Lehrstuhl für Informatik I
Informationsstrukturen und
wissensbasierte Systeme
Universität Würzburg

Prof. Dr. H. Noltemeier
Dr. S. O. Krumke

Würzburg, den 28. November 1997

Informationsstrukturen und wissensbasierte Systeme
(WS 97/98)

2. Übung mit Musterlösung

Aufgabe 4: (Verbesserter Algorithmus für MAX k-SAT)

In Aufgabe 2 wurde ein Algorithmus für MAX k-SAT entwickelt, der zu einer Instanz $I = (X, M)$ des Problems mit $m := |M|$ Klauseln immer eine Belegung der Atomformeln in X findet, die mindestens $m/2$ Klauseln erfüllt. In dieser Aufgabe soll ein Algorithmus entwickelt werden, der eine Belegung liefert, die mindestens $(1 - 2^{-k})m$ Klauseln erfüllt.

In dieser Aufgabe lassen wir zu, daß eine Klausel außer Literalen auch die Konstanten 0 und 1 enthält. Diese Literale haben die natürliche Bedeutung: Eine Klausel, welche 1 enthält, ist unter jeder Belegung wahr.

Wir nehmen in dieser Aufgabe zur Vereinfachung an, daß es keine Klausel gibt, die sowohl eine Atomformel A_i als auch die negierte Atomformel $\neg A_i$ enthält.

Sei $I = (X, M)$ eine Instanz von k-SAT mit $X = \{A_1, \ldots, A_m\}$ und $M = \{C_1, \ldots, C_m\}$. Wir bezeichnen mit $M_{A_i=0}$ diejenige Klauselmenge, die aus M entsteht, wenn man A_i in allen Klauseln durch 0 und $\neg A_i$ durch 1 ersetzt. Analog entsteht $M_{A_i=1}$ durch Ersetzen von A_i durch 1 und $\neg A_i$ durch 0. $M_{A_1=1}$ und $M_{A_1=0}$ sind Klauselmengen über $\{A_2, \ldots, A_n\}$.

(a) Sei $E[M] = (1 - 2^{-k})m$ der Erwartungswert der Anzahl Klauseln aus M, die eine zufällige Belegung erfüllt. Zeigen Sie, daß

$$E[M] = \frac{1}{2}(E[M_{A_1=1}] + E[M_{A_1=0}])$$

gilt. **(2 Punkte)**

(b) Wir betrachten nun den folgenden Algorithmus: Zuerst berechnet der Algorithmus $E[M_{A_1=1}]$ und $E[M_{A_1=0}]$. Dann weist er A_1 den Wahrheitswert t zu, der $E[M_{A_1=t}]$ maximiert. Dieser Vorgang wird dann mit der Klauselmenge $M_{A_1=t}$ und der Atomformel A_2 fortgesetzt, wobei ein Wahrheitswert für A_2 fixiert wird. Der Algorithmus fährt so fort, bis alle Atomformeln mit Wahrheitswerten belegt sind.

Beweisen Sie, daß die vom Algorithmus bei Abbruch gelieferte Belegung mindestens $(1 - 2^{-k})m$ Klauseln aus M erfüllt. **(4 Punkte)**

Aufgabe 5: (Hillclimbing und minimale aufspannende Bäume)

Sei $G = (V, E)$ ein endlicher ungerichteter und zusammenhängender Graph und $w\colon E \to \mathbb{R}$ eine Kantengewichtsfunktion. In dieser Aufgabe soll Hillclimbing benutzt werden, um in G eine minimalen aufspannenden Baum mit minimalem Gewicht zu finden.

Ein klassisches Verfahren zur Bestimmung eines minimalen aufspannenden Baumes ist das *Greedy-Verfahren* bzw. der *Algorithmus von Kruskal*. Dieses Verfahren sortiert die Kanten aus E aufsteigend bezüglich ihres Gewichtes. Der Algorithmus startet mit der leeren Kantenmenge $T := \varnothing$. Dann betrachtet er die Kanten aus E in der aufsteigend sortierten Reihenfolge (jede Kante einmal). Er fügt eine Kante zu T hinzu, wenn sie keinen Kreis in T erzeugt. Man kann zeigen, daß dieser Algorithmus immer einen minimalen aufspannenden Baum liefert. Sie dürfen diese Eigenschaft in dieser Aufgabe ohne Beweis benutzen.

