ETH zürich
Departement Informatik

Algorithmik für schwere Probleme

Prof. Dr. Dennis Komm
Dr. Hans-Joachim Böckenhauer
Dr. Richard Kralovič
`https://courses.algodi.inf.ethz.ch/schwereprob-2025`

Lösungsvorschläge – Blatt 2

Zürich, 4. März 2025

Lösung zu Aufgabe 2

Wir rekapitulieren zuerst kurz die Funktionsweise des unmodifizierten Algorithmus mit Laufzeit $\mathcal{O}(1.8393^n)$.

Die Eingabe ist eine Formel $\Phi \in$ 3KNF über n Variablen $\{x_1, \ldots, x_n\}$. Wie üblich gehen wir ohne Beschränkung der Allgemeinheit davon aus, dass jede Klausel jede Variable höchstens einmal enthält, egal ob negiert oder nicht. Der Algorithmus wählt eine beliebige Klausel C von Φ.

Nehmen wir zunächst an, dass die Klausel drei Literale enthält, also $C = (\ell_1 \vee \ell_2 \vee \ell_3)$.

Wir betrachten dann die folgenden drei partiellen Belegungen von Φ:

$$\alpha_1 \colon \ell_1 \mapsto 1, \qquad \alpha_2 \colon \begin{cases} \ell_1 \mapsto 0 \\ \ell_2 \mapsto 1 \end{cases} \qquad \text{und} \qquad \alpha_3 \colon \begin{cases} \ell_1 \mapsto 0 \\ \ell_2 \mapsto 0 \\ \ell_3 \mapsto 1 \end{cases}$$

Für jede partielle Belegung α_i bezeichnen wir mit $\Phi[\alpha_i]$ die Formel, die entsteht, wenn man aus Φ alle Klauseln löscht, in welchen α_i mindestens ein Literal auf 1 setzt, und in den verbleibenden Klauseln die Literale löscht, die α_i auf 0 setzt. Falls eine KNF-Formel ohne Klauseln entsteht, dann ist diese erfüllbar. Falls eine KNF-Formel mit einer leeren Klausel (also einer Klausel ohne Literale) entsteht, dann ist diese unerfüllbar.

In der Standardversion benützt man nun einfach die Tatsache

$$\Phi \text{ erfüllbar} \iff \Phi[\alpha_1] \text{ oder } \Phi[\alpha_2] \text{ oder } \Phi[\alpha_3] \text{ erfüllbar.} \tag{1}$$

Dies führt in dem Rekursionsschritt zu einer Verzweigung in drei Teilformeln, die jeweils mindestens 1, 2, und 3 Variablen weniger enthalten als Φ. Falls C nur zwei Literale enthält, kommt es analog zu einer Verzweigung in nur zwei Teilformeln mit mindestens 1 und 2 Variablen weniger; falls C nur 1 Literal enthält, wird die Berechnung auf eine einzige Teilformel mit mindestens einer Variablen weniger zurückgeführt.

(a) Die gewählte Klausel C einer 3KNF-Formel besteht aus höchstens 3 Literalen. In einer (partiellen) Belegung σ für diese Klausel gibt es die folgenden drei Möglichkeiten für jede Variable x in den höchstens 3 Literalen: (i) x wird durch σ nicht belegt, (ii) x wird durch σ mit 0 belegt, (iii) x wird durch σ mit 1 belegt. Es gibt also höchstens 3^3 (partielle) Belegungen für die gewählte Klausel C. Für jede solche Belegung σ kann man durch einfache Anwendung der Definition in Zeit $\mathcal{O}(k)$ prüfen, ob sie autark ist, wobei $k \in \mathcal{O}(n^3)$ die Anzahl der Klauseln bezeichnet. Falls es eine autarke Belegung σ für die Klausel C gibt, dann gilt

