_m\}, \{x_i | x_i \in \bigcup_{j=1}^{m} S_j\}, E)$, mit $E \ni \{x_i, S_j\}$ falls $x_i \in S_j$, ein Vertex-Cover mit der Mächtigkeit kleiner oder gleich m besitzt. Leider ist es häufig nicht effizient möglich, solch ein minimales Vertex Cover zu finden oder die Hall-Bedingung zu überprüfen, da der Aufwand dafür exponentiell ist (Min-VC ist (zumindest für nicht bipartite Graphen) NP-vollständig, und für die Hall-Bedingung müssen alle 2^m Teilmengen der S_i überprüft werden). Für gewisse Probleminstanzen kann man aber Vorwissen über deren Struktur ausnutzen und effizient entscheiden, ob sie ein Repräsentantensystem besitzen. Beispielhaft betrachten wir hier den Spezialfall, dass alle S_i die gleiche Mächtigkeit besitzen und jedes Element zur selben Anzahl von Mengen gehört.

Korollar 2.1 *Falls $S_1, \ldots, S_m$ jeweils r Elemente besitzen, $\left|\bigcup_{1\leq i\leq m} S_i\right| = n$ gilt und alle Elemente der S_i in der selben Anzahl von Mengen enthalten sind, also $|\{S_i|x_j \in S_i\}| = d$ für $1 \leq i \leq m$ und für alle $x_j \in \bigcup_{1\leq j\leq m} S_i$, haben $S_1, \ldots, S_m$ ein SDR.*

Beweis: Zähle die Anzahl des Enthaltenseins von Elementen in Mengen: Da die m S_i genau r Elemente besitzen, gilt $|S_1| + |S_2| + \ldots + |S_m| = m \cdot r$, und da jedes der n Elemente in genau d Mengen enthalten ist, gilt mit der Regel des doppelten Abzählens: $m \cdot r = n \cdot d$. Wegen $m \leq n$ muss $r \geq d$ gelten. Zeige nun, dass, falls $S_1, \ldots, S_m$ *kein* Repräsentantensystem besitzen, $r < d$ gilt, ein Widerspruch:
Wegen des Satzes von Hall verletzen $S_{i_1}, \ldots, S_{i_k}$ die Hall-Bedingung. Definiere nun $Y := S_{i_1} \cup S_{i_2} \cup \ldots \cup S_{i_k}$ mit $|Y| < k$. Durch erneutes doppeltes Abzählen erhält man $r \cdot k = \sum_{j=1}^{k} |S_{i_j}| = |Y| \cdot d < k \cdot d$, woraus der Widerspruch $r < d$ folgt.

Die Vorbedingung in Korollar 2.1 wirkt sehr scharf, dennoch ist es häufig möglich, viele Instanzen des Heiratsproblems auf solche Spezialfälle zu reduzieren. So kann man z.B. bei dem Heiratsbeispiel sukzessive die potentiellen Ehemänner aus dem Bekanntenkreis der Mädchen entfernen bis man eine einfache Struktur hat, mit der man Korollar 2.1 anwenden kann. Eine Nichtexistenz eines SDR lässt sich so natürlich nicht beweisen.
Man kann das Heiratsproblem weiter verallgemeinern und jedem Element bestimmte Eigenschaften zuordnen, die bei der Repräsentantenwahl berücksichtigt werden sollen. Betrachten wir dazu wieder das Beispiel der zu verheiratenden Mädchen und Jungen. Angenommen unter den n Jungen wären r äußerst unbeliebt, und eine Heirat mit einem unbeliebten Jungen soll möglichst vermieden werden. Ist es dann möglich, die Anzahl der unglücklichen Hochzeiten auf höchstens t zu begrenzen? Eine Antwort darauf liefert der Satz von Chvátal-Szemerédi (1988), in dem „schlechte“ Elemente rot und „gute“ blau gefärbt sind.