Als Zustandsmenge Z für das Hillclimbing-Verfahren wählen wir

$$Z := \{\, T : T \text{ ist aufspannender Baum von } G \,\}.$$

Für einen Zustand $z = T$ besteht die Menge seiner Nachfolger aus alle denjenigen aufspannenden Bäumen T', die durch Tausch einer Kante aus T entstehen, d.h. für die es $e \in T$ und $e' \notin T$ gibt, so daß $T' = (T \setminus \{e\}) \cup \{e'\}$ gilt.

Die Bewertung $\varphi(z)$ eines Zustandes $z = T$ ist das Gewicht des Baumes T. Der Hillclimbing-Algorithmus werde so modifiziert, daß er vom aktuellen Zustand z nur dann in einen Folgezustand $z' \in N(z)$ übergeht, wenn $\varphi(z') < \varphi(z)$ gilt.

(a) Zeigen Sie, daß der Hillclimbing-Algorithmus nach endlich vielen Schritten terminiert. Geben Sie eine möglichst gute obere Schranke für die Anzahl der Zustandsübergänge an. **(2 Punkte)**

(b) Beweisen Sie, daß der bei Abbruch des Hillclimbing-Algorithmus aktuelle Zustand $z = T$ ein minimaler aufspannender Baum von G ist.

(Hinweis: Benutzen Sie die Korrektheit des Algorithmus von Kruskal.) **(3 Punkte)**

Aufgabe 6: (Traveling Salesperson Problem (TSP))

Wir betrachten das Euklidische Traveling Salesperson Problem aus der Vorlesung. Gegeben ist eine Menge $S = \{S_1, \ldots, S_n\}$ von Punkten in der Euklidischen Ebene. Wir nehmen zur Vereinfachung der Aufgabe an, daß nicht drei Punkte aus S auf einer Geraden liegen.

Gesucht ist eine kürzeste Tour, die jeden Punkt genau einmal besucht, d.h. eine Permutation σ der Zahlen $\{1, \ldots, n\}$, so daß

$$\sum_{i=1}^{n-1} d_2(S_{\sigma(i)}, S_{\sigma(i+1)}) + d_2(S_{\sigma(n)}, S_{\sigma(1)})$$

minimiert wird. Hierbei ist d_2 die *Euklidische Metrik*. Der einfachen Notation halber nennen wir auch die entsprechende Folge $(S_{\sigma(1)}, \ldots, S_{\sigma(n)})$ eine Tour und bezeichnen sie ebenfalls mit σ.

(a) Wir nennen eine Tour σ *sich kreuzend*, wenn es $1 \leq i < j \leq n$ gibt, so daß sich die Strecken $\overline{S_{\sigma(i)}S_{\sigma(i+1)}}$ und $\overline{S_{\sigma(j)}S_{\sigma(j+1)}}$ in einem inneren Punkt der Strecken schneiden (wir setzen dabei $\sigma(n+1) := \sigma(1)$). Wenn σ nicht sich kreuzend ist, so heißt σ *kreuzungsfrei*.

Beweisen Sie, daß jede optimale Tour kreuzungsfrei ist. **(2 Punkte)**

(b) Sei opt die Länge einer kürzesten Tour. In der Vorlesung wurde bereits gezeigt, daß ein minimaler aufspannender Baum T im vollständigen Graphen G mit Eckenmenge S und euklidischen Kantengewichten nicht schwerer ist als opt.

Der *Grad* einer Ecke v in einem Graphen G ist die Anzahl der zu v inzidenten Kanten in G. Ein *Matching* in einem Graphen $H = (V, E)$ ist eine Teilmenge $M \subseteq E$ der Kanten, so daß keine zwei Kanten aus M einen gemeinsamen Endpunkt besitzen. Ein Matching heißt *perfekt*, wenn jede Ecke aus V Endpunkt einer Kante aus M ist.

