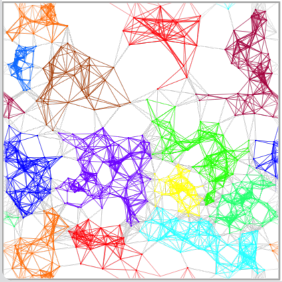
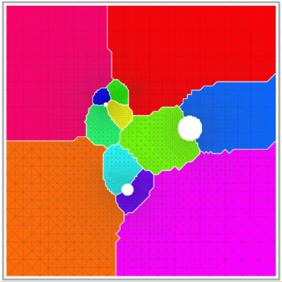
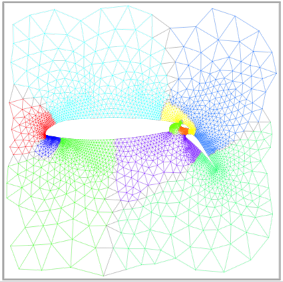
Algorithmische Methoden für schwere Optimierungsprobleme

Prof. Dr. Henning Meyerhenke

Institut für Theoretische Informatik

Vorlesung 6

Programm des Tages:

- Abschluss TSP
- (Max-)SAT

Inhalt

Das Erfüllbarkeitsproblem SAT
- Einführung und Komplexitätsergebnisse
- Randomisierte Algorithmen für Max-SAT

Einleitung

Eingabe: Boolesche Formel, z.B. in KNF

Ausgabe: Erfüllende Belegung oder Aussage, dass es keine gibt

Anwendungen:

- Hardware- und/oder Software-Verifikation
- Planen komplexer Projekte
- Automatische Schlussfolgerungen (*automatic reasoning*)
- Kombinatorischer Entwurf
- Lösen von Teilproblemen in vielen Domänen

Anwendungsszenario

Szenario:

- Sie sind Produktmanager bei BKW, den Badischen Kraftfahrzeugswerken
- Sie bestimmen, welche Konfigurationen von Autos gebaut werden können
- Beispiel: Automatische Scheinwerferreinigung nur in Verbindung mit Xenon-Scheinwerfern

Anwendungsszenario

Szenario:

- Sie sind Produktmanager bei BKW, den Badischen Kraftfahrzeugswerken
- Sie bestimmen, welche Konfigurationen von Autos gebaut werden können
- Beispiel: Automatische Scheinwerferreinigung nur in Verbindung mit Xenon-Scheinwerfern

Aufgabe:

- Kunden wählen Ihre Wunschkonfiguration aus einem “Katalog” (ggf. online)
- Sie assistieren bei der Erstellung einer Software, die prüft, ob die Konfiguration zulässig ist
- **Frage:** Wie modellieren Sie diese Konfigurationsprüfung? Geben Sie ein Beispiel an!

Komplexität

- SAT: Ist Boolesche Formel (in KNF) erfüllbar? $\mathcal{NP}$-vollständig!
- 3-SAT: Ist Boolesche Formel in KNF mit höchstens 3 Literalen pro Klausel erfüllbar? $\mathcal{NP}$-vollständig!
- 2-SAT: Ist Boolesche Formel in KNF mit höchstens 2 Literalen pro Klausel erfüllbar? 2-SAT $\in \mathcal{P}$

Komplexität

- SAT: Ist Boolesche Formel (in KNF) erfüllbar? $\mathcal{NP}$-vollständig!
- 3-SAT: Ist Boolesche Formel in KNF mit höchstens 3 Literalen pro Klausel erfüllbar? $\mathcal{NP}$-vollständig!
- 2-SAT: Ist Boolesche Formel in KNF mit höchstens 2 Literalen pro Klausel erfüllbar? 2-SAT $\in \mathcal{P}$

- Max-SAT: Bestimme maximale Anzahl erfüllbarer Klauseln einer Formel! $\mathcal{PP}$-vollständig!
- Maj-SAT: Ist Boolesche Formel von mehr als der Hälfte aller möglichen Belegungen erfüllbar? $\mathcal{PP}$-vollständig!

Klasse $\mathcal{PP}$

Klasse der Entscheidungen, die von einer probabilistischen Turingmaschine in Polynomialzeit lösbar ist (Fehlerwkt. $\leq 1/2$)

Weiterer Ablauf

- Problemstellung: Max-SAT
- Algorithmus 1 (Wiederholung aus Algo II):
  - randomisiert
  - relative Güte $3/2$
- Algorithmus 2:
  - Arithmetisierung als (I)LP
  - Randomisiertes Runden

Inhalt

Das Erfüllbarkeitsproblem SAT
- Einführung und Komplexitätsergebnisse
- Randomisierte Algorithmen für Max-SAT

Problemdefinition Max-SAT

Definition

- $V = x_1, \ldots, x_n$ Menge der Variablen
- Literal l: Variable $x_i \in V$ oder ihre Negation $\bar{x}_i$
- Klausel $C = l_1 \vee \cdots \vee l_k$: Oder-Verknüpfung von Literalen
- Boolesche (n, m)-Formel Φ: $C_1 \wedge \cdots \wedge C_m$ in konjunktiver Normalform

Problemdefinition Max-SAT

Definition

- $V = x_1, \ldots, x_n$ Menge der Variablen
- Literal l: Variable $x_i \in V$ oder ihre Negation $\bar{x}_i$
- Klausel $C = l_1 \vee \cdots \vee l_k$: Oder-Verknüpfung von Literalen
- Boolesche (n, m)-Formel Φ: $C_1 \wedge \cdots \wedge C_m$ in konjunktiver Normalform

Beispiel

$$\Phi(x_1, x_2, x_3) = \bar{x}_1 \wedge (x_1 \vee \bar{x}_2 \vee x_3) \wedge (x_1 \vee \bar{x}_3) \wedge (x_2 \vee x_3)$$

ist eine Boolesche $(3, 4)$-Formel in KNF.

