Strukturelle Untersuchungen zu Quantenturingmaschinen

Diplomarbeit

Ruhr-Universität Bochum

Andreas Bomke

13. Juni 2000

Inhaltsverzeichnis

Kapitel 1

Einführung

1.1 Historische Bemerkungen

Den Startschuß zur Beschäftigung mit der Quantenphysik unter dem Gesichtspunkt, Berechnungen durchzuführen, gab Feynman in [Fey82]. Dort stellte er die Frage, ob es eine effizientere Simulation eines Quantenphysikalischen Systems geben kann, als die triviale Simulation in Exponentialzeit.

Umgekehrt stellte er auch die Frage, ob wir mit einem Rechnermodell, welches Quantenphysikalische Phänomene zugrunde legt, Probleme in Polynomialzeit lösen können, für die wir keinen Polynomialzeitalgorithmus für Probabilistische Turingmaschinen (PTMs) kennen.

Deutsch hat danach in [Deu85] das Modell der Quantenturingmaschine (QTM) eingeführt und seine Mächtigkeit untersucht. Es ähnelt dem Modell der PTM, jedoch sind zu den Alternativen komplexe "Amplituden" assoziiert. Außerdem befindet sich eine QTM solange in einer Überlagerung von Konfigurationen, einer sogenannten "Superposition", bis wir sie "beobachten". PTMs wählen im Gegensatz dazu in jedem Schritt nur eine Alternative zufällig aus.

Im Gegensatz zu PTMs nehmen wir bei QTMs nicht nur durch die Festlegung des "Programms" der QTM, sondern auch durch die Beobachtungen Einfluß auf die Berechnung.

Es gibt eine wichtige Einschränkung, was die möglichen Berechnungen einer QTM angeht, damit sie konsistent zu den Gesetzen der Quantenphysik sein kann. Dies ist die Längenerhaltung, oder, äquivalent dazu, die Unitarität des Operators, der die zeitliche Entwicklung der QTM auf dem komplexen Vektorraum der Superpositionen von Konfigurationen beschreibt. Daraus ergibt sich als notwendige Bedingung insbesondere die Reversibilität, denn jede unitäre Transformationen ist invertierbar.

Genau wie bei PTMs müssen wir bei QTMs fragen, ob es für eine endliche Berechnung genügt, die Übergangsamplituden mit endlicher Genauigkeit zu spezifizieren, oder ob es QTMs gibt, die sich schon bei geringfügiger Ungenauigkeit ganz anders verhalten. [BV97] konnten zeigen, daß man die QTM als diskretes Rechnermodell auffassen kann, bei dem kein Gebrauch von dieser beliebigen Genauigkeit gemacht wird.

Sie zeigten in dieser Arbeit außerdem die Existenz einer effizienten Universellen Quantenturingmaschine, und somit eines theoretisch konstruierbaren Allzweck-Quantencomputers. Die Konstruktion ist jedoch wesentlich komplizierter als für klassische DTMs. Wegen der notwendigen Reversibilität sind für QTMs sogar scheinbar einfache Konstrukte, wie eine Zählschleife oder eine Verzweigung, nicht trivial.

Analog zum klassischen Fall, wo wir BPP als die Klasse aller effizient berechenbaren Probleme ansehen, können wir die Komplexitätsklasse BQP für Quantenturingmaschinen definieren. [BV97] konnten zeigen:

$$P \subseteq BPP \subseteq BQP \subseteq PSPACE$$

Da $P \stackrel{?}{\neq} PSPACE$ ein wichtiges offenes Problem der Komplexitätstheorie ist, wäre ein Beweis von "$BPP \neq BQP$" ein gewichtiges Resultat.

Wegen $BQP \subseteq PSPACE$ kann eine QTM keine Funktionen berechnen, die eine PTM nicht berechnen kann.

Die moderne Form der Church-Turing These lautet, daß jedes "vernünftige" Rechnermodell *effizient* von einer PTM simuliert werden kann (das heißt die Verlangsamung ist durch ein Polynom beschränkt). Da sich "vernünftig" nicht formal definieren läßt, handelt es sich bei dieser These nicht um eine formal nachprüfbare Aussage. Aber wir interpretieren intuitiv den Begriff "vernünftig" als "im Prinzip physikalisch realisierbar". Es scheiden also "analoge" Rechnermodelle, wie zum Beispiel Maschinen, die beliebig genaue Zahlen in einem Schritt manipulieren können, aus. Für ein algorithmisches Problem gibt es demnach zwischen verschiedenen "vernünftigen" Maschinenmodellen höchstens polynomielle Unterschiede in der Zeitkomplexität.

Die PTM basiert jedoch auf dem Modell der klassischen Physik, wohingegen die moderne Physik von einer quantenphysikalischen Struktur des Universums ausgeht. Eine wichtige offene Frage ist, ob diese These auch gilt, wenn wir QTMs als "vernünftig" ansehen.

Des weiteren gibt es Untersuchungen dazu, ob das QTM-Modell sinnvoll ist, in dem Sinn, daß Standard-Variationen wie zum Beispiel das Zulassen mehrerer Bänder oder anderer Kopfbewegungen, polynomiell äquivalent sind. Yao konnte dies in [Yao93] bestätigen und zeigte auch die Äquivalenz zu einem anderen Quantenrechnermodell, den "Quantenschaltkreisen". Die Konstruktionen sind jedoch wesentlich komplizierter als im klassischen Fall.

Ein wesentliches Problem bei dem Versuch, QTMs physikalisch zu realisieren, ist der Einfluß der Umgebung. Eine Superposition wird nämlich nur so lange aufrecht erhalten, wir wir sie nicht anschauen.

Ein wichtiges Phänomen bei Quantenrechnern ist der sogenannte "Quanten-Parallelismus", den wir an folgendem Beispiel informell demonstrieren wollen: gegeben sei eine QTM M die eine Funktion $\{0, \ldots, 2^m - 1\} \to \{0, \ldots, 2^n - 1\}$ berechnet und dabei für jede Eingabe die gleiche Laufzeit L besitzt. Dann gibt es eine QTM, die zur Eingabe 1 zunächst in $\tilde{L}(m)$ Schritten eine gleichverteilte Überlagerung aller Eingaben generiert und dann M laufen läßt. Sie berechnet also alle Ergebnisse "gleichzeitig" in nur $\tilde{L}(m) + L$ Schritten und erzielt damit eine exponentielle Beschleunigung.

Das paradoxe daran ist, daß wir uns diese Überlagerung der Ergebnisse nicht ansehen sollten. Wenn wir dies tun, so erhalten wir ein zufällig gewähltes Ergebnis, und die Überlagerung kollabiert zu einer konventionellen Konfiguration. Wir haben dabei zudem keine Wahlmöglichkeit zwischen den einzelnen sich überlagernden Konfigurationen. Allerdings ist zu jeder Konfiguration eine Beobachtungswahrscheinlichkeit assoziiert, nämlich das Betragsquadrat ihrer komplexen Amplitude.

Eine Ziel bei der Programmierung ist es daher, die Amplitude des gesuchten Ergebnisses so zu verstärken, daß es die Superposition dominiert.

Adleman et al untersuchten in [ADH94], welchen Einfluß die Wahl des Grundbereichs der Übergangsamplituden auf die Mächtigkeit des Rechnermodells hat. Sie zeigten, daß eine endliche Menge an Übergangsamplituden ausreicht, um jede von einer QTM akzeptierbare Sprache zu akzeptieren.

Es gibt außerdem zahlreiche Untersuchungen zu sogenannten Orakel-QTMs, auf die wir aber nicht näher eingehen wollen.

Ein wichtiges Ergebnis, welches das Interesse an Quantencomputern beflügelt hat, war die Vorstellung effizienter Quantenalgorithmen zur Faktorisierung ganzer Zahlen und zur Berechnung des diskreten Logarithmus durch Shor in [Sho94]. Wenn es reale Quantencomputer gäbe, hätte dies also wichtige Konsequenzen für diejenigen kryptographischen Systeme, die auf der vermuteten Härte dieser Probleme beruhen.

1.2 Aufbau der Arbeit

In dieser Arbeit werden wir, unter anderem, einen Zeithierarchiesatz für ein noch einzuführendes Quantenrechnermodell beweisen.

Es gibt für das Modell der deterministischen Turingmaschine einen allgemeinen Hierarchiesatz, den wir zur Motivation zunächst zitieren wollen. Details sind in [Rei90] nachzulesen.

Eine Hierarchie von Komplexitätsklassen ist eine echt aufsteigende Inklusionskette von Komplexitätsklassen mit anwachsender Ressourcenschranke. Ein Hierarchiesatz beantwortet die Frage, wie weit man eine Ressorcenschranke höchstens vergrößern muß, um eine größere Komplexitätsklasse zu erhalten, das heißt, mehr Funktionen berechnen oder mehr Sprachen akzeptieren zu können.

Seien C, D beliebige Ressourcen (zum Beispiel DTime oder DSpace) und $T_1, T_2 : \mathbb{N} \to \mathbb{N}$. Ferner seien folgende drei Voraussetzungen erfüllt:

- $D(T_2)$ ist *universell* für $C(T_1)$
  (Das heißt, es existiert eine Universelle Maschine, die von ihrer Ressource D höchstens T_2 verbraucht, und jede Maschine, die von ihrer Ressource C höchstens T_1 verbraucht, simuliert.)

- D ist bezüglich C T_2*-kontrollierbar*
  (Das heißt, die Universelle Maschine mit Ressource D, stellt bei der Simulation von Maschinen mit Ressource C sicher, daß sie die Schranke T_2 nicht überschreitet.)[1]

- $D(T_2)$ ist *abgeschlossen gegenüber Komplementbildung*
  (Das heißt, $L \in D(T_2) \Rightarrow \overline{L} \in D(T_2)$)

Dann gilt

$$C(T_1) \subsetneqq D(T_2)$$

Die Beweisidee basiert auf einer Diagonalisierungstechnik, bei der die zu simulierenden Maschinen auf ihre eigene Codierung angesetzt werden.

Es genügt für diese Anwendung die Existenz einer Universellen Maschine, die nur in der Lage ist, Maschinen M, angesetzt auf deren eigene Codierung $x = G(M)$, zu simulieren.

[1] Im allgemeinen genügt hierzu die Konstruierbarkeit von T_2: Bei Platzschranken wird vorab der zulässige Bandbereich markiert, und bei Zeitschranken werden die Schritte mitgezählt.

Für den ausführlichen Beweis sei auf [Rei90] verwiesen. In Kapitel 5 werden wir jedoch die gleiche Beweisidee verwenden. Wie wir sehen werden, tauchen bei Quantenturingmaschinen (QTMs) Probleme auf, die den Nachweis der drei genannten Voraussetzungen verkomplizieren. Einerseits handelt es sich dabei um Schwierigkeiten, die aus der "Unhandlichkeit" des QTM-Modells resultieren, aber andererseits begegnen wir prinzipiellen Schwierigkeiten, die bereits beim Probabilistischen Turingmaschinen-Modell auftreten. Wir werden zu gegebener Zeit darauf zurück kommen.

Zwei konkrete Beispiele für Hierarchiesätze im klassischen Fall sind:
Für eine zeitkonstruierbare Funktion T_2 mit $T_2 = \omega(T_1 \cdot \log T_1)$ gilt:

$$DTime(T_1) \subsetneqq DTime(T_2)$$

Für eine platzkonstruierbare Funktion $S_2 \geq log$ mit $S_2 = \omega(S_1)$ gilt:

$$DSpace(S_1) \subsetneqq DSpace(S_2)$$

Diese Arbeit ist wie folgt gegliedert:

Zunächst definieren wir das QTM-Modell und gehen auf die besonderen Eigenarten, Schwierigkeiten und Unterschiede zum klassischen Fall ein (Kapitel 2).

Mit dem Nachweis der drei genannten - gegebenenfalls modifizierten - Voraussetzungen beschäftigen wir uns in den Kapiteln über den Komplementabschluß (Kapitel 3) und über die Universelle QTM (Kapitel 4).

Zum Schluß fügen wir dies alles zu einem Zeithierarchiesatz für QTMs zusammen (Kapitel 5).

Es sei schon einmal darauf hingewiesen, daß viele Sachverhalte, die bei deterministischen oder probabilistischen Turingmaschinen trivial und daher intuitiv benutzbar sind, bei QTMs - wenn es überhaupt bekannt ist - nur unter speziellen Voraussetzungen möglich sind. So muß zum Beispiel die Verkettung zweier zeitbeschränkter QTMs[2] nicht notwendig wieder zeitbeschränkt sein.

1.3 Verwendete Notation

Wir wollen zunächst einige grundlegende Begriffe und grundsätzliche Zusammenhänge einführen. Für eventuelle Beweise sei auf [BV97] verwiesen.

[2] Bei unserer - naheliegenden - Definition von Zeitbeschränktheit.

Im weiteren Verlauf dieser Arbeit verwenden wir zudem die übliche Terminologie und Standardzusammenhänge aus der Theorie der Deterministischen und Probabilisitschen Turingmaschinen und der Komplexitätsanalyse.

Definition 1.3.1. *Ein unitärer Vektorraum ist ein $\mathbb{C}$-Vektorraum $\mathcal{V}$ mit einem Skalarprodukt $(\cdot,\cdot) : \mathcal{V} \times \mathcal{V} \to \mathbb{C}$. Das Skalarprodukt induziert eine Norm auf $\mathcal{V}$ gemäß $||x|| := (x,x)^{1/2}$. Wir verwenden in dieser Arbeit stets das euklidische Skalarprodukt.*

Für den Rest dieses Abschnitts bezeichne $\mathcal{V}$ einen unitären Vektorraum.

Definition 1.3.2. *Sei U ein linearer Operator auf $\mathcal{V}$. Falls ein linearer Operator U^* auf $\mathcal{V}$ existiert, so daß für alle $x, y \in \mathcal{V}$ $(U^*x, y) = (x, Uy)$ gilt, dann heißt er zu U adjungiert. Wenn ein solcher Operator existiert, dann ist er einzigartig. Im endlichdimensionalen Fall existiert er stets und ist in Matrixdarstellung gegeben durch die komplex konjugiert transponierte Matrix.*

Ein linearer Operator U auf $\mathcal{V}$ heißt **unitär**, *genau dann wenn sein adjungierter Operator existiert und $U^*U = UU^* = I$ erfüllt.*

Korollar 1.3.3. (1) *Für eine Matrix $U \in \mathbb{C}^{d\times d}$ gilt: U ist unitär $\Leftrightarrow$ Die Zeilen von U sind paarweise orthonormal $\Leftrightarrow$ Die Spalten von U sind paarweise orthonormal*

(2) *Für einen linearen Operator U auf $\mathcal{V}$ gilt: U ist unitär, genau dann wenn U^* existiert und unitär ist.*

(3) *Für einen linearen Operator U auf $\mathcal{V}$ gilt: Falls U^* und $(U^*)^*$ existieren, dann ist $U = (U^*)^*$ wegen $(U^*)^* = (U^*)^* \cdot (U^* \cdot U) = ((U^*)^* \cdot U^*) \cdot U = U$.*

Definition 1.3.4. *Die Norm eines linearen Operators U auf $\mathcal{V}$ ist definiert als $||U|| := \sup\limits_{x\in\mathcal{V}:\,||x||=1} ||Ux||$. Analog definieren wir für endlichdimensionale Matrizen die Matrixnorm. U heißt beschränkt, genau dann wenn $||U|| < \infty$.*

Fakt 1.3.5. *Sei U ein unitärer Operator auf $\mathcal{V}$, dann gilt $||U|| = ||U^*|| = 1$.*

$U = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ hat die Norm 1, aber ist offensichtlich nicht unitär. Daher ist dies ein Beispiel dafür, daß die Rückrichtung nicht gilt.

Fakt 1.3.6. *Seien U_1, U_1', U_2, U_2' unitäre Transformationen auf $\mathcal{V}$, dann gilt*

$$||U_1'U_2' - U_1U_2|| \le ||U_1' - U_1|| + ||U_2' - U_2||$$

Das heißt, wenn wir die U_i' als Approximation der U_i auffassen, so verhält sich der Fehler bei der Multiplikation additiv.

Kapitel 2

Quantenturingmaschinen

2.1 Definition

Wir führen die QTM als Modifikation des Modells der probabilistischen Turingmaschine[1] (PTM) ein. Ein wesentlicher Unterschied wird sein, daß die zu den Konfigurationsübergängen gehörigen Parameter nicht mehr Übergangswahrscheinlichkeiten aus dem Intervall $[0, 1]$ sind, sondern komplexe Zahlen. Die Wahrscheinlichkeit, mit der eine Beobachtung eine bestimmte Konfiguration liefert, ist nicht einfach dieser Parameter, sondern das Quadrat seines Betrages. Wir werden daher lineare Superpositionen betrachten, deren euklidische Länge 1 ist, und müssen sicherstellen, daß diese Länge während der Rechnung erhalten bleibt.

Wir wollen zuerst die Definition einer deterministischen Turingmaschine (DTM) wiederholen, um die grundlegenden Begriffe und Konventionen einzuführen:

Definition 2.1.1 (DTM). *Eine deterministische Turingmaschine ist definiert durch ein Tripel $M = (\Sigma, Q, \delta)$ wobei gilt:*
$\Sigma = \Sigma_1 \times \ldots \times \Sigma_k$ mit $k \geq 1$ ist ein endliches Alphabet mit einem ausgezeichneten Blank-Symbol $\underbrace{\#; \ldots ; \#}_{k\ mal} \in \Sigma$. Wir sagen dann M hat k **Spuren**.
Q ist eine endliche Menge von Zuständen mit einem ausgezeichneten Startzustand q_0 und einem ausgezeichneten Endzustand $q_f \neq q_0$.
$\delta : Q \times \Sigma \to \Sigma \times Q \times \{L, R\}$ ist eine deterministische **Übergangsfunktion**.
Ferner besitzt M ein beidseitig unendliches Band aus Zellen, die mit $\mathbb{Z}$ indiziert sind, einen Schreib-/Lesekopf der sich entlang des Bandes bewegen

[1] Für eine kurze Einführung in probabilistische Maschinenmodelle siehe zum Beispiel lecture 7 in [Gol99].

kann, sowie eine endliche Kontrolle die sich stets in einem definierten Zustand befindet.
Eine **Konfiguration** *der Turingmaschine ist eine vollständige Beschreibung des Bandinhalts, insbesondere des aktuellen Bandsymbols unter dem Kopf, der Kopfposition und des aktuellen Zustands. Diese Beschreibung ist endlich, denn zu jedem Zeitpunkt enthalten nur endlich viele Zellen einen von Blank verschiedenen Inhalt, wie sich aus der Definition ergeben wird.*
Für eine Konfiguration c der DTM ist die **Nachfolgekonfiguration** *c' definiert durch die Anwendung der Übergangsfunktion auf den aktuellen Zustand und das aktuelle Bandsymbol unter dem Kopf in der offensichtlichen Weise.*
Die **Startkonfiguration** *von M ist dadurch charakterisiert, daß sich der Kopf auf Zelle 0, genannt* **Startzelle**, *befindet und der aktuelle Zustand q_0 ist. Wir sagen die Startkonfiguration hat* **Eingabe** *$x \in (\Sigma_1 \setminus \#)^*$, falls auf der ersten Spur, beginnend mit der Startzelle, der String x steht, dieser von Blanks umgeben ist, und alle anderen Spuren nur Blanks enthalten.*
Die DTM **hält** *wenn sie in den Zustand q_f gelangt. Eine Konfiguration mit aktuellem Zustand q_f nennen wir* **Endkonfiguration**. *Die* **Laufzeit** *für Eingabe x sei die Anzahl der Übergänge, die M auf Eingabe x durchführt, bis sie hält (eventuell unendlich).*
Wenn M hält, dann ist ihre **Ausgabe** *der String aus Σ_k^*, der sich dann aus dem Bandinhalt der letzten Spur vom linkesten von Blank verschiedenen bis zum rechtesten von Blank verschiedenen Symbol ergibt, oder der leere String, falls die gesamte letzte Spur nur Blanks enthält.*
Eine DTM, die für jede Eingabe hält, berechnet auf diese Weise eine Funktion von $(\Sigma_1 \setminus \#)^*$ nach Σ_k^*.

Wir haben die Konvention getroffen, nur Eingaben aus $(\Sigma_1 \setminus \#)^*$ zu betrachten, damit die DTM Anfang und Ende des Eingabestrings eindeutig an den begrenzenden Blanks erkennen kann. Beachte, daß dies keine Einschränkung ist, da wir gegebenenfalls zum Alphabet $\Sigma' := \Sigma \cup \{\tilde{\#}\}$ mit ausgezeichnetem Blanksymbol $\tilde{\#}$ übergehen können, wenn $\# \in \Sigma$ ist und wir Eingaben aus Σ^* betrachten möchten.

Quantenturingmaschinen befinden sich zu jedem Zeitpunkt in einer komplexen Linearkombination von Konfigurationen. Eine Momentaufnahme einer QTM besteht also aus der Beschreibung aller Konfigurationen der aktuellen Superposition und ihrer zugehörigen Amplitude ungleich Null. Formal stellt sich die Situation wie folgt dar.

Definition 2.1.2 (QTM). *Sei $\tilde{\mathbb{C}}$ die Menge aller $\alpha \in \mathbb{C}$ für die eine DTM existiert, die Real- und Imaginärteil von α auf 2^{-n} genau in durch ein Polynom in n beschränkter Zeit berechnen kann. Eine Quantenturingmaschine*

(QTM) M ist definiert durch ein Tripel (Σ, Q, δ), wobei Σ und Q wie in Definition 2.1.1 sind, und

$$\delta : Q \times \Sigma \to \tilde{\mathbb{C}}^{\Sigma \times Q \times \{L,R\}}$$

eine sogenannte Quanten-Übergangsfunktion ist.
Wir verwenden die Notation $\delta(p, \sigma, \tau, q, d)$ für die **Amplitude** *$\delta(p, \sigma)(\tau, q, d)$. Band, Kopf, Konfiguration, Startkonfiguration und Endkonfiguration seien genauso definiert wie in unserer DTM-Definition 2.1.1.*
Sei $\mathcal{S}$ der unitäre Vektorraum der endlichen *$\mathbb{C}$-linearen Kombinationen von Konfigurationen von M mit der euklidischen Norm. Ein Element $\Phi \in \mathcal{S}$ heißt* **Superposition** *von M. Die QTM M definiert einen linearen Operator $\mathbf{U_M} : \mathcal{S} \to \mathcal{S}$, genannt Zeitentwicklungsoperator von M, wie folgt: Angenommen, M befindet sich in einer Superposition, die nur aus einer Konfiguration c mit Amplitude 1 besteht. c befinde sich im aktuellen Zustand p mit aktuellem Symbol σ. Dann geht M in einem Schritt in die Superposition $\Psi = \sum_{(\tau,q,d) \in \Sigma \times Q \times \{L,R\}} \delta(p, \sigma, \tau, q, d)\, c_{(\tau,q,d)}$ über. Dabei ist $c_{(\tau,q,d)}$ die aus dem entsprechenden Übergang resultierende Nachfolgekonfiguration von c. Wenn wir diese Abbildung auf ganz $\mathcal{S}$ $\mathbb{C}$-linear fortsetzen, erhalten wir U_M. Eine Anwendung dieses Operators auf die aktuelle Superposition nennen wir einen* **Schritt** *von M.*
Eine **Startsuperposition** *ist eine Superposition, die nur aus einer Startkonfiguration mit der Amplitude 1 besteht.*

Um die Übergangsamplituden möglichst wenig einzuschränken, wählen wir sie aus $\tilde{\mathbb{C}}$. Eine Einschränkung ist nötig, denn andernfalls (für $\mathbb{C}$) könnten wir mit ihnen schwer berechenbare, oder sogar unberechenbare Zusatzinformationen codieren. Dieses Phänomen tritt auch bei PTMs auf.

Die Konfigurationen von M bilden eine orthonormale Basis von $\mathcal{S}$ und wir können bezüglich dieser Standardbasis jede Superposition als Vektor mit komplexen Einträgen darstellen. U_M können wir uns denken als unendlich dimensionale "quadratische Matrix", denn die Menge der Konfigurationen ist abzählbar. Beachte daß die Superposition von M zu jedem Zeitpunkt eine endliche Linearkombination von Konfigurationen ist. Die Matrix zu U_M enthält in jeder Spalte nur endlich viele Einträge ungleich Null, denn die möglichen Übergänge sind lokal, ihre Anzahl ist durch $2 \cdot |Q| \cdot |\Sigma|$ beschränkt. Der "support" der Superpositionen kann jedoch exponentiell mit der Anzahl der Berechnungsschritte anwachsen.

Beachte, daß selbst für einen auf $n < \infty$ Zellen beschränkten Bandbereich mit einem zweielementigen Alphabet, die Anzahl der möglichen Bandinhalte bereits 2^n beträgt, und wir zur Beschreibung einer einzigen Bandinhalts-Superposition schon 2^n komplexe Zahlen benötigen.[2]

Definition 2.1.3. *Eine QTM M ist* **wohlgeformt** $:\Leftrightarrow$ *ihr Zeitentwickluns-operator U_M erhält die euklidische Länge.*

Wohlgeformtheit ist eine aus der Physik motivierte, notwendige Bedingung, die eine QTM erfüllen muß, damit ihre zeitliche Entwicklung konsistent zu den Gesetzen der Quantenphysik sein kann.

Diese Forderung schränkt unsere Möglichkeiten, QTMs zu "programmieren", erheblich ein, denn Längenerhaltung ist eine globale Eigenschaft. Das heißt, wir müssen bei jeder noch so kleinen Änderung der Übergangsfunktion einer wohlgeformten QTM die Auswirkung auf die Wohlgeformtheit beachten.

Definition 2.1.4. *Eine* **Beobachtung** *einer wohlgeformten QTM M in der Superposition $\Psi = \sum_i \alpha_i c_i$ liefert die Konfiguration c_i mit der Wahrscheinlichkeit $|\alpha_i|^2$ und die Superposition von M ändert sich dabei zu $\Psi' = c_i$.*
Eine **partielle Beobachtung** *ergibt analog einen spezifizierbaren Teil einer zufällig aus der Superposition ausgewählten Konfiguration, wobei die Superposition zur Linearkombination derjenigen Konfigurationen kollabiert, die mit dem Ergebnis der Beobachtung konsistent sind, und die euklidische Länge der resultierenden Superposition wird auf Einheitslänge hochskaliert.*

Formal läßt sich dies wie folgt definieren (siehe [Gru99]):
Zu einer Konfiguration c bezeichne $c(i)$ das i-te Bandsymbol. Für eine endliche Indexmenge $I \subset \mathbb{N}$ definiere folgende Äquivalenzrelation auf der Menge der Konfigurationen: $c_i \sim_I c_j :\Leftrightarrow \forall k \in I : c_i(k) = c_j(k)$. Bezeichne mit $[c]_I$ die Äquivalenzklasse zu c bezüglich $\sim_I$.

Die aktuelle Superposition sei $\Psi = \sum_i \alpha_i c_i \in \mathcal{S}$, dann liefert die partielle Beobachtung der Menge von Zellen, deren Index in I liegt, die Äquivalenzklasse $[c]_I$ mit der Wahrscheinlichkeit $Pr([c]_I) = \sum\limits_{c_i \in [c]_I} |\alpha_i|^2$, und die Superposition verändert sich dadurch zu $\frac{1}{\sqrt{\sum\limits_{c_j \in [c]_I} |\alpha_j|^2}} \sum\limits_{c_j \in [c]_I} \alpha_j c_j$.

