Kapitel 5 Die Probabilistische Methode

2 Grundprinzipien:

1. Ist X eine Zufallsvariable, dann gibt es $x_0 \leq E[X]$ und $x_1 \geq E[X]$ mit $Prob[X = x_0] > 0$, $Prob[X = x_1] > 0$

2. Ist die Wahrscheinlichkeit, dass ein zufällig ausgewähltes Objekt eine Eigenschaft 45 hat, ungleich 0, so muß es ein Objekt mit der Eigenschaft 45 geben.

Anwendung 1: Maximum Satisfiability

- Seien $x_1, \ldots, x_n$ aussagenlogische Variablen.
- Ein Literal ist eine Formel der Gestalt x_i oder $\neg x_i$ $(1 \leq i \leq n)$.
- Eine Klausel ist eine Formel der Gestalt $(y_1 \vee y_2 \vee \cdots \vee y_k)$ $(k \geq 1)$

Satz 1: Sei Φ eine Menge von m Klauseln. Dann gibt es eine Wahrheitsbelegung für $x_1, \ldots, x_n$ so dass mindestens $\frac{m}{2}$ viele Klauseln aus Φ wahr sind.

O.B.d.A enthält keine Klausel eine Variable x_i und $\neg x_i$

Beweis: (mittels der probabilistischen Methode).

Setze jede Variable $\overbrace{x_i}$ mit Wahrscheinlichkeit $\frac{1}{2}$ auf 0 und mit Wahrscheinlichkeit $\frac{1}{2}$ auf 1.

Für $1 \leq i \leq m$ definiere Zufallsvariable

$$Z_i = \begin{cases} 1 & \text{falls sich i-te Klausel zu wahr auswertet} \\ 0 & \text{sonst} \end{cases}$$

Ist C eine Klausel mit k Literalen so gilt:

$$\text{Prob}[C \text{ ist erfüllt}] = 1 - 2^{-k}$$

$$\geq \frac{1}{2} \quad (\text{da } k \geq 1)$$

$$\rightarrow E[Z_i] \geq \frac{1}{2}$$

$$\rightarrow E[\text{Anzahl d. erfüllten Klauseln}]$$

$$= \sum_{i=1}^{m} E[Z_i] \geq \frac{m}{2}. \qquad \square$$

Bemerkung:

- Satz 1 kann nicht verbessert werden: $\Phi = \{(x_1), (\neg x_1)\}$
- Falls jede Klausel aus Φ mind. k Literale enthält, so folgt aus obigen Beweis, dass es eine Wahrheitsbelegung gibt, welche mind. $(1-2^{-k}) \cdot m$ viele Klauseln erfüllt.

MAX-SAT ist das folgende Optimierungsproblem:

INPUT: Menge von Klauseln Φ

ZIEL: Finde Wahrheitsbelegung, welche möglichst viele Klauseln aus Φ erfüllt.

Sei A ein Algorithmus, der für eine gegebene Klauselmenge eine Wahrheitsbelegung berechnet.

Für eine Klauselmenge Φ sei

$m_A(\Phi)$ = Anzahl der Klauseln, die A bei Eingabe Φ erfüllt

$m_*(\Phi)$ = maximale Zahl von (durch eine bestimmte Wahrheitsbelegung) gleichzeitig erfüllbaren Klauseln von Φ.

Die <u>Approximationsrate</u> von A ist

$$\text{infimum}\left\{ \frac{m_A(\Phi)}{m_*(\Phi)} \;\middle|\; \Phi \text{ beliebige Klauselmenge} \right\}$$

Ist A ein randomisierter Algorithmus so ersetzen wir in obiger Definition $m_A(\Phi)$ durch $E[m_A(\Phi)]$

Bemerkungen:

- A ist "gut", falls die Approximationsrate von A nahe bei 1 liegt.
- Aus dem Beweis von Satz 1 ergibt sich ein randomisierter Algorithmus mit Approximationsrate $\geq \frac{1}{2}$.

<u>Satz 2</u>: Es gibt einen randomisierten Algorithmus für MAX-SAT, dessen Approximationsrate mind. $\frac{3}{4}$ ist.

Beachte: Enthält die Input-Klauselmenge Φ nur Klauseln mit mind. 2 Literalen, so hat der Algorithmus

aus dem Beweis von Satz 1 (Würfeln der Wahrheitsbelegung) eine Approximationsrate von mind.
$1 - 2^{-2} = \frac{3}{4}$.

Wir konstruieren deshalb einen zweiten randomisierten Algorithmus für MAX-SAT, der speziell bei vielen Klauseln mit nur einem Literal gut ist.

Sei $\phi = \{C_1, \dots, C_m\}$ unsere Input-Klauselmenge, seien $x_1, \dots, x_n$ die aussagenlogischen Formeln in ϕ.

Sei $C_j^+ = \{ i \mid 1 \le i \le n, x_i \text{ kommt in } C_j \text{ vor} \}$

$C_j^- = \{ i \mid 1 \le i \le n, \neg x_i \text{ kommt in } C_j \text{ vor} \}$

Formulierung von MAX-SAT als Integer Linear Program

Ordne jeder aussagenlogischen Variablen x_i eine Integer-Variable $y_i \in \mathbb{N}$ zu, und ordne jeder Klausel C_j eine Integer-Variable $z_j \in \mathbb{N}$ zu.