Satz 2.2 *Die Mengen $S_1, \ldots, S_m$ haben genau dann ein System von verschiedenen Vertretern mit höchstens t roten Elementen, wenn sie ein SDR besitzen und für alle $1 \leq k \leq m$ die Vereinigung von k beliebigen Mengen mindestens $k - t$ blaue Elemente besitzt.*

Beweis „$\Rightarrow$“: Sei R die Menge der roten Elemente mit $|R| > t$ und $r := |R| - t$. Erweitere die S_i zu $S_1, \ldots, S_m, S_{m+1}, \ldots, S_{m+r}$, wobei $S_{m+1}, \ldots, S_{m+r}$ r Kopien von R sind. Zeige nun, dass die *erweiterte* Mengenfolge ein SDR besitzt, da dann für die ersten Mengen nur noch $|R| - r = t$ rote Elemente übrig bleiben, sie besitzen also ein SDR mit höchstens t roten Repräsentanten: Die erweiterte Mengenfolge mit Indizes $I \subseteq \{1, \ldots, m + r\}$ und $|I| = k$ erfüllt die Hall-Bedingung, da für beliebige k der ersten m Mengen deren Vereinigung höchstens $k - t$ blaue Elemente besitzt(*):

$$\left|\bigcup_{i\in I} S_i\right| = \left|\bigcup_{i\in I} (S_i - R)\right| + |R| \geq^{(*)} k - t + |R| \geq k = |I|$$

„$\Leftarrow$“: Betrachte die Vereinigung von k beliebigen S_i. Höchstens t von ihnen besitzen ein rotes Element als Repräsentant, ansonsten hätten wir einen Widerspruch. Da alle S_i ein SDR besitzen, hat die Vereinigung der k Mengen wegen (1) mindestens $k - t$ blaue Elemente. □

3 Lateinische Rechtecke

Eine Anwendung von Korollar 2.1 sind die so genannten „Lateinischen Rechtecke“, die wie z.B. die magischen Quadrate zu den ältesten kombinatorischen Problemen gehören. Bereits 2800 v. Chr. befasste sich ein chinesische Mathematiker mit diesen Strukturen. Auch Leonard Euler befasste sich intensiv mit lateinischen Quadraten und analysierte zusätzliche Aspekte, wie z.B. Orthogonalität.

Definition 3.1 *Ein lateinisches Rechteck ist eine $r \times n$ Matrix mit $r \leq n$, in der die Zahlen $1, 2, \ldots, n$ in jeder Zeile genau und in jeder Spalte höchstens einmal auftreten. Falls $r = n$ ist, nennt man diese Matrix lateinisches Quadrat.*

Die Aufgabe besteht nun daraus, die anfangs noch leere Matrix mit Zahlen aufzufüllen, ohne die Definition zu verletzten. Bereits ab n Einträgen kann es bereits unmöglich werden, die Matrix komplett aufzufüllen (Abbildung 1). Wenn man aber das Rechteck sukzessive, Zeile für Zeile, aufbaut und dabei jede Zeile komplett füllt, ist es immer möglich jedes Lateinische Rechteck zu einem Quadrat zu erweitern, was Ryser 1951 unter Verwendung des Satzes von Hall bewies:

Satz 3.1 *Jedes $r \times n$ Lateinische Rechteck kann zu einem $(r + 1) \times n$ Lateinische Rechteck erweitert werden, falls $r < n$.*

Beweis: Definiere für das $r \times n$ Lateinische Rechteck R die Mengen S_i mit $1 \leq i \leq n$ und $S_i = \{x_i | x_i$ ist *nicht* in der *i*-ten Spalte von *R* enthalten$\}$. Zeige nun, dass $S_1, \ldots, S_n$ ein SDR besitzen und definiere diese Repräsentanten als neue $r + 1$-te Zeile $(x_1, x_2, \ldots, x_n)$ von R. Korollar 2.1 ist auf $S_1, \ldots, S_n$ anwendbar, denn

1. $S_1, \ldots, S_n$ sind $n - r$ elementige Teilmengen einer n-elementigen Menge
2. jedes Element $x_j \in \bigcup_{1 \leq i \leq n} S_i$ ist in genau $n - r$ S_i enthalten, da es bereits in r anderen Spalten eingefügt wurde

Aus 1., 2. und dem Korollar folgt die Behauptung. □

Weitere Anwendung finden Lateinische Quadrate z.B. in der Scheduling-Theorie, in statistischen Experimenten und in kryptographischen Protokollen [2].