Beispiel 2.1.5. *Betrachte für eine wohlgeformte QTM M die Beobachtung der Startzelle in der aktuellen Superposition $\Psi = \sum_i \alpha_i c_i$. Sei $\sigma \in \Sigma$ und C_σ die Menge der Konfigurationen von M mit einem σ in der Startzelle. Dann*

[2] Erstaunlicherweise motiviert die Quantenphysik die Annahme, daß es in der Natur einen Mechanismus gibt, der einen so gewaltigen Aufwand treibt, um sich an diese Superposition zu "erinnern" und sie sogar instantan zu aktualisieren. Diese Vorstellung ist gegenläufig zu unserer alltäglichen Vorstellung, die auf der klassischen Physik beruht (wie auch DTMs).

liefert die Beobachtung ein σ mit Wahrscheinlichkeit $Pr[\sigma] := \sum_{i:c_i \in C_\sigma} |\alpha_i|^2$ *und die resultierende Superposition ist* $\Psi' = \frac{1}{\sqrt{Pr[\sigma]}} \sum_{i:c_i \in C_\sigma} \alpha_i c_i$.

Die Wohlgeformtheitsbedingung sagt also nichts anderes, als daß zu jedem Zeitpunkt die Summe der Beobachtungswahrscheinlichkeiten aller in einer Superposition vertretenen Konfigurationen 1 sein muß. Beachte, daß wir aufgrund obiger Definition *nur bei wohlgeformten QTMs* von einer Beobachtung sprechen können.

Es ist nicht offensichtlich, welchen Einfluß auf die Mächtigkeit des Maschinenmodells das Zulassen von Beobachtungen während der Rechnung hat. Im Gegensatz zu DTMs können wir die Rechnung einer QTM außer durch die Wahl der Eingabe, prinzipiell auch durch Beobachtungen beeinflussen.

Ethan Bernstein hat in [Ber] gezeigt, daß wir uns ohne Beschränkung der Allgemeinheit zur Vereinfachung darauf beschränken können, nur *eine* Beobachtung zuzulassen.

Definition 2.1.6. *Seien M und M' zwei QTMs. Wir sagen M'* **simuliert** *M mit Verlangsamung f und Genauigkeit $\varepsilon > 0$, wenn für alle Eingaben x und Zeitpunkte $T > 0$ gilt:*

Sei $\mathcal{D}$ die Wahrscheinlichkeitsverteilung, so daß die Beobachtung von M auf Eingabe x nach T Schritten eine Probe aus $\mathcal{D}$ liefert. Sei $\mathcal{D}'$ die Wahrscheinlichkeitsverteilung, so daß die Beobachtung von M' auf Eingabe x nach $f(T)$ Schritten eine Probe aus $\mathcal{D}'$ liefert. Dann ist

$$|\mathcal{D} - \mathcal{D}'| := \frac{1}{2} \sum_c |\mathcal{D}(c) - \mathcal{D}'(c)| \leq \varepsilon$$

Die zugrundeliegende Vorstellung ist, daß wir zwei QTMs als in der Anwendung funktionell äquivalent ansehen, wenn die zu einer Eingabe generierten Wahrscheinlichkeitsverteilungen hinreichend nah zueinander sind.

Lemma 2.1.7. *Seien $\Phi, \Psi \in \mathcal{S}$ mit $||\Phi||, ||\Psi|| = 1$ und $||\Phi - \Psi|| \leq \varepsilon$. Seien $\mathcal{D}, \mathcal{D}'$ die Wahrscheinlichkeitsverteilungen die aus den Beobachtungen von Φ und Ψ resultieren, dann ist $|\mathcal{D} - \mathcal{D}'| \leq 4\varepsilon$.*

Beweis. [BV97] □

Dies erlaubt es uns, die Genauigkeit einer Simulation durch die Berechnung des euklidischen Abstands zwischen der gewünschten und der tatsächlichen Superposition zu bestimmen.

Eine äquivalente Sichtweise des Begriffs der Wohlgeformtheit ergibt sich aus folgendem Resultat:

Theorem 2.1.8. *Für eine QTM M gilt: M ist wohlgeformt $\Leftrightarrow$ U_M ist unitär*

Beweis. siehe Anhang A in [BV97] □

Ein wesentlicher Unterschied der QTMs gegenüber Probabilistischen Turingmaschinen (PTMs) ist die Möglichkeit des Auftretens sogenannter destruktiver Interferenz. Die Rechnung einer QTM können wir uns genau wie die einer PTM als Menge sich überlagernder Berechnungspfade vorstellen. Bei einer PTM ergibt sich die Wahrscheinlichkeit, in T Schritten von einer bestimmten Konfiguration zu einer anderen zu gelangen, als die Summe über die Wahrscheinlichkeiten aller möglichen Berechnungspfade der Länge T zwischen diesen beiden Konfigurationen.[3] Bei QTMs gilt das gleiche für die komplexen Amplituden, was sich nicht notwendig auf die Beobachtungswahrscheinlichkeiten überträgt. Betrachte etwa folgendes Beispiel, welches auch den Einfluß der Beobachtungen verdeutlicht:

Beispiel 2.1.9 (Hadamard Transformation).
$U := \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$ *ist offensichtlich eine unitäre Transformation auf* $\mathbb{C}^2$.
Die Superposition $\Psi := \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1 \cdot e_1 + 0 \cdot e_2$ *wird durch eine Anwendung von* U *transformiert in die Superposition* $\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}} e_1 + \frac{1}{\sqrt{2}} e_2$, *und eine Beobachtung würde den ersten Basisvektor mit der gleichen Wahrscheinlichkeit liefern wie den zweiten, nämlich mit* $|\frac{1}{\sqrt{2}}|^2 = \frac{1}{2}$. *Unabhängig davon, wie die Beobachtung ausgegangen ist, sehen wir nach einer nochmaligen Anwendung von* U, *bei einer anschließenden Beobachtung, wiederum beide Basisvektoren mit der gleichen Wahrscheinlichkeit, da* $|\frac{1}{\sqrt{2}}|^2 = |-\frac{1}{\sqrt{2}}|^2 = \frac{1}{2}$.

Wenn wir jedoch zwischendurch nicht beobachten, sondern U *zweimal hintereinander auf* Ψ *anwenden, so erhalten wir wegen* $U^2 = I$ *wieder* Ψ, *und eine Beobachtung liefert den ersten Basisvektor mit der Wahrscheinlichkeit* 1 *und den zweiten mit der Wahrscheinlichkeit* 0. *Die beiden Berechnungspfade, die zum ersten Basisvektor führen, interferieren konstruktiv und verstärken sich zu einer Amplitude von* 1, *und die beiden Berechnungspfade die zum zweiten Basisvektor führen, interferieren destruktiv und löschen sich gegenseitig aus. Beachte auch, daß jeder der vier Berechnungspfade nur eine Wahrscheinlichkeit von* $\frac{1}{4}$ *hat (das Betragsquadrat des Produkts der zu den einzelnen Schritten entlang des Pfades gehörigen Amplituden).*

[3] Dabei ist die Wahrscheinlichkeit eines Berechnungspfades das Produkt der zu den einzelnen Schritten entlang des Pfades gehörigen Übergangswahrscheinlichkeiten

2.2 Erforderliche Präzision bei einer QTM

Der Wertebereich der Quanten-Übergangsfunktion enthält komplexe Amplituden. Falls die Mächtigkeit des Maschinenmodells entscheidend davon abhängen würde, daß man diese beliebige Genauigkeit ausnutzt, dann wäre unser QTM-Modell das Modell eines analogen Rechners. Wir können nicht erwarten, daß sich Amplituden mit einer besseren Genauigkeit als $\frac{1}{Poly(T)}$ physikalisch repräsentieren lassen, wenn T die Anzahl der auszuführenden Schritte ist. Dies entspricht gerade $O(\log T)$ Bits Genauigkeit je Amplitude, die für eine hinreichend genaue Berechnung ausreichen müssen, damit wir die QTM als Modell eines diskreten Rechners auffassen können.

Theorem 2.2.1. *Sei M eine QTM und $T > 0$. Falls $\Phi_0, \tilde{\Phi}_0, \ldots, \Phi_T, \tilde{\Phi}_T$ Superpositionen von M sind mit*

$$\begin{aligned} \forall 0 \leq i \leq T &\;:\; \|\Phi_i - \tilde{\Phi}_i\| \leq \varepsilon \\ \forall 0 < i \leq T &\;:\; \Phi_i = U_M \tilde{\Phi}_{i-1} \end{aligned}$$

dann gilt $\|\tilde{\Phi}_T - U_M^T \Phi_0\| \leq T\varepsilon$

Beweis. [BV97] □

Das heißt der Fehler ε, den wir in jedem Schritt machen, addiert sich in T Schritten lediglich zu $T\varepsilon$.

Definition 2.2.2. *Zwei QTMs M und M' sind ε-**nah**, wenn sie die gleiche Zustandsmenge sowie das gleiche Alphabet haben, und für jedes Paar korrespondierender Übergangsamplituden der Betrag ihrer Differenz kleiner oder gleich ε ist.*
Weder M noch M' müssen dabei wohlgeformt sein.

Theorem 2.2.3. *Zwei QTMs M, M' mit gleichem Alphabet Σ und gleicher Zustandsmenge Q seien ε-nah. Dann gilt:*

$$\|U_M - U_{M'}\| \leq 2 \cdot |\Sigma| \cdot |Q| \cdot \varepsilon$$

Beweis. [BV97] □

Korollar 2.2.4. *Sei $M = (\Sigma, Q, \delta)$ eine wohlgeformte QTM und M' eine QTM die $\frac{\varepsilon}{24 \cdot |\Sigma| \cdot |Q| \cdot T}$-nah an M ist, für ein $\varepsilon > 0$. Dann gilt: M' simuliert M für T Schritte mit Fehler ε und Verlangsamung $f(t) = t$. M' muß dabei nicht wohlgeformt sein.*

Beweis. [BV97] □

Beachte, daß $O(\log T)$ Bits Genauigkeit je Amplitude genügen, um M' so anzugeben, daß es $\frac{\varepsilon}{24 \cdot |\Sigma| \cdot |Q| \cdot T}$-nah an M ist.

2.3 Ein- und Ausgabekonventionen bei QTMs

Wenn die Ergebnisse zweier Berechnungspfade interferieren sollen, müssen sie genau die gleiche Länge aufweisen. Die Amplituden zweier Konfigurationen am Ende solcher Berechnungspfade addieren sich nur dann, wenn die Konfigurationen identisch sind. Dies bedingt insbesondere gleiche Kopfposition und gleiche Ausrichtung des Bandinhalts. Um dies zu handhaben führen wir einige Konventionen und Begriffe ein.

Definition 2.3.1. *Eine QTM M* **hält** *auf Eingabe $x \in (\Sigma_1 \setminus \#)^*$ mit* **Laufzeit** *T, wenn die Superposition zur Zeit T nur Endkonfigurationen enthält und zu jeder Zeit kleiner T die Superposition keine Endkonfiguration enthält. Die Superposition zur Zeit T heißt dann* **Endsuperposition** *von M auf Eingabe x. M ist $T(n)$*-**zeitbeschränkt**, *wenn sie auf jeder Eingabe x mit durch $T(|x|)$ beschränkter Laufzeit hält. Eine* **Polynomialzeit-QTM** *ist eine wohlgeformte $T(n)$-zeitbeschränkte QTM, für ein Polynom $T(n)$.*

Diese Definition zusammen mit der Definition der Beobachtung erlaubt uns, folgende Konvention für die "reale" Ausführung einer wohlgeformten QTM M zu treffen:
Nach jeder Anwendung des Zeitentwicklungsoperators führen wir eine partielle Beobachtung des aktuellen Zustands durch, um festzustellen ob sich die QTM in einer Endsuperposition befindet oder nicht.[4] Dazu sei die Repräsentation des aktuellen Zustands so, daß sich q_f in einem ausgezeichneten "Bit" eindeutig von allen anderen Zuständen unterscheidet. Wenn die Beobachtung dieses "Bits" q_f ergibt, so stoppen wir die Ausführung der QTM, und andernfalls fahren wir mit der nächsten Anwendung von U_M fort. Wenn sich die QTM zu jedem Zeitpunkt entweder in einer Endsuperposition befindet, oder in einer Superposition, bei der keine Konfiguration im Endzustand ist, dann ändern diese Beobachtungen die jeweils aktuelle Superposition nicht.

Beachte, daß sich eine QTM die *nicht* hält, in Superpositionen befinden kann, in denen einige Konfigurationen Endkonfigurationen sind und andere nicht. Gemäß unserer Konvention, Halten festzustellen, ändert sich die Superposition in solchen Fällen über die Anwendung des Zeitentwicklungsoperators hinaus durch die Beobachtung. Zudem kennen wir keine Methode, um dies effizient festzustellen. Ein derartiges Verhalten ist unerwünscht und wir werden in der Regel nur zeitbeschränkte QTMs betrachten.

[4] Beachte, daß wir für die Laufzeit diese Beobachtungen nicht separat als Rechenschritte zählen, sondern nur die Anzahl der Anwendungen des Zeitentwicklungsoperators U_M.

Wir werden gelegentlich auch von "Halten" auf beliebigen "Eingabesuperpositionen" sprechen, allerdings wird eine zeitbeschränkte QTM im allgemeinen auf Überlagerungen von Startkonfigurationen als "Eingabe" nicht halten, denn die einzelnen sich überlagernden Berechnungspfade können zu verschiedenen Zeitpunkten die jeweilige Endsuperposition erreichen.

Für eine Eingabe, auf der eine QTM hält, können wir die Ausgabe als die Superposition des Bandinhalts nach dem Halten ansehen. Dabei kommt es im Gegensatz zu DTMs jedoch auch auf die Kopfposition und die Ausrichtung des Bandinhalts relativ zur Startzelle an. Daher werden wir oft QTMs betrachten, bei denen in jeder Konfiguration einer Endsuperposition das am weitesten links stehende von Blank verschiedene Symbol auf der Startzelle steht und der Kopf darauf zeigt.

Definition 2.3.2. *Eine QTM heißt* **wohlbenehmend***, wenn sie für jede Eingabe in einer Endsuperposition hält, in der bei jeder Konfiguration der Kopf auf derselben Zelle steht. Falls dies die Startzelle ist, so heißt die Maschine* **stationär***.*
Eine DTM heißt stationär, wenn sie für jede Eingabe mit dem Kopf auf der Startzelle hält.

Die folgende Eigenschaft steht im Zusammenhang zu der Idee, zwei QTMs hintereinander auszuführen, indem wir die von $q_f^{(1)}$ ausgehenden Übergänge zu $q_0^{(2)}$ führen lassen (siehe Lemma 2.5.6). Dabei wird uns Satz 2.7.2 garantieren, daß bei einer zusätzlich wohlgeformten QTM keine anderen Übergänge mit von Null verschiedener Amplitude zu q_0 führen, als die von q_f ausgehenden.

Definition 2.3.3. *Eine QTM (oder DTM)* $M = (\Sigma, Q, \delta)$ *ist in* **Normalform***, genau dann wenn für alle* $\sigma \in \Sigma$ *gilt:*

$$\begin{array}{rcl} \delta(q_f, \sigma, \sigma, q_0, R) & = & 1 \\ \forall (\tau, q, d) \in \Sigma \times Q \times \{L, R\} \setminus \{(\sigma, q_0, R)\} : \delta(q_f, \sigma, \tau, q, d) & = & 0 \end{array}$$

Definition 2.3.4. *Eine QTM* $M = (\Sigma, Q, \delta)$ *heißt* **unidirektional***, wenn jeder Zustand nur aus einer Richtung erreicht werden kann, das heißt*

$$\forall p_1, p_2, q \in Q,\ \sigma_1, \sigma_2, \tau_1, \tau_2 \in \Sigma,\ d_1, d_2 \in \{L, R\} :$$
$$\delta(p_1, \sigma_1, \tau_1, \mathbf{q}, d_1) \neq 0 \neq \delta(p_2, \sigma_2, \tau_2, \mathbf{q}, d_2) \quad \Rightarrow \quad d_1 = d_2$$

Wir werden später sehen, daß sich jede QTM effizient durch eine unidirektionale QTM simulieren läßt. Unidirektionalität wird sich als ein wichtiges Konzept bei der Konstruktion der Universellen QTM erweisen.

2.4 Reversible Berechnungen

Eine naheliegende Frage ist nun, welche Arten von Berechnungen man mit einer QTM durchführen kann. Wir betrachten dazu in diesem Abschnitt die deterministischen Berechnungen, wie sie von einer DTM durchgeführt werden können.

Definition 2.4.1. *Eine* **reversible Turingmaschine** *(rTM) ist eine DTM, bei der jede Konfiguration höchstens einen Vorgänger hat.*

Notwendige und Hinreichende Bedingungen hierfür sind:

Theorem 2.4.2. *Eine DTM $M = (\Sigma, Q, \delta)$ ist reversibel, genau dann wenn die beiden folgenden Bedingungen erfüllt sind:*

(1) *Jeder Zustand von M kann nur aus einer Richtung erreicht werden, das heißt, falls $\delta(p_1, \sigma_1) = (\tau_1, q, d_1)$ und $\delta(p_2, \sigma_2) = (\tau_2, q, d_2)$, dann ist $d_1 = d_2$.*

(2) *δ ist eine Bijektion auf $\Sigma \times Q$, wenn wir die Bewegungsrichtung ignorieren.*

Beweis. [BV97] □

Bennett hat in [Ben73] außerdem gezeigt, daß sich jede DTM durch eine reversible DTM effizient simulieren läßt.

Theorem 2.4.3. *Jede reversible Turingmaschine ist auch eine wohlgeformte QTM.*

Beweis. siehe Anhang B in [BV97]. Die Idee ist, zu zeigen, daß die abzählbar unendlich dimensionale Matrix M_U eine Permutationsmatrix ist. □

Damit ergibt sich das erfreuliche Resultat, daß sich jede effiziente deterministische Berechnung auch effizient auf einer QTM durchführen läßt.

Wir sind jedoch daran interessiert, die Möglichkeit der QTMs auszunutzen, auf Überlagerungen von Konfigurationen zu rechnen.

Und daher dürfen wir beliebige deterministische Berechnungen *nicht* einfach als Baustein für QTMs benutzen. Das Problem dabei ist, daß während der Rechnung einer QTM die aktuelle Superposition im allgemeinen nicht nur aus einer Konfiguration besteht. Wenn wir eine solche Superposition als "Eingabe" für eine andere QTM verwenden, so ist es möglich, daß die zu den verschiedenen Konfigurationen gehörigen Berechnungspfade unterschiedlich lang sind und die QTM nicht hält, obwohl sie laufzeitbeschränkt ist.

Das folgende Theorem zeigt, daß jede effiziente deterministische Berechnung, bei der die Länge der Ausgabe nur von der Länge der Eingabe, aber nicht von der Eingabe selbst abhängt, auch effizient von einer reversiblen DTM durchgeführt werden kann, deren Laufzeit nur von der Länge der Eingabe abhängt, und die darüber hinaus stationär und in Normalform ist.

Wenn wir also eine Superposition von Eingabestrings gleicher Länge (mit jeweils gleicher Kopfposition), die sich etwa als Ergebnis der Rechnung einer anderen QTM ergibt, als "Eingabe" für eine solche QTM verwenden, dann endet die Rechnung für jeden zu einem String korrespondierenden Pfad nach der gleichen Anzahl Schritte, und die resultierenden Strings interferieren.

Wenn eine DTM eine nicht-bijektive Funktion berechnet, dann kann es natürlich keine reversible DTM geben, die sie exakt simuliert. Es ist jedoch möglich, die Existenz zu beweisen, wenn wir diese Forderung dahingehend abschwächen, daß wir der reversiblen DTM zugestehen, zu einer Eingabe x die Ausgabe $x; f(x)$ zu berechnen.

Theorem 2.4.4 (Synchronisationstheorem).
Sei Σ ein endliches Alphabet, $f : \Sigma^* \to \Sigma^*$ eine von einer DTM in Polynomialzeit berechenbare Funktion und $|f(x)|$ hänge nur von $|x|$ ab. Dann existiert eine stationäre, reversible, Polynomialzeit-TM in Normalform, die zur Eingabe x die Ausgabe $x; f(x)$ berechnet und deren Laufzeit ausschließlich von der Eingabelänge $|x|$ abhängt.
Falls zusätzlich f^{-1} existiert und von einer DTM in Polynomialzeit berechnet werden kann, dann existiert sogar eine stationäre, reversible Polynomialzeit-TM in Normalform, die zur Eingabe x die Ausgabe $f(x)$ berechnet und deren Laufzeit nur von $|x|$ abhängt.

Beweis. siehe Anhang B in [BV97] □

2.5 Programmierprimitive

In diesem Abschnitt stellen wir einige Möglichkeiten dar, die bereits in [BV97] untersucht wurden, wie man eine QTM aus mehreren einfacheren QTMs zusammensetzen kann. Davon werden wir bei der Konstruktion von QTMs häufig Gebrauch machen. Die Verwendung elementarer Programmierprimitive wie Wiederholung, Verzweigung und Umkehrung einer reversiblen Berechnung ist bei DTMs sehr intuitiv möglich. Viele der Probleme, die uns dabei bei QTMs begegnen, träten jedoch bereits auf, wenn wir versuchen würden, eine reversible Turingmaschine aus anderen reversiblen Turingmaschinen zusammenzusetzen.

In diesem Fall genügt es jedoch, zunächst eine nicht notwendig reversible DTM zu konstruieren und dann auszunutzen, daß sich diese effizient durch eine reversible DTM simulieren läßt.

Wenn wir QTMs als Bausteine betrachten, so ergeben sich weitere Probleme aus dem uns schon aus dem vorherigen Abschnitt bekannten Wunsch nach Laufzeitgleichheit für verschiedene (sich überlagernde) Eingaben gleicher Länge. Darüber hinaus müssen wir auch die Kopfposition und die Ausrichtung des Bandinhalts in den sich überlagernden (End-)konfigurationen berücksichtigen.

Zunächst einige grundlegende Resultate mit offensichtlichen Beweisen:

Lemma 2.5.1. *Zu jeder QTM (oder rTM) $M = (\Sigma, Q, \delta)$ und jeder Menge Σ' existiert eine QTM (rTM) $M' = (\Sigma \times \Sigma', Q, \delta')$, die sich genau wie M verhält und ihre zweite Spur unverändert läßt. Wohlgeformtheit bleibt dabei erhalten.*

Lemma 2.5.2. *Zu jeder QTM (oder rTM) $M = (\Sigma_1 \times \ldots \times \Sigma_k, Q, \delta)$ und jeder Permutation $\pi : [1, k] \to [1, k]$ existiert eine QTM (rTM) $M' = (\Sigma_{\pi(1)} \times \ldots \times \Sigma_{\pi(k)}, Q, \delta')$, die sich genau wie M verhält, außer daß ihre Spuren gemäß π permutiert sind. Wohlgeformtheit bleibt dabei erhalten.*

Aufgrund dieser beiden Lemmata können wir mit der Reihenfolge der Spuren und der Anzahl etwaiger zusätzlicher Hilfsspuren flexibel umgehen. So können wir für Zwischenberechnungen jede Spur als Ein- oder Ausgabespur nutzen. Außerdem garantieren uns diese beiden Lemmata zu zwei beliebigen wohlgeformten QTMs die Existenz zweier entsprechender wohlgeformter QTMs mit gleichem Alphabet.

Lemma 2.5.3. *Sei M eine wohlgeformte QTM (rTM). Wenn wir die hereinführenden oder die herausführenden Übergänge zwischen einem Paar von Zuständen in M vertauschen, so erhalten wir wieder eine wohlgeformte QTM (rTM).*

Für Zustände p_1, p_2 läßt sich das Vertauschen von herausführenden Übergängen formal schreiben als $\delta'(p_1, \sigma) := \delta(p_2, \sigma)$, $\delta'(p_2, \sigma) := \delta(p_1, \sigma)$ und $\delta'(p, \sigma) := \delta(p, \sigma)$ für alle $p \neq p_1, p_2$. Das Vertauschen von hereinführenden Übergängen für Zustände q_1, q_2 lautet formal $\delta'(p, \sigma, \tau, q_1, d) := \delta(p, \sigma, \tau, q_2, d)$, $\delta'(p, \sigma, \tau, q_2, d) := \delta(p, \sigma, \tau, q_1, d)$ und $\delta'(p, \sigma, \tau, q, d) := \delta(p, \sigma, \tau, q, d)$ für alle $q \neq q_1, q_2$.

Lemma 2.5.4 (Vereinigung). *Seien $M_1 = (\Sigma, Q_1, \delta_1), M_2 = (\Sigma, Q_2, \delta_2)$ zwei wohlgeformte QTMs (rTMs) mit gleichem Alphabet und disjunkten Zustandsmengen, dann ist die Vereinigung der beiden Maschinen $M = (\Sigma, Q_1 \cup$*

$Q_2, \delta_1 \cup \delta_2$), wobei wir den neuen Startzustand beliebig aus den beiden Startzuständen wählen dürfen und der neue Endzustand dadurch festgelegt ist, ebenfalls eine wohlgeformte QTM (rTM).

Die letzten beiden Lemmata erlauben es, eine Maschine anstelle eines Zustands q in eine andere Maschine einzufügen. Die Idee ist dabei, die hereinführenden Übergänge von q und $q_{2,0}$ sowie die herausführenden Übergänge von q und $q_{2,f}$ zu vertauschen. Diese Konstruktion erhält nur dann Wohlgeformtheit, wenn die Maschinen in Normalform waren.

Lemma 2.5.5 (Einfügung). *Seien M_1 und M_2 wohlgeformte QTMs in Normalform (rTMs) mit gleichem Alphabet und disjunkten Zustandsmengen. Sei q ein Zustand von M_1, dann existiert eine wohlgeformte QTM M in Normalform, die sich genau wie M_1 verhält, außer daß sie jedes mal wenn M_1 in den Zustand q übergeht, stattdessen die Maschine M_2 laufen läßt.*

Beweis. [BV97] □

Im allgemeinen stören sich dabei jedoch die Berechnungen der beiden Maschinen. Wir werden diese Technik daher nur in zwei speziellen Szenarien anwenden: Zum einen bei der Verkettung, wo wir die zweite Maschine für den Endzustand einer wohlbenehmenden QTM einfügen und die Berechnung der ersten Maschine bereits abgeschlossen ist, bevor die zweite beginnt. Zum anderen wenn wir zwei Maschinen haben, die auf verschiedenen Spuren arbeiten, wo als einzige Störquelle die Kopfposition verbleibt.

Lemma 2.5.6 (Verkettung). *Seien M_1 und M_2 wohlgeformte, wohlbenehmende QTMs in Normalform (rTMs) mit gleichem Alphabet und disjunkten Zustandsmengen. Dann gibt es eine wohlgeformte QTM in Normalform (rTM), die die Berechnung von M_1 gefolgt von der Berechnung von M_2 durchführt.*

Beweis. [BV97] □

Beachte, daß die zweite QTM als Eingabe eine Superposition von Konfigurationen erhält, die sich im gleichen Zustand mit dem Kopf auf der selben Position befinden. Falls sie für verschiedene Bandinhalte in dieser Superposition unterschiedliche Laufzeiten besitzt, dann *muß die zusammengefügte QTM nicht halten*, selbst wenn die beiden ursprünglichen QTMs wohlbenehmend waren. Das heißt, die Verkettung zweier wohlbenehmender QTMs muß nicht wohlbenehmend sein.

Als nächstes betrachten wir die Konstruktion einer bedingten Verzweigung.

