Seminar Mathematik & Informatik
P $\stackrel{?}{=}$ NP mit Orakeln

Benjamin Frowein

Wintersemester 2001/02

Zusammenfassung

Im ersten Teil untersuchen wir den Aufbau der Komplexitätsklasse NP zum einen unter der Voraussetzung, daß P=NP gilt, zum anderen, daß diese Klassen verschieden sind. Im zweiten Teil führen wir den Begriff der Orakelmaschine ein und definieren eine relativierte Fassung der Komplexitätsklassen. Wir werden zeigen, daß es Orakel gibt, die zu verschiedenen Antworten auf die (relativierte) Frage P=NP? führen.
Die Darstellung orientiert sich an Kapitel 14 in [1].

1 Die Struktur von NP

Lemma 1.1 *Sei* $L_2 \in \mathbf{P}$ *und* L_1 *reduzibel auf* L_2 *durch eine Polynomzeit-Reduktion. Dann gilt* $L_1 \in \mathbf{P}$.

Satz 1.2 *Sei* L *eine* **NP***-vollständige Sprache.*

(i) Ist $L \in \mathbf{P}$, *dann gilt* $\mathbf{P} = \mathbf{NP}$.

(ii) Ist $L \notin \mathbf{P}$, *dann gilt* $L' \notin \mathbf{P}$ *für alle* **NP***-vollständigen Sprachen* L'.

Satz 1.3 *Angenommen, es gilt* $\mathbf{P} = \mathbf{NP}$. *Dann ist jede Sprache* $L \in \mathbf{P} \setminus \{\emptyset, \Sigma^*\}$ *bereits* **NP***-vollständig.*

Da die Frage, ob $\mathbf{P} = \mathbf{NP}$ gilt, immer noch offen ist, sind zunächst drei verschiedene Möglichkeiten der Struktur von **NP** denkbar (vergleiche Abbildung 1.1):

(a) Es gibt eine Sprache in $\mathbf{NP} - \mathbf{P}$, welche nicht **NP**-vollständig ist.

(b) Jede Sprache liegt entweder in **P** oder ist **NP**-vollständig.

(c) $\mathbf{P} = \mathbf{NP}$ und damit ist jede Sprache in **P** **NP**-vollständig.

Wir werden zeigen, daß der Fall (b) nicht eintreten kann.

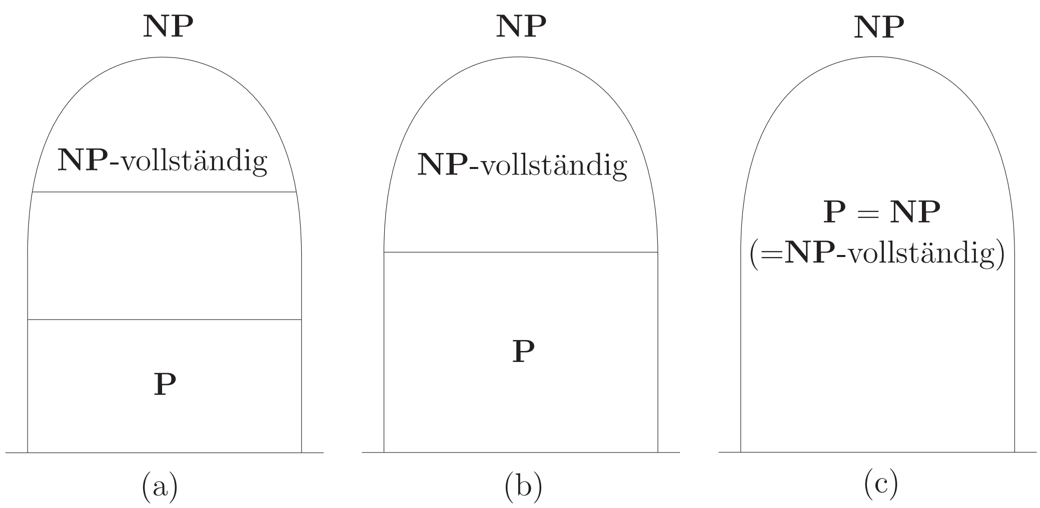

Abbildung 1.1: Die drei mögliche Strukturen für **NP**

Satz 1.4 *Angenommen* $\mathbf{P} \neq \mathbf{NP}$. *Dann gibt es eine Sprache in* **NP** *welche weder in* **P** *liegt noch* **NP**-*vollständig ist.*

2 $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ mit Orakeln

Wir wollen versuchen, alternative Universen zu konstruieren, in denen wir die $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ Frage lösen können. Tatsächlich konstruieren wir zwei Welten, in welchen diese Frage gegensätzliche Antworten besitzt.

