Effiziente Hermitesche Quantorenelimination

Lorenz Alexander Gilch

30. September 2003

Eingereicht als Diplomarbeit am Lehrstuhl für Mathematik
Prof. Dr. V. Weispfenning an der Fakultät für Mathematik und Informatik
der Universität Passau

Zusammenfassung

In dieser Arbeit wird ein Verfahren zur Quantorenelimination von Formeln erster Stufe über der Sprache der angeordneten Ringe durch Zählen reeller Nullstellen von Idealen vorgestellt. Im Vordergrund steht dabei, wie das Verfahren effizient gemacht werden kann. Es werden dazu Strukturkonstanten, welche zum effizienten Rechnen in Restklassenalgebren beitragen, vorgestellt sowie eine Verallgemeinerung von diesen. Außerdem werden einige weitere neu entworfene Optimierungsmöglichkeiten präsentiert, so daß die entstehenden Ergebnisformeln von möglichst geringer Komplexität gehalten werden. Ferner wird eine generische Variante dieses Eliminationsverfahrens erläutert, welche es erlaubt, gewisse, anhand geometrischer Beispiele auch anschaulich interpretierbare Annahmen über Variablen machen zu dürfen. Außerdem werden anhand typischer Fragestellungen in der Roboterkinematik praktische Anwendungen des Verfahrens demonstriert.

Inhaltsverzeichnis

Einleitung

In dieser Arbeit wird ein Algorithmus zur Quantorenelimination von Formeln erster Stufe vorgestellt, welcher auf dem Zählen der reellen Nullstellen von Idealen unter Nebenbedingungen basiert. Dieser Algorithmus wurde in [15] von Volker Weispfenning veröffentlicht. Der Hauptteil der vorliegenden Arbeit beschäftigt sich damit, das Eliminationsverfahren effizient zu machen. Es werden vom Autor entworfene Optimierungsmöglichkeiten des Algorithmus und Anwendungen in der Roboterkinematik und Geometrie vorgestellt.

Diese Arbeit gliedert sich in folgende Kapitel: Kapitel 1 schafft die mathematischen Grundlagen für das Eliminationsverfahren. In Kapitel 2 wird eine Möglichkeit zum effizienten Rechnen in Restklassenalgebren vorgestellt. Kapitel 3 stellt den eingangs erwähnten Quantoreneliminationsalgorithmus vor. In Kapitel 4 wird gezeigt, wie man diesen Algorithmus zur generischen Quantorenelimination verwenden kann. Kapitel 5 liefert Größenabschätzungen im Zusammenhang mit Typformeln, aus welchen im wesentlichen die Ergebnisformeln beim Eliminationsverfahren zusammengebaut werden. Außerdem werden in diesem Kapitel vom Autor neu entworfene Optimierungsmöglichkeiten des Eliminationsalgorithmus vorgestellt. Kapitel 6 zeigt Anwendungen des Eliminationsverfahrens auf Fragestellungen in der Roboterkinematik. In Kapitel 7 werden ausführlich Testbeispiele zum Eliminationsalgorithmus durchgerechnet und die Komplexität der Ergebnisformeln miteinander verglichen, wenn man bei deren Berechnungen die vorgestellten Optimierungsmöglichkeiten verwendet bzw. nicht verwendet.

Sämtliche vorgestellte Verfahren und Algorithmen in dieser Arbeit wurden im Computeralgebrasystem REDUCE 3.7. implementiert. Alle Beispiele wurden dabei auf einem Rechner mit einem Pentium IV-Prozessor mit 2GHz durchgerechnet. Der zur Verfügung stehende Arbeitsspeicher betrug jeweils 128MB.

Kapitel 1

Mathematische Grundlagen

Es werden nun in diesem Kapitel die mathematischen Grundlagen für den Quantoreneliminationsalgorithmus vorgestellt und geschaffen. Auf größere Beweise wird dabei verzichtet und auf [2] verwiesen.