Lemma 2.5.7 (Verzweigung). *Seien M_1 und M_2 wohlgeformte, wohlbenehmende QTMs in Normalform (rTMs) mit gleichem Alphabet und disjunkten Zustandsmengen. Dann gibt es eine wohlgeformte, wohlbenehmende QTM M in Normalform (rTM), die sich wie folgt verhält: Wenn die zweite Spur leer ist, dann führt sie M_1 auf ihrer ersten Spur aus und läßt dabei die zweite Spur leer. Wenn die zweite Spur eine 1 in der Startzelle enthält und alle anderen Zellen der zweiten Spur Blanks enthalten, dann führt sie M_2 auf ihrer ersten Spur aus und hinterläßt die 1 auf der zweiten Spur dort, wo der Kopf am Ende der Rechnung steht (alle anderen Zellen der zweiten Spur enthalten wieder Blanks). In jedem Fall benötigt M exakt 4 zusätzliche Rechenschritte gegenüber dem angewandten M_i.*

Die Beweisidee besteht darin, eine reversible DTM zu konstruieren, die die gewünschte Verzweigung und Zusammenführung leistet, und dann die beiden Maschinen M_1,M_2 anstelle entsprechender Zustände einzufügen. Die Zusammenführung ist erforderlich, da eine TM nur einen Endzustand besitzen kann. Damit dies reversibel geschieht, schreiben wir das Bit, das zur Entscheidung führte, welche Maschine ausgeführt wird, zurück auf das Band, wenn wir in den eindeutigen Endzustand übergehen.

Beweisdetails. siehe [BV97] □

Damit zwei Berechnungspfade einer QTM interferieren, müssen sie zur selben Zeit zwei identische Konfigurationen erreichen. Um diese Identität zu gewährleisten, ist es oft nötig, jegliche bei Zwischenberechnungen entstandenen Bandbeschriftungen wieder zu "löschen", denn Berechnungspfade mit gleichem Ergebnis, aber verschiedenen sonstigen Bandinschriften interferieren nicht miteinander. Dies erreichen wir durch Umkehrung einer Teilberechnung, denn "Löschen" ist keine unitäre Operation.

Da der Zeitentwicklungsoperator einer wohlgeformten QTM unitär ist, würde die Anwendung des adjungierten Operators die Berechnung wie gewünscht umkehren. Das Problem dabei ist, daß dieser inverse Operator nicht der Zeitentwicklungsoperator einer QTM sein kann, denn er repräsentiert Konfigurationsübergänge, bei denen Bandsymbole links oder rechts von der aktuellen Kopfposition manipuliert werden. Für den Fall einer unidirektionalen QTM gelingt die Konstruktion dennoch, denn da jeder Zustand nur aus einer Bewegungsrichtung erreicht werden kann, können wir die Umkehrung der Kopfbewegung einen Schritt im voraus gegenüber der Umkehrung der Bandinhalts- und Zustandsänderung vornehmen. Dadurch steht der Kopf zum richtigen Zeitpunkt wieder auf der zu ändernden Zelle.

Definition 2.5.8. *Seien M_1 und M_2 QTMs mit gleichem Alphabet. Dann sagen wir M_2* **kehrt die Berechnung von M_1 um**, *wenn unter Identifikation des Endzustands von M_1 mit dem Startzustand von M_2 und des Startzustands von M_1 mit dem Endzustand von M_2 folgendes gilt: Für jede Eingabe x auf der M_1 hält, bezeichne c_x die Startkonfiguration und ϕ_x die Endsuperposition von M_1 auf Eingabe x. Dann hält M_2 auf der Eingabesuperposition ϕ_x in einer Endsuperposition, die nur aus der Konfiguration c_x besteht.*

Lemma 2.5.9 (Umkehrung). *Für jede wohlgeformte, unidirektionale QTM in Normalform (rTM in Normalform) M existiert eine wohlgeformte QTM in Normalform (rTM in Normalform) M', die die Berechnung von M umkehrt und dabei zwei zusätzliche Rechenschritte benötigt.*

Beweisdetails. siehe [BV97] □

Als letztes betrachten wir die Konstruktion einer Zählschleife. Eine reversible Endlosschleife zu konstruieren ist trivial. Bei einer endlichen Anzahl an Wiederholungen müssen wir jedoch darauf achten, daß Ein- und Ausgang reversibel sind. Die Beweisidee besteht wiederum darin, zunächst eine reversible DTM zu konstruieren, die die gewünschte Wiederholung leistet, und dann die zu wiederholende QTM M anstelle eines besonderen Zustands einzufügen. Dabei tauchen folgende Probleme auf:

- Die reversible DTM hinterläßt das Zwischenergebnis ihrer Rechnung auf dem Band, während die zu wiederholende QTM läuft.
- Falls M nicht stationär ist, kann sie das korrekte Verhalten der reversiblen DTM stören.
- Selbst wenn M stationär ist, muß sie gar nicht unbedingt halten, da sie ab dem zweiten Aufruf auf einer Superposition von Eingaben unterschiedlicher Länge gestartet werden könnte.

Wir können daher keine allgemeingültigen Aussagen über das Verhalten der gesamten Maschine machen, wenn wir M nicht kennen. Wir müssen uns deshalb bei jeder konkreten Anwendung Gedanken zum Verhalten machen.

Lemma 2.5.10 (Looping Lemma). *Es existiert eine stationäre, reversible DTM M in Normalform und eine Konstante $c \in \mathbb{N}$ mit folgender Eigenschaft: Für jede binär codierte positive ganze Zahl $k > 0$ als Eingabe läuft M $O(k \log^c k)$ Schritte und hält mit unverändertem Bandinhalt. Außerdem besitzt M einen besonderen Zustand q^*, der exakt k mal besucht wird, und dabei befindet sich der Kopf jeweils auf der Startzelle.*

Beweis. siehe Anhang C in [BV97] □

2.6 Lokale Wohlgeformtheitsbedingungen

Bisher können wir die Wohlgeformtheit einer QTM nur anhand ihres Zeitentwicklungsoperators nachprüfen. Da dieser jedoch auf einem unendlich dimensionalen Vektorraum operiert, ist dies zu unhandlich, um eine QTM anhand ihrer Übergangsfunktion auf Wohlgeformtheit zu untersuchen. Die nun folgenden lokalen, notwendigen und hinreichenden Bedingungen werden für uns bei der Konstruktion einer geeigneten QTM in Kapitel 3 äußerst nützlich sein. Dabei handelt es sich um endlich viele Bedingungen an die Quantenübergangsfunktion. Sie ergeben sich - anschaulich gesprochen - im Wesentlichen daraus, daß die "Spalten" der unendlich dimensionalen Matrix U_M paarweise orthonormal sein müssen.

Theorem 2.6.1 (Lokale Wohlgeformtheitsbedingungen).
Eine QTM $M = (\Sigma, Q, \delta)$ ist wohlgeformt, genau dann wenn sämtliche der folgenden Bedingungen erfüllt sind:

(i) **Einheitslänge**

$$\forall (p, \sigma) \in Q \times \Sigma : \sum_{(\tau,q,d) \in \Sigma \times Q \times \{L,R\}} |\delta(p, \sigma, \tau, q, d)|^2 = 1$$

(ii) **Orthogonalität**

$$\forall (p_1, \sigma_1) \neq (p_2, \sigma_2) \in Q \times \Sigma :$$
$$\sum_{(\tau,q,d) \in \Sigma \times Q \times \{L,R\}} \delta^*(p_1, \sigma_1, \tau, q, d) \cdot \delta(p_2, \sigma_2, \tau, q, d) = 0$$

(iii) **Separabilität**

$$\forall (p_1, \sigma_1, \sigma_1'), (p_2, \sigma_2, \sigma_2') \in Q \times \Sigma \times \Sigma :$$
$$\sum_{p \in Q} \delta^*(p_1, \sigma_1, \sigma_1', p, R) \cdot \delta(p_2, \sigma_2, \sigma_2', p, L) = 0$$

Beweis. [BV97] □

Korollar und Definition 2.6.2 (Unidirektional Wohlgeformt).
Eine unidirektionale QTM $M = (\Sigma, Q, \delta)$ ist wohlgeformt, genau dann wenn die lineare Transformation U auf $\mathbb{C}^{Q \times \Sigma}$, die sich aus δ ergibt, wenn wir die Bewegungsrichtung ignorieren, unitär ist.

Zu einer unidirektionalen QTM M bezeichnen wir die Matrix mit $|\Sigma \times Q|$ Zeilen und ebenso vielen Spalten, deren Einträge die Amplituden von δ endlich codieren, als **Turingtafel**, *und den Vektor der Länge $|Q|$ mit binären Einträgen, welche die eindeutige Bewegunsrichtung codieren, mit der der jeweilige Zustand nur erreicht werden kann, als* **Richtungsstring**.

Beweis. [BV97] □

Diese Eigenschaft spielt bei der Konstruktion der Universellen QTM eine entscheidende Rolle. Beachte, daß U eine unitäre Transformation auf einem *endlichdimensionalen* unitären Vektorraum ist, wenn M wohlgeformt ist. Zusammen mit der Information, mit welcher eindeutigen Bewegungsrichtung die einzelnen Zustände erreicht werden, legt U die Übergangsfunktion δ vollständig fest.

Jede QTM mit partiell definierter Übergangsfunktion, die die euklidische Länge von Superpositionen aus ihrem Definitionsbereich erhält, läßt sich zu einer wohlgeformten QTM vervollständigen:

Definition 2.6.3. *Eine* **partielle QTM** *ist eine QTM $M = (\Sigma, Q, \delta)$ deren Übergangsfunktion δ nur auf einer Teilmenge $S \subseteq Q \times \Sigma$ definiert ist, das heißt für alle $(p, \sigma), (q, \tau) \in S$, $d \in \{L, R\}$ ist $\delta(p, \sigma, \tau, q, d)$ definiert und für alle $((p, \sigma), (q, \tau)) \notin S \times S$, $d \in \{L, R\}$ ist $\delta(p, \sigma, \tau, q, d)$ undefiniert.*
Wenn δ die drei lokalen Wohlgeformtheitsbedingungen aus Theorem 2.6.1 auf ihrem Definitionsbereich erfüllt, heißt M **wohlgeformte partielle QTM**.

Satz 2.6.4 (Vervollständigungssatz). *Sei $M = (\Sigma, Q, \delta)$ eine wohlgeformte partielle QTM. Dann existiert eine wohlgeformte QTM $M' = (\Sigma, Q, \delta')$ bei der δ' mit δ auf dem Definitionsbereich von δ übereinstimmt.*

Beweis. [BV97] □

Sei M eine auf S definierte, wohlgeformte partielle QTM. Dann kann die vervollständigte QTM M', ausgehend von einem Paar aus aktuellem Zustand und aktuellem Symbol unter dem Kopf, welches in S ist, keine anderen Nachfolgekonfigurationen erreichen als M. Formal:

Lemma 2.6.5. *Sei $M = (\Sigma, Q, \delta)$ eine auf $S \subseteq Q \times \Sigma$ definierte, wohlgeformte partielle QTM und $M' = (\Sigma, Q, \delta')$ die Vervollständigung gemäß Satz 2.6.4. Dann gilt für alle $(p, \sigma) \in S$, $(q, \tau) \in (Q \times \Sigma) \setminus S$, $d \in \{L, R\}$: $\delta'(p, \sigma, \tau, q, d) = 0$.*

Beweis. Sei $D := \{L, R\}$. M erfüllt auf ihrem Definitionsbereich die drei lokalen Wohlgeformtheitsbedingungen aus Theorem 2.6.1. Sei $(p, \sigma) \in S$ dann folgt aus der Einheitslängenbedingung $\sum_{(q,\tau,d) \in S \times D} |\delta(p, \sigma, \tau, q, d)|^2 = 1$

M' ist wohlgeformt und damit folgt aus der Einheitslängenbedingung für M':

$$\begin{aligned}
1 &= \sum_{(q,\tau,d)\in Q\times\Sigma\times D} |\delta'(p,\sigma,\tau,q,d)|^2 \\
&= \sum_{(q,\tau,d)\in S\times D} |\delta'(p,\sigma,\tau,q,d)|^2 + \sum_{(q,\tau,d)\in((Q\times\Sigma)\setminus S)\times D} |\delta'(p,\sigma,\tau,q,d)|^2 \\
&= \sum_{(q,\tau,d)\in S\times D} |\delta(p,\sigma,\tau,q,d)|^2 + \sum_{(q,\tau,d)\in((Q\times\Sigma)\setminus S)\times D} |\delta'(p,\sigma,\tau,q,d)|^2 \\
&= 1 + \sum_{(q,\tau,d)\in((Q\times\Sigma)\setminus S)\times D} |\delta'(p,\sigma,\tau,q,d)|^2 \\
\Rightarrow &\quad \forall (q,\tau) \in (Q\times\Sigma)\setminus S\,,\ d\in D : \delta'(p,\sigma,\tau,q,d) = 0
\end{aligned}$$

□

2.7 Wichtige Hilfsresultate

Lemma 2.7.1 (Unidirection Lemma).
Zu jeder QTM M existiert eine unidirektionale QTM M', die M mit einer Verlangsamung um den Faktor 5 fehlerfrei simuliert. Falls M wohlbenehmend und in Normalform, so auch M'.

Beweis. [BV97] □

Beachte daß eine entsprechende Konstruktion bei DTMs trivial ist: Für $M = (\Sigma, Q, \delta)$ sei $Q' := \{q' : q \in Q\}$ und $M' = (\Sigma, Q \cup Q', \delta')$ wobei δ' wie folgt definiert ist:

Für $(p,\sigma) \in (Q \cup Q') \times \Sigma$ sei $(\tau_{p,\sigma}, q_{p,\sigma}, d_{p,\sigma}) := \begin{cases} \delta(p,\sigma) & \text{falls } p \in Q \\ \delta(\tilde{p},\sigma) & \text{falls } p = \tilde{p}' \in Q' \end{cases}$

und $\delta'(p,\sigma) := \begin{cases} (\tau_{p,\sigma}, q_{p,\sigma}, d_{p,\sigma}) & \text{falls } d_{p,\sigma} = L \\ (\tau_{p,\sigma}, q'_{p,\sigma}, d_{p,\sigma}) & \text{falls } d_{p,\sigma} = R \end{cases}$

Dann ist M' unidirektional und simuliert M fehlerfrei mit Verlangsamung um den Faktor 1. M' ist jedoch nicht reversibel, da sie auf jedem Zustandspaar q, q' gleich operiert.

Der Zustand q_0 wird von einer wohlgeformten QTM in Normalform *nur* von q_f aus erreicht:

Satz 2.7.2 (Normalformsatz).
Ist $M = (\Sigma, Q, \delta)$ eine wohlgeformte QTM in Normalform, dann gilt

$$\forall p \in Q \setminus \{q_f\}\,,\ \sigma,\tau \in \Sigma\,,\ d \in \{L,R\}\ :\ \delta(p,\sigma,\tau,q_0,d) = 0$$

Beweis. Seien $\sigma, \tau \in \Sigma$ und $p \in Q \setminus \{q_f\}$ beliebig. Aus der Orthogonalitätsbedingung folgt

$$\begin{aligned} 0 &= \sum_{(\rho,q,d)\in\Sigma\times Q\times\{L,R\}} \delta^*(q_f,\tau,\rho,q,d)\cdot\delta(p,\sigma,\rho,q,d) \\ &= \delta^*(q_f,\tau,\tau,q_0,R)\cdot\delta(p,\sigma,\tau,q_0,R) \\ &= \delta(p,\sigma,\tau,q_0,R) \quad \text{da } M \text{ in Normalform} \end{aligned}$$

Aus der Separabilitätsbedingung folgt

$$\begin{aligned} 0 &= \sum_{q\in Q} \delta^*(q_f,\sigma,\sigma,q,R)\cdot\delta(p,\sigma,\tau,q,L) \\ &= \delta^*(q_f,\sigma,\sigma,q_0,R)\cdot\delta(p,\sigma,\tau,q_0,L) \\ &= \delta(p,\sigma,\tau,q_0,L) \quad \text{da } M \text{ in Normalform} \end{aligned}$$

□

Definition 2.7.3. *Sei M eine wohlgeformte QTM und $x \in (\Sigma_1 \setminus \#)^*$. Falls M auf x hält, so definiere $T^M(x)$ als die Laufzeit für Eingabe x. Jede Konfiguration aus der Superposition zum Zeitpunkt $T^M(x)$ befindet sich dabei also im Endzustand. Falls M auf x nicht hält so sei $T^M(x) := \infty$.*
Offensichtlich ist für alle $x \in (\Sigma_1 \setminus \#)^*$ $T^M(x) > 0$ da $q_0 \neq q_f$. Falls M $T(n)$-zeitbeschränkt ist, so gilt für alle $x \in (\Sigma_1 \setminus \#)^* : T^M(x) \leq T(|x|)$.

Lemma 2.7.4. *Sei M eine wohlgeformte QTM in Normalform und $x \in (\Sigma_1 \setminus \#)^*$ eine Eingabe auf der M hält. Dann gilt für alle $0 \leq t \leq T^M(x)$ und jede Konfiguration c_t aus der Superposition der Rechnung von M auf x zum Zeitpunkt t mit Amplitude ungleich Null:*
Der aktuelle Zustand in c_t ist q_0, genau dann wenn $t = 0$ ist, und er ist q_f, genau dann wenn $t = T^M(x)$ ist.

Beweis. Der aktuelle Zustand in c_t kann für $0 \leq t < T^M(x)$ nicht q_f sein, denn M hält auf x zur Zeit $T^M(x)$ und befindet sich zuvor nur in Superpositionen, bei der keine Konfiguration eine Endkonfiguration ist.

Damit folgt aus dem Normalformsatz 2.7.2, daß sich keine Konfiguration aus der Superposition zum Zeitpunkt $0 < t \leq T^M(x)$ im Zustand q_0 befindet, da er im ersten Berechnungsschritt verlassen wird und nur der Endzustand, der erst im Schritt $T^M(x)$ erreicht wird, zu ihm zurück führt. □

2.8 Die QTM als Sprachakzeptor

Im klassischen Fall wird **BPP** als die Klasse aller effizient berechenbaren Sprachen angesehen. Im folgenden definieren wir ihr Analogon im Quantenfall, BQP.

Definition 2.8.1. *Sei $M = (\Sigma, Q, \delta)$ eine zeitbeschränkte, stationäre Mehrspur-QTM in Normalform und $\Sigma = \Sigma_1 \times \ldots \times \Sigma_k$ mit $\Sigma_k = \{\#, 0, 1\}$. Wir lassen M auf Eingabe $x \in (\Sigma_1 \setminus \#)^*$ laufen und bezeichnen mit p die Wahrscheinlichkeit, mit der die Beobachtung der letzten Spur der Startzelle nach dem Halten eine 1 liefert. Dann sagen wir M* **akzeptiert** *x mit Wahrscheinlichkeit p und* **verwirft** *x mit Wahrscheinlichkeit $1 - p$.*

Betrachte eine beliebige Sprache $L \subseteq (\Sigma_1 \setminus \#)^*$. Wir sagen M akzeptiert L mit Wahrscheinlichkeit p, falls M jedes $x \in L$ mit einer Wahrscheinlichkeit von mindestens p akzeptiert und *jedes $x \in (\Sigma_1 \setminus \#)^* \setminus L$ mit einer Wahrscheinlichkeit von mindestens p verwirft.*

Für $p = 1$ sagen wir M akzeptiert L exakt.
Wir definieren die Sprachklassen error-free quantum (polynomial) time *und* bounded-error quantum (polynomial) time *wie folgt:*

$$\begin{aligned}
\mathbf{EQP} &:= \{L \subseteq (\Sigma_1 \setminus \#)^* \ : \ \exists \text{ Polynomialzeit-QTM obigen Typs, die } L \text{ exakt akzeptiert}\} \\
\mathbf{EQTime(T(n))} &:= \{L \subseteq (\Sigma_1 \setminus \#)^* \ : \ \exists\ T(n)\text{-laufzeitbeschränkte QTM obigen Typs, die } L \text{ exakt akzeptiert}\} \\
\mathbf{BQP} &:= \{L \subseteq (\Sigma_1 \setminus \#)^* \ : \ \exists \text{ Polynomialzeit-QTM obigen Typs, die } L \text{ mit Wahrscheinlichkeit } 2/3 \text{ akzeptiert}\} \\
\mathbf{BQTime(T(n))} &:= \{L \subseteq (\Sigma_1 \setminus \#)^* \ : \ \exists\ T(n)\text{-laufzeitbeschränkte QTM obigen Typs, die } L \text{ mit Wahrscheinlichkeit } 2/3 \text{ akzeptiert}\}
\end{aligned}$$

Beachte, daß das eigentliche Zufallsexperiment bei zeitbeschränkten QTMs, nach dieser Definition, die Beobachtung nach dem Halten ist. Im Gegensatz dazu wählen probabilistische Turingmaschinen in jedem Schritt zufällig eine der möglichen Nachfolgekonfigurationen aus und befinden sich somit nie in Überlagerungen von Konfigurationen.

Wir werden in Zukunft nicht jedes mal erwähnen, daß $\Sigma_k = \{\#, 0, 1\}$ ist, wenn wir von "akzeptieren" sprechen, sondern setzen dies in solchen Fällen implizit voraus.

Definition 2.8.2. *Seien $\widehat{\mathbf{BQP}}$ und $\widehat{\mathbf{BQTime}}(\mathbf{T(n)})$ die Sprachklassen, die genau wie BQP und $BQTime(T(n))$ definiert sind, außer daß die QTMs* nicht notwendig in Normalform *sein müssen. Offensichtlich gilt $BQP \subseteq \widehat{BQP}$ und $BQTime(T(n)) \subseteq \widehat{BQTime}(T(n))$.*

Lemma 2.8.3. *Sei* $\mathbf{BQTime^*(T(n))}$ *die Sprachklasse, die genau wie* $BQTime(T(n))$ *definiert ist, außer daß die QTMs* zusätzlich unidirektional *sein müssen. Dann folgt aus Lemma 2.7.1:*

$$BQTime^*(T(n)) \subseteq BQTime(T(n)) \subseteq BQTime^*(5 \cdot T(n))$$

Bemerkung 2.8.4 (Einschränkung des Grundbereichs).
Sprachen aus $\widehat{BQTime}(T(n)$ verlangen die Existenz einer QTM, die *alle* Eingaben aus $(\Sigma_1 \backslash \#)^*$ separiert in solche, die sie mit einer Wahrscheinlichkeit in $[0, \frac{1}{3}]$ akzeptiert, und solche, die sie mit einer Wahrscheinlichkeit in $[\frac{2}{3}, 1]$ akzeptiert.

Ein wesentlicher Bestandteil unseres Hierarchiesatzbeweises wird die Benutzung einer Universellen QTM sein. Wir müssen dazu eine QTM konstruieren, die als Eingabe x die Codierung einer QTM M_x erhält und daraufhin M_x hinreichend genau auf Eingabe x simuliert. Dabei garantieren wir, daß sie M_x nur für eine bestimmte Anzahl Schritte $(T'(n))$ simuliert und dabei insgesamt $T(n) = Poly(T'(n))$-zeitbeschränkt ist.

Unser Wunsch ist es, daß eine Sprache existiert, von der die Universelle Maschine bezeugt, daß sie in $\widehat{BQTime}(T(n))$ ist. Wir werden sehen, daß wir die Schranke $\frac{2}{3}$ durch eine beliebige Schranke aus dem offenen Intervall $(\frac{1}{2}, 1)$ ersetzen können, indem wir eine sogenannte Boosting-Technik anwenden. Dies ist auch nötig, denn die Universelle Maschine simuliert M_x nicht fehlerfrei.

Es tritt jedoch das Problem auf, daß einerseits "gute" Eingaben x, nämlich Codierungen von Maschinen M_x, die die Eingabe x mit einer Wahrscheinlichkeit in $[0, \frac{1}{3}] \cup [\frac{2}{3}, 1]$ akzeptieren, mit einem hinreichend kleinen Fehler ε simuliert werden sollen. Andererseits müssen aber auch "schlechte" Eingaben x, also insbesondere solche, für die die Akzeptanzwahrscheinlichkeit von M_x auf x nicht von $\frac{1}{2}$ weg beschränkt ist, mit einer Wahrscheinlichkeit in $[0, \frac{1}{3} + \varepsilon] \cup [\frac{2}{3} - \varepsilon, 1]$ akzeptiert werden. Formal können wir dies wie folgt ausdrücken:

Sei $p(x) := Pr[M_x$ akzeptiert x in höchstens $T'(|x|)$ Schritten$]$, für x aus einer noch zu präzisierenden Menge von Codierungen wohlgeformter, $T'(n)$-zeitbeschränkter QTMs. Dann muß die Universelle QTM $\tilde{M}$ für solche x insbesondere leisten:

$$\begin{aligned} h(x) &:= Pr[\tilde{M} \text{ akzeptiert } x \text{ in höchstens } T(|x|) \text{ Schritten}] \\ &\overset{!}{\in} \begin{cases} [\frac{2}{3} - \varepsilon, 1] & \text{falls } p(x) \geq 2/3 \\ [0, \frac{1}{3} + \varepsilon] & \text{falls } p(x) \leq 1/3 \\ [0, \frac{1}{3} + \varepsilon] \cup [\frac{2}{3} - \varepsilon, 1] & \text{sonst} \end{cases} \end{aligned}$$

Wenn wir der Universellen QTM exponentiell in der Eingabelänge viel Laufzeit zugestehen würden, könnte sie mit den Resultaten aus Abschnitt 2.4 mit einer deterministischen Rechnung $p(x)$ exakt bestimmen, indem sie sämtliche Berechnungspfade (also exponentiell viele) nacheinander simuliert. Dann könnte sie abhängig von $p(x)$ deterministisch entscheiden, ob eine 0 oder eine 1 auf die letzte Spur der Startzelle geschrieben werden muß.

Wir kennen jedoch keine Möglichkeit, $p(x)$ mit einer QTM in Polynomialzeit exakt zu bestimmen, und müssen uns daher mit der Berechnung einer Schätzung $\tilde{p}(x)$ zufrieden geben.

Sei $\tilde{h}(\alpha) := \min_{x:p(x)=\alpha} h(x)$, dann muß $\tilde{h}(\cdot)$ auf dem Intervall $[0,1]$ mindestens eine Unstetigkeitsstelle besitzen, wenn $h(\cdot)$ obige Bedingung erfüllt. Wenn wir bei der Konstruktion von $\tilde{M}$ nach dem Schema "akzeptiere x genau dann wenn $\tilde{p}(x) > c$", für eine Konstante c, vorgehen, so ist $\tilde{h}(\cdot)$ stetig.

Es ist ein offenes Problem, wie eine QTM für die Codierung einer QTM als Eingabe, in Polynomialzeit das oben genannte Akzeptanzverhalten realisieren soll. Dieses Problem ist allerdings nicht spezifisch für QTMs, sondern tritt bereits bei Probabilistischen Turingmaschinen auf.

Eine andere Schwierigkeit ergibt sich daraus, daß $\tilde{M}$ die Codierung einer nicht laufzeitbeschränkten QTM mit von $\frac{1}{2}$ weg beschränkter Wahrscheinlichkeit verwerfen muß. Sie müßte dazu nach jedem simulierten Schritt die Repräsentation der zu simulierenden Maschine daraufhin überprüfen ob entweder alle oder gar keiner der sich überlagernden Zustände der Endzustand ist. Es ist unklar, wie $\tilde{M}$ dies während der Simulation in Polynomialzeit mit hinreichender Sicherheit feststellen kann.

Aus diesen Gründen schränken wir den Grundbereich ein, auf dem wir eine von $\frac{1}{2}$ weg beschränkte Akzeptanzwahrscheinlichkeit fordern. Die folgende Definition formalisiert dies. In Kapitel 4 werden wir dann eine geeignete Wahl des Grundbereichs treffen, so daß die Universelle QTM auf diesem Grundbereich obige Bedingung erfüllt.

