§4 Das PCP Theorem und seine Konsequenzen

- PCP = probabilistically checkable proofs
- PCP Theorem: eine neue Charakterisierung von NP
- zentrales Resultat der aktuellen TI (hat die Entwicklung der TI in den letzten 25 Jahren maßgeblich beeinflusst, z.B. die „unique games conjecture").
- wichtige Konsequenz des PCP-Theorems: Viele Optimierungsprobleme sind nicht nur NP-schwer exakt zu lösen, auch das Finden von guten, approximativen Lösungen ist oft NP-schwer.

Def. von PCP:

Haben gesehen, dass $NP \subseteq IP(1)$ gilt. Um eine Charakterisierung von NP zu finden, sollten wir den Verifizierer weiter einschränken:

1. Einschränkung: Der Verifizierer hat nur $r(n)$ Zufallsbits zur Verfügung.

2. Einschränkung: Der Verifizierer kann nur $q(n)$ Bits des Beweises sehen.

<u>Def.:</u> Seien $r: \mathbb{N} \to \mathbb{N}$, $q: \mathbb{N} \to \mathbb{N}$ Fkt.
Eine Sprache $L \subseteq \{0,1\}^*$ gehört zur Klasse $PCP(r(n), q(n))$, falls es eine probabilistische pzb. TM V gibt, die bei Eingabe von $x \in \{0,1\}^n$ $O(r(n))$ zufällige Bits verwenden darf und die höchstens $O(q(n))$ Bits eines Zertifikats $y \in \{0,1\}^*$ lesen darf, so dass für alle $x \in \{0,1\}^n$ gilt:

$$x \in L \Rightarrow \exists\, y \in \{0,1\}^* : P_{r_r}[V(x,y,r) = 1] = 1$$

$$x \notin L \Rightarrow \forall\, y \in \{0,1\}^* : P_{r_r}[V(x,y,r) = 0] \geq \frac{1}{2}.$$

<u>Klar:</u> (i) $PCP(0,0) = P$

(ii) $PCP(poly(n), 0) = coRP$

(iii) $PCP(0, poly(n)) = NP$,

wobei für Mengen $\mathcal{R}, \mathcal{Q}$ von Fkt. $f: \mathbb{N} \to \mathbb{N}$

$$PCP(\mathcal{R}, \mathcal{Q}) = \bigcup_{\substack{r \in \mathcal{R}, \\ q \in \mathcal{Q}}} PCP(r(n), q(n)).$$

<u>Bem.</u>: (zur Beweislänge von $y \in \{0,1\}^*$)

Falls V $O(r(n))$ zufällige Bits zur Verfügung hat, dann hat er Zugriff auf höchstens

$$p(n)\, 2^{O(r(n))} = 2^{O(r(n) + \log n)}, \quad \text{wobei } p \text{ ein Polynom}$$

Bitpositionen im Zertifikat y. D.h. wir können o.B.d.A. annehmen, dass $y \in \{0,1\}^{O(r(n) + \log n)}$ ist.

Insbesondere gilt $PCP(\log n, 1) \subseteq NP$,

außerdem $PCP(\log n, poly(n)) = NP$.

<u>Theorem</u> (PCP Theorem,
Arora, Lund, Motwani, Sudan, Szegedy, 1992)

$$NP = PCP(\log n, 1)$$

D.h. man kann die Korrektheit eines Zertifikats (mit großer W.keit) überprüfen, obwohl man es sich nur an konstant vielen Bits anschaut.

<u>Bem.</u>: Zur Zeit gibt es zwei Beweise des PCP Theorems:
Der ursprüngliche ist z.B. in der Dissertation von Arora vollständig zu finden (ca. 50 Seiten).
Dinur (2005) hat den ursprünglichen Beweis deutlich vereinfacht (ca. 25 Seiten).

Wir werden nur ein paar Ideen des Beweises uns ansehen.

<u>Doch vorher</u>: Konsequenzen des PCP Theorems für Optimierungsprobleme

Def.: (ε-ROBUST-3-SAT)

Sei $\varepsilon > 0$. Definiere die Sprache ε-ROBUST-3-SAT wie folgt: Als Eingabe sind alle 3-KNF-Formeln F möglich, die entweder erfüllbar sind oder bei der jede Belegung wenigstens einen ε-Anteil aller Klauseln nicht erfüllt. Formeln gehören zu ε-ROBUST-3-SAT, wenn sie erfüllbar sind.

[ε-ROBUST-3-SAT ist ein „Promise-Problem". Man bekommt das Versprechen, dass die Eingabe nicht aus nichterfüllbaren Formeln besteht, bei denen aber echt mehr als ein $(1-\varepsilon)$-Anteil der Klauseln erfüllt werden kann; D.h. anstatt $L \subseteq \Sigma^*$ betrachten wir $L \subseteq P \subseteq \Sigma^*$ für eine echte Teilmenge P. Falls wir L in polynomieller Zeit als Teilmenge von P erkennen können, dann können wir es auch als Teilmenge von Σ^*].

Satz $\exists \varepsilon > 0$: ε-ROBUST-3-SAT $\in$ NPC

$\iff PCP(\log n, 1) = NP$.