1.1 Grundlagen in der Idealtheorie

In diesem Abschnitt werden grundlegende Eigenschaften von Idealen in Polynomringen vorgestellt. Dabei wird man für einen Körper $\mathbb{K}$ der Charakteristik 0 zunächst den Polynomring $\mathbb{K}[\underline{X}] := \mathbb{K}[X_1, \ldots, X_n]$ in den Variablen $X_1, \ldots, X_n$ betrachten. Für $f \in \mathbb{K}[\underline{X}]$ und $\underline{c} \in \mathbb{K}^n$ bezeichnet $f(\underline{c})$ die Auswertung von f an der Stelle $\underline{c}$.

$T(\underline{X})$ bezeichnet die Menge aller Terme in den Variablen $X_1, \ldots, X_n$. Zu einem Element $f \in \mathbb{K}[\underline{X}]$ bezeichnet dann $T(f)$ die Menge aller in f enthaltenen Terme mit nicht-verschwindenden Koeffizienten. Zu $F \subseteq \mathbb{K}[\underline{X}]$ wird dann $T(F) := \bigcup_{f \in F} T(f)$ definiert.

Für ein $f \in \mathbb{K}[\underline{X}]$ bzw. für $F \subseteq \mathbb{K}[\underline{X}]$ und eine gegebene Termordnung $<$ bezeichnet $HT(f)$ bzw. $HT(F)$ den höchsten Term von f bzw. die Menge der höchsten Terme von F. Der höchste Koeffizient von f bzgl. $<$ wird mit $HK(f)$ notiert.

Zunächst zwei Definitionen:

Definition 1.1 (Normalform) *Sei $F \subseteq \mathbb{K}[\underline{X}]$ endlich, $<$ eine beliebige Termordnung, $h \in \mathbb{K}[\underline{X}]$. Dann ist h in Normalform (modulo F), falls h nicht mehr mit F reduziert werden kann.*

Bemerkung: Eine Linearkombination von Normalformen modulo einer Menge $F \subseteq \mathbb{K}[\underline{X}]$ ist wieder in Normalform modulo F.

Definition 1.2 (Reduzierte Menge) *Sei $F \subseteq \mathbb{K}[\underline{X}]$ endlich, $<$ eine beliebige Termordnung. Dann ist F eine* reduzierte Menge*, falls jedes $f \in F$ in Normalform modulo $F \backslash \{f\}$ ist.*

Bemerkung: In der Literatur wird auch vorausgesetzt, daß in einer reduzierten Menge die höchsten Koeffizienten der einzelnen Polynome zu 1 normiert sind. Diese zusätzliche Anforderung wird hier fallengelassen, da sie für alle weiteren Überlegungen in dieser Arbeit irrelevant ist.

Folgender Satz ist ein wichtiges Hilfsmittel, um zu charakterisieren, wann eine Menge $G \subseteq \mathbb{K}[\underline{X}]$ eine Gröbnerbasis ist:

Satz 1.3 *Sei $I \subseteq \mathbb{K}[\underline{X}]$ ein Ideal, $G \subseteq I$ endlich mit $0 \notin G$, $<$ eine Termordnung. Dann sind folgende Aussagen äquivalent:*

(i) G ist eine Gröbnerbasis von I bzgl. $<$.

(ii) $HT(I) \subseteq mult(HT(G))$

(iii) Jedes $f \in \mathbb{K}[\underline{X}]$ besitzt eine eindeutige Normalform modulo G.

(iv) Jedes $f \in I$ läßt sich mit G auf das Nullpolynom reduzieren.

Beweis: Siehe [2], Proposition 5.38 mit Theorem 5.35. □

Wenn man zu einem Ideal $I \subseteq \mathbb{K}[\underline{X}]$ eine Gröbnerbasis $G \subseteq I$ gegeben hat, so ist es sicher wünschenswert, daß G nur so groß wie nötig ist. Denn umso größer G ist, desto aufwendiger werden ggf. Normalformberechnungen, da dazu die Menge $HT(G)$ betrachtet werden muß. In Abbildung 1.1 wird ein Algorithmus vorgestellt, welcher eine gegebene Gröbnerbasis so verkleinert, daß die in Satz 1.3 (ii) aufgeführte charakteristische Eigenschaft einer Gröbnerbasis erhalten bleibt.