Sei D die Menge derjenigen Ecken aus T, die ungeraden Grad besitzen. Im vollständigen Graphen H mit Eckenmenge D und euklidischen Kantengewichten sei M^* ein perfektes Matching mit minimalem Gewicht (Ein solches Matching kann durch klassische Methoden effizient bestimmt werden). Beweisen Sie, daß das Gewicht des Matchings M^* höchstens $\mathsf{opt}/2$ ist. **(2 Punkte)**

(c) Der Satz von Euler besagt, daß es in einem endlichen Graphen H, in dem jede Ecke geraden Grad besitzt, einen *Eulerschen* Kreis gibt, d.h. einen Kreis, der jede Kante genau einmal durchläuft. In diesem Fall heißt H dann auch *Eulersch*.

Sei H der Graph mit Eckenmenge S und allen Kanten aus T und M^*. Zeigen Sie, daß H Eulersch ist. **(1 Punkt)**

(d) Sei K ein Eulerscher Kreis im Graphen H aus (c) (Ein solcher Kreis kann ebenfalls durch bekannte Methoden effizient gefunden werden). Zeigen Sie, wie man aus K effizient eine Tour σ konstruieren kann, die Länge höchstens $3/2 \cdot \mathrm{OPT}$ besitzt. **(3 Punkte)**

Aufgabe 7: (A^*-Algorithmus)

Sei V eine Zustandsmenge mit Nachbarschaftsbeziehungen $N(v)$ $(v \in V)$. Implizit wird dadurch der gerichtete Graph $G = (V, R)$ mit $R = \{\, (v, w) : w \in N(v) \,\}$ definiert. Es sei $k\colon R \to \mathbb{R}$ eine Kantenbewertung. Gegeben sei ferner eine Startmenge $S \subseteq V$ und eine ausgezeichnete Menge $F \subseteq V$, die sogenannte „Zielmenge“.

(a) Finden Sie ein Beispiel eines endlichen Graphen, wo für $\mathrm{OPT} := \inf\{\, h^*(s) : s \in S \,\}$ gilt: $\mathrm{OPT} = -\infty$. **(1 Punkt)**

(b) Geben Sie ein Beispiel eines Graphen an, bei dem zwar $-\infty < \mathrm{OPT} < +\infty$ gilt, dennoch aber kein kostenminimaler Weg von der Startmenge S in die Zielmenge F existiert. **(1 Punkt)**

Abgabetermin: Dienstag, 25. November 1997, bis 10^{00} Uhr in den dafür vorgesehenen Kästen vor der Teilbibliothek Mathematik/Informatik

Lösungen

Aufgabe 4: (Verbesserter Algorithmus für MAX k-SAT)

(a) Der Erwartungswert $E[M]$ ist die Summe der Wahrscheinlichkeiten für die Erfülltheit der einzelnen Klauseln aus M
Für jede einzelne Klausel C gilt einer der Fälle:

Fall 1: Die atomare Formel A_1 kommt nicht in in C vor (d.h. weder als positives noch als negatives Literal). Für eine zufällige Belegung $\mathcal{B}$ gilt dann:
$P(\mathcal{B} \models C_{A_1=1}) = 1 - 2^{-k}$
$P(\mathcal{B} \models C_{A_1=0}) = 1 - 2^{-k}$

Fall 2: Die atomare Formel A_1 kommt als positives Literal vor
$P(\mathcal{B} \models C_{A_1=1}) = 1$
$P(\mathcal{B} \models C_{A_1=0}) = 1 - 2^{-k+1}$

Fall 2: Die atomare Formel A_1 kommt als negatives Literal vor
$P(\mathcal{B} \models C_{A_1=1}) = 1 - 2^{-k+1}$
$P(\mathcal{B} \models C_{A_1=0}) = 1$

Seien in M: a Klauseln ohne die atomare Formel A_1, b Klauseln mit A_1 als positivem Literal, c Klauseln mit A_1 als negativem Literal. Also ist $a+b+c = m$ und es gelten folgende Beziehungen:

$$E[M_{A_1=1}] = a \cdot (1 - 2^{-k}) + b \cdot 1 + c \cdot (1 - 2^{-k+1})$$
$$E[M_{A_1=0}] = a \cdot (1 - 2^{-k}) + b \cdot (1 - 2^{-k+1}) + c \cdot 1$$