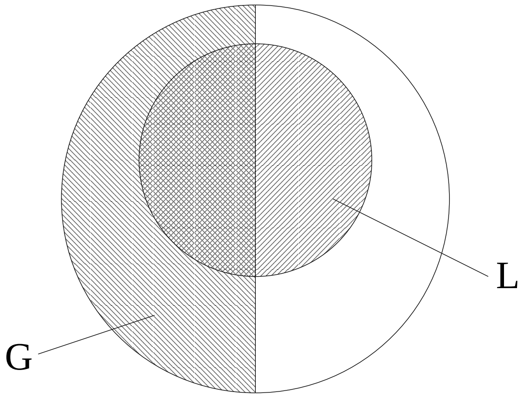

Abbildung 2.1: Einschränkung des Grundbereichs $(\Sigma_1 \setminus \#)^*$ auf G

Der äußere Kreis enthält alle Strings aus $(\Sigma_1 \setminus \#)^*$. L ist eine Sprache aus $(\Sigma_1 \setminus \#)^*$. Die Strings aus $(\Sigma_1 \setminus \#)^* \setminus G$ interessieren uns nicht. Die Strings aus G müssen jedoch entweder mit hinreichend großer $(G \cap L)$, oder mit hinreichend kleiner $(G \setminus L)$ Wahrscheinlichkeit akzeptiert werden.

Definition 2.8.5 (Spezielle Sprachklassen).
Sei $\Sigma := \Sigma_1 \times \ldots \times \Sigma_k$, $G \subseteq (\Sigma_1 \setminus \#)^*$, $T : \mathbb{N} \to \mathbb{N}$ *und* $\frac{1}{2} < \alpha \leq 1$, *dann definieren wir für* $L \subseteq (\Sigma_1 \setminus \#)^*$

$$L \in \mathbf{BQTime_{G,\alpha}(T(n))} :\Leftrightarrow$$

$\exists$ $T(n)$*-zeitbeschränkte, wohlgeformte, stationäre Mehrspur-QTM in Normalform* M*, so daß für alle* $x \in G$ *gilt*

(i) $x \in L \Rightarrow M$ *akzeptiert* x *mit Wahrscheinlichkeit* $\geq \alpha$

(ii) $x \notin L \Rightarrow M$ *akzeptiert* x *mit Wahrscheinlichkeit* $\leq 1 - \alpha$

Von einem solchen M *sagen wir, es* **bezeugt** $L \in BQTime_{G,\alpha}(T(n))$.

Die Klasse $\mathbf{BQTime^*_{G,\alpha}(T(n))}$ *sei wie* $BQTime_{G,\alpha}(T(n))$ *definiert, außer daß die QTMs* zusätzlich unidirektional *sein müssen.*
Die Klasse $\mathbf{\widehat{BQTime}_{G,\alpha}(T(n))}$ *sei wie* $BQTime_{G,\alpha}(T(n))$ *definiert, außer daß die QTMs* nicht notwendig in Normalform *sein müssen.*

Wenn der Parameter α *gleich* $\frac{2}{3}$ *ist, so notieren wir ihn nicht und schreiben zum Beispiel nur* $\mathbf{BQTime_G(T(n))}$ *statt* $BQTime_{G,\frac{2}{3}}(T(n))$.
Analog verwenden wir "P" statt "Time(T(n))" wie in Definition 2.8.1.

Beachte, daß wir für "$\Rightarrow$" in den beiden Fällen (i),(ii) genauso gut auch "$\Leftrightarrow$" schreiben könnten, da die jeweils zugeordneten Mengen eine Partition von G bilden.

Beachte auch, daß M auf jeder Eingabe $x \in (\Sigma_1 \setminus \#)^*$, also nicht nur für solche aus G, nach höchstens $T(|x|)$ Schritten halten muß.

Offensichtlich gilt für $G \subseteq (\Sigma_1 \setminus \#)^*$:
$BQTime(T(n)) \subseteq BQTime_G(T(n))$ und
$EQTime(T(n)) \subseteq BQTime_{G,1}(T(n))$.
Dabei gilt für $G = (\Sigma_1 \setminus \#)^*$ jeweils "$=$" statt "$\subseteq$".

Außerdem gilt für $G \subseteq (\Sigma_1 \setminus \#)^*$:
$L \in BQTime_{G,\alpha}(T(n)) \Leftrightarrow L \cap G \in BQTime_{G,\alpha}(T(n))$.

Intuitiv können wir uns den Unterschied zwischen dem Fall $G \subsetneqq (\Sigma_1 \setminus \#)^*$ und dem ursprünglichen Fall $G = (\Sigma_1 \setminus \#)^*$ so vorstellen, daß wir den QTMs für Eingaben aus $(\Sigma_1 \setminus \#)^* \setminus G$ eine beliebige Akzeptanzwahrscheinlichkeit zugestehen, also etwa auch eine Akzeptanzwahrscheinlichkeit von $\frac{1}{2}$, was einem zufälligen Münzwurf entspräche.

Bemerkung 2.8.6. *Die Normalform der bezeugenden QTMs (und auch ihre Stationärität), sowie die Wahl der Schranke $\frac{2}{3}$ sind insofern nicht wesentlich, als daß wir in Kapitel 4 mit Hilfe der Universellen QTM und Boosting zeigen werden, daß für alle $\frac{1}{2} < \alpha, \beta < 1$ und $G \subseteq (\Sigma_1 \setminus \#)^*$ gilt: $BQP_{G,\beta} = BQP_{G,\alpha} = \widehat{BQP_{G,\alpha}}$. Dabei könnten wir nicht nur auf die Normalformeigenschaft, sondern auch auf die Stationärität der bezeugenden QTMs verzichten.*

Auf der anderen Seite ist die Normalformeigenschaft bei der Konstruktion von QTMs mit Hilfe der Programmierprimitive aus Abschnitt 2.5 wesentlich. Die obige Gleichheit sagt gewissermaßen nur etwas über die Akzeptanzwahrscheinlichkeit und die Laufzeit. Darüber, welche Eigenschaften die Bandinhalte erfüllen, macht sie keine Aussage. Daher haben wir zum Beispiel bei einer Verkettung keine Garantie dafür, daß die zweite QTM auf der von der ersten QTM produzierten Superposition hält.

Kapitel 3

Komplementabschlußeigenschaft

3.1 Begriffsbildung und Motivation

Ein wesentlicher Bestand unseres Hierarchiesatzbeweises wird die Anwendung des folgenden Satzes sein:

Satz 3.1.1 (Komplementabschluß).
Seien $G \subseteq (\Sigma_1 \setminus \#)^*$, $\frac{1}{2} < \alpha \leq 1$ und $T : \mathbb{N} \to \mathbb{N}$ beliebig. Dann gilt

$$L \in BQTime_{G,\alpha}(T(n)) \Rightarrow \overline{L} := (\Sigma_1 \setminus \#)^* \setminus L \in \widehat{BQTime}_{G,\alpha}(T(n))$$

Da der Beweis recht technisch ist, stellen wir vorab die wesentlichen Ideen kurz dar. Anschließend definieren wir eine partielle QTM M' und zeigen dann im zweiten Abschnitt, daß sie die gewünschten Wohlgeformtheits- und Akzeptanzeigenschaften hat. Zur Vereinfachung der Notation schreiben wir bei Mehrspurmaschinen, wenn dies nicht zu Verwechslungen führt, für $x; y \in \Sigma_a \times \Sigma_b$ kurz xy.

Wir wollen die Definition von M' zunächst informell motivieren. Wenn M eine QTM ist, die $L \in BQTime_{G,\alpha}(T(n))$ bezeugt, dann müssen wir M' so konstruieren, daß sie $\overline{L} \in \widehat{BQTime}_{G,\alpha}(T(n))$ bezeugt. M' darf also im allgemeinen auf ihrer Eingabe keinen Schritt länger rechnen als M. Daher führen wir eine Schritt-für-Schritt "Simulation" durch.

Anschaulich gesprochen ist es dabei wesentlich, daß Nachfolgekonfigurationen verschiedener Konfigurationen der aktuellen Superposition bei M' genau dann gleich sind, wenn dies für die "korrespondierenden" Nachfolgekonfigurationen bei M gilt. Andernfalls können wir nicht erwarten, daß die Rechnung von M' im Wesentlichen - bis auf invertiertes Akzeptanzverhalten - mit der von M übereinstimmt.

Die naheliegende Idee, M' immer dann eine 1 auf die letzte Spur schreiben zu lassen, wenn M dorthin ein $\#$ oder eine 0 schreibt, eine 0 schreiben zu lassen, wenn M eine 1 schreibt, und sich ansonsten genau wie M zu verhalten, verletzt die Orthogonalitätsbedingung. Das Problem dabei ist auch, daß M', wenn der Bandinhalt der letzten Spur der aktuellen Zelle eine 1 ist, nicht "weiß", welchen Übergang M in der entsprechenden Konfiguration ausführen würde, denn M könnte dort auf ein $\#$ oder eine 0 treffen und sich jeweils unterschiedlich verhalten.

Um dieses Problem zu beheben, geben wir M' eine zusätzliche Spur mit dem Alphabet $\{\#, 0, 1\}$ vor der letzten Spur, und wenden die Bijektion $\# \mapsto 11, 0 \mapsto 01, 1 \mapsto 00$ zwischen den Symbolen der letzten Spur bei M und den letzten beiden Spuren bei M' an. Ein Problem bleibt dabei bestehen: M' kann nicht unterscheiden zwischen Zellen, die auf der letzten Spur ein $\#$ enthalten, weil sie noch nie besucht wurden, und solchen, die dort mit einem $\#$ im Verlauf der Rechnung beschrieben wurden. M' müßte sich, wenn sie auf $\#\#$ in den letzten beiden Spuren trifft, genauso verhalten, wie wenn sie auf 11 trifft, denn für M sind diese beiden Fälle identisch. Wenn wir jedoch die Übergangsfunktion von M' auf diese Weise definieren, dann verletzt sie die Orthogonalitätsbedingung.

Wir können aber von der Forderung, daß M' immer dann auf die letzte Spur eine 1 schreiben soll, wenn M ein $\#$ schreibt, abrücken, denn zur Entscheidung, ob die Maschine akzeptiert, wird nach dem Halten lediglich die Startzelle beobachtet. Es genügt daher, die obige Bijektion nur für Symbole auf der Startzelle zu fordern und außerhalb der Startzelle mit der Bijektion $\# \mapsto \#\#, 0 \mapsto \#0, 1 \mapsto \#1$ zu arbeiten. Dem ersten Schritt kommt dabei die Aufgabe zu, die Startzelle entsprechend zu "markieren". Wir werden sehen, daß q_0 nur zu Anfang angenommen wird und dabei auf den Bandinhalt $\#\#$ in den letzten beiden Spuren trifft. Ab dem zweiten Schritt kann M' dann anhand des Inhalts der vorletzten Spur entscheiden, ob sie sich auf der Startzelle befindet ($\{0, 1\}$) oder nicht ($\#$) und welche der beiden Bijektionen anzuwenden ist.

Bemerkung 3.1.2. *Wenn wir M' mehr Rechenschritte als M zugestehen würden, könnten wir versuchen, M' mit Hilfe des Verkettungslemmas 2.5.6 aus M zu konstruieren. Für unsere Anwendung ist es jedoch wesentlich, daß die Zeitschranke identisch ist.*

Wenn $\Sigma_k = \{\#, 1\}$ wäre, so könnten wir die zu schreibenden Symbole einfach vertauschen. Wir können jedoch nicht erwarten, daß eine solche QTM eine QTM mit $\Sigma_k = \{\#, 0, 1\}$ ohne Verlangsamung simulieren kann.

Um M' formal zu definieren wählen wir für den Rest des Kapitels beliebige $G \subseteq (\Sigma_1 \setminus \#)^*$, $\frac{1}{2} < \alpha \leq 1$ und $T : \mathbb{N} \to \mathbb{N}$ sowie $L \in BQTime_{G,\alpha}(T(n))$ und fixieren sie.

Definition 3.1.3. *Sei $M = (\Sigma_1 \times \ldots \times \Sigma_k, Q, \delta)$ eine wohlgeformte, stationäre, $T(n)$-zeitbeschränkte Mehrspur-QTM in Normalform mit $\Sigma_k = \{\#, 0, 1\}$, so daß M jedes $x \in G \cap L$ mit einer Wahrscheinlichkeit von mindestens α akzeptiert, und jedes $x \in G \setminus L :$ mit einer Wahrscheinlichkeit von höchstens $1 - \alpha$ akzeptiert. Seien $q_0, q_f \in Q$ der Start- respektive Endzustand von M. Sei*

$D := \{L, R\}$, $\Sigma := \Sigma_1 \times \ldots \times \Sigma_k$

$\Sigma' := \Sigma_1 \times \ldots \times \Sigma_{k-1} \times \Sigma_k \times \Sigma_k$

$\Sigma^B := \Sigma_1 \times \ldots \times \Sigma_{k-1} \times \{00, 01, 11\}$

$\Sigma^\# := \Sigma_1 \times \ldots \times \Sigma_{k-1} \times \{\#\} \times \Sigma_k$

$\Sigma^E := \Sigma^B \cup \Sigma^\#$

$\mathbf{S := Q \times \Sigma^E \subseteq Q \times \Sigma'}$

Sei $\eta : \{11, 01, 00\} \to \{\#, 0, 1\}$ die Bijektion $11 \mapsto \#, 01 \mapsto 0, 00 \mapsto 1$

und $\theta : \{\#\#, \#0, \#1\} \to \{\#, 0, 1\}$ die Bijektion $\#\# \mapsto \#, \#0 \mapsto 0, \#1 \mapsto 1$

Wir definieren eine partielle Mehrspur-QTM $M' = (\Sigma', Q, \delta')$ auf S mit Startzustand q_0 und Endzustand q_f wie folgt:

$\forall a, \tilde{a} \in \Sigma_1 \times \ldots \times \Sigma_{k-1},\ r, s \in \Sigma_k,\ bc, \tilde{b}\tilde{c} \in \{00, 01, 11\},\ q \in Q,\ p \in Q \setminus \{q_0, q_f\},\ d \in D,\ x \in \Sigma^E$ sei

Übergangsfunktion	*Typ*
$\delta'(q_0, a\#r, \tilde{a}\tilde{b}\tilde{c}, q, d) := \delta(q_0, ar, \tilde{a}\eta(\tilde{b}\tilde{c}), q, d)$	1
$\delta'(q_0, a\#r, \tilde{a}\#s, q, d) := 0$	$1'$
$\delta'(p, a\#r, \tilde{a}\#s, q, d) := \delta(p, ar, \tilde{a}s, q, d)$	2
$\delta'(p, a\#r, \tilde{a}\tilde{b}\tilde{c}, q, d) := 0$	$2'$
$\delta'(p, abc, \tilde{a}\tilde{b}\tilde{c}, q, d) := \delta(p, a\eta(bc), \tilde{a}\eta(\tilde{b}\tilde{c}), q, d)$	3
$\delta'(p, abc, \tilde{a}\#s, q, d) := 0$	$3'$
$\delta'(q_f, x, x, q_0, R) := 1$	4
$\delta'(q_f, x, \tilde{x}, \tilde{q}, \tilde{d}) := 0 \qquad \forall(\tilde{q}, \tilde{x}, \tilde{d}) \in (S \times D) \setminus \{(q_0, x, R)\}$	$4'$
$\delta'(q_0, abc, \tilde{a}\#s, q, d) := \delta(q_0, a\eta(bc), \tilde{a}s, q, d)$	5
$\delta'(q_0, abc, \tilde{a}\tilde{b}\tilde{c}, q, d) := 0$	$5'$

Offensichtlich läßt sich jedes Quintupel aus dem Definitionsbereich $Q \times \Sigma^E \times \Sigma^E \times Q \times D$ genau einem Typ zuordnen, und damit ist M' eine partielle QTM bezüglich S. Wir benutzen die Klassifikation der Argumente von δ' in Typen, um im folgenden weniger schreiben zu müssen.

Bemerkung: Wie wir später sehen werden, gilt für jede Eingabe $x \in (\Sigma_1 \setminus \#)^*$ und jede Konfiguration einer Superposition von M' zum Zeitpunkt $t < T^{M'}(x)$: das Paar aus aktuellem Zustand und aktuellem Symbol unter dem Kopf (q, σ) ist ein Element von $((Q \setminus \{q_0\}) \times \Sigma^E) \cup (\{q_0\} \times (\Sigma_1 \times \ldots \times \Sigma_{k-1} \times \{\#\} \times \{\#\}))$. Wir nehmen in S jedoch alle Paare aus $\{q_0\} \times \Sigma^E$ auf, damit wir die Typen $4/4'$ so definieren können wie wir es getan haben. Andernfalls wäre es unklar, wie $\delta'(q_f, x)$ für $x \notin (\Sigma_1 \times \ldots \times \Sigma_{k-1} \times \{\#\} \times \{\#\})$ definiert werden muß, damit M' partiell wohlgeformt ist (dank unserer Definition von $4/4'$ können wir ausnutzen, daß der Normalformsatz für M garantiert, daß q_0 in M' von keinem anderen Zustand als q_f erreicht wird). Diese Wahl von S erfordert wiederum die Definition von $\delta'(q_0, x)$ für alle $x \in \Sigma^E$ (Typen 1/1'/5/5').

Intuitiv können wir uns vorstellen, daß Typ $1/1'$ dem ersten Schritt einer Rechnung entspricht, Typ $2/2'$ der Normalfall während der Rechnung ist, solange wir nicht auf der Startzelle sind, und Typ $3/3'$ auftritt, wenn wir während der Rechnung die Startzelle erreichen. Die Typen $4/4'/5/5'$ spielen wegen obiger Bemerkung keine Rolle für die Rechnung von M' auf einer Eingabe $x \in \Sigma^E$, denn sie kommen erst nach dem Halten in Frage. Ihre Definition ist derart, daß M' partiell wohlgeformt ist, wie der nächste Satz zeigt.

Die in den verschiedenen Fällen angewandten Bijektionen stellt die folgende Tabelle noch einmal im Überblick dar:

von $\downarrow$ nach $\rightarrow$	$Q \times \Sigma^\#$	$Q \times \Sigma^B$
$\{q_0\} \times \Sigma^\#$	0 (Typ $1'$)	θ, η (Typ 1)
$\{q_0\} \times \Sigma^B$	η, θ (Typ 5)	0 (Typ $5'$)
$(Q \setminus \{q_0, q_f\}) \times \Sigma^\#$	θ, θ (Typ 2)	0 (Typ $2'$)
$(Q \setminus \{q_0, q_f\}) \times \Sigma^B$	0 (Typ $3'$)	η, η (Typ 3)
$\{q_f\} \times (\Sigma^\# \cup \Sigma^B)$	$\in \{0, 1\}$ gemäß (Typ $4/4'$)	

In den Zeilen sind die möglichen Paare aus aktuellem Zustand und aktuellem Symbol unter dem Kopf gruppiert, und in den Spalten die möglichen Paare aus Folgezustand und zu schreibendem Symbol.

Dabei bedeutet ein Eintrag 0, daß die entsprechenden Amplituden 0 sind, und die Einträge (μ, ν) mit $\mu, \nu \in \{\eta, \theta\}$ bedeuten, daß die Bijektion μ auf die letzten beiden Spuren des aktuellen Symbols angewandt wird, und die Bijektion ν auf die letzten beiden Spuren des zu schreibenden Symbols. In Klammern ist der jeweilige Übergangstyp angegeben.

Beachte, daß die Bewegungsrichtung von M' in allen Fällen aus der Bewegungsrichtung von M hervorgeht.

3.2 Beweise

Satz 3.2.1. *Die partielle QTM M' aus Definition 3.1.3 ist wohlgeformt.*

Beweis. Seien
$\alpha : \Sigma' \to \Sigma_k \times \ldots \times \Sigma_{k-1},\ a_1 \ldots a_{k+1} \mapsto a_1 \ldots a_{k-1}$
$\beta : \Sigma' \to \Sigma_k \times \Sigma_k,\ a_1 \ldots a_{k+1} \mapsto a_k a_{k+1}$
$\gamma : \Sigma' \to \Sigma_k,\ a_1 \ldots a_{k+1} \mapsto a_{k+1}$

(i) Einheitslänge

zu zeigen: $\forall (p, \sigma) \in S : (S1) := \sum_{(q,\tau,d) \in S \times D} |\delta'(p, \sigma, \tau, q, d)|^2 \stackrel{!}{=} 1$

1. Fall $(p, \sigma) = (q_0, a\#r)$ mit $a \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $r \in \Sigma_k$

$$\begin{aligned}
(S1) &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} |\underbrace{\delta'(q_0, a\#r, \tau, q, d)}_{=\delta(q_0, ar, \alpha(\tau)\eta(\beta(\tau)), q, d) \text{ da vom Typ 1}}|^2 + \\
&\quad \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} |\underbrace{\delta'(q_0, a\#r, \tau, q, d)}_{=0 \text{ da vom Typ } 1'}|^2 \\
&= \sum_{(q,\rho,d)\in Q\times\Sigma\times D} |\delta(q_0, ar, \rho, q, d)|^2 \quad \text{da } \eta \text{ Bijektion} \\
&= 1 \quad \text{da } \delta \text{ wohlgeformt}
\end{aligned}$$

2. Fall $(p, \sigma) = (q_0, abc)$ mit $a \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $bc \in \{00, 01, 11\}$

$$\begin{aligned}
(S1) &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} |\underbrace{\delta'(q_0, abc, \tau, q, d)}_{=0 \text{ da vom Typ } 5'}|^2 + \\
&\quad \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} |\underbrace{\delta'(q_0, abc, \tau, q, d)}_{=\delta(q_0, a\eta(bc), \alpha(\tau)\gamma(\tau), q, d) \text{ da vom Typ 5}}|^2 \\
&= \sum_{(q,\rho,d)\in Q\times\Sigma\times D} |\delta(q_0, a\eta(bc), \rho, q, d)|^2 \\
&= 1 \quad \text{da } \delta \text{ wohlgeformt}
\end{aligned}$$

3. Fall $(p, \sigma) = (p, a\#r)$ mit $a \in \Sigma_1 \times \ldots \times \Sigma_{k-1},\ r \in \Sigma_k,\ p \in Q \backslash \{q_0, q_f\}$

$$\begin{aligned}
(S1) &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} |\underbrace{\delta'(p, a\#r, \tau, q, d)}_{=0 \text{ da vom Typ } 2'}|^2 + \\
&\quad \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} |\underbrace{\delta'(p, a\#r, \tau, q, d)}_{=\delta(p, ar, \alpha(\tau)\gamma(\tau), q, d) \text{ da vom Typ 2}}|^2
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{(q,\rho,d)\in Q\times\Sigma\times D} |\delta(p, ar, \rho, q, d)|^2 \\
&= 1 \quad \text{da } \delta \text{ wohlgeformt}
\end{aligned}
$$

4. Fall $(p, \sigma) = (p, abc)$ mit $a \in \Sigma_1 \times \ldots \times \Sigma_{k-1},\ bc \in \{00, 01, 11\},\ p \in Q \setminus \{q_0, q_f\}$

$$
\begin{aligned}
(S1) &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} |\underbrace{\delta'(p, abc, \tau, q, d)}_{=\delta(p,a\eta(bc),\alpha(\tau)\eta(\beta(\tau)),q,d) \text{ da vom Typ 3}}|^2 + \\
&\quad \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} |\underbrace{\delta'(p, abc, \tau, q, d)}_{=0 \text{ da vom Typ } 3'}|^2 \\
&= \sum_{(q,\rho,d)\in Q\times\Sigma\times D} |\delta(p, a\eta(bc), \rho, q, d)|^2 \quad \text{da } \eta \text{ Bijektion} \\
&= 1 \quad \text{da } \delta \text{ wohlgeformt}
\end{aligned}
$$

5. Fall $(p, \sigma) = (q_f, x)$ mit $x \in \Sigma^E$

$$
\begin{aligned}
(S1) &= |\underbrace{\delta'(q_f, x, x, q_0, R)}_{=1 \text{ da vom Typ 4}}|^2 + \sum_{(q,\tau,d)\in Q\times\Sigma^E\times D\setminus\{(q_0,x,R)\}} |\underbrace{\delta'(q_f, x, \tau, q, d)}_{=0 \text{ da vom Typ } 4'}|^2 \\
&= 1
\end{aligned}
$$

(ii) Orthogonalität

<u>zu zeigen</u>: $\forall (p_1, \sigma_1) \neq (p_2, \sigma_2) \in S$:

$$(S2) := \sum_{(q,\tau,d)\in S\times D} \delta'^*(p_1, \sigma_1, \tau, q, d)\, \delta'(p_2, \sigma_2, \tau, q, d) \stackrel{!}{=} 0$$

Falls $(p_2, \sigma_2) \in \{q_f\} \times \Sigma^E$, so ist $(S2)$ von der Form $\delta^*(p_1, *_1, *_2, q_0, R)$, wobei die $*_i$ eindeutig durch p_1, σ_1 und σ_2 bestimmt sind. Für $p_1 \neq q_f$ ist dies Null wegen Normalformsatz 2.7.2 und wir brauchen den Fall $p_2 = q_f$ für $p_1 \neq q_f$ im Folgenden nicht mehr aufzuführen.