1	5	2	4	?
				3

5	4	3	2	1
2	1	5	4	3
4	3	2	1	5
1	5	4	3	2
3	2	1	5	4

Abbildung 1: unvollständiges Lateinisches Rechteck, und vollständiges Lateinisches Quadrat

4 Matchings in bipartiten Graphen

Das Heiratsproblem kann allgemein als Matching-Problem in einem bipartiten Graphen $G = (A, B, E)$ betrachtet werden, wobei A die Menge der Frauen, B die der Jungen ist und $(x, y) \in E$ für ein $x \in A, y \in B$, falls x und y sich kennen. Ein Matching $M \subseteq E$ in G ist eine Menge von Kanten, die keinen gemeinsamen Knoten haben. Die Kanten und ihre inzidenten Knoten in M nennt man *gematched*, die anderen *frei*. Man nennt M ein Matching von *A nach B*, falls jeder Knoten aus A in M enthalten ist. Für das Heiratsproblem suchen wir also ein Matching M von der Menge der Mädchen in die Menge der Jungen, da dann, falls M die Mächtigkeit $|A|$ besitzt, alle Mädchen verheiratet werden können. Weil solch ein Matching aber häufig nicht existiert und da das Anwenden der Hall-Bedingung oder des Satzes von König zu aufwendig ist, benötigt man effizientere Algorithmen zum Finden eines maximalen Matchings in G. Die Berechnung von M mit einem Greedy-Ansatz anzugehen ist zwar mit $O(|E|)$ äußerst effizient, liefert aber nicht immer eine optimale Lösung (Abbildung 2). Die Idee dabei wäre, immer eine Kante zu wählen, die zwei Knoten mit möglichst geringem Knotengrad verbindet.

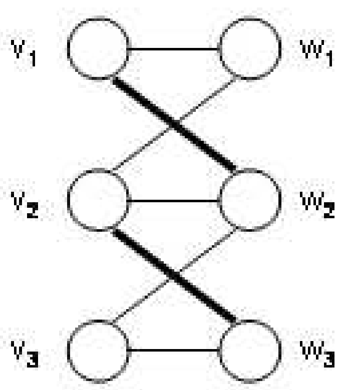

Abbildung 2: Gegenbeispiel zur Matching Berechnung mit Greedy-Ansatz

Das Beispiel aus Abbildung 3 zeigt, dass man mit einer lokalen Nachbarschaftstransformation, nämlich dem „Verschieben" des Matchings nach unten und der Aufnahme von (v_1, w_1) in das neue Matching, ein größeres Matching erhält. Dieses Prinzip der sukzessiven Verbesserung durch lokale Operationen nennt man *lokale Suche*. Beim Maximum Matching-Problem ist die lokale Nachbarschaftstransformation das Finden und Verbessern eines *erweiternden Pfades* (engl. augmenting path).

Definition 4.1 *Ein erweiternder Pfad in einem bipartiten Graph $G = (A, B, E)$ bezüglich eines Matchings M ist ein Pfad, dessen Kanten abwechselnd gematched und frei sind, und dessen Start- und Zielknoten nicht im Matching liegen.*

Ein Graph, der einen erweiternden Pfad besitzt, kann folglich kein maximales Matching besitzen, da man alle Matching-Kanten um je eine Kante auf dem Pfad nach hinten verschieben und die erste Kante ins Matching aufnehmen kann, weil sowohl Ziel- und Endknoten frei sind (siehe Abbildung 3). Berger hat 1957 gezeigt, dass die Nichtexistenz eines solchen Pfades eine hinreichende Bedingung für die Optimalität eines Matchings ist.