Idee: $z_j = \begin{cases} 1 & \text{falls } C_j \text{ erfüllt ist} \\ 0 & \text{sonst.} \end{cases}$

$\longrightarrow$ Maximiere $\sum_{j=1}^{m} z_j$

unter folgenden Randbedingungen (*)

$\forall i \, \forall j : 0 \le y_i, z_j \le 1$

$\forall j : \sum_{i \in C_j^+} y_i + \sum_{i \in C_j^-} (1 - y_i) \ge z_j$

Die <u>linear programming relaxation</u> von (*) erhalten wir, indem wir $y_i, z_j \in \mathbb{R}$ in (*) erlauben (d.h. $0 \leq y_i, z_j \leq 1$).

Diese kann mit Standardmethoden effizient (in Polynomialzeit) gelöst werden. Seien $\hat{y}_i, \hat{z}_j \in [0,1]$ die Werte, die wir so erhalten.

Beachte: $\sum_{j=1}^{m} \hat{z}_j \geq m_*(\phi)$

Wir wenden nun auf $\hat{y}_1, \dots, \hat{y}_n$ <u>randomisiertes Runden</u> an:

Setze y_i unabhängig von den anderen $y_{i'}$ mit Wahrscheinlichkeit $\hat{y}_i$ auf 1 und mit Wahrscheinlichkeit $1-\hat{y}_i$ auf 0.
↑ entspricht x_i = FALSE

Sei $\beta_k = 1 - (1 - \frac{1}{k})^k \geq 1 - \frac{1}{e}$

für $k \geq 1$.

<u>Lemma 1</u>: Sei C_j eine Klausel mit k Literalen. Die Wahrscheinlichkeit, dass C_j beim rand. Runden erfüllt wird, ist mind. $\beta_k \cdot \hat{z}_j$

Beweis: O.B.d.A. können wir $C_j = (x_1 \wedge x_2 \wedge \dots \wedge x_k)$ annehmen.

$\xrightarrow{(*)}$ $\hat{y}_1 + \dots + \hat{y}_k \geq \hat{z}_j$

Prob[C_j wird beim rand. Runden nicht erfüllt] =
Prob[$\hat{y}_1, \dots, \hat{y}_k$ werden auf 0 gerundet]

$$= \prod_{i=1}^{k} (1 - \hat{y}_i)$$

Es genügt somit zu zeigen:

$$1 - \prod_{i=1}^{k} (1 - \hat{y}_i) \geq \beta_k \cdot \hat{z}_j$$

$\prod_{i=1}^{k} (1 - \hat{y}_i)$ wird unter der Randbedingung $\hat{y}_1 + \dots + \hat{y}_k \geq \hat{z}_j$ (d.h $(1-\hat{y}_1) + \dots + (1-\hat{y}_k) \leq k - \hat{z}_j$) maximal für $\hat{y}_1 = \dots = \hat{y}_k = \hat{z}_j / k$

→ Es genügt zu zeigen:

(**) $\underbrace{1 - (1 - \frac{z}{k})^k}_{\text{konvex}} \geq \underbrace{\beta_k \cdot z}_{\text{linear}}$ falls $k \geq 1$, $z \in [0,1]$

↳ (**) muß nur für Randwerte $z=0$, $z=1$ gezeigt werden.

$z=0$: $1 - (1 - \frac{0}{k})^k \geq \beta_k \cdot 0$ ✓

$z=1$: $\underbrace{1 - (1 - \frac{1}{k})^k}_{\beta_k} \geq \beta_k \cdot 1$ ✓ □

Aus Lemma 1 folgt:

$E[\text{Anzahl d. Klauseln, die durch rand. Runden erfüllt werden}]$

$$= \sum_{j=1}^{m} \text{Prob}[C_j \text{ wird beim rand. Runden erfüllt}]$$

$$\overset{\text{Lemma 1}}{\geq} \sum_{j=1}^{m} \beta_k \cdot \hat{z}_j \geq \beta_k \cdot m_*(\phi)$$

$$\geq (1 - \frac{1}{e}) \cdot m_*(\phi)$$

Wir haben somit folgenden Satz gezeigt:

Satz 3 Randomisiertes Runden erreicht eine Approximationsrate (für MAX-SAT) von mind.
$1 - \frac{1}{e}$

Beachte: Dies ist bereits eine Verbesserung gegenüber der Approximationsrate von $\frac{1}{2}$, die sich bei zufälliger Belegung der Variablen ergibt. Unser Ziel einer Approximationsrate von mind. $\frac{3}{4}$ haben wir aber noch nicht erreicht.