1. Fall $(p_1, \sigma_1) = (q_0, a\#r)$ mit $a \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $r \in \Sigma_k$. Sei $(p_2, \sigma_2) \neq (p_1, \sigma_1)$ beliebig.

$$
\begin{aligned}
(S2) &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} \underbrace{\delta'^*(q_0, a\#r, \tau, q, d)}_{=\delta^*(q_0,ar,\alpha(\tau)\eta(\beta(\tau)),q,d) \text{ da vom Typ 1}} \delta'(p_2, \sigma_2, \tau, q, d) + \\
&\quad \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} \underbrace{\delta'^*(q_0, a\#r, \tau, q, d)}_{=0 \text{ da vom Typ } 1'} \delta'(p_2, \sigma_2, \tau, q, d) \\
&=: (S2)_1
\end{aligned}
$$

Eine Inspektion der Definition von δ' zeigt, daß wegen $\tau \in \Sigma^B$ für $(p_2, \sigma_2) \in ((Q \setminus \{q_0, q_f\}) \times \Sigma^{\#}) \cup (\{q_0\} \times \Sigma^B)$ gilt: $\delta'(p_2, \sigma_2, \tau, q, d) = 0$.
Falls $(p_2, \sigma_2) = (q_0, \tilde{a}\#\tilde{r})$ für ein $\tilde{a} \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und ein $\tilde{r} \in \Sigma_k$, dann ist $\tilde{a}\tilde{r} \neq ar$ und es gilt (Typ 1):

$$\begin{aligned}
(S2)_1 &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} \delta^*(q_0, ar, \alpha(\tau)\eta(\beta(\tau)), q, d) \cdot \\
&\qquad \delta(q_0, \tilde{a}\tilde{r}, \alpha(\tau)\eta(\beta(\tau)), q, d) \\
&= \sum_{(q,\rho,d)\in Q\times\Sigma\times D} \delta^*(q_0, ar, \rho, q, d)\, \delta(q_0, \tilde{a}\tilde{r}, \rho, q, d) \\
&= 0 \quad \text{da } \delta \text{ wohlgeformt und } \tilde{a}\tilde{r} \neq ar
\end{aligned}$$

Falls $(p_2, \sigma_2) = (p, \tilde{a}\tilde{b}\tilde{c})$ für ein $p \in Q \setminus \{q_0, q_f\}$, $\tilde{a} \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $\tilde{b}\tilde{c} \in \{00, 01, 11\}$, gilt (Typ 3):

$$\begin{aligned}
(S2)_1 &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} \delta^*(q_0, ar, \alpha(\tau)\eta(\beta(\tau)), q, d) \cdot \\
&\qquad \delta(p, \tilde{a}\eta(\tilde{b}\tilde{c}), \alpha(\tau)\eta(\beta(\tau)), q, d) \\
&= \sum_{(q,\rho,d)\in Q\times\Sigma\times D} \delta^*(q_0, ar, \rho, q, d)\, \delta(p, \tilde{a}\eta(\tilde{b}\tilde{c}), \rho, q, d) \\
&= 0 \quad \text{da } \delta \text{ wohlgeformt und } p \neq q_0
\end{aligned}$$

2. Fall $(p_1, \sigma_1) = (q_0, abc)$ mit $a \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $bc \in \{00, 01, 11\}$.
Sei $(p_2, \sigma_2) \neq (p_1, \sigma_1)$ beliebig.

$$\begin{aligned}
(S2) &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} \underbrace{\delta'^*(q_0, abc, \tau, q, d)}_{=0 \text{ da vom Typ } 5'} \delta'(p_2, \sigma_2, \tau, q, d) + \\
&\quad \sum_{(q,\tau,d)\in Q\times\Sigma^{\#}\times D} \underbrace{\delta'^*(q_0, abc, \tau, q, d)}_{=\delta^*(q_0, a\eta(bc), \alpha(\tau)\gamma(\tau), q, d) \text{ da vom Typ } 5} \delta'(p_2, \sigma_2, \tau, q, d) \\
&=: (S2)_2
\end{aligned}$$

Für $(p_2, \sigma_2) \in ((Q \setminus \{q_0, q_f\}) \times \Sigma^B) \cup (\{q_0\} \times \Sigma^{\#})$ gilt wegen $\tau \in \Sigma^{\#}$: $\delta'(p_2, \sigma_2, \tau, q, d) = 0$.
Falls $(p_2, \sigma_2) = (p, \tilde{a}\#\tilde{r})$ für ein $p \in Q \setminus \{q_0, q_f\}$, $\tilde{a} \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $\tilde{r} \in \Sigma_k$, gilt (Typ 2):

$$(S2)_2 = \sum_{(q,\tau,d)\in Q\times\Sigma^{\#}\times D} \delta^*(q_0, a\eta(bc), \alpha(\tau)\gamma(\tau), q, d) \cdot$$

$$
\begin{aligned}
& \delta(p, \tilde{a}\tilde{r}, \alpha(\tau)\gamma(\tau), q, d) \\
= & \sum_{(q,\rho,d)\in Q\times\Sigma\times D} \delta^*(q_0, \eta(bc), \rho, q, d)\delta(p, \tilde{a}\tilde{r}, \rho, q, d) \\
= & \; 0 \quad \text{da } \delta \text{ wohlgeformt und } p \neq q_0
\end{aligned}
$$

Falls $(p_2, \sigma_2) = (q_0, \tilde{a}\tilde{b}\tilde{c})$ für ein $\tilde{a} \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $\tilde{b}\tilde{c} \in \{00, 01, 11\}$, dann ist $\tilde{a}\tilde{b}\tilde{c} \neq abc$ und es gilt (Typ 5):

$$
\begin{aligned}
(S2)_2 = & \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} \delta^*(q_0, a\eta(bc), \alpha(\tau)\gamma(\tau)), q, d) \cdot \\
& \delta(q_0, \tilde{a}\eta(\tilde{b}\tilde{c}), \alpha(\tau)\gamma(\tau)), q, d) \\
= & \sum_{(q,\rho,d)\in Q\times\Sigma\times D} \delta^*(q_0, a\eta(bc), \rho, q, d)\, \delta(q_0, \tilde{a}\eta(\tilde{b}\tilde{c}), \rho, q, d) \\
= & \; 0 \quad \text{da } \delta \text{ wohlgeformt und } a\eta(bc) \neq \tilde{a}\eta(\tilde{b}\tilde{c})
\end{aligned}
$$

3. Fall $(p_1, \sigma_1) = (p, a\#r)$ mit $a \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$, $r \in \Sigma_k$, $p \in Q \setminus \{q_0, q_f\}$. Sei $(p_2, \sigma_2) \neq (p_1, \sigma_1)$ beliebig.

$$
\begin{aligned}
(S2) = & \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D} \underbrace{\delta'^*(p, a\#r, \tau, q, d)}_{=0 \text{ da vom Typ } 2'} \delta'(p_2, \sigma_2, \tau, q, d) + \\
& \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} \underbrace{\delta'^*(p, a\#r, \tau, q, d)}_{=\delta^*(p, ar, \alpha(\tau)\gamma(\tau), q, d) \text{ da vom Typ 2}} \delta'(p_2, \sigma_2, \tau, q, d) \\
=: & \; (S2)_4
\end{aligned}
$$

Für $(p_2, \sigma_2) \in ((Q \setminus \{q_0, q_f\}) \times \Sigma^B) \cup (\{q_0\} \times \Sigma^\#)$ gilt wegen $\tau \in \Sigma^\#$: $\delta'(p_2, \sigma_2, \tau, q, d) = 0$.
Falls $(p_2, \sigma_2) = (\tilde{p}, \tilde{a}\#\tilde{r})$ mit $\tilde{p} \in Q \setminus \{q_0, q_f\}$, $\tilde{a} \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $\tilde{r} \in \Sigma_k$, dann ist $(\tilde{p}, \tilde{a}\#\tilde{r}) \neq (p, a\#r)$ und es gilt (Typ 2):

$$
\begin{aligned}
(S2)_4 = & \sum_{(q,\tau,d)\in Q\times\Sigma^\#\times D} \delta^*(p, ar, \alpha(\tau)\gamma(\tau), q, d) \cdot \\
& \delta(\tilde{p}, \tilde{a}\tilde{r}, \alpha(\tau)\gamma(\tau), q, d) \\
= & \sum_{(q,\rho,d)\in Q\times\Sigma\times D} \delta^*(p, ar, \rho, q, d)\, \delta(\tilde{p}, \tilde{a}\tilde{r}, \rho, q, d) \\
= & \; 0 \quad \text{da } \delta \text{ wohlgeformt und } (p, ar) \neq (\tilde{p}, \tilde{a}\tilde{r})
\end{aligned}
$$

Der Fall $(p_2, \sigma_2) = (q_0, \tilde{a}\tilde{b}\tilde{c})$ mit $\tilde{a} \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$ und $\tilde{b}\tilde{c} \in \{00, 01, 11\}$ ist symmetrisch zum bereits gezeigten Fall 2 und ergibt daher $(S2) = 0$.

4. Fall $(p_1,\sigma_1)=(p,abc)$ mit $a\in\Sigma_1\times\ldots\times\Sigma_{k-1}$, $bc\in\{00,01,11\}$, $p\in Q\setminus\{q_0,q_f\}$. Sei $(p_2,\sigma_2)\neq(p_1,\sigma_1)$ beliebig.

Die Fälle $(p_2,\sigma_2)\in((Q\setminus\{q_0,q_f\})\times\Sigma^{\#})\cup(\{q_0\}\times\Sigma^E)$ sind symmetrisch zu bereits gezeigten Fällen und ergeben daher $(S2)=0$.
Es bleibt noch der Fall $(p_2,\sigma_2)=(\tilde{p},\tilde{a}\tilde{b}\tilde{c})$ mit $\tilde{p}\in Q\setminus\{q_0,q_f\}$, $\tilde{a}\in\Sigma_1\times\ldots\times\Sigma_{k-1}$ und $\tilde{b}\tilde{c}\in\{00,01,11\}$. Dann ist $(\tilde{p},\tilde{a}\tilde{b}\tilde{c})\neq(p,abc)$ und es folgt

$$\begin{aligned}
(S2) &= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D}\underbrace{\delta'^*(p,abc,\tau,q,d)}_{=\delta^*(p,a\eta(bc),\alpha(\tau)\eta(\beta(\tau)),q,d)\text{ da vom Typ 3}}\cdot\\
&\qquad \delta'(\tilde{p},\tilde{a}\tilde{b}\tilde{c},\tau,q,d)+\\
&\qquad \sum_{(q,\tau,d)\in Q\times\Sigma^{\#}\times D}\underbrace{\delta'^*(p,abc,\tau,q,d)}_{=0\text{ da vom Typ }3'}\delta'(\tilde{p},\tilde{a}\tilde{b}\tilde{c},\tau,q,d)\\
&= \sum_{(q,\tau,d)\in Q\times\Sigma^B\times D}\delta^*(p,a\eta(bc),\alpha(\tau)\gamma(\beta(\tau)),q,d)\cdot\\
&\qquad \delta(\tilde{p},\tilde{a}\eta(\tilde{b}\tilde{c}),\alpha(\tau)\eta(\beta(\tau)),q,d)\\
&= \sum_{(q,\rho,d)\in Q\times\Sigma\times D}\delta^*(p,a\eta(bc),\rho,q,d)\,\delta(\tilde{p},\tilde{a}\eta(\tilde{b}\tilde{c}),\rho,q,d)\\
&= 0\quad\text{da }\delta\text{ wohlgeformt und }(\tilde{p},\tilde{a}\eta(\tilde{b}\tilde{c}))\neq(p,a\eta(bc))
\end{aligned}$$

5. Fall $(p_1,\sigma_1)=(q_f,x)$ mit $x\in\Sigma^E$. Sei $(p_2,\sigma_2)\neq(p_1,\sigma_1)$ beliebig.

$$\begin{aligned}
(S2) &= \sum_{(q,\tau,d)\in Q\times\Sigma^E\times D\setminus\{(q_0,x,R)\}}\underbrace{\delta'^*(q_f,x,\tau,q,d)}_{=0\text{ da vom Typ }4'}\delta'(p_2,\sigma_2,\tau,q,d)+\\
&\qquad \underbrace{\delta'^*(q_f,x,x,q_0,R)}_{=1\text{ da vom Typ 4}}\delta'(p_2,\sigma_2,x,q_0,R)\\
&= \delta'(p_2,\sigma_2,x,q_0,R)\\
&= 0\quad\text{denn:}
\end{aligned}$$

<u>Falls</u> $p_2=q_f$, so ist $\sigma_2\neq x$ und $\delta'(q_f,\sigma_2,x,q_0,R)=0$, da vom Typ $4'$
<u>Falls</u> $p_2\neq q_f$ so ist $\delta'(p_2,\sigma_2,x,q_0,R)$ von der Form $\delta(p_2,*_1,*_2,q_0,R)$ und dies ist Null wegen Normalformsatz 2.7.2.

(iii) Separabilität
<u>zu zeigen</u>: $\forall(p_1,\sigma_1,\sigma_1'),(p_2,\sigma_2,\sigma_2')\in S\times\Sigma^E$:
$(S3):=\sum_{p\in Q}\delta'^*(p_1,\sigma_1,\sigma_1',p,R)\,\delta'(p_2,\sigma_2,\sigma_2',p,L)\overset{!}{=}0$

Die folgende Tabelle listet alle möglichen Typkombinationen auf:

	Typ $\rightarrow$	$1'$	1	5	$5'$	2	$2'$	$3'$	3	$4/4'$
Typ $\downarrow$	$(p_1,\sigma_1,\sigma_1') = \rightarrow$ $(p_2,\sigma_2,\sigma_2') = \downarrow$	$q_0, a\#r, \tilde{a}\#\tilde{r}$	$q_0, a\#r, \tilde{a}\tilde{b}\tilde{c}$	$q_0, abc, \tilde{a}\#\tilde{r}$	$q_0, abc, \tilde{a}\tilde{b}\tilde{c}$	$p_a, a\#r, \tilde{a}\#\tilde{r}$	$p_a, a\#r, \tilde{a}\tilde{b}\tilde{c}$	$p_a, abc, \tilde{a}\#\tilde{r}$	$p_a, abc, \tilde{a}\tilde{b}\tilde{c}$	$q_f, x, \tilde{x}$
$1'$	$q_0, f\#s, \tilde{f}\#\tilde{s}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$
1	$q_0, f\#s, \tilde{f}\tilde{g}\tilde{h}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{E}$
5	$q_0, fgh, \tilde{f}\#\tilde{s}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{E}$
$5'$	$q_0, fgh, \tilde{f}\tilde{g}\tilde{h}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$
2	$p_b, f\#s, \tilde{f}\#\tilde{s}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{E}$
$2'$	$p_b, f\#s, \tilde{f}\tilde{g}\tilde{h}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$
$3'$	$p_b, fgh, \tilde{f}\#\tilde{s}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{B}$	$\mathcal{B}$
3	$p_b, fgh, \tilde{f}\tilde{g}\tilde{h}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{C}$	$\mathcal{E}$
$4/4'$	$q_f, y, \tilde{y}$	$\mathcal{A}$	$\mathcal{D}$	$\mathcal{D}$	$\mathcal{A}$	$\mathcal{D}$	$\mathcal{A}$	$\mathcal{A}$	$\mathcal{D}$	$\mathcal{D}$

Dabei ist $p_a, p_b \in Q \setminus \{q_0, q_f\}$, $a, \tilde{a}, f, \tilde{f} \in \Sigma_1 \times \ldots \times \Sigma_{k-1}$, $r, \tilde{r}, s, \tilde{s} \in \Sigma_k$, $bc, \tilde{b}\tilde{c}, gh, \tilde{g}\tilde{h} \in \{00, 01, 11\}$, $x, \tilde{x}, y, \tilde{y} \in \Sigma^E$ und die Einträge bedeuten:

$\mathcal{A}$: $(S3) = 0$ denn $\forall q \in Q, d \in D : \delta'^*(p_1, \sigma_1, \sigma_1', q, d) = 0$

$\mathcal{B}$: es gilt nicht $\mathcal{A}$, aber $(S3) = 0$, denn $\forall q \in Q, d \in D : \delta'(p_2, \sigma_2, \sigma_2', q, d) = 0$

$\mathcal{C}$: es gilt nicht $\mathcal{A}, \mathcal{B}$, aber $(S3) = 0$ folgt aus der entsprechenden Eigenschaft für δ denn es ist von der Form $\sum_{p \in Q} \delta^*(p_1, *_1, *_2, p, R) \cdot \delta(p_2, *_3, *_4, p, L)$ wobei die $*_i$ eindeutig durch $p_1, \sigma_1, \sigma_1', p_2, \sigma_2, \sigma_2'$ bestimmt sind

$\mathcal{D}$: es gilt nicht $\mathcal{A}, \mathcal{B}, \mathcal{C}$, aber $(S3) = 0$, denn $\forall q \in Q : \delta'(q_f, y, \tilde{y}, q, L) = 0$, da vom Typ $4'$

$\mathcal{E}$: es gilt nicht $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$, aber $(S3) = 0$ folgt aus dem Normalformsatz 2.7.2, denn es ist von der Form $\delta(p_2, *_5, *_6, q_0, L)$, wobei die $*_i$ eindeutig durch p_2, σ_2, σ_2' bestimmt sind und für die betreffenden Typen 1,2,3 und 5 $p_2 \neq q_f$ gilt.

Es ist also in allen Fällen (S3)=0

Damit ist nach Theorem 2.6.1 M' auf ihrem Definitionsbereich wohlgeformt. □

Korollar 3.2.2. *M' läßt sich zu einer wohlgeformten QTM vervollständigen, die angesetzt auf eine beliebige Eingabe $x \in (\Sigma_1 \setminus \#)^*$ genau die gleichen Berechnungsschritte wie M' auf x durchführt.*

Beweis. Wegen Satz 3.2.1 und Vervollständigungssatz 2.6.4 existiert eine auf ganz $Q \times \Sigma'$ definierte, wohlgeformte QTM, deren Übergangsfunktion auf S mit δ' übereinstimmt.

Für jede Eingabe $x \in (\Sigma_1 \setminus \#)^*$ enthält das Band zu Beginn nur Symbole aus $\Sigma^{\#}$. Wegen Satz 2.6.5 ergibt sich induktiv, daß die vervollständigte QTM auf Eingabe x zu jedem Zeitpunkt nur Symbole aus Σ^E auf das Band schreibt. Daher ist für jede Konfiguration der Superposition zu einem beliebigen Zeitpunkt das Paar von aktuellem Zustand und aktuellem Symbol unter dem Kopf in S enthalten.

Bei der Vervollständigung ändert sich also keine einzige Übergangsamplitude, die bei einer Rechnung auf einer Eingabe $x \in (\Sigma_1 \setminus \#)^*$ eingeht. Daher verhält sich die vervollständigte QTM für jede Eingabe genau wie M'. □

Zur Vereinfachung der Notation bezeichnen wir im folgenden diese auf ganz $Q \times \Sigma'$ definierte, wohlgeformte QTM wieder mit M'.

Der folgende Satz drückt aus, daß M' Schritt für Schritt im Wesentlichen die gleiche Rechnung durchführt wie M, mit dem Unterschied, daß sie genau das entgegengesetzte Akzeptanzverhalten aufweist.

Satz 3.2.3. *a) Für alle $x \in \Sigma_1^*$, alle $0 < t \leq T^M(x)$ und jede Konfiguration c' (mit Amplitude ungleich Null) aus der Superposition, die sich zum Zeitpunkt t aus der Rechnung von M' auf Eingabe x ergibt, gilt:*

Es gibt eine korrespondierende Konfiguration c aus der Superposition, die sich zum Zeitpunkt t aus der Rechnung von M auf Eingabe x ergibt, die die gleiche Amplitude wie c' aufweist, bei der die Kopfposition und der aktuelle Zustand mit denen in c' übereinstimmt, deren Bandinhalt auf den ersten $k-1$ Spuren gleich dem bei c' ist, und wo für die weitere(n) Spure(n) folgende Bijektion gilt:

- *Die letzten beiden Spuren der Startzelle in c' enthalten die Symbole $\eta^{-1}(s)$, wenn die letzte Spur der Startzelle in c das Symbol s enthält.*
- *Für jede andere Zelle der letzten beiden Spuren in c' gilt, daß sie $\#s$ enthält, wenn die letzte Spur der korrespondieren Zelle in c das Symbol s enthält.*

b) Die wohlgeformte QTM M' ist stationär und hält auf jeder Eingabe mit der gleichen Laufzeit wie M.
Der Beobachtung des Bandinhalts von M' nach dem Halten liegt eine bezüglich obiger Bijektion gleiche Wahrscheinlichkeitsverteilung wie der Beobachtung des Bandinhalts von M nach dem Halten zugrunde.

Bevor wir diesen Satz beweisen, stellen wir fest, daß hieraus der Komplementabschlußsatz 3.1.1 folgt.

Korollar 3.2.4 (beweist Satz 3.1.1). *Für die wohlgeformte, vervollständigte, stationäre QTM M' gilt mit den Bezeichnungen aus Definition 3.1.3:* M *bezeugt* $L \in BQTime_{G,\alpha}(T(n)) \Rightarrow M'$ *bezeugt* $\overline{L} \in \widehat{BQTime}_{G,\alpha}(T(n))$

Beweis. $\forall x \in G \cap L = G \setminus \overline{L} : M$ hält nach $T^M(x)$ Schritten, und die anschließende Beobachtung der letzten Spur der Startzelle liefert eine 1 mit einer Wahrscheinlichkeit von mindestens α. Daraus folgt mit Satz 3.2.3 $\forall x \in G \cap L : M'$ hält nach $T^M(x)$ Schritten, und die anschließende Beobachtung der letzten Spur der Startzelle liefert eine 1 mit einer Wahrscheinlichkeit von höchstens $1-\alpha$ (nämlich genau dann wenn M eine 0 oder ein # liefert, siehe Definition von η), das heißt, M' akzeptiert x mit einer Wahrscheinlichkeit von höchstens $1 - \alpha$.

$\forall x \in G \setminus L = G \cap \overline{L} : M$ hält nach $T^M(x)$ Schritten, und die anschliessende Beobachtung der letzten Spur der Startzelle liefert eine 0 oder ein # mit einer Wahrscheinlichkeit von mindestens α. Daraus folgt mit Satz 3.2.3 $\forall x \in G \setminus L : M'$ hält nach $T^M(x)$ Schritten, und die anschließende Beobachtung der letzten Spur der Startzelle liefert eine 1 mit einer Wahrscheinlichkeit von mindestens α (siehe Definition von η), das heißt, M' akzeptiert x mit einer Wahrscheinlichkeit von mindestens α.

□

Beweis von Satz 3.2.3. Sei $x \in \Sigma_1^*$ beliebig. Wir führen den Beweis von Teil a) per Vollständiger Induktion nach t.
$\underline{t=1}$: M' führt ausgehend von der Startsuperposition einen Übergang der Typen 1 und $1'$ aus, da die letzten k Spuren mit Blanks initialisiert sind. Sei c' eine beliebige Konfiguration aus der resultierenden Superposition mit Amplitude ungleich Null. c' hat in den letzten beiden Spuren der Startzelle den Bandinhalt $\eta^{-1}(s)$, wenn der korrespondierende Übergang von M, der sich aus der Definition von Typ 1 ergibt, zu einer Konfiguration c mit Bandinhalt s in der letzten Spur der Startzelle führt. Ferner hat c die gleiche Amplitude, den gleichen Zustand, die gleiche Kopfposition und den gleichen Bandinhalt auf den ersten $k-1$ Spuren wie c. Außerhalb der Startzelle gilt die gewünschte Bijektion zwischen den Bandinhalten der letzten beiden Spuren von M'

und der letzten Spur von M ebenfalls, da dort nur Blanks stehen.
$\underline{t \to t+1}$: Sei c' eine beliebige Konfiguration aus der Superposition von M' auf Eingabe x mit Amplitude ungleich Null. Die nach Induktionsvoraussetzung existierende korrespondierende Konfiguration von M bezeichnen wir mit c, den aktuellen Zustand in c und c' mit q, und den Bandinhalt der letzten Spur an der aktuellen Kopfposition in c mit s.
Da $t < T^M(x)$ folgt aus Lemma 2.7.4 angewandt auf M daß $q \neq q_f$, und wegen $t > 0$ auch $q \neq q_0$.
Falls sich der Kopf in c' auf der Startzelle befindet, so ist nach Induktionsvoraussetzung der Bandinhalt der letzten beiden Spuren an der aktuellen Kopfposition $\eta^{-1}(s)$ und M' führt einen Übergang vom Typ 3 aus. Für jede so entstandene Nachfolgekonfiguration von c' gibt es eine aus der Definition von Typ 3 resultierende korrespondierende Nachfolgekonfiguration von c bei M, die die gleiche Amplitude aufweist, die gleiche Kopfposition und den gleichen aktuellen Zustand hat. Gemäß der Induktionsvoraussetzung gilt hierfür außerhalb der Startzelle die gewünschte Bijektion zwischen den Bandinhalten, da bei den beiden korrespondierenden Übergängen nur die jeweilige Startzelle beschrieben und alle anderen Zellen unverändert gelassen werden. Für die Startzelle gilt die gewünschte Bijektion, da der Übergang von M' zum Bandinhalt $\tilde{a}\eta^{-1}(\tilde{s})$ führt, wenn der korrespondierende Übergang von M zum Bandinhalt $\tilde{a}\tilde{s}$ führt.
Falls sich der Kopf in c' nicht auf der Startzelle befindet, so ist nach Induktionsvoraussetzung der Bandinhalt der letzten beiden Spuren an der aktuellen Kopfposition in c' $\#s$ und M' führt einen Übergang vom Typ 2 aus. Für jede daraus hervorgehende Nachfolgekonfiguration gibt es wiederum eine korrespondierende Nachfolgekonfiguration von c gemäß der Definition von Typ 2. Sie weist die gleiche Amplitude, die gleiche Kopfposition und den gleichen aktuellen Zustand auf, und die Bandinhalte unterscheiden sich gemäß der gewünschten Bijektion: Außerhalb der aktuellen Kopfposition bei c' und c folgt dies aus der Induktionsvoraussetzung, da diese Zellen bei den beiden korrespondierenden Übergängen nicht verändert werden. Für die aktuelle Kopfposition folgt es aus der Definition von Typ 2, woraus hervorgeht daß M' zum Bandinhalt $\tilde{a}\#\tilde{s}$ führt, wenn der korrespondierende Übergang von M zum Bandinhalt $\tilde{a}\tilde{s}$ führt.
Damit ist Teil a) bewiesen.

Teil b) folgt, indem wir Teil a) für den Zeitpunkt $T^M(x)$ anwenden: Für eine beliebige Konfiguration c' aus der Superposition von M' auf x zum Zeitpunkt $T^M(x)$ existiert eine Konfiguration c aus der Superposition von M auf x zum Zeitpunkt $T^M(x)$, bei denen insbesondere die aktuelle Kopfposition und der aktuelle Zustand übereinstimmen. Aus der Definition von $T^M(x)$

folgt, daß der aktuelle Zustand in c, und damit auch der in c', q_f sein muß. Daraus folgt, daß jede Konfiguration aus der Superposition von M' auf x zum Zeitpunkt $T^M(x)$ im Zustand q_f sein muß und es sich somit um eine Endsuperposition handelt. Da M stationär ist (siehe Definition 3.1.3), zeigt der Kopf in c und damit auch der in c' auf die Startzelle und M' ist stationär, da c' beliebig war. Die Amplituden von c und c' stimmen überein, daher liegt den Beobachtungen eine bezüglich obiger Bijektion der Bandinhalte gleiche Wahrscheinlichkeitsverteilung zugrunde. $\square$

Kapitel 4

Die Universelle QTM und Boosting

4.1 Spezielle Codierungen

Aufgrund der Definition von Quantenturingmaschinen ist es naheliegend, sie zu codieren durch die aus dem klassischen Fall bekannte Codierung der DTM, die die Übergangsamplituden auf n Binärziffern genau in $Poly(n)$ Zeit berechnet. Eine solche Codierung nennen wir DTM-Codierung einer QTM.

Definition 4.1.1. *Sei M eine unidirektionale QTM mit Alphabet Σ und Zustandsmenge Q. Das Blank-Symbol erhalte die Nummer 1, der Startzustand die Nummer 1 und der Endzustand die Nummer 2. Die anderen Bandsymbole seien von 2 bis $|\Sigma|$ durchnummeriert und die anderen Zustände seien von 3 bis $|Q|$ durchnummeriert. Die DTM-Codierung der QTM M, $\mathcal{G}(\mathbf{M}) \in \{0,1,2,3\}^*$, sei folgender String:*

$1^{|\Sigma|}\,3$

Binärcodierung von $|Q|$, Trennsymbol 3

Codierung über $\{0,1,2\}$ der DTM, die in $Poly(n)$ Zeit die Übergangsamplituden unter Vernachlässigung der Bewegungsrichtung auf n Binärziffern genau berechnet.

Trennsymbol 3

String der Länge $|Q|$ über $\{0,1\}$, in dem die j-te Stelle die eindeutige Bewegungsrichtung codiert, mit der Zustand q_j nur betreten werden kann, wobei die 0 L codiert und die 1 R codiert.

Trennsymbol 3

Liste der binär codierten Nummern aller Symbolen, die eine 1 in der letzten Spur aufweisen, durch das Symbol 2 voneinander getrennt. Diese Liste nennen wir auch Akzeptanzliste.

Beachte, daß es eine Universelle DTM gibt, die mit quadratischer Verlangsamung die als Codierung vorliegende DTM ausführt (siehe [Rei90]).

Die unäre Codierung von $|\Sigma|$ garantiert, daß die Dimension der Turingtafel polynomiell in der Länge der Codierung der QTM ist, auch wenn die Akzeptanzliste sehr klein ist.[1] Bei binärer Codierung könnten extrem große Alphabete Σ dazu führen, daß eine Universelle Maschine die Turingtafel zur Eingabe $\mathcal{G}(M)$ nicht einmal auf konstant viele Binärziffern genau in Polynomialzeit aufschreiben kann.

Beachte, daß gemäß dem Synchronisationstheorem eine reversible Polynomialzeit-TM existiert, die zur Eingabe einer Codierung x, aus der Unärcodierung von $|\Sigma|$ ihre Binärcodierung berechnet. Durch voranstellen hinreichend vieler Nullen in der Ausgabe, so daß ihre Länge insgesamt $\lceil \log_2 |x| \rceil$ ist, erreichen wir, daß ihre Laufzeit nur von $|x|$ abhängt.