Problemdefinition Max-SAT

Definition

- $V = x_1, \ldots, x_n$ Menge der Variablen
- Literal l: Variable $x_i \in V$ oder ihre Negation $\bar{x}_i$
- Klausel $C = l_1 \vee \cdots \vee l_k$: Oder-Verknüpfung von Literalen
- Boolesche (n, m)-Formel Φ: $C_1 \wedge \cdots \wedge C_m$ in konjunktiver Normalform

Definition (Max-SAT)

- Probleminstanz Φ: Boolesche (n, m)-Formel
- Zulässige Lösung: Belegung b der Variablen mit Booleschen Wahrheitswerten
- Zielfunktion: Maximiere $\mathit{wahr}(b, \Phi) = |\{j | C_j \in \Phi, b(C_j) = \mathit{true}\}|$
- **Aufgabe:** Finde b^* mit $\mathit{wahr}(b^*, \Phi)$ maximal!

Beobachtungen

- Das klassische Erfüllbarkeitsproblem SAT fragt, ob es eine Belegung b mit $wahr(b, \Phi) = m$ gibt
- **Annahme:** In einer Klausel kommen keine Variablen doppelt vor

Beobachtung

Sei Φ eine Boolesche (n, m)-Formel in KNF. Dann ist

$$\max\{wahr((false, \ldots, false), \Phi), wahr((true, \ldots, true), \Phi)\} \geq \frac{1}{2} \cdot m$$

Beweis.

Siehe Übung! □

Algorithmus Rand-Max-SAT

Vorgehen

Für jede Variable $x_i \in V$ setze $x_i := \mathit{true}$ mit Wkt. $1/2$.

Theorem

Für jede Instanz Φ von Max-SAT mit m Klauseln,
in der jede Klausel mindestens k Literale enthält,
erfüllt der erwartete Wert der Lösung von Rand-Max-SAT:

$$\mathbb{E}[\mathit{wahr}(b,\Phi)] \geq \left(1 - \frac{1}{2^k}\right) \cdot m$$

Algorithmus Rand-Max-SAT

Vorgehen

Für jede Variable $x_i \in V$ setze $x_i := \mathit{true}$ mit Wkt. $1/2$.

Theorem

Für jede Instanz Φ von Max-SAT mit m Klauseln, in der jede Klausel mindestens k Literale enthält, erfüllt der erwartete Wert der Lösung von Rand-Max-SAT:

$$\mathbb{E}[\mathit{wahr}(b, \Phi)] \geq \left(1 - \frac{1}{2^k}\right) \cdot m$$

Corollary

Für jede Instanz Φ mit m Klauseln, in der jede Klausel mindestens k Literale enthält, gibt es eine Belegung b mit $\mathit{wahr}(b, \Phi) \geq (1 - \frac{1}{2^k})m$.

Relative Güte von Rand-Max-SAT

Corollary

Algorithmus Rand-Max-SAT hat für jede Boolesche (n, m)-Formel Φ mit mindestens k Literalen pro Klausel eine erwartete relative Güte von

$$\frac{OPT(\Phi)}{\mathbb{E}[\text{Rand-Max-SAT}(\Phi)]} \leq \frac{m}{(1 - \frac{1}{2^k})m} = \frac{1}{1 - \frac{1}{2^k}}$$

Beobachtung

Gut bei großem k, "schlecht" bei kleinem k

Relative Güte von Rand-Max-SAT

Corollary

Algorithmus Rand-Max-SAT hat für jede Boolesche (n, m)-Formel Φ mit mindestens k Literalen pro Klausel eine erwartete relative Güte von

$$\frac{OPT(\Phi)}{\mathbb{E}[\textit{Rand-Max-SAT}(\Phi)]} \leq \frac{m}{(1 - \frac{1}{2^k})m} = \frac{1}{1 - \frac{1}{2^k}}$$

Beobachtung

Gut bei großem k, "schlecht" bei kleinem k

Theorem (Chen, Friesen, Zhang 99)

Algorithmus Rand-Max-SAT garantiert für jede Boolesche (n, m)-Formel Φ eine erwartete relative Güte von $3/2$.

Randomisiertes Runden

Ziel: Algorithmus, der besser wird, wenn die längste Klausel kürzer wird

Randomisiertes Runden

Ziel: Algorithmus, der besser wird, wenn die längste Klausel kürzer wird

Vorgehen:

1. Max-SAT als ganzzahliges lineares Programm (ILP) schreiben
2. ILP zu LP relaxieren
3. LP lösen
4. Ganzzahlige Lösung durch randomisiertes Runden herstellen