Wir lassen nun unsere beiden rand. Algorithmen für MAX-SAT (zufällige Belegung, rand. Runden) nacheinander laufen, und nehmen diejenige Wahrheitsbelegung, welche mehr Klauseln erfüllt. Sei

$n_1 = E[$ Anzahl von Klauseln, die durch zufällige Belegung erfüllt werden $]$

$n_2 = E[$ Anzahl von Klauseln, die durch rand. Runden erfüllt werden $]$

Satz 4: $\max\{n_1, n_2\} \geq \frac{3}{4} \cdot m_*(\phi)$

Beweis: Es genügt zu zeigen:

$$\frac{n_1 + n_2}{2} \geq \frac{3}{4} \cdot \sum_{j=1}^{m} \hat{z}_j$$

Sei $S_k \subseteq \phi$ die Menge aller Klauseln mit genau k Literalen

Es gilt: $m_1 = \sum_{k\geq 1} \sum_{C_j \in S_k} (1-2^{-k})$

$\geq \sum_{k\geq 1} \sum_{C_j \in S_k} (1-2^{-k}) \cdot \underbrace{\hat{z}_j}_{\in [0,1]}$

$m_2 \geq \sum_{k\geq 1} \sum_{C_j \in S_k} \beta_k \cdot \hat{z}_j$

$\uparrow$ Lemma 1

$\Rightarrow \frac{m_1+m_2}{2} \geq \sum_{k\geq 1} \sum_{C_j \in S_k} \frac{1-2^{-k}+\beta_k}{2} \cdot \hat{z}_j$

Weiter gilt: $1-2^{-k}+\beta_k =$

$1-2^{-k}+1-(1-\frac{1}{k})^k =$

$2-2^{-k}-(1-\frac{1}{k})^k \geq \frac{3}{2}$ denn

$2^{-k}+(1-\frac{1}{k})^k \leq \frac{1}{2}$ für alle $k \in \mathbb{N}$

(für $k \in \{1,2\}$ gilt $2^{-k}+(1-\frac{1}{k})^k = \frac{1}{2}$,

für $k \geq 3$ gilt $2^{-k}+(1-\frac{1}{k})^k \leq \frac{1}{8}+\frac{1}{e}$

$\leq 1/2$)

$\longrightarrow \frac{m_1+m_2}{2} \geq \frac{3}{4} \cdot \sum_k \sum_{C_j \in S_k} \hat{z}_j$

$= \frac{3}{4} \cdot \sum_{j=1}^{m} \hat{z}_j$ □

Lovász Local Lemma

- Seien $\mathcal{E}_1, \dots, \mathcal{E}_n$ Ereignisse.
  Sei $T \subseteq \{1, \dots, n\}$ und $i \in \{1, \dots, n\}$.
- $\mathcal{E}_i$ ist unabhängig von der Menge $\{\mathcal{E}_j \mid j \in T\}$ falls für jede Teilmenge $U \subseteq T$ gilt:

$$\text{Prob}\left[\mathcal{E}_i \mid \bigcap_{j \in U} A_j\right] = \text{Prob}[\mathcal{E}_i]$$

wobei $A_j = \begin{cases} \mathcal{E}_j \text{ oder} \\ \overline{\mathcal{E}_j} \end{cases}$

- Ein gerichteter Graph G ist ein <u>Abhängigkeitsgraph</u> für $\mathcal{E}_1, \dots, \mathcal{E}_n$, falls gilt:

a) Die Knotenmenge von G ist $\mathcal{E}_1, \dots, \mathcal{E}_n$

b) Jedes Ereignis $\mathcal{E}_i$ ist unabhängig von der Menge $\{\mathcal{E}_j \mid (\mathcal{E}_i, \mathcal{E}_j)$ keine Kante in $G\}$

<u>Beispiele</u>:

- Der vollständige Graph auf $\mathcal{E}_1, \dots, \mathcal{E}_n$ (für alle i, j ist $\mathcal{E}_i \to \mathcal{E}_j$ eine Kante) ist stets ein Abhängigkeitsgraph für $\mathcal{E}_1, \dots, \mathcal{E}_n$.

- Seien $\mathcal{E}_1, \dots, \mathcal{E}_n$ paarweise unabhängig. Jeder Graph G mit folgender Eigenschaft ist

ein Abhängigkeitsgraph für $\mathcal{E}_1, \ldots, \mathcal{E}_n$: Für jedes Ereignis $\mathcal{E}_i$ gibt es genau ein Ereignis $\mathcal{E}_j$ $(j \neq i)$, so dass es in G keine Kante vom $\mathcal{E}_i$ nach $\mathcal{E}_j$ gibt.