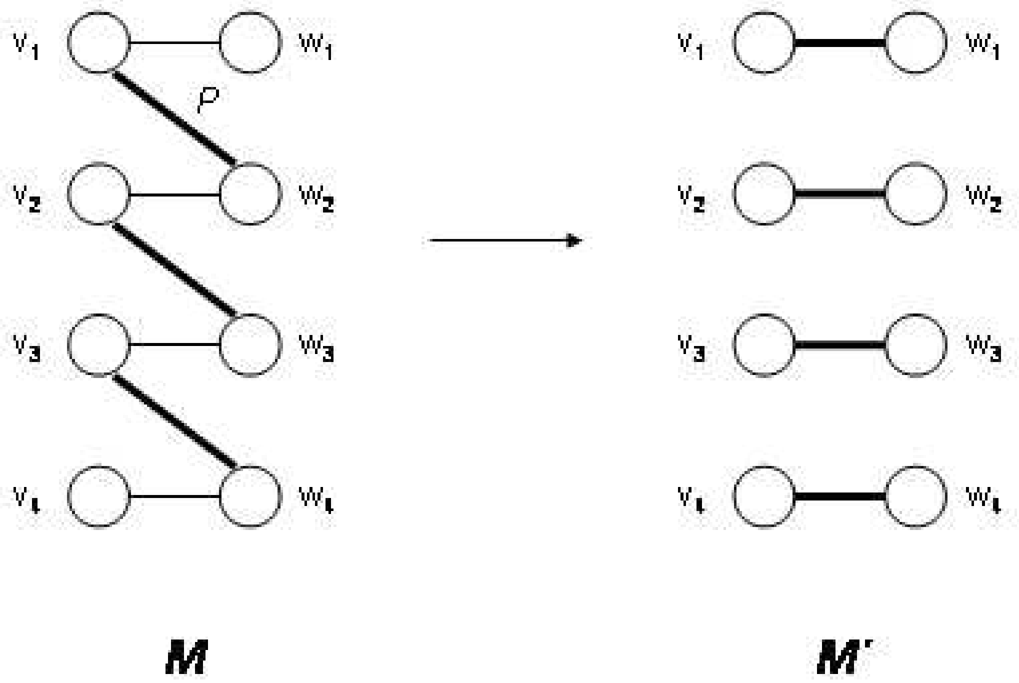

Abbildung 3: Erweiterung von M zu M' mittels eines erweiternden Pfades P

Satz 4.1 *Ein Matching M in einem Graph G ist genau dann maximal, wenn es keinen erweiternden Pfad in G gibt.*

Beweis: Aufgrund der möglichen Nachbarschaftstransformation auf dem erweiternden Pfad ist die Rückrichtung klar. Zeige deshalb für die Hinrichtung, dass ein Graph mit Matching M, welches kein Maximum Matching ist, einen erweiternden Pfad besitzt:
Angenommen es gäbe ein Matching M' mit $|M| < |M'|$. Bilde nun die exklusive Vereinigung $\oplus$ beider Mengen $H := M \oplus M' := (M \cup M') \setminus (M \cap M')$.

Da H aus der exklusiven Vereinigung beider Matchings besteht, kann keine Kante in zwei Matchings liegen, und wegen $|M| < |M'|$ muss es eine Zusammenhangskomponente G' geben, in der weniger Knoten zu M gehören als zu M' (*). Da alle Knoten in M und M' den Grad eins besitzen, haben alle Knoten in G' höchstens den Grad zwei und G' besteht aus einem Zyklus gerader Länge oder einem Pfad, welche beide keine gemeinsamen Knoten besitzen. G' besteht aus genau einem Pfad, da (*) gilt und G' zusammenhängend ist. Somit gibt es keinen Zyklus gerader Länge, der alle Knoten aus G' enthält. Betrachte nun diesen Pfad in G': Der Anfangs- und Endknoten dieses Pfades muss wegen (*) in M' liegen und auf einem solchen Pfad P können keine 2 Kanten desselben Matchings direkt aufeinander folgen, da sonst ein Knoten zweimal im selben Matching enthalten wären. Dieser Pfad ist bezüglich M ein erweiternder Pfad, da Anfangs- und Endknoten frei sind und die Kanten abwechselnd frei und gematched sind. □

