



Technische
Universität
Braunschweig



Theoretische Informatik 2

Arne Schmidt

Kapitel 10 – Die Klasse P

Die Klasse P



Welche Probleme lassen
sich in polynomieller Zeit
mit einer DTM lösen?

Zunächst: Probleme in P heißen auch *effizient lösbare Probleme*.

In der Praxis kann das aber trotzdem lange dauern. Mit Laufzeit $O(n^k)$ steigt die Zeit um 2^k , wenn die Eingabegröße verdoppelt wird.

PRIMES

PRIMES

Gegeben: Natürliche Zahl n .

Frage: Ist n eine Primzahl?

Theorem 10.1 (Agrawal, Kayal, Saxena 2002)

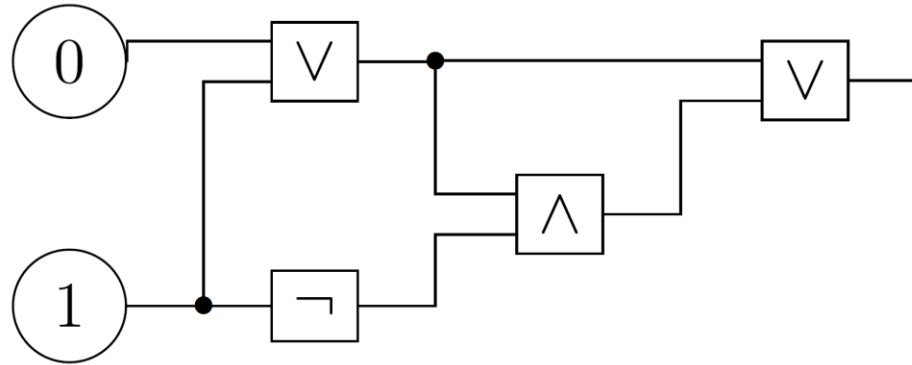
PRIMES ist in P.

Das Problem lässt sich in Zeit $O((\log n)^{6+\varepsilon})$ lösen, ist also in P.

Die Laufzeit ist allerdings bereits recht hoch für praktische Anwendungen.

Circuit Value Problem (CVP)

Ein Schaltkreis – Beispiel 10.2



Codierung C:

$$P_0 = 0$$

$$P_1 = 1$$

$$P_2 = P_1 \vee P_0$$

$$P_3 = \neg P_1$$

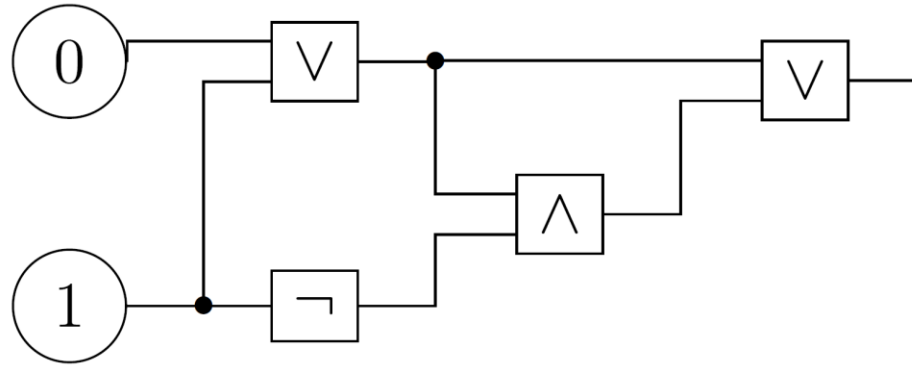
$$P_4 = P_3 \wedge P_2$$

$$P_5 = P_4 \vee P_2$$

Frage:

Welches Signal kommt am Ende heraus?

Boolescher Schaltkreis (Circuit)



Definition 10.3

Ein Boolescher Schaltkreis (Circuit) ist eine Sequenz von endlich viele Zuweisungen

$$P_0 = \dots, P_1 = \dots, P_n = \dots$$

wobei jede Zuweisung P_i mit $i \in \{0, \dots, n\}$ von einer der Formen ist:

$$P_i = 0 \mid P_i = 1 \mid P_i = \neg P_s \mid P_i = P_s \vee P_t \mid P_i = P_s \wedge P_t$$

mit $s, t < i$.

Definition 10.3

Ein Boolescher Schaltkreis (Circuit) ist eine Sequenz von endlich viele Zuweisungen

$$P_0 = \dots, P_1 = \dots, P_n = \dots$$

wobei jede Zuweisung P_i mit $i \in \{0, \dots, n\}$ von einer der Formen ist:

$$P_i = 0 \mid P_i = 1 \mid P_i = \neg P_s \mid P_i = P_s \vee P_t \mid P_i = P_s \wedge P_t$$

mit $s, t < i$.

CIRCUIT VALUE PROBLEM (CVP)

Gegeben: Ein Boolescher Schaltkreis C als Liste von Zuweisungen $P_0, \dots, P_n$.

Frage: Ist der Wert von P_n gleich 1?

Bemerkung 10.5:

Man kann jede Boolesche Formel in einen Booleschen Schaltkreis übertragen, allerdings ist das Auswerten von Booleschen Formeln in L, während CVP P-vollständig ist.

CVP ist P-vollständig

Theorem 10.6 (Ladner, 1975)

CVP ist P-vollständig (bzgl. Logspace-Reduktionen).

Lemma 10.7 „Membership“

CVP liegt in P.

CVP ist in P

Lemma 10.7 „Membership“

CVP liegt in P.

Beweis(skizze).

Annahme: Zuweisungen der CVP-Instanz sind durch # getrennt; die n Zuweisungen selbst sind binär codiert und geben somit eine Gesamtgröße von $O(n \log n)$.

Konstruiere eine Mehrband-Maschine:

- Band 1 enthält die Instanz
- Band 2 bewegt sich synchron zu Band 1 und speichert die Auswertungen.
- Band 3 speichert ab, welche Information wir holen sollen.

Wir definieren, wie bspw. für $P_i = \neg P_s$ verfahren wird.

CVP ist in P (II)

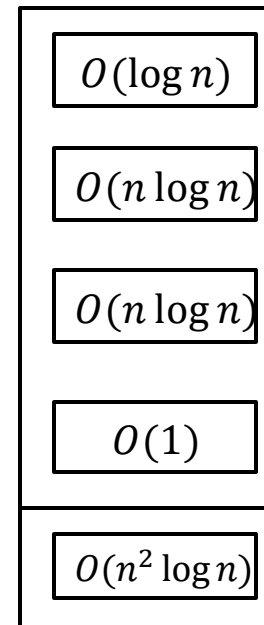
Lemma 10.7 „Membership“

CVP liegt in P.

Wir definieren, wie bspw. für $P_i = \neg P_s$ verfahren wird.

- Schreibe $\text{bin}(s)$ auf Band 3.
- Laufe auf Band 1 (und synchron Band 2) nach links, um Zuweisung für P_s zu finden und speichere die Negation im Kontrollzustand.
- Laufe auf Band 1 (und Band 2) wieder nach rechts bis P_i erreicht wird. (Zum Überlegen: Wie erkennen wir das?)
- Schreibe den im Kontrollzustand gespeicherten Wert an die entsprechende Stelle des zweiten Bandes.

Die Auswertung anderer Zuweisungen erfolgt ähnlich.



CVP ist P-vollständig

Lemma 10.8 „Hardness“

CVP ist P-schwer (bzgl. Logspace-Reduktionen).

Beweis:

Wir kennen kein P-schweres Problem, d.h. wir müssen **jedes** Problem aus P auf CVP reduzieren können.

Ab an die Tafel!