$$\Phi \text{ erfüllbar} \iff \Phi[\sigma] \text{ erfüllbar}\,. \tag{2}$$

Um (2) zu beweisen, zeigen wir zwei Implikationen. Falls $\Phi[\sigma]$ erfüllbar ist, dann ist offensichtlich auch Φ erfüllbar. Nehmen wir nun an, Φ sei erfüllbar. Dann gibt es eine Belegung α, die Φ erfüllt. Wir zeigen, dass die Belegung α auch die Formel $\Phi[\sigma]$ erfüllt. Da die Belegung σ autark ist, erfüllt sie alle Klauseln von Φ, die eine Variable von C enthalten. Somit besteht $\Phi[\sigma]$ aus einer Teilmenge unmodifizierter Klauseln von Φ, die dann jeweils durch α erfüllt sind, da sie auch in Φ vorkommen.

Falls es eine autarke Belegung σ gibt, dann berechnet der modifizierte Algorithmus die einzige Teilformel $\Phi[\sigma]$. Sonst berechnet der modifizierte Algorithmus dieselben Teilformeln $\Phi[\alpha_1]$, $\Phi[\alpha_2]$ und $\Phi[\alpha_3]$ wie der ursprüngliche Algorithmus.

Die Korrektheit des modifizierten Algorithmus folgt aus (1) und (2). Die Teilformeln $\Phi[\sigma]$ sowie $\Phi[\alpha_i]$ können in Zeit $\mathcal{O}(k)$ berechnet werden. Alle Teilformeln für einen Knoten des Suchbaums können somit in Zeit $\mathcal{O}(k) \subseteq \mathcal{O}(n^3)$ berechnet werden.

(b) Nehmen wir an, dass die Formel Φ aus $n \geq 4$ Variablen besteht. Falls es eine autarke Belegung σ für die gewählte Klausel C in der Formel Φ gibt, dann wird lediglich eine Teilformel $\Phi[\sigma]$ mit höchstens $n-1$ Variablen (da keine durch σ belegten Variablen in $\Phi[\sigma]$ vorkommen) berechnet. In diesem Fall gilt $T(n) \leq T(n-1)$ und $T'(n) \leq T(n-1)$.

Sonst werden bis zu drei Teilformeln $\Phi[\alpha_i]$, für $i \in \{1,2,3\}$, berechnet, wobei α_i genau i Variablen belegt. Nach Annahme ist keine der drei Belegungen α_i autark. Deshalb gibt es für jede Formel $\Phi[\alpha_i]$ eine Klausel C_i von Φ, in der mindestens eine Variable durch α_i belegt ist, die durch α_i jedoch nicht erfüllt wird. Falls alle Variablen in C_i durch α_i belegt sind, dann wird aus C_i eine leere Klausel in $\Phi[\alpha_i]$. Deshalb ist $\Phi[\alpha_i]$ in diesem Fall unerfüllbar und der Suchbaum wird in einem Blatt mit $\Phi[\alpha_i]$ abgebrochen. Sonst wird aus C_i eine Klausel mit höchstens 2 Literalen in $\Phi[\alpha_i]$ (da die durch α_i belegten Variablen gelöscht werden). Jede Teilformel $\Phi[\alpha_i]$ enthält somit eine Klausel mit höchstens 2 Literalen. Ausserdem enthält die Teilformel $\Phi[\alpha_i]$ höchstens $n-i$ Variablen, da keine durch α_i belegten Variablen in $\Phi[\alpha_i]$ vorkommen. Damit folgt, dass $T(n) \leq T'(n-1) + T'(n-2) + T'(n-3)$ gilt.

Falls die Formel Φ mindestens eine Klausel enthält, die aus höchstens zwei Literalen besteht, dann besteht die gewählte Klausel C ebenfalls aus höchstens zwei Literalen (da immer eine Klausel mit der kleinsten Anzahl der Literale gewählt wird). Somit werden nur bis zu zwei Teilformeln $\Phi[\alpha_i]$, $i \in \{1,2\}$, berechnet und mit $i \in \{1,2\}$ folgt nun, dass $T'(n) \leq T'(n-1) + T'(n-2)$ gilt.