Die nun definierte Sprache stellt die spezielle Wahl des eingeschränkten Grundbereichs für die Universelle Maschine dar:

Definition 4.1.2. *Sei $\mathcal{G}$ die Menge aller $\mathcal{G}(M)$ für solche unidirektionale, wohlgeformte, stationäre QTMs M in Normalform, die die Eingabe $\mathcal{G}(M)$ mit einer Wahrscheinlichkeit in $[0, \frac{1}{3}] \cup [\frac{2}{3}, 1]$ akzeptieren.*[2]

Beachte, daß insbesondere für jede QTM M die $L \in BQTime^*(T(n))$ für ein $L \subseteq (\Sigma_1 \setminus \#)^*$ und ein $T(\cdot) : \mathbb{N} \to \mathbb{N}$ bezeugt, $\mathcal{G}(M) \in \mathcal{G}$ gilt. Die Einschränkung des Grundbereichs läßt also nichttriviale Eingaben zu, nämlich insbesondere Codierungen von Maschinen, deren mögliche Akzeptanzwahrscheinlichkeiten in $[0, \frac{1}{3}] \cup [\frac{2}{3}, 1]$ liegen.

4.2 Boosting

Fakt 4.2.1 (Chernoff-Schranken). *Sei $k \geq 1$ und seien $X_1, \ldots, X_k$ unabhängige Bernoullivariablen mit $Pr[X_i = 1] = p$, $Pr[X_i = 0] = 1 - p$. Sei $S_k := \sum_{i=1}^{k} X_i$ dann ist der Erwartungswert $E[S_k] = \sum_{i=1}^{k} E[X_i] = pk$ und für $0 \leq \gamma \leq 1$ gilt:*

$$Pr[S_k > (p + \gamma)k] \leq e^{-2k\gamma^2}$$
$$Pr[S_k < (p - \gamma)k] \leq e^{-2k\gamma^2}$$

[1] Aus der syntaktischen Korrektheit folgt noch nicht, daß die von einer Codierung dargestellte QTM irgend welche Eigenschaften außer Unidirektionalität besitzt.

[2] Insbesondere *hält* eine solche QTM M auf Eingabe $\mathcal{G}(M)$, und zu jedem Zeitpunkt vor dem Halten befindet sich keine Endkonfiguration in der aktuellen Superposition.

Definition 4.2.2. *Eine Funktion $f : \mathbb{N} \to \mathbb{N}$ heißt* **zeitkonstruierbar**, *wenn eine $f(n)$-zeitbeschränkte 2-Band-DTM existiert, die die Binärcodierung von $f(n)$ aus dem Eingabestring 1^n berechnet.*[3]

Bennett, Bernstein, Brassard und Vazirani haben in [BBBV97] gezeigt, daß die genaue Wahl der Schranke α in $BQP_{G,\alpha}$ in folgendem Sinn willkürlich ist:

Theorem 4.2.3 (Boosting).
Seien $G \subseteq (\Sigma_1 \setminus \#)^*$, $T : \mathbb{N} \to \mathbb{N}$ mit $T(n) > n$ zeitkonstruierbar sowie $\varepsilon > 0$. Dann existiert ein Polynom $P(\cdot, \cdot)$ so daß:

$$BQTime_{G,\frac{2}{3}}(T(n)) \quad \subseteq \quad BQTime_{G,1-\varepsilon}(P(T(n), \log 1/\varepsilon))$$

Beweis. [BBBV97] □

Die Schwierigkeit, das Polynom $P(\cdot, \cdot)$ genauer zu bestimmen, liegt darin, daß eine Abschätzung der Laufzeit der verwendeten deterministischen Teilberechnungen noch nicht genügt, sondern wir auch die Verlangsamung, die aus der Anwendung des Synchronisationstheorems resultiert, kennen müßten.

Wir präzisieren den Beweis und zeigen mit der gleichen Beweistechnik:

Satz 4.2.4 (Boosting).
Seien $G \subseteq (\Sigma_1 \setminus \#)^*$, $T : \mathbb{N} \to \mathbb{N}$ mit $T(n) > n$ zeitkonstruierbar sowie $0 < \varepsilon < \frac{2}{3} - \frac{1}{2} = \frac{1}{6}$. Dann existiert ein Polynom $P(\cdot, \cdot)$ so daß:

$$BQTime_{G,\frac{2}{3}-\varepsilon}(T(n)) \quad \subseteq \quad BQTime_{G,\frac{2}{3}}(P(T(n), 1/\varepsilon))$$

Beweis. Die Idee ist, eine QTM zu konstruieren, die unabhängig voneinander k mal die QTM M auf der Eingabe x laufen läßt und dann das Majoritätsvotum der k Akzeptanzentscheidungen berechnet.

Sei M eine wohlgeformte, stationäre QTM in Normalform die $L \in BQTime_{G,\frac{2}{3}-\varepsilon}(T(n))$ bezeugt.
Sei $k' := \lceil \frac{\ln 3}{2(\frac{1}{6}-\varepsilon)^2} \rceil$ und $k := \begin{cases} k & \text{falls } k' \text{ ungerade} \\ k+1 & \text{falls } k' \text{ gerade} \end{cases}$
Wir definieren eine wohlgeformte, stationäre QTM in Normalform M' durch die folgenden drei Meta-Schritte. $x \in (\Sigma_1 \setminus \#)^*$ bezeichne die Eingabe.

(1) Benutze drei separate Arbeitsspuren und schreibe darauf die Binärcodierung von k, die Binärcodierung von $2m := 2T(|x|)$, und den String $(x\#^{2m-|x|})^{k-1}x$

[3] Jede Berechnung einer 2-Band DTM läßt sich mit quadratischer Verlangsamung auch von einer 1-Band DTM durchführen, siehe [Rei90].

(2) Wiederhole k mal eine Maschine die erst M laufen läßt und anschließend $2m$ Schritte nach rechts geht. Anschließend gehe wieder auf die Startzelle zurück.

(3) Berechne die Majorität der k Ergebnisse aus den Läufen von M und schreibe sie in die Startzelle. Das heißt, schreibe eine 1, wenn die Anzahl der Einsen mindestens $\lceil \frac{k}{2} \rceil$ beträgt, und eine 0 sonst.

Meta-Schritt 1 beschreibt eine in deterministischer Polynomialzeit berechenbare Funktion vom Eingabestring x auf einen dreispurigen Ausgabestring, wobei die Länge des Ergebnisses nur von k (impliziter Parameter) und $|x|$ abhängt. Gemäß Synchronisationstheorem 2.4.4 können wir diesen Meta-Schritt also mittels einer stationärer Polyomialzeit-QTM in Normalform ausführen, deren Laufzeit nur von k und $|x|$ abhängt. Beachte, daß die Endsuperposition bei Eingabe x nur aus einer Konfiguration besteht.

Für Meta-Schritt 2 benutzen wir zunächst eine stationäre Polynomialzeit-QTM in Normalform, die die Binärcodierungen von k und $2m$ auf den Arbeitsspuren um eine Position nach rechts verschiebt, wenn sie mit dem Kopf auf deren linkester Position gestartet wurde. Ihre Laufzeit hängt gemäß Synchronisationstheorem 2.4.4 nur von den Längen der Codierungen von k und $2m$, also nur von k und $|x|$, ab.

Wir fügen ans Ende dieser QTM einen Schritt an, der den Kopf nach rechts bewegt, dabei vom alten Endzustand in einen neuen Endzustand übergeht und den Bandinhalt nicht verändert. Wegen der Normalform der ursprünglichen QTM erhält dies die Wohlgeformtheit gemäß Verkettungslemma 2.5.6 und die resultierende QTM ist wohlbenehmend, wobei der Kopf nach dem Halten eine Zelle rechts neben der Startzelle, auf der linkesten Stelle der Codierungen von k und $2m$, steht.

Aufgrund dieser Eigenschaft funktioniert die Konstruktion des Looping Lemmas 2.5.10 angewandt auf diese nicht-stationäre QTM, für $2m$ Wiederholungen. Diese Anzahl bezieht die reversible DTM von der entsprechenden Hilfsspur als Eingabe. Dabei wird diese Spur von der reversiblen DTM nicht verändert, sondern nur von der eingefügten, zu wiederholenden QTM. Wir erhalten eine wohlbenehmende QTM in Normalform, die sich $2m$ Schritte nach rechts bewegt und dabei die Codierungen von k und $2m$ mit sich schleppt.

Die Lemmata 2.5.1 und 2.5.2 erlauben uns, eine Maschine, die dies leistet, hinter M gemäß Lemma 2.5.6 zu verketten und eine wohlgeformte wohlbenehmende QTM in Normalform zu erhalten.

Wir verwenden wieder die Konstruktion des Looping Lemmas 2.5.10 um eine QTM in Normalform zu erhalten, die M auf jeder der k Kopien der

Eingabe x laufen läßt. Dies ist möglich, obwohl die zu wiederholende QTM nicht stationär ist, da sie wohlbenehmend ist und der Kopf nach dem Halten gewissermaßen auf der "Startzelle" der rechts benachbarten Eingabekopie steht, wobei die Codierungen von k und m auf den beiden Arbeitsspuren ebenfalls so verschoben wurden, daß sie an dieser Kopfposition beginnen. Anschaulich gesprochen merkt die reversible DTM also den Unterschied gar nicht.

Zuletzt verketten wir diese Maschine mit einer Maschine, die analog zur bisherigen Konstruktion k mal eine Maschine wiederholt, welche wiederum $2m$ mal eine Maschine wiederholt, die die Codierungen von k und $2m$ einen Schritt mit sich nach links schleppt.

Die resultierende QTM ist dann insbesondere wieder stationär. Beachte auch, daß M stets nur auf Superpositionen gestartet wird, bei der der Bandbereich den sie während der Rechnung besucht, eindeutig ist. Das heißt in jeder Konfiguration stehen die gleichen Symbole (x und Codierungen von k und $2m$). Außerdem ist der aktuelle Zustand sowie die Kopfposition beim Ein- *und* Austritt aus Meta-Schritt 2 in diesem Sinn eindeutig.

Meta-Schritt 3 stellt eine Funktion von einem dreispurigen Eingabestring auf einen Ausgabestring der Länge 1 dar, die von einer Polynomialzeit-TM berechnet werden kann. Die Eingabe ist x und der gemeinsame Ausgabestring der Länge $2kT(|x|)$ der k Rechnungen von M aus den Meta-Schritten 1 bis 3. Die DTM kann zunächst m aus x berechnen und dann mittels k und m leicht die relevanten Positionen bestimmen, daraus die Majorität ermitteln und diese auf die Startzelle schreiben. Ihre Laufzeit ist polynomiell in der Eingabelänge und somit in k und $|x|$ beschränkt. Ihre Ausgabelänge ist sogar konstant. Daher existiert gemäß dem Synchronisationstheorem eine wohlgeformte, stationäre QTM in Normalform deren Laufzeit ein Polynom in k und $|x|$ ist und die Meta-Schritt 3 realisiert.

Die Berechnungspfade zu allen sich überlagernden Eingabebandinhalten terminieren also nach exakt der gleichen Anzahl Schritten mit dem Kopf auf der Startzelle in einem gemeinsamen Endzustand und können somit wie gewünscht interferieren.

Wegen der Lemmata 2.5.1 und 2.5.2 gibt es zu den drei Meta-Schritten wohlgeformte, stationäre Polynomialzeit-QTMs in Normalform mit gleichem Alphabet und wir können diese mittels Lemma 2.5.6 zu einer wohlgeformten QTM M' in Normalform verketten, deren Laufzeit durch $P(T(n), 1/\varepsilon)$ beschränkt ist, für ein geeignetes Polynom $P(\cdot, \cdot)$, welches sich aus den einzelnen Konstruktionsschritten ergibt.

M' ist auch stationär, denn die drei einzelnen QTMs waren stationär und ihre Laufzeit war für jede sich überlagernde Eingabe gleich.

Analysieren wir nun das Akzeptanzverhalten von M':

Für $x \notin G$ ist die Akzeptanzwahrscheinlichkeit irrelevant.
Sei $x \in G$, dann ist $p(x) := Pr[M \,\text{akzeptiert}\, x] \in [0, \frac{1}{3} + \varepsilon] \cup [\frac{2}{3} - \varepsilon, 1]$.

Sei $\tilde{p}(x)$ die Wahrscheinlichkeit mit der M' die Eingabe x akzeptiert.

Fixiere eine Eingabe x. Sei c die zufällig durch die Beobachtung nach dem Halten gewählte Konfiguration aus der Endsuperposition der Rechnung von M' auf Eingabe x.

Sei S_k die Anzahl der Einsen unter den Ergebnissen der k unabhängigen Läufe von M in der Konfiguration c und beachte daß k ungerade ist.

Dann gilt aufgrund des Majoritätsvotums $\tilde{p}(x) = Pr[S_k > k/2]$, wobei das Zufallsexperiment die Beobachtung nach dem Halten ist.
Falls $p(x) \leq \frac{1}{3} + \varepsilon$ so ist $x \in G \setminus L$ und aus der ersten Chernoff-Schranke folgt:

$$\tilde{p}(x) = Pr[S_k > \underbrace{\frac{1}{2} \cdot k}_{\Rightarrow \gamma = \frac{1}{2} - p \in [\frac{1}{6} - \varepsilon, \frac{1}{2}]}] \leq e^{-2k(\frac{1}{2}-p)^2} \leq e^{-2k(\frac{1}{6}-\varepsilon)^2}$$

$$\leq e^{-2 \cdot \frac{\ln 3}{2(\frac{1}{6}-\varepsilon)^2} \cdot (\frac{1}{6}-\varepsilon)^2} \leq e^{-\ln 3} = \frac{1}{3}$$

Falls $p(x) \geq \frac{2}{3} - \varepsilon$ so ist $x \in G \cap L$ und aus der zweiten Chernoff-Schranke folgt:

$$1 - \tilde{p}(x) = Pr[S_k < \underbrace{\frac{1}{2} \cdot k}_{\Rightarrow \gamma = p - \frac{1}{2} \in [\frac{1}{6} - \varepsilon, \frac{1}{2}]}] \leq e^{-2k(p-\frac{1}{2})^2} \leq e^{-2k(\frac{1}{6}-\varepsilon)^2} \leq e^{-\ln 3} = \frac{1}{3}$$

Also bezeugt M' daß $L \in BQTime_{G,\frac{2}{3}}(P(T(n), 1/\varepsilon))$

$\square$

Beachte, daß wir nicht gezeigt haben, daß die Klassen $EQTime$ mit den Klassen $BQTime$ polynomiell verknüpft sind. Die Boosting-Technik mit polynomiell vielen Wiederholungen läßt sich in diesem Fall nicht anwenden, denn egal wie oft man eine QTM mit beschränktem Fehler wiederholt, wird das Majoritätsvotum immer mit einer, wenn auch extrem kleinen Wahrscheinlichkeit falsch entscheiden und daher nie eine exakte Klassifikation der Eingabe erlauben.

4.3 Zerlegung endlichdim. Transformationen

Wir zitieren nun einige Sätze und Definitionen aus [BV97], die für die Konstruktion einer Universellen Maschine verwendet werden. Die Beweise sind dort nachzulesen.

In diesem Abschnitt betrachten wir stets den *endlichdimensionalen* unitären Vektorraum $\mathbb{C}^d$. Wir wählen die orthonormale Standardbasis $e_1, \ldots, e_d$ und können jeden linearen Operator auf $\mathbb{C}^d$ bezüglich dieser Basis als Matrix darstellen.

Definition 4.3.1. *Für $\varepsilon < 1$ heißt ein $U \in \mathbb{C}^{d \times d}$* **$\varepsilon$-nah an unitär**, *genau dann wenn eine unitäre Matrix $\tilde{U} \in \mathbb{C}^{d \times d}$ existiert, so daß $||U - \tilde{U}|| \leq \varepsilon$.*

Wenn U ε-nah an unitär ist, so folgt $1 - \varepsilon \leq ||U|| \leq 1 + \varepsilon$, aber nicht umgekehrt, denn $\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ hat die Norm 1, aber Abstand 1 von jeder unitären Matrix. Man kann jedoch zeigen, daß wenn U nah an unitar ist, die Zeilen von U nahezu Einheitslänge haben und nahezu paarweise orthogonal sind.

Lemma 4.3.2. *Sei $U \in \mathbb{C}^{d \times d}$ und für alle $1 \leq i, j \leq d$ gelte $|u_{i,j}| \leq \varepsilon$, dann gilt $||U|| \leq d\varepsilon$.*

Definition 4.3.3. *$U \in \mathbb{C}^{d \times d}$ heißt* **nahezu trivial**, *genau dann wenn eine der beiden folgenden Bedingungen erfüllt ist:*

(1) *U ist die Identität außer daß ein Diagonalelement gleich $e^{i\theta}$ ist für ein $\theta \in [0, 2\pi]$*

(2) *U ist die Identität bis auf eine Untermatrix in zwei verschiedenen Dimensionen, die eine Drehung um einen Winkel $\theta \in [0, 2\pi]$ darstellt*

Im folgenden bezeichne m die Eingabelänge.

Theorem 4.3.4. *Es gibt eine $Poly(d, \log \frac{1}{\varepsilon}, m)$-zeitbeschränkte DTM, die zur Eingabe $\varepsilon > 0$ und einer Codierung einer $\frac{\varepsilon}{2(10\sqrt{d})^d}$-nah an unitären Matrix $U \in \mathbb{C}^{d \times d}$, nahezu triviale Matrizen $U_1, \ldots, U_n \in \mathbb{C}^{d \times d}$, für ein Polynom n in d, berechnet so daß $||U - U_n \cdot \ldots \cdot U_1|| \leq \varepsilon$.*

Lemma 4.3.5. *Sei $\mathcal{R} := 2\pi \sum_{i=1}^{\infty} 2^{-2^i} < \pi$. Es existiert eine $Poly(\log \frac{1}{\varepsilon}, m)$-zeitbeschränkte DTM, die zur Eingabe $\theta \in [0, 2\pi]$ und $\varepsilon > 0$ eine natürliche Zahl $k \leq Poly(\frac{1}{\varepsilon})$ berechnet, so daß $|k\mathcal{R} - \theta| \bmod 2\pi \leq \varepsilon$.*

Definition 4.3.6. *M sei eine $Poly(1/\varepsilon, m)$-zeitbeschränkte QTM mit Alphabet $\{\#, 0, 1\}$ auf der ersten Spur und die erste Spur enthalte zu jedem Zeitpunkt nur Strings aus $\{0,1\}^k$ ab der Startzelle und Blanks sonstwo. Sei $d := 2^k$. Dann läßt sich die Menge der Superpositionen der ersten Spur mit $\mathbb{C}^d$ identifizieren.*

Zu jedem $U \in \mathbb{C}^{d\times d}$ bezeichne x_U eine eventuell näherungsweise Codierung. Wir sagen die QTM M **führt jede Transformation aus $\mathbb{C}^{d\times d}$ in Polynomialzeit mit Nähefaktor $c < 1$ aus** $:\Leftrightarrow$

Für jeden Genauigkeitsparameter $\varepsilon > 0$ und jedes $c\varepsilon$-nah an unitäre $U \in \mathbb{C}^{d\times d}$ existiert eine durch ein Polynom in $\frac{1}{\varepsilon}$ und der Länge $|x_U|$ beschränkte natürliche Zahl T, so daß für alle $\phi \in \mathbb{C}^d$ gilt:

M hält auf der Eingabesuperposition $\phi; x_U; \varepsilon$ nach exakt T Schritten mit dem Kopf auf der Startzelle in der Endsuperposition $U'(\phi); x_U; \varepsilon$ für ein unitäres $U' \in \mathbb{C}^{d\times d}$ mit $||U - U'|| \leq \varepsilon$

Beachte, daß sich selbstverständlich ε endlich codieren lassen muß.

Anschaulich gesprochen ist der Nähefaktor gerade der Faktor, um den die Codierung der auszuführenden Transformation "genauer" sein muß als der maximal auftretende Fehler bei der Rechnung von M.

Theorem 4.3.7 (Unitary Transformation Theorem). *Es existiert eine wohlgeformte, stationäre QTM in Normalform mit Alphabet $\{\#, 0, 1\}$ auf der ersten Spur, die jede Transformation $U \in \mathbb{C}^{d\times d}$ auf ihrer ersten Spur in Polynomialzeit mit Nähefaktor $\frac{1}{2(10\sqrt{d})^d}$ ausführt.*

Die Beweisidee ist, die als eine hinreichend genaue Codierung näherungsweise gegebene, endlichdimensionale, unitäre Transformation U mit Theorem 4.3.4 zu zerlegen in polynomiell viele, unitäre, nahezu triviale Matrizen, deren Produkt hinreichend nah an U ist. Und in einem zweiten Schritt diese nahezu trivialen Matrizen mit Lemma 4.3.5 näherungsweise zu zerlegen in ein Produkt polynomiell vieler trivialer Matrizen und Permutationsmatrizen. Die trivialen Matrizen sind dabei nahezu triviale Matrizen mit $\theta = \mathcal{R}$ und stellen die einzigen echten Quantenoperationen der Universellen QTM dar.

Der Aufwand für diese Zerlegung und die Zahl der Faktoren ist insgesamt polynomiell in d und $1/\varepsilon$.[4]

In dieser Konstruktion steckt die Hauptarbeit, auf dem Weg, die Existenz einer effizienten Universellen QTM zu beweisen.

[4]Bei unserer DTM-Codierung ist d durch ein Polynom in der Codierungslänge der gesamten QTM beschränkt.

4.4 Die Universelle Quantenturingmaschine

Unter Anwendung der Sätze aus dem vorigen Abschnitt haben [BV97] gezeigt:

Theorem 4.4.1.
Es gibt eine wohlgeformte, stationäre QTM in Normalform, die für die DTM-Codierung einer beliebigen wohlgeformten QTM M, jedes $\varepsilon > 0$, jeden String $x \in (\Sigma_1 \setminus \#)^*$ und jedes $T > 0$ als Eingabe, M auf Eingabe x mit Genauigkeit ε für T Schritte mit Verlangsamung $Poly(\cdot, 1/\varepsilon)$ simuliert.

Dabei wird zunächst mit Hilfe des Unidirection Lemmas zu einer wohlgeformten QTM eine unidirektionale QTM konstruiert, und diese dann simuliert. Ohne diesen Schritt könnten wir das Unitary Transformation Theorem nicht anwenden, denn die Zerlegung funktioniert nur für endlichdimensionale unitäre Transformationen.

Wir modifizieren dies, um eine Maschine M_U zu erhalten, die unseren Anforderungen im Hierarchiesatzbeweis genügt. Die Unterschiede sind:

(1) Die zu simulierenden Maschinen sind unidirektional.

(2) Die Eingabe ε ist implizit in M_U enthalten.

(3) Wenn die Eingabe x die Codierung einer QTM M_x ist, soll M_U M_x auf Eingabe x simulieren. M_U bekommt daher als Eingabe nur einmal x.

(4) Die Simulationsdauer ist implizit im M_U codiert als Funktion $T(n)$.

(5) M_U ist auf allen Eingaben aus $(\Sigma_1 \setminus \#)^*$ laufzeitbeschränkt und liefert für Eingaben aus einem geeignet gewählten Grundbereich, eine um höchstens ε abweichende Akzeptanzwahrscheinlichkeit gegenüber der Rechnung von M_x auf x.

(6) Für Eingaben x aus diesem Grundbereich soll M_U im Fall $T(|x|) > T^{M_x}(x)$ M_x auf x mit dem gleichen Akzeptanzverhalten simulieren, als wenn $T(|x|) = T^{M_x}(x)$ wäre.

Das Problem bei Punkt (6) ist, daß die Universelle QTM in Theorem 4.4.1, in einer Schleife genau $T := T(|x|)$ Simulationsschritte durchführt. Dadurch garantiert sie, daß die Simulation auch für "schlechte" Eingaben auf jeden Fall nach $T(|x|)$ Simulationsschritten terminiert (und sie selbst ihre Zeitschranke einhält).

Wenn wir innerhalb dieser Schleife auf Vorliegen eines Endzustands testen und gegebenenfalls die Schleife vorzeitig verlassen wollten, so ergäben sich folgende Probleme:

- Die QTM aus dem Looping Lemma 2.5.10 erlaubt keinen vorzeitigen Abbruch. Wir müßten die Existenz einer solchen QTM mit elementaren Methoden nachweisen.

- Wenn x eine QTM M_x codiert, die auf Eingabe x nicht hält, sondern innerhalb der ersten T Schritte eine Superposition erreicht, die sowohl Endkonfigurationen als auch Nicht-Endkonfigurationen enthält, dann verliert M_U die "Synchronität". Das heißt, die verschiedenen, sich überlagernden Berechnungspfade terminieren nicht zur gleichen Zeit.

Das zweite Problem ist das entscheidende, denn wir müssen unbedingt garantieren, daß M_U auf *allen* Eingaben hält, damit sie zeitbeschränkt ist.

Unsere Lösung wird sein, M_x mit einer "dummy"-QTM zu einer QTM M'_x zu verketten und dann M'_x zu simulieren. Die "dummy"-QTM wird eine hinreichend lange Rechnung durchführen, die den Bandinhalt nicht verändert. Sie dient nur dazu, die Rechnung hinreichend zu verlängern, so daß die "Akzeptanzwahrscheinlichkeit" der Eingabe x von M'_x nach $T(|x|)$ Schritten die gleiche ist, wie die von M_x nach $T^{M_x}(x) \leq T(|x|)$ Schritten.

Lemma 4.4.2 ("Nichtstuende" QTM). *Zu jedem Alphabet Σ mit einem ausgezeichneten Blanksymbol und jeder natürlichen Zahl $k > 0$ gibt es eine wohlgeformte, wohlbenehmende, unidirektionale QTM in Normalform $M^k_\Sigma = (\Sigma, \{\tilde{q}_0, \tilde{q}_1, \ldots, \tilde{q}_k\}, \delta)$ mit Startzustand $\tilde{q}_0$ und Endzustand $\tilde{q}_k$, deren Laufzeit für beliebige Eingaben durch k beschränkt ist und die den Bandinhalt nicht verändert.*

Beweis. Wähle folgende Übergangsfunktion:
Für alle Zustände p und Bandsymbole σ sei

$$\delta(p, \sigma, \tau, q, d) := \begin{cases} 1 & \text{falls } (\tau, q, d) = (\sigma, succ(p), R) \\ 0 & \text{sonst} \end{cases}, \text{ wobei}$$

$$succ(p) = \begin{cases} \tilde{q}_{i+1} & \text{falls } p = \tilde{q}_i \text{ für ein } 0 \leq i < k \\ \tilde{q}_0 & \text{falls } p = \tilde{q}_k \end{cases}.$$

Dann läßt sich die Wohlgeformtheit leicht anhand der lokalen Bedingungen 2.6.1 nachrechnen und die QTM erfüllt offensichtlich die genannten Eigenschaften. □

Es ist unklar, wie eine wohlgeformte QTM, die einfach eine Endlosschleife durchführt, definiert werden soll, so daß sie in Normalform ist. Und es ist unklar, wie eine QTM die nicht in Normalform ist, hinter eine wohlgeformte

QTM in Normalform verkettet werden kann, so daß die resultierende QTM wohlgeformt ist. Daher scheidet diese naheliegende Wahl der "dummy"-QTM aus.

Für uns ist es wichtig, daß offensichtlich eine DTM existiert, die zur Eingabe $|\Sigma|$ und k, die Codierung von M_Σ^k, in Form der Turingtafel und des Richtungsstrings gemäß Definition 2.6.2, effizient berechnet.