Lovász Local Lemma: Sei $G = (\{\mathcal{E}_1, \ldots, \mathcal{E}_n\}, E)$ ein Abhängigkeitsgraph für die Ereignisse $\mathcal{E}_1, \ldots, \mathcal{E}_n$. Angenommen es gibt Zahlen $x_i \in [0,1]$ $(1 \leq i \leq n)$ mit

$$\text{Prob}[\mathcal{E}_i] \leq x_i \cdot \prod_{(i,j) \in E} (1 - x_j)$$

Dann gilt: $\text{Prob}\left[\bigcap_{i=1}^{n} \overline{\mathcal{E}}_i\right] \geq \prod_{i=1}^{n} (1 - x_i)$

Beweis: Sei $S \subseteq \{1, \ldots, n\}$ und $i \notin S$

Behauptung: $\text{Prob}\left[\mathcal{E}_i \mid \bigcap_{j \in S} \overline{\mathcal{E}}_j\right] \leq x_i$

Beweis durch Induktion über $|S|$

$S = \emptyset$: $\text{Prob}\left[\mathcal{E}_i \mid \bigcap_{j \in S} \overline{\mathcal{E}}_j\right] =$

$$\text{Prob}[\mathcal{E}_i] \leq x_i \cdot \prod_{(i,j) \in E} (1 - x_j) \leq x_i$$

$S \neq \emptyset$: Sei $S_1 = \{j \in S \mid (i,j) \in E\}$

$$S_2 = S \setminus S_1$$

Wegen $\text{Prob}[A \mid B \cap C] = \frac{\text{Prob}[A \cap B \mid C]}{\text{Prob}[B \mid C]}$

gilt:

$$\mathrm{Prob}\Big[\varepsilon_i \,\Big|\, \bigcap_{j\in S} \bar{\varepsilon}_j\Big] = \mathrm{Prob}\Big[\varepsilon_i \,\Big|\, \bigcap_{j\in S_1} \bar{\varepsilon}_j \cap \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big]$$

$$= \frac{\mathrm{Prob}\big[\varepsilon_i \cap \bigcap_{j\in S_1} \bar{\varepsilon}_j \,\big|\, \bigcap_{j\in S_2} \bar{\varepsilon}_j\big]}{\mathrm{Prob}\big[\bigcap_{j\in S_1} \bar{\varepsilon}_j \,\big|\, \bigcap_{j\in S_2} \bar{\varepsilon}_j\big]} = \frac{\alpha}{\beta}$$

$$\alpha \le \mathrm{Prob}\Big[\varepsilon_i \,\Big|\, \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big] \overset{\nabla}{\underset{\uparrow}{=}} \mathrm{Prob}[\varepsilon_i]$$

↑ Abhängigkeitsgraph für $\varepsilon_{i_1}, \dots, \varepsilon_{i_r}$

$$\le x_i \cdot \prod_{(i,j)\in E} (1 - x_j)$$

Sei $S_1 = \{j_1, \dots, j_r\}$ $(r \ge 0)$

Falls $r = 0$, d.h. $S = S_2$:

$$\mathrm{Prob}\Big[\varepsilon_i \,\Big|\, \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big] = \mathrm{Prob}[\varepsilon_i] \le x_i \quad \checkmark$$

$S_2 = S$

Sei nun $r > 0$.

$$\beta = \mathrm{Prob}\Big[\bar{\varepsilon}_{j_1} \cap \dots \cap \bar{\varepsilon}_{j_r} \,\Big|\, \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big]$$

$$= \mathrm{Prob}\Big[\bar{\varepsilon}_{j_1} \,\Big|\, \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big] \cdot$$

$$\mathrm{Prob}\Big[\bar{\varepsilon}_{j_2} \,\Big|\, \bar{\varepsilon}_{j_1} \cap \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big] \cdot \dots \cdot$$

$$\mathrm{Prob}\Big[\bar{\varepsilon}_{j_r} \,\Big|\, \bar{\varepsilon}_{j_1} \cap \dots \cap \bar{\varepsilon}_{j_{r-1}} \cap \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big]$$

$$= \Big(1 - \overbrace{\mathrm{Prob}\Big[\varepsilon_{j_1} \,\Big|\, \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big]}^{\le x_{j_1}\ (IH)}\Big) \cdot$$

$$\Big(1 - \mathrm{Prob}\Big[\varepsilon_{j_2} \,\Big|\, \bar{\varepsilon}_{j_1} \cap \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big]\Big) \cdot \dots \cdot$$

$$\Big(1 - \mathrm{Prob}\Big[\varepsilon_{j_r} \,\Big|\, \bar{\varepsilon}_{j_1} \cap \dots \cap \bar{\varepsilon}_{j_{r-1}} \cap \bigcap_{j\in S_2} \bar{\varepsilon}_j\Big]\Big)$$

$$\ge (1 - x_{j_1}) \cdot (1 - x_{j_2}) \cdots (1 - x_{j_r})$$

↑ Induktionshypothese

$$\geq \prod_{(i,j)\in E} (1-x_j)$$

$$\rightarrow \frac{\alpha}{\beta} \leq \frac{x_i \cdot \prod_{(i,j)\in E} (1-x_j)}{\prod_{(i,j)\in E} (1-x_j)} = x_i$$

Die beweist die Behauptung

$\text{Prob}[\varepsilon_i \mid \bigcap_{j\in S} \overline{\varepsilon}_j] \leq x_i$ falls $i \notin S$.