Damit folgt:

$$\begin{aligned}
& \frac{1}{2}(E[M_{A_1=1}] + E[M_{A_1=0}]) \\
= \ & \frac{1}{2} \cdot (2 \cdot a \cdot (1 - 2^{-k}) + b \cdot (2 - 2^{-k+1}) + c \cdot (2 - 2^{-k+1})) \\
= \ & \frac{1}{2} \cdot (2 \cdot a \cdot (1 - 2^{-k}) + 2 \cdot b \cdot (1 - 2^{-k}) + 2 \cdot c \cdot (1 - 2^{-k})) \\
= \ & (a + b + c) \cdot (1 - 2^{-k}) = m(1 - 2^{-k}) \\
= \ & E[M] \qquad \square
\end{aligned}$$

Diese Gleichung zeigt, daß eine der beiden möglichen Belegungen der atomaren Formel A_1 den Erwartungswert, für die Erfülltheit der Klauselmenge nicht verkleinert.

(b) Wie gerade gezeigt kann man also durch eine geschickte Belegung einer atomaren Variablen einen Erwartungswert $\geq$ dem alten erhalten. Aufgabe (a) bewies uns den speziellen Fall bei der Belegung der ersten atomaren Variablen. Dieses Ergebnis möchte ich nun verallgemeinern auf die schrittweise Belegung aller atomaren Variablen.

Seien nun in M:

- S die Teilmenge der Klauseln aus M, die in vorangegangenen Schritten schon so belegt wurde, daß sie gültig sind.
- a die Anzahl der Klauseln aus $M \setminus S$, die die atomare Formel A_i nicht enthalten.
- b die Anzahl der Klauseln aus $M \setminus S$, die die atomare Formel A_i als positives Literal enthalten.
- c die Anzahl der Klauseln aus $M \setminus S$, die die atomare Formel A_i als negatives Literal enthalten.

Wir betrachten jetzt die möglicherweise schon früher durch Festsetzen einiger Variablen veränderten Klausel $C \in M$ und belegen nun die Variable A_i. Es sein nun q die Anzahl der schon festgelegten Variablen in der Klausel C, somit sind noch $r =_{def} k - q$ zu belegen. Es unterscheiden sich wieder folgende Fälle:

Fall 1: C war bereits erfüllt ($C \in S$), also vorher $P(\mathcal{B} \models C) = 1$
$P(\mathcal{B} \models C_{A_i=1}) = 1$
$P(\mathcal{B} \models C_{A_i=0}) = 1$
also gilt trivialerweise auch
$P(\mathcal{B} \models C) = \frac{1}{2} \cdot (P(\mathcal{B} \models C_{A_i=1}) + P(\mathcal{B} \models C_{A_i=0}))$

Fall 2: C war bisher nicht erfüllt und A_i kommt nicht in C vor (weder als positives noch als negatives Literal). Da es noch r zu belegende Variablen gibt, gilt vorher $P(\mathcal{B} \models C) = 1 - 2^{-r}$
$P(\mathcal{B} \models C_{A_i=1}) = 1 - 2^{-r}$
$P(\mathcal{B} \models C_{A_i=0}) = 1 - 2^{-r}$
also gilt wieder
$P(\mathcal{B} \models C) = \frac{1}{2} \cdot (P(\mathcal{B} \models C_{A_i=1}) + P(\mathcal{B} \models C_{A_i=0})$

Fall 3: C war bisher nicht erfüllt und A_i kommt in C als positives Literal vor. Also gilt vorher $P(\mathcal{B} \models C) = 1 - 2^{-r}$
$P(\mathcal{B} \models C_{A_i=1}) = 1$
$P(\mathcal{B} \models C_{A_i=0}) = 1 - 2^{-r+1}$
diesmal:

$$\begin{aligned} P(\mathcal{B} \models C) &= 1 - 2^{-r} \\ &= \frac{1}{2} \cdot (2 - 2^{-r+1}) \\ &= \frac{1}{2} \cdot (1 + 1 - 2^{-r+1}) \\ &= \frac{1}{2} \cdot (P(\mathcal{B} \models C_{A_i=1}) + P(\mathcal{B} \models C_{A_i=0}) \end{aligned}$$