Je nachdem, ob es eine autarke Belegung für die gewählte Klausel C gibt oder nicht, haben wir Abschätzungen $T(n) \leq T(n-1)$ und $T'(n) \leq T(n-1)$ bzw. $T(n) \leq T'(n-1) + T'(n-2) + T'(n-3)$ und $T'(n) \leq T'(n-1) + T'(n-2)$ hergeleitet. Daraus folgt

$$\begin{aligned} T(n) &\leq \max\{T(n-1),\ T'(n-1) + T'(n-2) + T'(n-3)\}\,, \\ T'(n) &\leq \max\{T(n-1),\ T'(n-1) + T'(n-2)\}\,. \end{aligned}$$

(c) Wir beweisen die Abschätzungen $T(n) \leq 3 \cdot \phi^n$ und $T'(n) \leq 2 \cdot \phi^n$ per Induktion.

Aus $T(n) = T'(n) = 1$ für $n \leq 3$ folgen sofort die Abschätzungen $T(n) \leq 3 \cdot \phi^n$ und $T'(n) \leq 2 \cdot \phi^n$ für $n \leq 3$.

Sei nun $n \geq 4$ beliebig. Als Induktionsannahme nehmen wir an, die Abschätzungen gelten für $n-1$, $n-2$ und $n-3$. Wenn wir diese Induktionsannahme in die Abschätzung aus Teilaufgabe (b) einsetzen, erhalten wir

$$\begin{array}{rcl} T(n) & \leq & \max\{3 \cdot \phi^{n-1},\ 2 \cdot \phi^{n-1} + 2 \cdot \phi^{n-2} + 2 \cdot \phi^{n-3}\}\,, \\ T'(n) & \leq & \max\{3 \cdot \phi^{n-1},\ 2 \cdot \phi^{n-1} + 2 \cdot \phi^{n-2}\}\,. \end{array}$$

Aus $\phi^2 = \phi + 1$ folgt, dass $\phi^n = \phi^{n-1} + \phi^{n-2}$ und $\phi^{n-1} = \phi^{n-2} + \phi^{n-3}$. Daraus folgt

$$\begin{array}{rcl} \max\{3 \cdot \phi^{n-1},\ 2 \cdot \phi^{n-1} + 2 \cdot \phi^{n-2} + 2 \cdot \phi^{n-3}\} & = & 4 \cdot \phi^{n-1}\,, \\ \max\{3 \cdot \phi^{n-1},\ 2 \cdot \phi^{n-1} + 2 \cdot \phi^{n-2}\} & = & \max\{3 \cdot \phi^{n-1},\ 2 \cdot \phi^n\}\,. \end{array}$$

Schliesslich überprüfen wir erfolgreich, dass

$$\begin{array}{rcl} 4 \cdot \phi^{n-1} & \leq & 3 \cdot \phi^n\,, \\ \max\{3 \cdot \phi^{n-1},\ 2 \cdot \phi^n\} & \leq & 2 \cdot \phi^n\,, \end{array}$$

da $\phi > \frac{3}{2} > \frac{4}{3}$.

(d) Nach Teilaufgabe (c) besteht der Suchbaum des modifizierten Algoritmus für eine Formel Φ mit n Variablen aus $T(n) \leq 3 \cdot \phi^n$ Blättern. Jedes Blatt hat höchstens n Vorgänger in dem Suchbaum, da jede Teilformel weniger Variablen als die ursprüngliche Formel hat. Somit gibt es insgesamt höchstens $T(n) \cdot (n+1)$ Knoten in dem Suchbaum. Da die Teilformeln für einen Knoten des Suchbaums in Zeit $\mathcal{O}(n^3)$ berechnet werden, läuft der modifizierte Algorithmus in Zeit $\mathcal{O}(T(n) \cdot n^4) \subseteq \mathcal{O}^*(T(n)) \subseteq \mathcal{O}^*(\phi^n)$.