Es folgt: $\text{Prob}[\bigcap_{j=1}^{m} \overline{\varepsilon}_j] =$

$(1-\text{Prob}[\varepsilon_1])(1-\text{Prob}[\varepsilon_2 \mid \overline{\varepsilon}_1]) \cdots$

$\cdots (1-\text{Prob}[\varepsilon_m \mid \bigcap_{j=1}^{m-1} \overline{\varepsilon}_j]) \geq$

$\prod_{i=1}^{m} (1-x_i)$ □

(p<1)

Korollar: Seien $\varepsilon_1, \ldots, \varepsilon_m$ Ereignisse mit $\text{Prob}[\varepsilon_i] \leq p$ für $1 \leq i \leq m$, so dass jeden ε_i unabhängig zu einer Teilmenge $F_i \subseteq \{\varepsilon_1, \ldots, \varepsilon_m\}$ mit $|F_i| \geq m-d$ ist. Sei $e \cdot p \cdot (d+1) \leq 1$. Dann gilt $\text{Prob}[\bigcap_{i=1}^{m} \overline{\varepsilon}_i] > 0$.

Beweis: Es muß $d \geq 1$ gelten (ε_i ist abhängig von ε_i, da $\text{Prob}[\varepsilon_i] < 1$).

Es existiert ein Abhängigkeitsgraph für $\varepsilon_1, \ldots, \varepsilon_m$, so dass jeder ε_i höchstens d Nachbarn hat.

Sei $x_i = \frac{1}{d+1} < 1$ für $1 \leq i \leq m$.

$\rightarrow x_i (1-x_i)^d = \frac{1}{d+1} \left(1-\frac{1}{d+1}\right)^d$

Es gilt: $\left(1-\frac{1}{d+1}\right)^d > \frac{1}{e}$:

dies ist äquivalent zu

$$\frac{d+1}{d} < e^{1/d}$$

Es gilt:

$$\frac{d+1}{d} = 1 + \frac{1}{d} < 1 + \frac{1}{d} + \frac{1}{2!}\left(\frac{1}{d}\right)^2 + \frac{1}{3!}\left(\frac{1}{d}\right)^3 + \dots = e^{1/d}$$

- Also gilt:

$$x_i \prod_{(i,j)\in E} (1-x_j) \geq x_i (1-x_i)^d$$

$$= \frac{1}{d+1}\left(1-\frac{1}{d+1}\right)^d > \frac{1}{d+1}\cdot\frac{1}{e} \geq p$$

$$\geq \text{Prob}[\mathcal{E}_i].$$

Aus Lovász Local Lemma folgt:

$$\text{Prob}\left[\bigcap_{i=1}^{m} \bar{\mathcal{E}}_i\right] \geq \prod_{i=1}^{m} (1-x_i) > 0. \quad \square$$

Eine Anwendung von Lovász Local Lemma auf k-SAT.

<u>Satz 1</u> Sei $\phi = \{C_1, \dots, C_m\}$ eine Menge von Klauseln, wobei jedes C_i aus genau k Literalen besteht. Angenommen jede Variable x_j kommt (negiert oder unnegiert in höchstens $2^{k/50}$ Klauseln vor und $k \geq 4$.

Dann ist ϕ erfüllbar.

<u>Beweis</u>: Setze jede Variable mit Wahrscheinlichkeit $\frac{1}{2}$ auf 0 oder 1.

Sei $\mathcal{E}_i$ das Ereignis, dass C_i <u>nicht</u> erfüllt wird.

Wir müssen zeigen:

$$\text{Prob}\left[\bigcap_{i=1}^{m} \bar{\mathcal{E}}_i\right] > 0.$$

- Sei F_i die Menge aller Ereignisse $\mathcal{E}_j$, so dass C_i und C_j keine gemeinsame Variable haben.

  $\Rightarrow \mathcal{E}_i$ ist unabhängig von F_i.
- Es gibt höchstens $d := k \cdot 2^{k/50}$ Klauseln, die mit C_i eine gemeinsame Variable haben.

$\Rightarrow |F_i| \geq m - d$.

- Außerdem gilt $\text{Prob}[\mathcal{E}_i] = 2^{-k}$
- Nach Korollar zu Lovász Local Lemma genügt es also zu zeigen:

$$\underbrace{e \cdot 2^{-k} \left(k \cdot 2^{k/50} + 1\right)}_{} \leq 1$$

$$= e \cdot \left(k \cdot 2^{-\frac{49}{50} \cdot k} + 2^{-k}\right) \leq 1 \quad \text{für } k \geq 4$$

□

- Sei $\phi = \{C_1, \ldots, C_m\}$ wieder eine k-SAT Instanz, die die Bedingungen aus Satz 1 erfüllt.
- Wir wollen nun mittels eines randomisierten Algorithmus eine erfüllende Wahrheitsbelegung finden.
- Unser Algorithmus arbeitet in 2 Phasen:
  - In Phase 1 wird für eine bestimmte Menge von Variablen der Wahrheitswert zufällig ermittelt.
  - In Phase 2 wird für die restlichen Variablen der Wahrheitswert deterministisch ermittelt.
- Wir betrachten k im Folgenden als eine feste Konstante.