Fall 4: C war bisher nicht erfüllt und A_i kommt in C als negatives Literal vor. Also gilt vorher $P(\mathcal{B} \models C) = 1 - 2^{-r}$
$P(\mathcal{B} \models C_{A_i=1}) = 1 - 2^{-r+1}$
$P(\mathcal{B} \models C_{A_i=0}) = 1$

abschließend:

$$
\begin{aligned}
P(\mathcal{B} \models C) &= 1 - 2^{-r} \\
&= \frac{1}{2} \cdot (2 - 2^{-r+1}) \\
&= \frac{1}{2} \cdot (1 - 2^{-r+1} + 1) \\
&= \frac{1}{2} \cdot (P(\mathcal{B} \models C_{A_i=1}) + P(\mathcal{B} \models C_{A_i=0})
\end{aligned}
$$

Da die Wahrscheinlichkeiten bei geschickter schrittweiser Belegung für jede Klausel nie kleiner werden, als sie schon waren ist auch der Erwartungswert nach Belegung aller freien Variablen größer oder gleich dem ursprünglichen, nämlich $(1 - 2^{-k}) \cdot m$. Da jedoch schon jede Atomformel mit einem Wahrheitswert belegt ist ist dies auch schon der „wirkliche“ Wert der mindestens erfüllten Klauseln.

Aufgabe 5: (Hillclimbing und minimale aufspannende Bäume)

(a) Da der modifizierte Hillclimbing-Algorithmus von einem Zustand z nur dann zu einem neuen Zustand z' übergeht, wenn $\varphi(z') < \varphi(z)$ gilt, ist eine spätere Rückkehr zu z ausgeschlossen, weil sonst $\varphi(z) < \varphi(z)$ gelten müsste.
Gleichzeitig ist die Menge Z der Zustände endlich, da G ein endlicher Graph ist, und somit auch nur endlich viele spannenden Bäume in G existieren.
Da Z endlich ist, und kein Zustand mehrmals angenommen wird, können nur endlich viele Zustandsübergänge erfolgen. Der Algorithmus terminiert also nach endlich vielen Schritten.
Sei T der spannende Baum, der Startzustand des HC-Algorithmus ist. Sei weiterhin $n := |V|$ und $m := |E|$. Für jede Kante $e \in T$ gilt: Es gibt höchstens $m-1$ Kanten, die ein geringeres Gewicht haben als e, d.h. ausgehend von der Kante e, kann es höchsten zu einer Folge von $m-1$ Ersetzungen durch jeweils leichtere Kanten kommen. Da dies für alle $n-1$ Kanten aus T gilt, terminiert der modifizierte HC-Algorithmus spätestens nach $(n-1)(m-1)$ Zustandsübergängen.

(b) Sei T der spannende Baum bei Abbruch des HC-Algorithmus und T^* ein minimaler spannender Baum, wie er zum Beispiel vom Kruskal-Algorithmus geliefert wird. Wir werden im Folgenden zeigen, daß das Gesamtgewicht von T gleich dem Gesamtgewicht von T^* ist, indem wir ein Verfahren angeben, bei dem wir jede Kante aus T durch eine gleichschwere Kante aus T^* ersetzen, so daß der sich ergebende Graph immernoch ein spannender Baum ist.

Sei $e \in T$ und $e \notin T^*$; wenn wir e aus T entfernen, zerfällt der spannende Baum in genau zwei Zusammenhangskomponenten ZK_1 und ZK_2. Sei $\sigma(ZK_1, ZK_2)$ der zugehörige Schnitt auf dem Graphen G. Wir betrachten nun einmal, was passiert, wenn wir die Kante e zu T^* hinzufügen:

Da T^* ein spannender Baum ist, entsteht durch das Hinzufügen einer Kante ein einfacher Kreis $k = (e_1, \ldots, e_k)$ mit o.B.d.A. $e_1 = e$. Der Kreis

k muss eine Kante $e_i \in \sigma(ZK_1, ZK_2)$ mit $e_i \neq e$ und $e_i \notin T$ enthalten. Wir können e in T durch e_i ersetzen, und $(T \setminus \{e\}) \cup \{e_i\} := T'$ ist wieder ein zusammenhängender Graph mit $|V| - 1$ Kanten und somit ein spannender Baum.