Dieser Satz liefert eine ganze Reihe von Anwendungsmöglichkeiten, da er nicht nur auf bipartite Graphen anwendbar ist, sondern in einer Abwandlung auch auf Flussnetzwerke. Der Ford-Fulkerson-Algorithmus berechnet einen maximalen Fluss in einem solchen Netzwerk, indem er nach erweiternden Pfaden sucht, die einen größeren Fluss ermöglichen. Es ist sogar möglich, mit dem Ford-Fulkerson-Algorithmus das Maximum-Matching-Problem und damit auch das Heiratsproblem effizient zu lösen wie Abbildung 4 illustriert: Definiere zwei zusätzliche Knoten als Quelle und Senke, verbinde die Quelle mit allen Knoten aus A und die Senke mit allen Knoten aus B. Nachdem alle Kanten von links nach rechts gerichtet worden sind und ihnen die Kapazität eins zugeordnet worden ist, berechne einen maximalen Fluss in dem Netzwerk. Die Kanten des Maximalen Flusses zwischen A und B bilden dann das Maximum Matching. Für weitere Details und Korrektheitsbeweise sei auf [3] verwiesen.

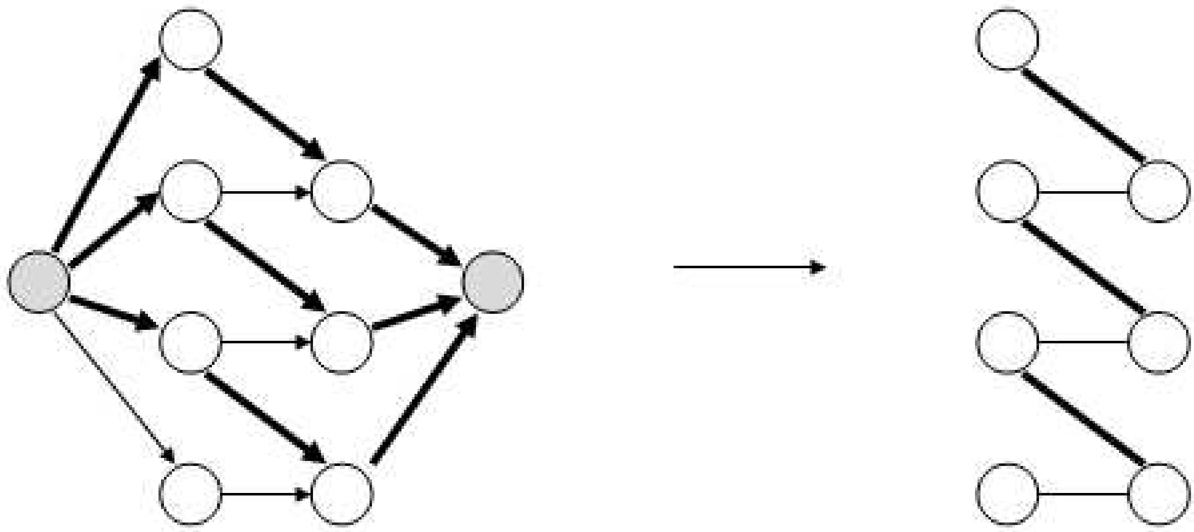

Abbildung 4: Berechnung eines Maximum Matchings anhand des Ford-Fulkerson-Algorithmus

Da in Netzwerken die erweiternden Pfade anders definiert sind als im Satz von Berger, verwenden wir hier einen anderen Algorithmus zum Lösen des Maximum-Matching-Problems, welcher erweiternde Pfade mittels Tiefensuche findet:

ALGORITHMUS FÜR DAS MAXIMUM-MATCHING-PROBLEM:

Eingabe: $G = (A, B, E)$
  $\{M$ ist das aktuelle Matching$\}$
Schritt 1: $M := \emptyset$;
Schritt 2: $\{$Richte alle Kanten $e \in M$ von A nach B und die restlichen von B nach $A\}$
  for all $e \in E$ **do**
    if $e \in M$ **then**
      $e := (a, b)$; $\{$für $a \in A$ und $b \in B\}$
    else $e := (b, a)$;
Schritt 3: $\{$definiere A_0 und B_0, die alle freien Knoten aus A und B enthalten$\}$
  $A_0 := B_0 := \emptyset$;
  for all $v \in A$ und $v \notin M$ **do**
    $A_0 := A_0 \cup \{v\}$;
  for all $v \in B$ und $v \notin M$ **do**
    $B_0 := B_0 \cup \{v\}$;
Schritt 4: **for all** $v \in B_0$ **do**
  starte Tiefensuche von v bis ein Pfad von v zu w für ein $w \in B_0$ gefunden wurde
  $\{$dieser Pfad ist ein erweiternder Pfad$\}$
Schritt 5: **if** erweiternder Pfad gefunden **then**
  begin
    update M;
    gehe zu Schritt 2;
  end
  else Output M;