Dafür definieren wir im nächsten Abschnitt zuerst, was wir unter Orakelmaschinen verstehen wollen:

2.1 Orakelmaschinen

Definition 2.1 *(Orakelmaschinen)*

Sei $A \subseteq \Sigma^*$ eine beliebige Sprache. Die Orakelmaschine M^A sei wie eine übliche Turingmaschine definiert. Zusätzlich besitzt M^A ein weiteres Band, das sogenannte Frage-Band und drei weitere Zustände $q_?$, q_{JA}, q_{NEIN}. A heißt Orakel der Maschine M^A. Von dem Frage-Zustand bewegt sich die Maschine M^A entweder nach q_{JA} oder q_{NEIN}, falls die aktuelle Zeichenkette auf dem Frage-Band in A enthalten ist, bzw. nicht. Die Antwortzustände erlauben der Maschine, diese Antwort in seiner weiteren Berechnung zu benutzen. Die Orakakelmaschine kann das Frage-Band beliebig beschreiben und beliebig oft in den Frage-Zustand $q_?$-Zustand wechseln.

Die Berechnung von M^A mit Eingabe x wird mit $M^A(x)$ bezeichnet.

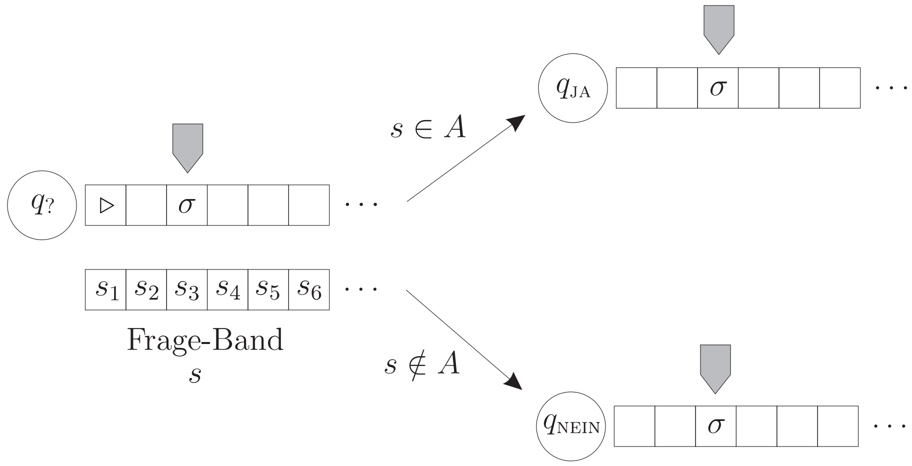

Abbildung 2.2: Orakelmaschinen

Definition 2.2 *(Zeitkomplexität für Orakelmaschinen)*

Die Zeitkomplexität für Orakelmaschinen wird genau so definiert wie für gewöhnliche Turingmaschinen. Jeder Frage-Schritt zählt wie ein gewöhnlicher Schritt. Auch nichtdeterministische Turing-Maschinen mit Orakel können auf dieselbe Weise definiert werden.

$\mathbf{TIME}^A(f(n))$ *sei die Klasse der Sprachen, die von Orakelmaschinen mit Orakel A mit Zeitschranke $f(n)$ entschieden werden. Alle anderen Zeitkomplexitätsklassen werden entsprechend relativiert:* $\mathbf{P}^A$, $\mathbf{NP}^A$, $\ldots$.

2.2 Relativierungen der $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ - Frage

Wir erinnern zunächst an ein folgendes Ergebnis, welches wir im weiteren Verlaufe benötigen:

Satz 2.3 *(Savitch's Theorem) ERREICHBARKEIT* $\in \mathbf{SPACE}(\log^2 n)$.

Korrolar 2.4 *Für jede geeignete Funktion* $f(n) \geq \log n$ *gilt* $\mathbf{NSPACE}(f(n)) \subseteq \mathbf{SPACE}(f^2(n))$. *Insbesondere gilt:* $\mathbf{PSPACE} = \mathbf{NPSPACE}$.

Satz 2.5 *Es existiert ein Orakel A, für welches* $\mathbf{P}^A = \mathbf{NP}^A$ *gilt.*

Satz 2.6 *Es existiert ein Orakel B, für welches* $\mathbf{P}^B \neq \mathbf{NP}^B$ *gilt.*

Wir erwähnen hier noch weitere Resultate ohne Beweis. Als Referenz sei [2] genannt.

Satz 2.7 *Es existiert ein Orakel C, so daß* $\mathbf{NP}^C$ *nicht unter Komplementbildung abgeschlossen ist.*

Lemma 2.8 *Falls* $\mathbf{NP} \neq co\mathbf{NP}$ *folgt* $\mathbf{P} \neq \mathbf{NP}$.

Satz 2.9 *Es existiert ein Orakel D, so daß* $\mathbf{P}^D \neq \mathbf{NP}^D$ *aber* $\mathbf{NP}^D$ *unter Komplementen abgeschlossen ist.*

Satz 2.10 *Es existiert ein Orakel E, so daß* $\mathbf{P}^E \neq \mathbf{NP}^E$ *aber* $\mathbf{P}^E = \mathbf{NP}^E \cap \mathbf{coNP}^E$.