Phase 1:

G-Klauseln := $\emptyset$ (Menge der gefährlichen Klauseln)

G-Var := $\emptyset$ (Menge der gefährlichen Variablen)

Sei $\{x_1, \ldots, x_n\}$ die Menge aller Variablen

for i := 1 to n do
  if $x_i \notin$ G-Var then
    setze x_i mit Wahrscheinlichkeit $\frac{1}{2}$ auf 0 oder 1, und setze diesen Wahrheitswert in allen Klauseln für x_i ein.
    G-Klauseln := $\{C_j \mid C_j$ ist noch nicht erfüllt und für $k/2$ viele Literale von C_j ist der Wahrheitswert schon festgelegt$\}$
    G-Var := $\{x_j \mid x_j$ kommt in einer Klausel aus G-Klauseln (negiert oder unnegiert) vor und der Wahrheitswert von x_j ist noch nicht festgelegt$\}$
  endif
endfor.

- Nach Phase 1 hat eine Klausel C_j überlebt, falls sie noch nicht erfüllt ist.

Lemma 1 Prob[C_j überlebt] $\leq d \cdot 2^{-k/2}$, wobei $d := k \cdot 2^{k/50}$

Beweis: Es gilt:

C_j überlebt $\Leftrightarrow$

(1) C_j wird gefährlich (d.h. C_j wird in die Menge G-Klauseln der gefährlichen Klauseln aufgenommen) oder

(2) die Variablen aus C_j, deren

nicht Wahrheitswert nach Phase 1 noch festgelegt wurde, kommen in gefährlichen Klauseln vor.

Damit ergibt sich:

- C_j überlebt $\overset{(*)}{\Rightarrow}$
  Eine Klausel C_i, die mit C_j eine Variable gemeinsam hat, wird gefährlich.
- $\text{Prob}[C_i \text{ wird gefährlich}] \overset{(**)}{\leq} 2^{-k/2}$

Aus (*) und (**) folgt:

$\text{Prob}[C_j \text{ überlebt}] \leq d \cdot 2^{-k/2}$. □

Beachte: Überlebt eine Klausel C_i Phase 1, so ist für mindestens $k/2$ viele Literale von C_i nach Phase 1 der Wahrheitswert noch nicht festgelegt.

Lemma 2: Es gibt eine Wahrheitsbelegung für die Variablen, deren Wahrheitswert in Phase 1 nicht festgelegt wurde, welche die überlebenden Klauseln erfüllt.

Beweis: gleicher Beweis wie für Satz 1, wobei 2^{-k} durch $2^{-k/2}$ ersetzt werden muß. □

- Sei G der Abhängigkeitsgraph auf den Klauseln:
  - Knotenmenge $= \{C_1, \ldots, C_m\}$
  - Es gibt eine Kante zwischen C_i und C_j, falls C_i und C_j eine gemeinsame Variable haben.
- Sei G' der induzierte Teilgraph von G, der durch die überlebenden Klauseln induziert wird.

Lemma 3: Es gibt eine Konstante $\gamma \geq 0$ mit:

$\text{Prob}[$ Jede Zusammenhangskomponente von G' hat höchstens $\gamma \cdot \log(m)$ viele Knoten $] = 1 - o(1)$.

Beweis:

(1) Seien $C_1, \ldots, C_r$ Klauseln mit $\text{Distanz}_G(C_i, C_j) \geq 4$ für $i \neq j$, $i,j \in \{1, \ldots, r\}$. Dann gilt:

$$\text{Prob}\left[\bigcap_{i=1}^{r} C_i \text{ überlebt}\right] \leq (d \cdot 2^{-k/2})^r$$

Beweis von (1): Wenn C_i überlebt, muß es eine Klausel D_i geben mit:
- D_i wird gefährlich
- $\text{Distanz}_G(C_i, D_i) \leq 1$.

- Es gibt höchstens d^r viele Möglichkeiten, ein Tupel $(D_1, \ldots, D_r)$ mit $\text{Distanz}_G(C_i, D_i) \leq 1$ für alle $i \in \{1, \ldots, r\}$ auszuwählen.
- Für jedes solche Tupel $(D_1, \ldots, D_r)$ gilt:

$\text{Distanz}(D_i, D_j) \geq 2$ für $i \neq j$.

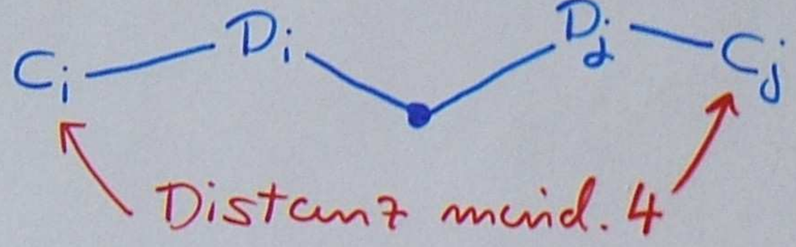
$$\Rightarrow \text{Prob}\left[\bigcap_{i=1}^{r} D_i \text{ wird gefährlich}\right] \leq (2^{-k/2})^r = 2^{-r \cdot k/2}$$

Damit ergibt sich:

$$\text{Prob}\left[\bigcap_{i=1}^{r} C_i \text{ überlebt}\right] \leq (d \cdot 2^{-k/2})^r$$

Dies beweist (1).