Was können wir nun über das Gewicht $w(e_i)$ aussagen?

1.) $w(e_i) \geq w(e)$
Wäre $w(e_i) < w(e)$, dann wäre auch das Gesamtgewicht von T' geringer als das Gesamtgewicht von T. Somit wäre T' ein spannender Baum mit $\varphi(T') < \varphi(T)$. Da aber T' in der Nachfolgermenge von T liegt, hätte der modifizierte HC-Algorithmus nicht mit T terminieren dürfen, sondern zum Zustand T' übergehen müssen.

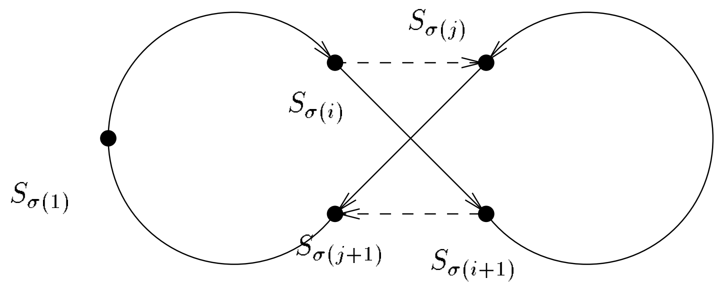
2.) $w(e_i) \leq w(e)$
Wäre $w(e_i) > w(e)$, dann könnte man e_i in T^* durch e ersetzen, und da e_i und e auf einem gemeinsamen Kreis k liegen, wäre $(T^* \setminus \{e_i\}) \cup \{e\}$ wieder ein zusammenhängender Graph mit $|V| - 1$ Kanten, also ein spannender Baum, mit einem veringertem Gesamtgewicht. Dies wäre aber ein Widerspruch zur Annahme, dass T^* ein minimaler spannender Baum ist.

Folglich muß $w(e_i) = w(e)$ gelten und somit ist das Gesamtgewicht von T' gleich dem Gewicht von T.
Ersetzt man nun sukzessive alle Kanten aus T durch gleichschwere Kanten aus T^*, so erhält man einen spannenden Baum T'', der das gleiche Gewicht hat wie T. Da T'' aber nur noch Kanten aus T^* enthält muß $T^* = T''$ gelten. Also war T bereits ein minimaler spannender Baum. □

Aufgabe 6: (Traveling Salesperson Problem (TSP))

(a) Beweis durch Widerspruch:

Sei eine optimale nicht kreuzungsfreie Tour gegeben.



Entferne aus diese Tour die beiden Strecken $a_1 := (S_{\sigma(i)}, S_{\sigma(i+1)})$ und $b_1 := (S_{\sigma(j)}, S_{\sigma(j+1)})$. Füge stattdessen die beiden Strecken $a_2 := (S_{\sigma(i)}, S_{\sigma(j)})$ und $b_2 := (S_{\sigma(i+1)}, S_{\sigma(j+1)})$ ein und drehe auf dem Weg von $S_{\sigma(i+1)}$ nach $S_{\sigma(j)}$ die Richtung um.

Wegen der Dreiecksungleichung in der Euklidischen Ebene ist die so erhaltene Tour kürzer als die Ausgangstour im Widerspruch zur Optimalität. Es gilt $a_2 \leq a_1$, $b_2 \leq b_1$ und im nicht entarteten Fall sind die Ungleichungen sogar strikt.

Daher ist jede optimale Tour kreuzungsfrei.

(b) Sei $T_{opt} = (S_{\sigma(1)}, \cdots, S_{\sigma(n)})$ eine optimale Tour der Länge opt. Mittels „Shortcuts" kann man aus T_{opt} eine Tour T_D konstruieren, die nur Ecken aus D besucht. Es gilt daher $l(T_{opt}) \geq l(T_D)$. Insbesondere ist $m := |D|$ gerade, da in einem Graphen die Anzahl der Ecken mit ungeradem Grad gerade ist. Sei $T_D = (d_1, \cdots, d_m)$ eine solche Tour.