Das Hauptresultat dieses Abschnitts ist:

Theorem 4.4.3. *Für jeden Genauigkeitsparameter $\varepsilon > 0$ und jede zeitkonstruierbare, polynomiell beschränkte Simulationsdauer $T : \mathbb{N} \to \mathbb{N}$ mit $T(n) > n$, gibt es ein Polynom $P(\cdot, \cdot)$ sowie eine wohlgeformte, $P(T(n), 1/\varepsilon)$-zeitbeschränkte, stationäre QTM in Normalform M_U, die für jede Codierung $x \in \mathcal{G}$ als Eingabe folgende Akzeptanzwahrscheinlichkeit realisiert:*

Falls die von x codierte Maschine M_x auf Eingabe x nicht in $T(|x|)$ Schritten hält, so verwirft M_U die Eingabe x mit einer Wahrscheinlichkeit von mindestens $1 - \varepsilon$.

Falls M_x auf Eingabe x in höchstens $T(|x|)$ Schritten hält, so bezeichne $p(x)$ die Wahrscheinlichkeit, mit der M_x x akzeptiert. Dann weicht die Wahrscheinlichkeit, mit der M_U x akzeptiert, von $p(x)$ um höchstens ε ab.

Beweis. In Anlehnung an den Beweis des Theorems 4.4.1 konstruieren wir die QTM M_U wie folgt:

Da wir nur unidirektionale QTMs simulieren wollen, brauchen wir die Konstruktion einer unidirektionalen QTM aus einer wohlgeformten QTM, wie dies in einer ersten Phase der QTM in Theorem 4.4.1 erforderlich war, nicht durchzuführen.

Wir müssen jedoch die impliziten Eingaben berechnen, die in Theorem 4.4.1 noch explizit waren. Außerdem müssen wir die Eingabe auf syntaktische Korrektheit testen. Wenn die Eingabe syntaktisch korrekt ist, müssen wir die Rechnung der davon codierten, zu simulierenden QTM M_x so verlängern, daß wir, falls $T^{M_x}(x) < T(|x|)$ ist, ohne die Wahrscheinlichkeit mit der x akzeptiert wird zu verändern, genau $T(|x|)$ Schritte simulieren können.

Angenommen x codiert eine QTM M_x, dann sei die *Superzellenrepräsentation* einer Konfiguration (und als Überlagerung dann auch der Superposition) der zu simulierenden QTM M_x wie folgt:

Wir verwenden eine Spur der Universellen QTM, um die jeweiligen Konfigurationen von M_x zu simulieren.

Da das Alphabet und die Zustandsmenge von M_x eine beliebige endliche Größe haben können, benutzen wir für jede Zelle von M_x ein Tupel von

$\lceil \log_2 (|Q|+1) \rceil \cdot \lceil \log_2 |\Sigma| \rceil$ Zellen der Universellen QTM, genannt Superzelle, um sie zu simulieren.

Jede Superzelle enthält ein Paar (p, σ), wobei $\sigma \in [1, |\Sigma|]$ den Inhalt der entsprechenden Zelle von M_x repräsentiert und $p \in [0, |Q|]$ den Zustand von M_x repräsentiert, wenn der Kopf von M_x gerade auf dieser Zelle steht, und $p = 0$ sonst.

Da sich der Kopf von M_x in $T(|x|)$ Schritten höchstens um $T(|x|)$ von der Startzelle entfernen kann, benötigen wir zur Simulation nur Superzellen für die relevanten $2T(|x|) + 1$ Zellen des Bandes von M_x. Die Enden seien geeignet markiert.

M_U setzt sich aus einer vorbereitenden Phase und der eigentlichen Simulationsphase zusammen.

Die vorbereitende Phase setzt sich aus einer Bedingten Verzweigung nach der deterministischen Überprüfung der Syntaktischen Korrektheit, und 7 weiteren Meta-Schritten zusammen. Diese 7 Meta-Schritte führen allesamt nur deterministische Berechnungen durch und halten in einer Endsuperposition mit nur einer Konfiguration.

Die Übergangsfunktion einer wohlgeformten, unidirektionalen QTM liefert gemäß Korollar 2.6.2, wenn wir die Bewegungsrichtung ignorieren, eine unitäre Transformation der Dimension $|Q \times \Sigma|$ von Superpositionen des aktuellen Zustands und des aktuellen Bandsymbols auf Superpositionen des neuen Zustands und des neuen aktuellen Bandsymbols.

Daher können wir zur eigentlichen Simulation das Unitary Transformation Theorem 4.3.7 anwenden. Die Simulationsphase ist identisch mit der im Beweis von Theorem 4.4.1.

Die Meta-Schritte von M_U sind im einzelnen:

(1) Überprüfe zunächst, ob x eine syntaktisch korrekte Codierung ist und verwerfe andernfalls.

Bezeichne im folgenden die durch x codierte unidirektionale QTM mit M_x.

Es folgen 7 deterministische Meta-Schritte:

(2) Schreibe die Binärcodierungen von $|\Sigma|$ und $|Q|$ aus der Codierung x auf je eine Hilfsspur.

(3) Berechne $T(|x|)$ aus x und schreibe das Ergebnis auf eine Hilfsspur.

(4) Berechne $d := |\Sigma| \cdot (|Q| + T(|x|))$ und schreibe es auf eine Hilfsspur.

(5) Verwende die DTM aus der Codierung von x und berechne die Turingtafel von M_x mit Einträgen auf mindestens $\frac{\varepsilon}{8T(|x|)(10\sqrt{d})^{d+2}}$ genau.

Die von dieser Turingtafel zusammen mit dem Richtungsstring aus der Codierung x dargestellte QTM bezeichne im folgenden mit $M^{(1)}$.

(6) Berechne die Turingtafel und den Richtungsstring von $M_{\Sigma}^{T(|x|)}$ für das $|\Sigma|$ aus der Codierung x.

Berechne die Turingtafel und den Richtungsstring, die die Verkettung von $M^{(1)}$ mit $M_{\Sigma}^{T(|x|)}$ codieren, gemäß der Konstruktion aus Lemma 2.5.6, und schreibe sie auf jeweils eine Hilfsspur. Beachte, daß bei der Verkettung die Unidirektionalität erhalten bleibt.

Die so entstandene Transformation bezeichnen wir im folgenden mit U', den so entstandenen Richtungsstring mit s', und die davon dargestellte QTM mit $M^{(2)}$.

(7) Schreibe auf eine weitere Hilfsspur $\gamma := \frac{\varepsilon}{8T(|x|)}$.

(8) Berechne aus x die analog zu oben definierte Superzellenrepräsentation der Startkonfiguration von $M^{(2)}$ auf Eingabe x.[5]

An diese Vorbereitungsphase schließt sich die Simulationsphase an:

(9) Benutze die Maschine aus dem Looping Lemma 2.5.10, um $T(|x|)$ mal die folgende QTM zu wiederholen:

- Bewege den aktuellen Zustand und das aktuelle Symbol der zu simulierenden Maschine $p; \sigma$ auf einen leeren Arbeitsbereich einer Hilfsspur, der ab der Startzelle von M_U beginnt, und hinterlasse eine spezielle Markierung.
- Wende mittels der Maschine aus dem Unitary Transformation Theorem 4.3.7 die oben konstruierte Transformation U' auf $p; \sigma$ an und erhalte eine Superposition von neuem Zustand und neuem Symbol $q; \tau$. Der Genauigkeitsparameter sei dabei obiges γ.
- Kehre den vorletzten Schritt um, indem die so entstandene Superposition zurück auf die markierte Superzelle bewegt wird (und der Arbeitsbereich wieder geleert wird).

[5]Das heißt, x ist die Eingabe für die zu simulierende QTM.

- Bewege die Zustandscodierung q eine Superzelle nach links oder rechts, je nachdem, welche Richtung das q-te Bit des neuen Richtungsstrings s' anzeigt.

(10) Untersuche, ob der aktuelle Zustand der simulierten Maschine ein Zustand $\tilde{q}_i$ der angefügten QTM $M_\Sigma^{T(|x|)}$ ist, und schreibe andernfalls[6] eine 0 auf die letzte Spur der Startzelle von M_U.

Andernfalls untersuche, ob der Teil der Simulationsspur, der den Bandinhalt der simulierten Startzelle codiert, ein Symbol aus der Akzeptanzliste der Codierung x enthält, und schreibe eine 1 falls ja und eine 0 falls nein.

Beachte, daß für $x \in \mathcal{G}$ sowohl M_x, als auch die Verkettung von M_x mit $M_\Sigma^{T(|x|)}$ wohlgeformt, wohlbenehmend und in Normalform sind. $M^{(1)}$ und $M^{(2)}$ sind jedoch nur Näherungen, die im allgemeinen noch nicht einmal wohlgeformt sind. Zudem wird $M^{(2)}$ mit der QTM aus dem Unitary Transformation Theorem im allgemeinen auch nicht fehlerfrei simuliert.

Die für die *Laufzeit* $T^{M_x}(x)$ möglichen Fälle illustriert Abbildung 4.1:

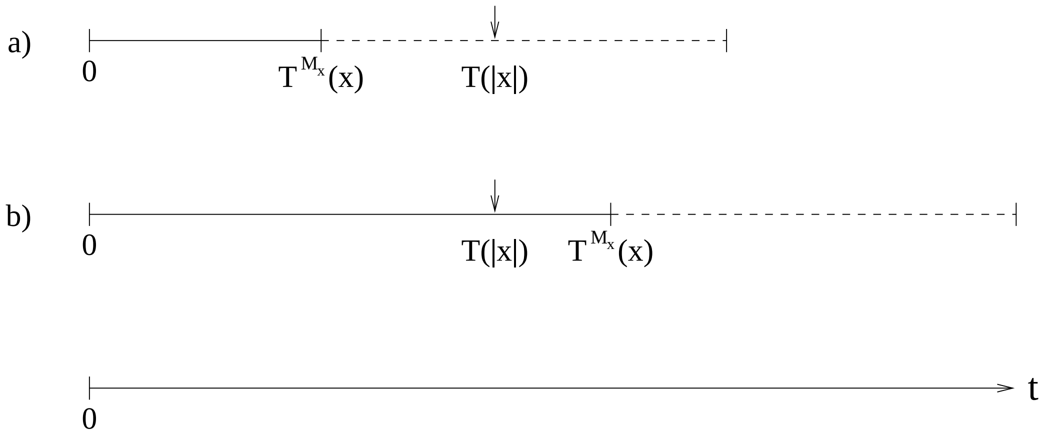

Abbildung 4.1: Haltezeitpunkte von M_x auf Eingabe x

Wir möchten eigentlich M_x auf x für höchstens $T(|x|)$ Schritte simulieren und dabei verwerfen, falls M_x in dieser Zeit nicht hält. Andernfalls möchten wir die Wahrscheinlichkeit $p(x)$, mit der M_x x akzeptiert, bis auf ε genau bestimmen.

[6] Das heißt, falls M_x noch nicht gehalten hat

Da unsere Konstruktion aus den oben dargestellten Gründen jedoch immer *genau* $T(|x|)$ Schritte simuliert, verketten wir die "Nichtstuende" QTM (dargestellt durch das gestrichelte Zeitintervall) hinter M_x. Sie verändert nicht die Wahrscheinlichkeit, mit der wir eine 1 auf der letzten Spur der Startzelle beobachten. Wir erhalten die QTM $M^{(2)}$, deren frühestmöglicher Haltezeitpunkt hinter $T(|x|)$ liegt.

Falls M_x rechtzeitig hält (Fall a)), so akzeptiert die $T(|x|)$-Schritte-Simulation von $M^{(2)}$ die Eingabe x, mit einer Wahrscheinlichkeit, die von $p(x)$ um höchstens ε abweicht.

Falls M_x nicht rechtzeitig hält (Fall b)), so erkennt dies Meta-Schritt 10, mit Genauigkeit ε, am aktuellen Zustand. Daher verwerfen wir mit Wahrscheinlichkeit $1-\varepsilon$.

Wir gehen nun auf die einzelnen Meta-Schritte näher ein:

ad 1) Gemäß Synchronisationstheorem[7] existiert eine wohlgeformte, stationäre Polynomialzeit-QTM M_1 in Normalform, die die syntaktische Korrektheit der Eingabe x überprüft und entsprechend die Ausgabe 0 oder 1 produziert.

Bezeichne mit M^{Rest} die wohlgeformte, stationäre Polynomialzeit-QTM in Normalform, die die Meta-Schritte 2 bis 10 realisiert, deren Existenz wir im folgenden noch zeigen werden.

Dann liefert uns das Verzweigungslemma 2.5.7 die Existenz einer wohlgeformten, wohlbenehmenden QTM M^V in Normalform, die M^{Rest} bedingt ausführt. Verkettungslemma 2.5.6 liefert uns dann M_U als Verkettung von M^V hinter M_1, wenn wir die Alphabete gemäß den Lemmata 2.5.1 und 2.5.2 angeglichen haben.

ad 2,3,4) Wir können das Synchronisationstheorem anwenden, wenn wir die Ausgaben mit führenden Nullen ergänzen, so daß ihre Länge nur von $|x|$ abhängt. So ergänzen wir die Binärcodierungen von $|\Sigma|$ und $|Q|$ auf je $\lceil \log_2 |x| \rceil$ Stellen und die Binärcodierung von d auf $l(|x|) := \lceil \log_2 |x| \rceil + \lceil \log_2 (|x| + T(|x|)) \rceil$ Stellen. Die Länge der Codierung von $T(|x|)$ hängt ohnehin nur von $|x|$ ab.

ad 5) Benutze die Universelle DTM, die zur Eingabe einer Codierung einer DTM und eines Strings z, diese DTM auf Eingabe z fehlerfrei mit polynomieller Verlangsamung simuliert.

[7]Die Verlangsamung durch die Anwendung des Synchronisationstheorems ist höchstens polynomiell

Erzeuge damit die Turingtafel und verwende für die einzelnen Einträge ein Fixkommaformat mit nur von $|x|$ abhängiger Länge, die polynomiell in $|x|$ ist, so daß die darstellbare Genauigkeit mindestens so gut wie die geforderte Genauigkeit ist. Nennen wir die daraus resultierende Codierungslänge einer Amplitude $\tilde{l}(|x|)$, so ergänzen wir die gesamte Ausgabe um so viele Nullen, daß ihre Gesamtlänge $L(|x|) := \tilde{l}(|x|) \cdot 2^{2 \cdot l(|x|)}$ ist.

Dann hängt die Länge der gesamten Ausgabe nur von $|x|$ ab und ist durch ein Polynom in $|x|$ beschränkt, so daß das Synchronisationstheorem die Existenz der gewünschten QTM liefert.

ad 6) Die Länge der Codierung der Turingtafel U' ist offensichtlich beschränkt durch $L(|x|)$, denn ihre Dimension ist gerade d. Wir fügen wieder genau so viele Nullen hinzu, daß die Ausgabe der DTM, die U' berechnet, die Länge $L(|x|)$ hat. Der Richtungsstring verlängert sich durch die Verkettung genau um $T(|x|)$.

Daher hängt die gesamte Ausgabelänge nur von $|x|$ ab und ist durch ein Polynom in $|x|$ beschränkt.

Offensichtlich lassen sich die gewünschten Berechnungen deterministisch in polynomiell in $|x|$ viel Zeit durchführen und das Synchronisationstheorem liefert die Existenz der gewünschten QTM.

Beachte, daß sich die Genauigkeit der Codierung der Amplituden durch diese Manipulation nicht verschlechtert.

Die QTM $M^{(2)}$ besitzt für $x \in \mathcal{G}$ folgende Eigenschaft, wie sich leicht mit Vollständiger Induktion zeigen läßt:

Sei $T^{M_x(x)} \leq T \leq T(|x|)$ ein beliebiger Zeitpunkt.

Sei C_1 die Menge der Konfigurationen mit Amplitude ungleich Null aus der Superposition von $M^{(1)}$ auf Eingabe x zur Zeit $T^{M_x}(x)$.

Sei C_2 die Menge der Konfigurationen mit Amplitude ungleich Null aus der Superposition von $M^{(2)}$ auf Eingabe x zur Zeit T.

Dann sind C_1, C_2 endlich, und es existiert eine Bijektion λ zwischen C_1 und C_2, so daß für jede Konfiguration c in C_1 eine Konfiguration $c' = \lambda(c)$ in C_2 existiert, für die gilt:

- Die Amplitude von c' in der Superposition von $M^{(2)}$ auf x zur Zeit T ist gleich der Amplitude von c in der Superposition von $M^{(1)}$ auf x zur Zeit $T^{M_x}(x)$.

- Die Bandinhalte von c und c' stimmen überein.[8]

[8] Beachte, daß gemäß Konstruktion die Alphabete von $M^{(2)}$ und $M^{(1)}$ gleich sind.

Insbesondere folgt, daß für $T^{M_x}(x) \leq T(|x|)$ die Wahrscheinlichkeit, in der letzten Spur der Startzelle bei einer Beobachtung von $M^{(2)}$ auf x zur Zeit $T(|x|)$ eine 1 zu sehen, gleich der Wahrscheinlichkeit, in der letzten Spur der Startzelle bei einer Beobachtung von $M^{(1)}$ auf x zur Zeit $T^{M_x}(x)$ eine 1 zu sehen, ist.

$M^{(2)}$ ist also für Eingaben aus $\mathcal{G}$ eine genauso gute Näherung an die Akzeptanzwahrscheinlichkeit von M_x wie es $M^{(1)}$ war. Der Unterschied liegt darin, daß wir $M^{(2)}$ in jedem Fall exakt $T(x)$ Schritte simulieren lassen, und dann beobachten.

ad 7) Wir verwenden wieder hinreichend viele führende Nullen, so daß die Länge der Codierung von γ nur von $|x|$ abhängt, und wenden das Synchronisationstheorem an.

ad 8) Wir ergänzen die Superzellenrepräsentation außerhalb der $2T(|x|)+1$ Superzellen um hinreichend viele Nullen, so daß die Gesamtlänge nur von $|x|$ abhängt, und wenden das Synchronisationstheorem an. Die Gesamtlänge läßt sich dabei wieder leicht durch Abschätzen von $|\Sigma|$ und $|Q|$ durch $|x|$ berechnen.

Beachte, daß sämtliche QTMs zu den Meta-Schritten 2 bis 8 wohlgeformt, stationär und in Normalform sind. Gemäß den Lemmata 2.5.1 und 2.5.2 können wir annehmen, daß sie außerdem ein gemeinsames Alphabet haben.

Gemäß Verkettungslemma 2.5.6 lassen sie sich daher zu einer wohlgeformten QTM in Normalform verketten.

Diese QTM führt effektiv nur eine deterministische Berechnung durch und hält in Polynomialzeit, angesetzt auf die Eingabe x, in einer Endsuperposition, die nur aus einer einzigen Konfiguration besteht, bei der der Kopf auf die Startzelle zeigt.

Kommen wir nun zur eigentlichen Simulationsphase:

ad 9) Dieser Schritt ist genau wie im Beweis von 4.4.1 in [BV97]: Der erste, dritte und vierte Unterschritt läßt sich gemäß dem Synchronisationstheorem von wohlgeformten, stationären Polynomialzeit-QTMs in Normalform durchführen, deren Laufzeiten nur von $T(|x|)$ und damit nur von $|x|$ abhängen.

Der erste Unterschritt dient dazu, jeweils den Inhalt der Superzelle, auf die der Kopf der zu simulierenden Maschine zeigt, auf einen ausgezeichneten Arbeitsbereich zu transportieren.

Der zweite Unterschritt stellt die einzige Stelle dar, an der bei der Konstruktion von M_U Amplituden ungleich 0 und 1 eingehen.

Für den Genauigkeitsparameter γ kann er mit der QTM aus dem Unitary Transformation Theorem 4.3.7, in $Poly(|x|, 1/\gamma) = Poly(|x|, 1/\varepsilon)$ Zeit ausgeführt werden und transformiert die Superposition des aktuellen Symbols und Zustands in die Nachfolgesuperposition des neuen Symbols und Zustands.[9]

Beachte, daß dabei alle sich überlagernden Berechnungspfade nach exakt gleich vielen Schritten mit dem Kopf auf der Startzelle terminieren.

Der dritte Unterschritt bewegt die entstandene Superposition wieder auf die ursprüngliche Superzelle zurück, und der vierte Unterschritt bewegt den simulierten Kopf entsprechend dem Richtungsstring s' der zu simulierenden Maschine um eine Superzelle nach links oder rechts.

Mit den Lemmata 2.5.1 und 2.5.2 können wir das Verkettungslemma 2.5.6 anwenden und erhalten eine wohlgeformte QTM in Normalform, die die vier Unterschritte realisiert. Ihre Laufzeit hängt nur polynomiell von $T(|x|)$ und $1/\varepsilon$ ab, da dies auch für die einzelnen QTMs galt.

Dank der Synchronität des zweiten Unterschritts ist auch die verkettete QTM stationär.

Das Looping Lemma erlaubt ihre $T(|x|)$-fache Wiederholung, denn die zu wiederholende, wohlgeformte, stationäre QTM in Normalform arbeitet auf ihren eigenen Spuren und hinterläßt keine unerwünschten Ergebnisse von Zwischenberechnungen.

Die Eingabe der Anzahl der Wiederholungen kommt dabei von der entsprechenden Hilfsspur. Die resultierende QTM realisiert Schritt 9 und ist $Poly(T(|x|), 1/\varepsilon)$-zeitbeschränkt.

ad 10) Es handelt sich offensichtlich um eine effiziente, deterministische Berechnung, deren Ausgabelänge konstant 1 ist und daher nur von der Eingabelänge abhängt. Das Synchronisationstheorem liefert uns daher die gewünschte QTM, die auch auf Überlagerungen als Eingabe wie gewünscht arbeitet.

---

[9] Diese Konstruktion ist nur möglich, weil QTMs nach unserer Definition - genau wie DTMs und PTMs - in einem Schritt nur lokale Änderungen vornehmen können. Beachte, daß die ausgeführte Transformation nur eine Näherung an U' ist.

Wir verketten nun mit Lemma 2.5.1, 2.5.2 und 2.5.6 die QTM aus Meta-Schritt 9 mit der QTM aus Meta-Schritt 10, und die resultierende QTM hinter die QTM der Meta-Schritte 2 bis 8, und erhalten M^{Rest}. Die zu Anfang genannte Konstruktion liefert dann die Existenz einer QTM M_U mit den gewünschten Eigenschaften[10], bis auf die Simulationseigenschaft.

Um diese zu zeigen, analysieren wir nun die erzielte Simulationsgenauigkeit:

Fall a) Nehmen wir zunächst an, daß $x \in \mathcal{G}$ ist und M_x auf x in höchstens $T(|x|)$ Schritten hält. Bezeichne mit $p(x)$ die Wahrscheinlichkeit, mit der M_x x akzeptiert.

Bezeichne mit $\tilde{U}$ die endlichdimensionale unitäre Transformation gemäß Korollar 2.6.2, die sich aus der Verkettung von M_x mit $M_\Sigma^{T(|x|)}$ ergibt. Die zugehörige QTM ist wohlgeformt, wohlbenehmend und in Normalform. Die Beobachtung der letzten Spur ihrer Startzelle nach genau $T(|x|)$ Schritten liefert eine 1 mit der Wahrscheinlichkeit $p(x)$.

Jeder Eintrag ihrer Codierung U' hat die Genauigkeit $\frac{\varepsilon}{8T(|x|)(10\sqrt{d})^{d+2}}$ und die dargestellte Transformation ist $\frac{\varepsilon}{16T(|x|)(10\sqrt{d})^d}$-nah an der unitären Transformation $\tilde{U}$ wegen Lemma 4.3.2 und da für alle $d \geq 1$ gilt: $\frac{1}{8(10\sqrt{d})^2} < \frac{1}{16(\sqrt{d})^2}$.

Es ist $\frac{\varepsilon}{16T(|x|)(10\sqrt{d})^d} = \frac{1}{2(10\sqrt{d})^d} \cdot \gamma$ und Theorem 4.3.7 liefert je Simulationsschritt eine Genauigkeit von $\gamma = \frac{\varepsilon}{8T(|x|)}$, das heißt, die in jedem Schritt t angewendete unitäre Transformation U''_t ist $\frac{\varepsilon}{4T(|x|)}$-nah an $\tilde{U}$, denn für alle $d \geq 1$ gilt:
$||U''_t - \tilde{U}|| \leq ||U''_t - \tilde{U}|| + ||\tilde{U} - U'|| \leq \gamma + \frac{\varepsilon}{16T(|x|)(10\sqrt{d})^d} \leq \frac{\varepsilon}{8T(|x|)} + \frac{\varepsilon}{8T(|x|)} = \frac{\varepsilon}{4T(|x|)}$

Fakt 1.3.6 sagt uns dann, daß nach $T(|x|)$ Schritten insgesamt eine unitäre Transformation angewandt wurde, die $\frac{\varepsilon}{4}$-nah an der Transformation $\tilde{U}^{T(|x|)}$ ist.

Lemma 2.1.7 garantiert dann, daß die Beobachtung der Simulationsspur nach dem Halten eine Wahrscheinlichkeitsverteilung liefert, deren Abstand von der Wahrscheinlichkeitsverteilung, die der Beobachtung von M_x auf Eingabe x nach dem Halten zugrunde liegt, höchstens ε ist.

Fall b) Nehmen wir nun an, daß $x \in \mathcal{G}$ ist und M_x auf x nicht in $T(|x|)$ Schritten hält.

[10] Beachte, daß dies - auch was die Simulationsphase angeht - für jede syntaktisch korrekte Eingabe gilt, nicht nur für $x \in \mathcal{G}$

Dann erkennt Meta-Schritt 10 mit der soeben hergeleiteten Genauigkeit ε, daß der aktuelle Zustand der zu simulierenden QTM $M^{(2)}$ nicht in $\{\tilde{q}_0, \ldots, \tilde{q}_{T(|x|)}\}$ ist.

Fall c) Nehmen wir nun an, daß x zwar syntaktisch korrekt ist, aber $x \notin \mathcal{G}$ gilt.

Dann produziert die Simulation zwar unsinnige Ergebnisse, aber die Laufzeit von M_U ist dennoch durch ein Polynom in $T(|x|)$ und $1/\varepsilon$ beschränkt, denn der einzige kritische Punkt ist die Anwendung der QTM aus dem Unitary Transformation Theorem. Aus ihrer Konstruktion geht jedoch hervor, daß sie für alle Eingaben laufzeitbeschränkt ist, nicht nur für hinreichend nah an unitäre. Sie liefert dann zwar keine Garantie für irgend eine Simulationsgenauigkeit, aber wir müssen von M_U für Eingaben $x \notin \mathcal{G}$ auch nichts weiter als Laufzeitbeschränktheit zeigen.

Fall d) Falls x syntaktisch inkorrekt ist, so erkennt dies die QTM aus Schritt 1 absolut zuverlässig und beendet die Rechnung von M_U. Die Akzeptanzwahrscheinlichkeit ist dabei für uns wegen $x \notin \mathcal{G}$ nicht von Interesse. □

4.5 Folgerungen

Mit Hilfe der Boosting-Konstruktion können wir nun zeigen, daß die Zeitkomplexitätsklassen für verschiedene Akzeptanzparameter $\alpha \in (\frac{1}{2}, 1)$ polynomiell verknüpft sind. Ohne dieses Resultat, welches auch für PTMs gilt, müßten wir uns fragen, ob unser QTM-Modell überhaupt sinnvoll ist, denn die spezielle Wahl von $\alpha = \frac{2}{3}$ ist willkürlich und muß gerechtfertigt werden.

Außerdem können wir mit Hilfe der Universellen QTM zeigen, daß die Komplexitätsklasse zu QTMs in Normalform, mit der Komplexitätsklasse zu QTMs, die nicht notwendig in Normalform sind, ebenfalls polynomiell verknüpft ist. Dies bezieht sich jedoch zunächst nur auf das Akzeptanzverhalten von Sprachen. Gleiches können wir von der Stationärität zeigen.