Satz 2.11 *Es existiert ein Orakel F, so daß* $\mathbf{P}^F \subset \mathbf{NP}^F \cap \mathbf{coNP}^F \subset \mathbf{NP}^F$.

Veranschaulichungen der letzten sechs Sätze finden sich in Abbildung 2.3, von oben links reihenweise nach unten betrachtet, wieder.

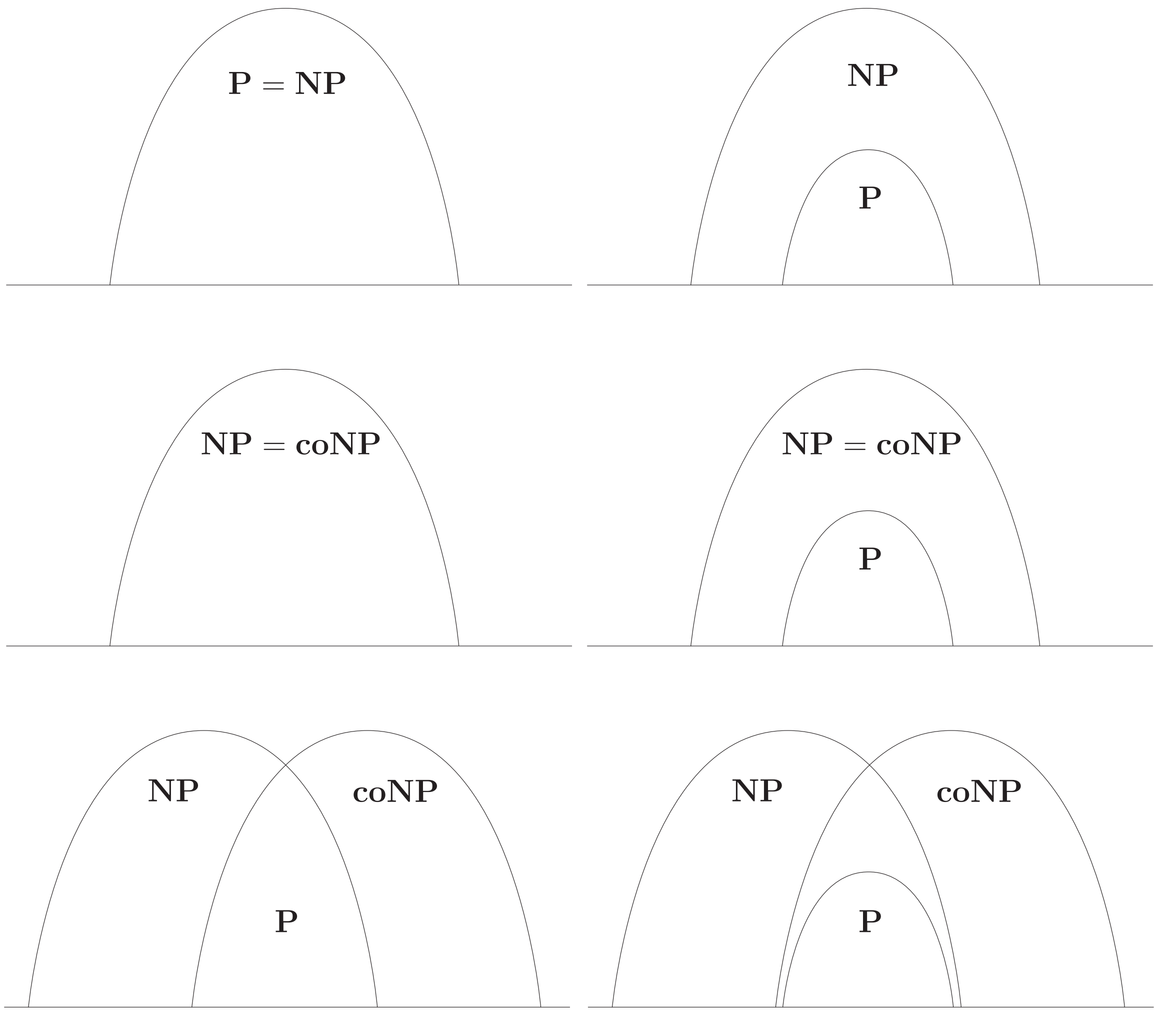

Abbildung 2.3: Alle konsistenten Inklusionen bezüglich **P**,**NP** und **coNP**.

Literatur

[1] Christos H. Papadimitriou, *Computational Complexity*, Addison-Wesley, 1995.

[2] T. Baker, J. Gill, R. Solovay, *Relativizations of the* $P \stackrel{?}{=} NP$ *question*, SIAM J. Computing 4, pp. 431-442, 1975.

[3] Ingo Wegener, *Theoretische Informatik - eine algorithmische Einführung*, Teubner, 1999.