Lemma 1.4 *Sei G eine Gröbnerbasis eines Ideals $I \subseteq \mathbb{K}[\underline{X}]$ bzgl. einer beliebigen Termordnung $<$. Dann berechnet der Algorithmus **SMALLERGB** aus Abbildung 1.1 zu G eine Gröbnerbasis G' von I bzgl. $<$ mit $G' \subseteq G$. Insbesondere ist kein $g' \in G'$ topreduzibel modulo $G' \backslash \{g'\}$.*

Beweis:
Beh.1: Der Algorithmus terminiert.

Beweis der Beh.1: Die Termination des Algorithmus ist offensichtlich, da eine gegebene Gröbnerbasis G endlich ist und in jedem Durchlauf der *while*-Schleife die Menge F um genau ein Element verkleinert wird, also schließlich leer wird.

Beschreibung: Verkleinerung einer gegebenen Gröbnerbasis

Eingabe: Gröbnerbasis G bzgl. Termordnung $<$

Output Gröbnerbasis $G' \subseteq G$ bzgl. Termordnung $<$ mit $Id(G) = Id(G')$

procedure SMALLERGB(G)
begin
$F := G$
$G' := \emptyset$
while $F \neq \emptyset$ **do**
  $f :=$ „Wähle $f \in F$."
  $F := F \backslash \{f\}$
  if $HT(g) \nmid HT(f)$ für alle $g \in F$ **and**
    $HT(g') \nmid HT(f)$ für alle $g' \in G'$ **then**
      $G' := G' \cup \{f\}$
  end
end
return G'
end SMALLERGB

Abbildung 1.1: Verkleinerung einer Gröbnerbasis

Beh.2: Die Menge $mult(HT(F \cup G'))$ ist eine Invariante der *while*-Schleife.

Beweis der Beh.2: Vor dem allerersten Eintritt in die *while*-Schleife sind $F = G$ und $G' = \emptyset$ und damit

$$mult(HT(F \cup G')) = mult(HT(G)).$$

Gelte nun zu Beginn eines Schleifendurchlaufs diese Gleichung. In einem Schleifendurchlauf wird genau dann ein Polynom $f \in F$ aus $F \cup G'$ entfernt, wenn ein $g \in (F \cup G') \backslash \{f\}$ existiert mit $HT(g)|HT(f)$. In diesem Falle wird also $F \cup G'$ nach dem Durchlauf zu $(F \cup G') \backslash \{f\}$ und es gilt dabei

$$mult(HT(G)) = mult(HT(F \cup G')) = mult\big(HT((F \cup G') \backslash \{f\})\big).$$

Somit gilt in diesem Fall die Gleichung auch nach Beendigung des Schleifendurchlaufs. Wird im aktuellen Schleifendurchlauf kein Element aus $F \cup G'$ entfernt, so gilt die Gleichung $mult(HT(F \cup G')) = mult(HT(G))$ trivialerweise auch nach dem Schleifendurchlauf.

Daraus folgt Beh.2.

Nach dem letzten *while*-Schleifendurchlauf ist $F = \emptyset$, also gilt mit Satz 1.3 und Beh.2 für die Ergebnismenge G' des Algorithmus:

$$mult(HT(I)) \subseteq mult(HT(G)) = mult(HT(G')).$$

Nach Satz 1.3 ist wegen $G' \subseteq G \subseteq I$ nun G' ebenfalls eine Gröbnerbasis von I bzgl. $<$.