Veranschaulichung:



(2) Definiere Graph G_4 wie folgt:

- Knotenmenge = $\{C_1, \dots, C_m\}$
- Es gibt Kante zwischen C_i und C_j in $G_4 \Leftrightarrow \text{Distanz}_G(C_i, C_j) = 4$.

Bes: Es gibt höchstens $m \cdot d^{8r}$ viele Teilmengen $U \subseteq \{C_1, \dots, C_m\}$ mit $|U| = r$

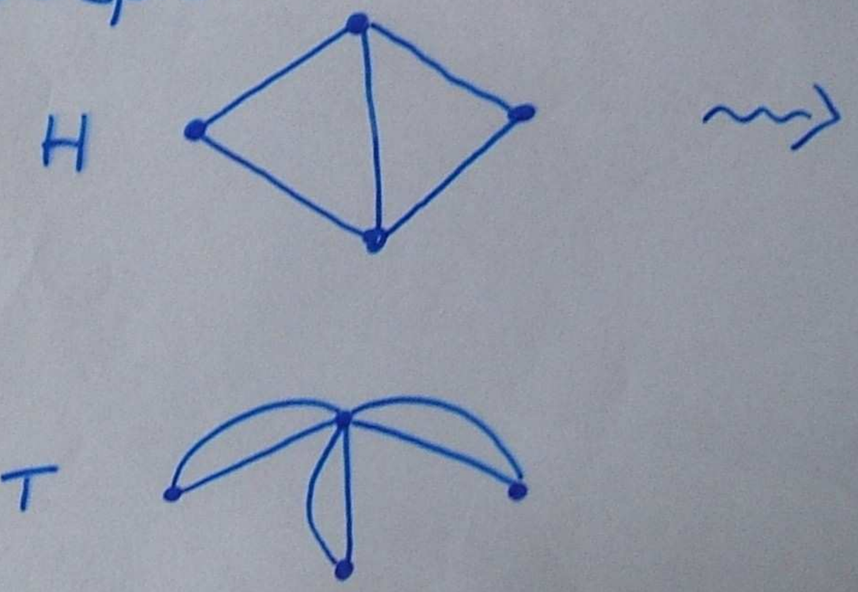
- U ist eine zusammenhängende Knotenmenge von G_4. (d.h. G_4 induziert auf U ist zusammenhängend)

Beweis von (2): Sei $U \subseteq \{C_1, \dots, C_m\}$ mit $|U| = r$ und U zusammenhängend in G_4.

- Beachte: Jeder Knoten in G_4 hat höchstens d^4 Nachbarn.
- Sei H der durch U induzierte Teilgraph von G_4.
- H zusammenhängend. $\rightarrow$

Wir können einen Spannbaum T für H auswählen.
Dupliziere in T jede Kante.

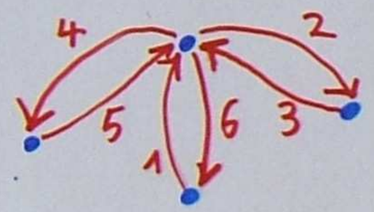
Beispiel:



Jeder Knoten von T ist mit einer geraden Anzahl von Kanten adjazent.

⇒ T besitzt einen Eulerkreis.
(einen Pfad, in dem jede Kante genau einmal durchlaufen wird.

Beispiel:



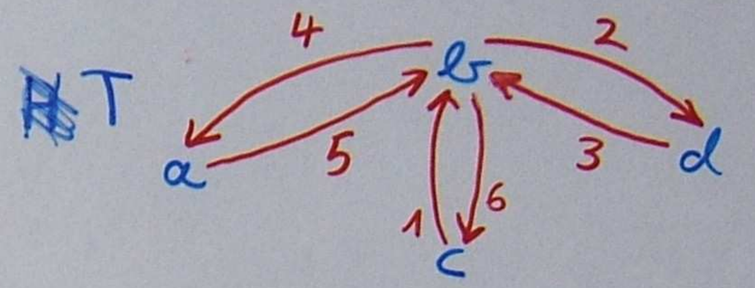
Beachte:

- Solch ein Eulerkreis legt die Knotenmenge U eindeutig fest und
- kann durch ein Tupel

$(v, a_1, \dots, a_{2(n-1)})$ beschrieben werden.

Startknoten — aus $\{1, \dots, d^4\}$: spezifiziert Nachbarknoten.

Beispiel:



$n = 4$

$U = \{a, b, c, d\}$

Sei a = 1. Nachbar von b
c = 2. Nachbar von b
d = 3. Nachbar von b

⇒ Obige Eulerpfad wird beschrieben durch $(c, 1, 3, 1, 1, 1, 2)$.