Laufzeit: Da das Matching nach jeder Iteration von Schritt 2 um mindestens eins vergrößert wird, und die Tiefensuche auf höchstens $n/2$ Knoten angewendet wird, bleibt für eine Gesamtlaufzeit von $O(n^3)$ nur noch zu zeigen, dass die Tiefensuche in Schritt 4 linear bezüglich der Knoten ist: Da freie und gematchte Kanten unterschiedlich gerichtet sind, muss ein Tiefensuchlauf abwechselnd aus freien Kanten und Kanten des Matchings bestehen. Der Tiefensuchbaum hat im schlechtesten Falle deshalb höchstens $2 \cdot \frac{n}{2} = n$ Kanten, woraus die Gesamtlaufzeit von $O(n^3)$ folgt.

Mittlerweile sind schnellere Algorithmen bekannt, wie z.B. der von Hopcroft und Karp (1973), welcher in $O(n^{5/2})$ läuft [4]. Zur Berechnung von Maximum Matchings in nichtbipartiten Graphen ist ebenfalls der Satz von Berger anwendbar, aber die Suche nach erweiternden Pfaden ist erheblich aufwändiger als in bipartiten Graphen, da ein Aufteilen und entgegengesetztes Richten der Kanten nicht mehr möglich ist [5].

5 Zusammenfassung, Ausblick

Das Finden von Systemen von verschiedenen Vertretern ist ein klassisches Problem der Kombinatorik, schon lange vor dem Beweis von Hall befassten sich Mathematiker mit den zugehörigen kombinatorischen Strukturen. Da sich das Heiratsproblem als Maximum Matching-Problem formulieren lässt, ist es möglich, mit dem Erweiternden-Pfad-Algorithmus ein Repräsentantensystem effizient zu berechnen, ohne die Hall-Bedingung nachweisen zu müssen.
Man kann das Heiratsproblem noch weiter generalisieren, z.B. indem man Präferenzen der Mädchen bezüglich der Jungen einführt und versucht, jedem Mädchen einen Ehepartner mit möglichst hoher Präferenz zuzuordnen. Das zugehörige Matching Problem ist das *Maximum weighted Matching Problem*, ein weiteres grundlegendes Problem aus der Optimierung, welches auch in der Praxis, z.B. im Operations Research, zahlreiche Anwendungsmöglichkeiten hat. In der Algorithmik werden Maximum Matchings gerne als Hilfsmittel zur Konstruktion von anderen Algorithmen, wie z.B. von Approximationalgorithmen für schwere Probleme, verwendet. Ein herausragender Vertreter ist der Algorithmus von Cristofides, welcher eine Lösung des $\Delta - TSP$ liefert, die höchstens 50% schlechter als die beste Lösung ist. Eine direkte Anwendung des Satzes von König ist ein anderer Approximationsalgorithmus zum Lösen des Vertex-Cover Problems: Man nimmt alle Knoten des Maximum Matchings in das Vertex-Cover auf, und erhält somit eine Lösung, die höchstens doppelt so teuer wie die optimale ist [6].

Literatur

[1] Jukna, S.: *Extremal Combinatorics*, S.: 55 - 63, Springer, 2001

[2] Dénes, J., Keedwell, A.D.: *Latin squares and their applications*, Academic Press, 1974

[3] Cormen, T., Leiserson, C., Rivest, R.:*Introduction to Algorithms*, MIT Press, und McGraw-Hill, 1990

[4] Hopcroft, J., Karp, R.: *An n5/2 Algorithm for Maximum Matchings in Bipartite Graphs*, SIAM J. Comput. 2(4): S:225-231, 1973

[5] Galil, Z.:*Efficient algorithms for finding maximum matchings in graphs*, ACM Computing Surveys, Volume 18, S: 23 - 38, 1986

[6] Hromkovič, J.: *Algorithmics for Hard Problems*, S.: 268 - 270, 294 - 305, Springer, 2003