Diese Tour T_D induziert zwei perfekte Matchings auf D, die jeweils genau $\frac{m}{2}$ Kanten enthalten:

(a) $M_1 : (d_1, d_2), (d_3, d_4), \cdots, (d_{m-1}, d_m)$

(b) $M_2 : (d_2, d_3), (d_4, d_5), \cdots, (d_m, d_1)$

Für das Gewicht der beiden Matchings gilt:

$l(M_1) + l(M_2) = l(T_D) \leq l(T_{opt}) = \mathsf{opt}$.

Daher gilt entweder $l(M_1) \leq \frac{1}{2} \cdot l(T_D)$ oder $l(M_2) \leq \frac{1}{2} \cdot l(T_D)$.

Also existiert ein perfektes Matching, dessen Gewicht höchstens $\mathsf{opt}/2$ ist

(c) Der Graph H mit Eckenmenge S und allen Kanten aus T und M^* ist *eulersch*. Um dies zu zeigen, ist muß man nachweisen, daß jede Ecke in H geraden Grad besitzt.

Für alle Ecken v_i aus T mit geradem Grad gilt, daß diese mit keiner Kante des perfekten Matchings inzidieren (Nach der Wahl von M^* bzw. nach Angabe). Daher bleibt der gerade Grad von v_i erhalten.

Betrachte daher nur die Ecken v_j von T, die ungeraden Grad besitzen. Für diese wird auf dem zugehörigen vollständigen Graphen ein perfektes Matching bestimmt. Dieses existiert, da in einem Graphen die Anzahl der Ecken mit ungeradem Grad gerade ist. Zu jeder dieser Ecken ist genau eine Kante des Matchings inzident. Daher hat v_j in H geraden Grad.

(d) Sei K ein Eulerscher Kreis im Graphen H aus Teilaufgabe (c). Dieser durchläuft jede Kante vom T und aus M^* genau einmal.

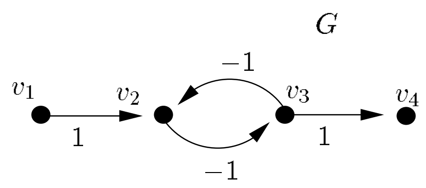
Aus (b) ist bekannt, daß die Summe der Kantengewichte in T höchstens opt und in M^* höchstens $\mathsf{opt}/2$ beträgt.

Der Eulersche Weg hat also eine Länge von höchstens $\frac{3}{2} \cdot \mathsf{opt}$.

Durch Verwenden von „Shortcuts" wie in der Vorlesung, um mehrmaliges Besuchen von Ecken $S_{\sigma(j)}$ mit $j \neq 1, j \neq n$ zu vermeiden, wird die Tour nicht schlechter, da wir uns in der Euklidischen Ebene befinden.

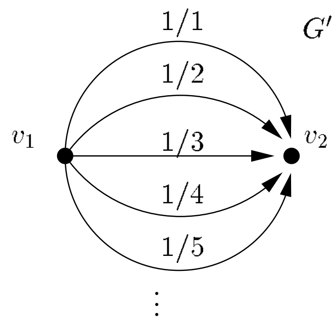
Aufgabe 7: (A^*-Algorithmus)

(a) Gegeben sei folgender Graph G:



Sei $S := \{v_1\}$ und $F := \{v_4\}$ Der Graph enthält einen Kreis negativer Länge und deshalb gilt:
$\text{OPT} = \inf\{h^*(s) : s \in S\} = h^*(v_1) = -\infty$

(b) Gegeben sei folgender unendlicher Graph $G' = (V, R)$:



$|R| = \infty$ mit $k(r_i) := 1/i \quad ; i \in \mathbb{N}$. Sei $S := \{v_1\}$ und $F := \{v_2\}$
Dann gilt $\text{OPT} = \inf\{h^*(s) : s \in S\} = h^*(v_1) = 0$ Aber es gibt dennoch keinen Weg von v_1 nach v_2 mit minimalen Kosten, da für keine Kante $r \in R$ gilt $k(r) = 0$.