Es folgt:

Teilmengen $U \subseteq \{C_1, \dots, C_m\}$ mit
- $|U| = n$
- U zusammenhängend in G_4

$\leq m \cdot (d^4)^{2(n-1)} \leq m \cdot d^{8n}$ □

↑ # Folgen d. Länge $2(n-1)$ über $\{1, \dots, d^4\}$

Ein 4-Baum ist eine Teilmenge $T \subseteq \{c_1, \dots, c_m\}$ von Knoten von G mit:

- $\text{Distanz}_G(c_i, c_j) \geq 4$ für alle $c_i, c_j \in T$ mit $i \neq j$
- T bildet eine zusammenhängende Teilmenge von G_4

(3). Es gibt eine Konstante b mit:

$\text{Prob}[$ Es gibt einen 4-Baum T mit (i) $|T| \geq b \cdot \log(m)$ und (ii) alle Klauseln aus T überleben Phase 1 $] = \sigma(1)$

Beweis von (3):

Es gilt folgendes:

$\text{Prob}[$ Es gibt einen 4-Baum T mit: (i) $|T| \geq b \cdot \log(m)$ und (ii) alle Klauseln aus T überleben $]$

$(\leq)$ $=$ $\text{Prob}[$ Es gibt einen 4-Baum T mit: (i) $|T| = b \cdot \log(m)$ und (ii) alle Klauseln aus T überleben $]$

$$\leq \underbrace{(d \cdot 2^{-k/2})^{b \cdot \log(m)}}_{\text{aus (1) mit } r = b \cdot \log(m)} \cdot \underbrace{m \cdot d^{8 \cdot b \cdot \log(m)}}_{\text{aus (2) mit } r = b \cdot \log(m)}$$

$$= m \cdot (d^9 \cdot 2^{-k/2})^{b \cdot \log(m)} =$$

$$= m \cdot (k^9 \cdot 2^{9/50 \cdot k} \cdot 2^{-k/2})^{b \cdot \log(m)}$$

$$= m \cdot \underbrace{(k^9 \cdot 2^{-16/50 \cdot k})}_{\text{Konstante}}^{b \cdot \log(m)}$$

$= m \cdot \varepsilon^{\log(m)} = m \cdot m^{\log(\varepsilon)}$

für $\varepsilon = (k^9 \cdot 2^{-16/50 \cdot k})^{b}$

Beachte:

$b \geq 0$ nahe bei 0 $\rightarrow$

$\varepsilon \geq 0$ nahe bei 0 $\rightarrow$

$m \cdot m^{\log(\varepsilon)} = o(1)$. Dies beweist (3).

Wir beenden nun den Beweis von Lemma 3:

Sei H eine beliebige Zusammenhangskomponente von G'.

Sei T ein maximaler (bzgl. Inklusion) 4-Baum in H.

$\Rightarrow$ Für jeden Knoten C_i vom H gibt es einen Knoten $C_j \in T$ mit: $\text{Distanz}_H(C_i, C_j) \leq 3$.

(hätte C_i von allen Knoten $C_j \in T$ Distanz mindestens 4, so wäre T kein maximaler 4-Baum!)

$\rightarrow$ #Knoten von $H \leq 3 \cdot d^3 \cdot |T|$

Sei b die Konstante aus (3). $\rightarrow$

Prob [Es existiert eine Zusammenhangskomponente H von G' mit #Knoten von $H \geq \underbrace{3 \cdot b \cdot d^3}_{\text{Konstante}} \cdot \log(m)$]

$\leq$ Prob [Es existiert ein 4-Baum T mit $|T| \geq b \cdot \log(m)$ und alle Klauseln aus T überleben Phase 1] $= o(1)$ (nach (3))

□

• Wir können nun <u>Phase 2</u> beschreiben

- Wir müssen eine Wahrheitsbelegung für die Variablen, deren Wahrheitswert in Phase 1 nicht festgelegt wurde, finden, welche die überlebenden Klauseln erfüllt (existiert nach Lemma 2).

- Seien $H_1, \dots, H_\ell$ die Zusammenhangskomponenten von $G' =$ der von den überlebenden Klauseln induzierte Teilgraph des Abhängigkeitsgraph G.

↳ Mit Wahrscheinlichkeit $1 - o(1)$ gilt (Lemma 3)

(*) $\forall 1 \le i \le \ell$: H_i besteht aus höchstens $\gamma \cdot \log(m)$ Klauseln (Konstante)

Außerdem: $i \neq j \rightarrow H_i$ und H_j haben keine gemeinsame Variablen.

→ Es genügt, jede Klauselmenge H_i "einzeln zu erfüllen"

• Gelte nun (*). (falls (*) nicht nach Phase 1 gilt, wiederholen wir Phase 1 so oft, bis (*) gilt. Im Mittel wird dies ≤ 2 mal geschehen)

• Beachte: H_i enthält höchstens $\underbrace{k \cdot \gamma}_{\text{Konstante}} \cdot \log(m)$ viele noch nicht belegte Variablen.

```
for i := 1 to ℓ do
    überprüfe alle ≤ 2^(k·γ·log(m))
    = m^(k·γ) (ein Polynom in m) vielen
    Wahrheitsbelegungen für die noch
    unbelegten Variablen aus Hi, und
    teste so die Erfüllbarkeit von Hi
    finden so eine erfüllende Belegung
endfor
```