Lemma 4.5.1. *Seien $G \subseteq (\Sigma_1 \setminus \#)^*$, $T : \mathbb{N} \to \mathbb{N}$ mit $T(n) > n$ zeitkonstruierbar sowie $\frac{1}{2} < \alpha, \beta < 1$. Dann existieren Polynome $P(\cdot), P'(\cdot)$ (die von $1/\alpha$ und $1/\beta$ polynomiell abhängen), so daß*

$$\begin{aligned} BQTime_{G,\alpha}(T(n)) &\subseteq BQTime_{G,\beta}(P(T(n))) \qquad (4.1)\\ BQTime_{G,\alpha}(T(n)) &\subseteq \widehat{BQTime}_{G,\alpha}(T(n)) \qquad (4.2)\\ BQTime_{G,\alpha}(P'(T(n))) &\supseteq \widehat{BQTime}_{G,\alpha}(T(n)) \qquad (4.3) \end{aligned}$$

Dabei muß die $L \in \widehat{BQTime}_{G,\alpha}(T(n))$ "bezeugende" QTM in 4.3 nicht notwendig stationär sein. Außerdem gilt

$$BQP_{G,\beta} = BQP_{G,\alpha} = \widehat{BQP_{G,\alpha}} \tag{4.4}$$

Beweis.
Falls $\alpha \geq \beta$ so gilt trivialerweise $BQTime_{G,\alpha}(T(n)) \subseteq BQTime_{G,\beta}(T(n))$

Falls $\alpha < \beta$:
Wegen Satz 4.2.4 ist für $\frac{1}{2} < \alpha < \frac{2}{3}$
$BQTime_{G,\alpha}(T(n)) \subseteq BQTime_{G,\frac{2}{3}}(P_1(T(n)))$ für ein Polynom $P_1(\cdot)$
und wegen Theorem 4.2.3 ist für $\frac{2}{3} < \beta < 1$
$BQTime_{G,\frac{2}{3}}(T(n)) \subseteq BQTime_{G,\beta}(P_2(T(n)))$ für ein Polynom $P_2(\cdot)$.

Zusammen folgt für $\frac{1}{2} < \alpha \leq \frac{2}{3} \leq \beta < 1$
$BQTime_{G,\alpha}(T(n)) \subseteq BQTime_{G,\beta}(P_3(T(n)))$ für ein Polynom $P_3(\cdot)$.

Für $\frac{1}{2} < \alpha < \beta < \frac{2}{3}$ folgt aus Satz 4.2.4 für ein Polynom $P_4(\cdot)$:
$BQTime_{G,\alpha}(T(n)) \subseteq BQTime_{G,\frac{2}{3}}(P_4(T(n))) \overset{\text{trivial}}{\subseteq} BQTime_{G,\beta}(P_4(T(n)))$.

Für $\frac{2}{3} < \alpha < \beta < 1$ folgt aus Theorem 4.2.3 für ein Polynom $P_5(\cdot)$:
$BQTime_{G,\alpha}(T(n)) \overset{\text{trivial}}{\subseteq} BQTime_{G,\frac{2}{3}}(T(n)) \subseteq BQTime_{G,\beta}(P_5(T(n)))$

Damit ist Inklusion 4.1 gezeigt.

Inklusion 4.2 ist klar, denn wenn M eine QTM in Normalform ist, die $L \in BQTime_{G,\alpha}(T(n))$ bezeugt, dann bezeugt sie auch $L \in \widehat{BQTime}_{G,\alpha}(T(n))$.

Zu Inklusion 4.3: M sei eine wohlgeformte, nicht notwendig stationäre QTM die $L \in \widehat{BQTime}_{G,\alpha}(T(n))$ "bezeugt". Gemäß Theorem 4.4.1 existiert dann eine wohlgeformte, stationäre QTM in Normalform M', die zur Eingabe $\varepsilon := (\alpha - \frac{1}{2})/2$, einer Codierung von M, x und $T(|x|)$, eine Simulation von M auf Eingabe x mit Genauigkeit ε für $T(|x|)$ Schritte mit Verlangsamung $Poly(\cdot, 1/\varepsilon)$ durchführt. Mit einer Konstruktion wie im Beweis von Theorem 4.4.3 erhalten wir daraus eine wohlgeformte, $Poly(T(|x|), 1/\varepsilon)$-zeitbeschränkte, stationäre QTM in Normalform M'', die als Eingabe nur x erhält und die anderen Eingaben implizit enthält, sowie auf Eingabe x eine Akzeptanzwahrscheinlichkeit realisiert, die von der Akzeptanzwahrscheinlichkeit der $T(n)$-zeitbeschränkten QTM M auf Eingabe x um höchstens ε abweicht.[11]

[11]Beachte, daß der Unterschied zu der Konstruktion in Theorem 4.4.3 nur darin besteht, daß die zu simulierende QTM implizit gegeben ist und die Eingabe für diese zu simulierende Maschine explizit einzugeben ist. Die zu simulierende QTM wird also auf beliebige Eingaben aus G angesetzt, besitzt jedoch auch auf ganz G die Separationseigenschaft, da sie $L \in \widehat{BQTime}_{G,\alpha}(T(n))$ bezeugt.

M'' bezeugt daher $L \in BQTime_{G,\alpha-\varepsilon}(P_6(T(n)))$ für ein Polynom $P_6(\cdot)$, und es ist $\alpha - \varepsilon = \frac{\alpha+1/2}{2} > \frac{1}{2}$ da $\alpha > \frac{1}{2}$. Zusammen mit Inklusion 4.1 folgt damit Inklusion 4.3.

Die Gleichungen 4.4 folgen trivialerweise aus den ersten 3 Inklusionen, da ein Polynom angewandt auf ein Polynom wieder ein Polynom ergibt.

□

Die Aussage zum Komplementabschluß aus Kapitel 3 läßt sich mit diesen Ergebnissen auf Komplexitätsklassen mit unbestimmter polynomieller Laufzeitschranke übertragen, wobei der Typ der bezeugenden QTMs jedoch gleich bleiben kann.

Definition 4.5.2. *Für eine beliebige Sprachklasse $\mathcal{C}$ über dem Alphabet $(\Sigma_1 \setminus \#)^*$ sei* $\mathbf{co}\mathcal{C} := \{L \subseteq (\Sigma_1 \setminus \#)^* : (\Sigma_1 \setminus \#)^* \setminus L \in \mathcal{C}\}$

Korollar 4.5.3. *Für $G \subseteq (\Sigma_1 \setminus \#)^*$, $\frac{1}{2} < \alpha < 1$ und $T : \mathbb{N} \to \mathbb{N}$ gilt:*

$$BQP_{G,\alpha} = co\widehat{BQP_{G,\alpha}} = coBQP_{G,\alpha}$$

Beweis. Zur ersten Gleichung:
"$\subseteq$" ist genau die Aussage von Satz 3.1.1
"$\supseteq$": Sei $L \in co\widehat{BQP_{G,\alpha}}$. Dann ist $(\Sigma_1 \setminus \#)^* \setminus L \in \widehat{BQP_{G,\alpha}}$ und nach Gleichung 4.4 in Lemma 4.5.1 auch $(\Sigma_1 \setminus \#)^* \setminus L \in BQP_{G,\alpha}$. Nach Satz 3.1.1 ist dann $(\Sigma_1 \setminus \#)^* \setminus ((\Sigma_1 \setminus \#)^* \setminus L) = L \in \widehat{BQP_{G,\alpha}}$ und letztlich liefert Gleichung 4.4 $L \in BQP_{G,\alpha}$.

Zur zweiten Gleichung:
"$\subseteq$": Sei $L \in co\widehat{BQP_{G,\alpha}}$. Dann ist $(\Sigma_1 \setminus \#)^* \setminus L \in \widehat{BQP_{G,\alpha}}$ und nach Gleichung 4.4 auch $(\Sigma_1 \setminus \#)^* \setminus L \in BQP_{G,\alpha}$.
"$\supseteq$": Sei $L \in coBQP_{G,\alpha}$ und M eine QTM, die $(\Sigma_1 \setminus \#)^* \setminus L \in BQP_{G,\alpha}$ bezeugt, dann bezeugt M auch $(\Sigma_1 \setminus \#)^* \setminus L \in \widehat{BQP_{G,\alpha}}$. □

Kapitel 5

Hierarchiesätze

5.1 Quantenzeitkomplexitätsklassen

Satz 5.1.1 (Zeithierarchiesatz). *Sei* $T_1 : \mathbb{N} \to \mathbb{N}$ *zeitkonstruierbar und* $T_1(n) > n$*. Dann existiert ein Polynom* $P(\cdot)$*, so daß gilt:*

$$BQTime_{\mathcal{G}}(T_1(n)) \subsetneqq BQTime_{\mathcal{G}}(P(T_1(n)))$$

Beweis. Sei $BQ\widehat{Time^*_G}(T(n))$ genau wie $BQ\widehat{Time_G}(T(n))$ definiert, außer daß die bezeugenden QTMs zusätzlich unidirektional sein müssen.

Wir zeigen zunächst die Existenz eines Polynoms $P''(\cdot)$, so daß gilt:

$$\widehat{BQTime}^*_{\mathcal{G}}(5T_1(n)) \subsetneqq \widehat{BQTime}_{\mathcal{G}}(P''(T_1(n)))$$

"$\subseteq$" ist trivial für $P''(m) \geq 5m$.
"$\neq$": Betrachte die universelle QTM M_U aus Theorem 4.4.3 mit implizitem Genauigkeitsparameter $\frac{1}{7}$ und impliziter Simulationsdauer $5T_1(n)$. Dann existiert ein Polynom $P'(\cdot)$, so daß M_U wohlgeformt, $P'(T_1(n))$-zeitbeschränkt, stationär und in Normalform ist.

Wir wenden die Konstruktion der Boosting-QTM aus Satz 4.2.4 für $\varepsilon = \frac{1}{7}$ auf M_U an, und erhalten die Existenz eines Polynoms $P''(\cdot)$ mit $P''(m) \geq 5m$ für alle $m \in \mathbb{N}$, sowie die Existenz einer wohlgeformten, $P''(T_1(n))$-zeitbeschränkten, stationären QTM M_2 in Normalform mit folgendem Akzeptanzverhalten für $x \in \mathcal{G}$:

Falls die von x codierte QTM M_x auf der Eingabe x in $5T_1(|x|)$ Schritten nicht hält, so verwirft M_2 x mit einer Wahrscheinlichkeit von mindesten $\frac{2}{3}$.

Falls M_x x in $5T_1(|x|)$ Schritten akzeptiert, so gilt:

M_2 akzeptiert x mit einer Wahrscheinlichkeit von mindestens $\frac{2}{3}$, falls M_x x mit einer Wahrscheinlichkeit von mindestens $\frac{2}{3}$ akzeptiert.

Und M_2 verwirft x mit einer Wahrscheinlichkeit von mindestens $\frac{2}{3}$, falls M_x x mit einer Wahrscheinlichkeit von mindestens $\frac{2}{3}$ verwirft.

Beachte, daß dies für $x \in \mathcal{G}$ alle möglichen Fälle sind. Also gilt mit

$$L := \{x \in \mathcal{G} : M_2 \text{ akzeptiert } x \text{ mit einer Wahrscheinlichkeit von mindestens } 2/3 \text{ in Zeit } P''(T_1(|x|))\}$$

M_2 bezeugt[1] $L \in BQTime_{\mathcal{G}}(P''(T_1(n)))$.

Und es gilt:

$$L = \{x \in \mathcal{G} : M_x \text{ akzeptiert } x \text{ mit einer Wahrscheinlichkeit von mindestens } 2/3 \text{ in Zeit } 5T_1(|x|)\}$$

Denn die Inklusion "$\supseteq$" haben wir oben schon festgestellt. Wir zeigen nun noch "$\subseteq$":

Sei $x \in L$. Angenommen M_x akzeptiert x *nicht* mit einer Wahrscheinlichkeit von mindestens $\frac{2}{3}$ in Zeit $5T_1(|x|)$.

Dann folgt wegen $x \in \mathcal{G}$, daß M_x x mit Wahrscheinlichkeit $\geq \frac{2}{3}$ in Zeit $5T_1(|x|)$ verwirft, oder daß M_x auf x nicht in Zeit $5T_1(|x|)$ hält.

Falls aber M_x auf x nicht in Zeit $5T_1(|x|)$ hält, so würde M_2 dies erkennen und x mit Wahrscheinlichkeit $\geq \frac{2}{3}$ verwerfen, und es wäre $x \notin L$.

Also muß M_x x in Zeit $5T_1(|x|)$ mit Wahrscheinlichkeit $\geq \frac{2}{3}$ verwerfen. Aber dann würde M_2 x ebenfalls mit Wahrscheinlichkeit $\geq \frac{2}{3}$ verwerfen.

Also muß die Annahme falsch gewesen sein und wir haben die Inklusion "$\subseteq$" ebenfalls gezeigt.

Wegen Satz 3.1.1 folgt $\overline{L} \in \widehat{BQTime}_{\mathcal{G}}(P''(T_1(n)))$.

Wir zeigen nun $\overline{L} \notin \widehat{BQTime}^*_{\mathcal{G}}(5T_1(n))$.[2]

Angenommen, es gäbe eine unidirektionale, wohlgeformte, $5T_1(n)$-zeitbeschränkte, stationäre QTM M', die $\overline{L} \in \widehat{BQTime}^*_{\mathcal{G}}(5T_1(n))$ bezeugt.

[1] Beachte, daß hier die Wahl des Grundbereichs entscheidend eingeht.

[2] Hierbei wird entscheidend sein, daß die das Komplement akzeptierende Maschine die gleiche Laufzeitschranke wie M_2 besitzt, damit wir das Diagonalisierungsargument anwenden können.

Betrachte die Codierung $x' := \mathcal{G}(M') \in \mathcal{G}$ von M', dann folgt aus der Akzeptanzeigenschaft von M_2:

$$\begin{array}{ll} & x' \in \mathcal{G} \cap \overline{L} = \mathcal{G} \setminus L \\ \Rightarrow & M_2 \text{ verwirft } x' \text{ mit Wahrscheinlichkeit} \geq 2/3 \text{ in Zeit } P''(T_1(|x|)) \\ \Rightarrow & M' \text{ verwirft } x' \text{ mit Wahrscheinlichkeit} \geq 2/3 \text{ in Zeit } T_1(|x|) \end{array}$$

Dabei ergibt sich die zweite Implikation, wegen $x \in \mathcal{G}$, aus den zu Anfang genannten drei möglichen Fällen, und da M' nach unserer Annahme $5T_1(n)$-zeitbeschränkt ist.

Aus der Akzeptanzeigenschaft von M_2 folgt außerdem:

$$\begin{array}{ll} & x' \in \mathcal{G} \setminus \overline{L} = \mathcal{G} \cap L \\ \Rightarrow & M' \text{ akzeptiert } x' \text{ mit Wahrscheinlichkeit} \geq 2/3 \text{ in Zeit } 5T_1(|x|) \end{array}$$

Insgesamt ergibt sich somit ein Widerspruch zur Annahme, daß M' jedes $y \in \mathcal{G} \cap \overline{L}$ in Zeit $5T_1(|x|)$ mit Wahrscheinlichkeit $\geq 2/3$ akzeptiert und jedes $y \in \mathcal{G} \setminus \overline{L}$ in Zeit $5T_1(|x|)$ mit Wahrscheinlichkeit $\geq 2/3$ verwirft.

Daher folgt $\overline{L} \notin \widehat{BQTime}^*_{\mathcal{G}}(5T_1(n))$ und wir haben die behauptete Ungleichheit gezeigt.

Als nächstes zeigen wir: $BQTime_{\mathcal{G}}(T_1(n)) \subseteq \widehat{BQTime}^*_{\mathcal{G}}(5T_1(n))$

Sei $L'' \in BQTime_{\mathcal{G}}(T_1(n))$ und M'' bezeuge dies. Dann gibt es gemäß Unidirection Lemma 2.8.3 eine unidirektionale, stationäre QTM in Normalform M''', die M'' fehlerfrei mit Verlangsamung um den Faktor 5 simuliert. M''' bezeugt daher insbesondere $L'' \in \widehat{BQTime}^*_{\mathcal{G}}(5T_1(n))$ und wir haben die behauptete Inklusion gezeigt.

Nach Lemma 4.5.1 (Inklusion 4.3) existiert ein Polynom $P(\cdot) \geq P''(\cdot)$, so daß $\widehat{BQTime}_{\mathcal{G}}(P''(T_1(n))) \subseteq BQTime_{\mathcal{G}}(P(T_1(n)))$.

Damit haben wir die Aussage des Hierarchiesatzes bewiesen, denn wir haben insgesamt gezeigt:

$$\begin{array}{rcl} BQTime_{\mathcal{G}}(T_1(n)) & \subseteq & \widehat{BQTime}^*_{\mathcal{G}}(5T_1(n)) \\ & \subsetneqq & \widehat{BQTime}_{\mathcal{G}}(P''(T_1(n))) \\ & \subseteq & BQTime_{\mathcal{G}}(P(T_1(n))) \end{array}$$

□

5.2 Ausblick

Wir haben mit diesem Hierarchiesatz nun gezeigt, daß es auch bei QTMs keine "größte" Zeitschranke gibt, in der alle überhaupt von einer QTM akzeptierbaren Sprachen akzeptiert werden könnten. Zumindest haben wir dies für einen festen, eingeschränkten Grundbereich gezeigt.

Wir erhalten eine Hierarchie von Quanten-Zeitkomplexitätsklassen durch wiederholte Anwendung des Hierarchiesatzes:

Es existieren Polynome $T_i : \mathbb{N} \to \mathbb{N}$ für $i \geq 1$, mit $n < T_1(n) < T_2(n) < T_3(n) < \ldots$ so daß

$$BQTime_{\mathcal{G}}(T_1(n)) \subsetneqq BQTime_{\mathcal{G}}(T_2(n)) \subsetneqq BQTime_{\mathcal{G}}(T_3(n)) \subsetneqq \ldots$$

Außerdem folgt, daß wir zu einer gegebenen Quanten-Zeitkomplexitätsklasse mit dem Grundbereich $\mathcal{G}$, die Zeitschranke nur um ein Polynom vergrößern müssen, um garantiert mehr Sprachen akzeptieren zu können.

Unser Boosting-Theorem gilt jedoch für beliebige Grundbereiche und für beliebige Separationsparameter $\alpha \in (\frac{1}{2}, 1)$. Daraus ergab sich das Resultat

$$BQP_{G,\beta} = BQP_{G,\alpha} = \widehat{BQP_{G,\alpha}}$$

welches wir von Probabilistischen Turingmaschinen her kennen.

Wir haben in Kapitel 3 die Komplementabschlußeigenschaft, in dem Sinn daß der Typ der QTM gleich bleibt, nur für die Klassen mit beliebiger polynomieller Zeitschranke $BQP_{G,\alpha}$ gezeigt. Für unsere Anwendung im Hierarchiesatzbeweis genügte uns die Konstruktion einer QTM M', die nicht notwendig in Normalform ist und mit der gleichen Zeitschranke das zu einer gegebenen QTM M inverse Akzeptanzverhalten realisiert.

Es ist interessant zu bemerken, daß unser Beweis in Kapitel 3 nur deshalb keine QTM M' in Normalform liefert, weil der Vervollständigungssatz 2.6.4 uns keine Normalform garantiert.

Es gilt jedoch folgende notwendige Bedingung:

Lemma 5.2.1. *Sei $M = (\Sigma, Q, \delta)$ eine partielle wohlgeformte QTM auf S. Die gemäß Satz 2.6.4 vervollständigte, wohlgeformte QTM M' sei in Normalform. Dann gilt:*

$$(q_0, \sigma) \in S \Rightarrow \begin{cases} (q_f, \sigma) \in S \text{ oder} \\ \forall (q, \tau) \in S, d \in \{L, R\} : \delta(q, \tau, \sigma, q_0, d) = 0 \end{cases} \tag{5.1}$$

Beweis. Sei $(q_0, \sigma) \in S$. Falls $(q_f, \sigma) \notin S$, so gilt für alle $(q, \tau) \in S$ wegen der Orthogonalitätsbedingung für M':

$$\begin{aligned} 0 &= \sum_{(\tilde{\tau},\tilde{q},\tilde{d}) \in \Sigma \times Q \times \{L,R\}} \delta'^*(q, \tau, \tilde{\tau}, \tilde{q}, \tilde{d})\, \delta'(q_f, \sigma, \tilde{\tau}, \tilde{q}, \tilde{d}) \\ &= \delta'^*(q, \tau, \sigma, q_0, R) \quad \text{da } M' \text{ in Normalform} \\ &= \delta^*(q, \tau, \sigma, q_0, R) \quad \text{da } (q, \tau) \in S \end{aligned}$$

Und da M' in Normalform ist, gilt auch für alle $(q, \tau) \in S$:

$$\begin{aligned} 0 &= \delta'(q, \tau, \sigma, q_0, L) \\ &= \delta(q, \tau, \sigma, q_0, L) \quad \text{da } (q_0, \sigma) \in S \end{aligned}$$

□

Wir vermuten, daß diese Bedingung 5.1, zusammen mit der Bedingung

$$(q_0, \sigma), (q_f, \sigma) \in S \Rightarrow \delta(q_0, \sigma, \sigma, q_f, R) = 1 \tag{5.2}$$

hinreichend ist, in dem Sinn, daß sich jede partiell wohlgeformte QTM, die diese Bedingungen erfüllt, zu einer wohlgeformten QTM *in Normalform* vervollständigen läßt.

Dann würde die Konstruktion aus Kapitel 3 genügen, um zu beweisen:

$$BQTime_{G,\alpha}(T(n)) = coBQTime_{G,\alpha}(T(n))$$

Eine weitere interessante Frage ist, das Polynom $P(\cdot)$ aus dem Hierarchiesatz, welches die einer QTM zusätzlich zuzugestehende Zeit angibt, damit sie garantiert mehr Sprachen akzeptieren kann, genauer zu bestimmen.

Dazu müßten die Verlangsamungen bei der Universellen QTM und beim Boosting genauer bestimmt werden.

In diesem Rahmen könnten wir auch untersuchen, welchen Einfluß der Parameter der Simulationsgenauigkeit auf den Zeitbedarf hat. So könnte eine andere Wahl als $\varepsilon = \frac{1}{7}$ im Hierarchiesatzbeweis, den dominierenden Zeitaufwand zwischen Boosting und Simulation verlagern.

Beachte, daß wir, um eine Chance zu haben, die Verlangsamung bei der Universellen QTM genau zu bestimmen, ein modifiziertes QTM-Modell wählen müßten. Denn in unserem Modell kennen wir den genauen Zeitbedarf der DTM, die die Amplituden beliebig genau berechnet, nicht. Die für die Simulation erforderliche Genauigkeit hängt jedoch von der Eingabelänge ab, und ist somit nicht als konstant anzusehen.

Dabei ist zu untersuchen, wie das neue QTM-Modell mit unserem Modell verknüpft ist. Von einem sinnvollen Modell sollten wir erwarten können, daß es polynomiell mit unserem Modell verknüpft ist. Dabei verlagert sich das Problem darauf, dieses Polynom, welches die Verlangsamung bei einer wechselseitigen Simulation der beiden Maschinenmodelle angibt, genau zu bestimmen.

Ein wichtiges Konstrukt, dessen genaue Verlangsamung wir ebenfalls bestimmen müßten, ist das Synchronisationstheorem. Es wird häufig benutzt, um zu begründen, warum wir bestimmte deterministische Berechnungen als Unterroutine benutzen dürfen.

Außerdem müßten wir den Zeitbedarf der zahlreichen, für Zwischenberechnungen benutzten DTMs, möglichst genau bestimmen.

Idealerweise könnten wir einen Hierarchiesatz auf dem uneingeschränkten Grundbereich $(\Sigma_1 \setminus \#)^*$ formulieren. Wir vermuten jedoch, daß die in der Motivation zur Einschränkung des Grundbereichs genannte Schwierigkeit, "schlechte" Codierungen, die als Eingabe vorliegen, in Quanten-Polynomialzeit zu erkennen, sich zurückführen läßt auf offene Probleme der Komplexitätstheorie. Eine Lösung unseres Problems müßte dazu effizient in eine Lösung eines solchen offenen Problems transformiert werden können.

So konnten Fenner et al in [FGHP98] zeigen, daß die Probleme, zu entscheiden, ob eine Akzeptanzwahrscheinlichkeit ungleich Null ist, und zu entscheiden, ob die Amplitude einer Konfiguration in einer gegebenen Superposition ungleich Null ist, in diesem Sinn hart sind.

Wir könnten zudem untersuchen, ob sich der Grundbereich $\mathcal{G}$ eventuell noch vergrößern läßt.

Wir könnten ferner versuchen, Hierarchiesätze für Platzkomplexitätsklassen zu zeigen. Ein Problem dabei ist allerdings die geeignete Definition platzbeschränkter QTMs - wir müssen vermutlich die Beobachtungs-Konvention modifizieren. Die Schwierigkeit entsteht, wenn einige Konfigurationen einer Superposition die Platzschranke einhalten und andere nicht. Wir können dies vermutlich nur durch eine Beobachtung zuverlässig erkennen.

Das Problem bei den Programmierprimitiven in Abschnitt 2.5 ist, daß sie keine Aussage über die Zeitbeschränktheit der resultierenden QTM machen, wo immer eine QTM für einen Zustand eingefügt wird. Der Grund hierfür liegt in den sich überlagernden Berechnungspfaden.

Wir könnten versuchen, geeignete Eigenschaften von QTMs zu finden, und zugehörige Sprachklassen zu definieren, so daß diese Eigenschaft bei Verkettung, Alternativer Ausführung, Wiederholung und ähnlichen Primitiven erhalten bleibt.

Idealerweise gäbe es zu beliebigen wohlgeformten QTMs solche QTMs, die diese Eigenschaften haben, und mit polynomieller Verlangsamung möglichst niedrigen Grades, in einem zu definierenden Sinn das gleiche berechnen.

Literaturverzeichnis

[ADH94] L. Adleman, J. DeMarrais, and M. Huang. Quantum computability. manuscript, 1994.

[BBBV97] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997.

[Ben73] Charles H. Bennett. Logical reversibility of computation. *IBM J. Res. Develop.*, 17:525–532, 1973.

[Ber] Ethan Bernstein. Quantum complexity theory. PhD dissertation, U.C. Berkeley, to appear.

[BV97] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM J. Comput.*, 26(5):1411–1473, 1997.

[Deu85] D. Deutsch. Quantum theory, the church-turing principle and the universal quantum computer. *Proc. R. Soc. Lond*, A400:97–117, 1985.

[Fey82] R. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21:467–488, 1982.

[FGHP98] Stephen Fenner, Frederic Green, Steven Homer, and Randall Pruim. Quantum NP is hard for PH. *Proceedings of 6th Italian Conference on Theoretical Computer Science*, pages 241–252, 1998.

[Gol99] Oded Goldreich. Introduction to complexity theory (lecture notes - two semester course). http://www.wisdom.weizmann.ac.il /home/oded/public_html/cc.html, 1999.

[Gru99] Jozef Gruska. *Quantum Computing*. McGraw-Hill, 1999.

[Rei90] Karl Rüdiger Reischuk. *Einführung in die Komplexitätstheorie.* B.G. Teubner Stuttgart, 1990.

[Sho94] P. Shor. Algorithms for quantum computation: Discrete log and factoring. *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science*, 1994.

[Yao93] A. Yao. Quantum circuit complexity. *Proceedings of the 34th Annual IEEE Symposium on Foundations of Computer Science*, pages 352–361, 1993.