Beh.3: Eine weitere Schleifeninvariante ist die Aussage, daß

$$HT(g_1') \nmid HT(g_2') \text{ für alle } g_1', g_2' \in G' \text{ mit } g_1' \neq g_2'.$$

Beweis der Beh.3: Diese Aussage gilt vor dem ersten Schleifendurchlauf trivialerweise, da $G' = \emptyset$. Gelte nun diese Aussage vor einem Schleifendurchlauf. Sei G'_{vor} die Menge G' zum Zeitpunkt des Beginns und G'_{nach} die Menge G' am Ende des aktuellen Schleifendurchlaufs, d.h. für G'_{vor} ist die behauptete Aussage gültig. Falls $G'_{vor} = G'_{nach}$, gilt die behauptete Aussage auch nach dem Schleifendurchlauf.
Sei nun in dem aktuellen Schleifendurchlauf $f \in F$ so gewählt, daß

$$\begin{array}{lll} HT(g) \nmid HT(f) & \text{für alle} & g \in F\backslash\{f\} \text{ und} \\ HT(g') \nmid HT(f) & \text{für alle} & g' \in G'_{vor}. \end{array}$$

Damit wird G' zu $G' \cup \{f\}$, d.h. $G'_{nach} = G'_{vor} \cup \{f\}$. Zu zeigen bleibt also nur noch, daß

$$HT(f) \nmid HT(g') \text{ und } HT(g') \nmid HT(f) \text{ für alle } g' \in G'_{vor}.$$

Die Beziehung $HT(g') \nmid HT(f)$ für alle $g' \in G'_{vor}$ ist offensichtlich nach Wahl von f. Angenommen es gilt $HT(f) | HT(g')$ für ein $g' \in G'_{vor}$. In einem der vorherigen Schleifendurchläufe wurde g' zu G' hinzugefügt, wobei zu diesem Zeitpunkt noch $f \in F$ war. Daher musste in jenem Schleifendurchlauf die Bedingung $HT(f) \nmid HT(g')$ erfüllt sein, ein Widerspruch zur Annahme. Also gilt die behauptete Invariante vor und nach jedem Durchlauf der *while*-Schleife.

Die Aussage aus Beh.3 ist insbesondere nach dem allerletzten Schleifendurchlauf gültig. Die Ergebnismenge G' des Algorithmus besitzt also die Eigenschaft, daß jedes Element $g' \in G'$ nicht mehr topreduzibel modulo $G'\backslash\{g'\}$ ist. □

Es folgen vier weitere wichtige Definitionen mit dazugehörigen Eigenschaften:

Definition 1.5 (Reduzierte Gröbnerbasis) *Sei $G \subseteq \mathbb{K}[\underline{X}]$ endlich. Dann ist G eine reduzierte Gröbnerbasis bzgl. einer Termordnung $<$, falls G eine reduzierte Menge und eine Gröbnerbasis bzgl. $<$ ist.*

Bemerkung: Es läßt sich aus jeder Gröbnerbasis G bzgl. $<$ eine reduzierte Gröbnerbasis G_{red} bzgl. $<$ berechnen, so daß $Id(G) = Id(G_{red})$. Dies geschieht durch Interreduktion der Elemente von G.

Definition 1.6 (Menge der reduzierten Terme) *Sei $I \subseteq \mathbb{K}[\underline{X}]$ ein Ideal und $<$ eine beliebige Termordnung. Dann ist die* Menge der reduzierten Terme *$RT(I)$ von I (bzgl. $<$) gegeben durch die Menge*

$$RT(I) := T(\underline{X}) \setminus HT(I).$$

Lemma 1.7 *Sei $I \subseteq \mathbb{K}[\underline{X}]$ ein Ideal und G eine Gröbnerbasis von I bzgl. einer beliebigen Termordnung $<$. Dann ist*

$$RT(I) = \big\{t \in T(\underline{X}) \;\big|\; t \text{ in Normalform modulo } G\big\},$$

d.h. die Menge der reduzierten Terme von I ist die Menge aller Terme, die sich mittels G nicht mehr reduzieren lassen.

Beweis: Siehe [2], Lemma 6.51. □

Bemerkung: Jede Normalform modulo einer Gröbnerbasis G läßt sich als Linearkombination von reduzierten Termen aus $RT(Id(G))$ darstellen.

Definition 1.8 (Unabhängige Variablen & Dimension eines Ideals) *Sei $I \subseteq \mathbb{K}[\underline{X}]$ ein Ideal und $<$ eine beliebige Termordnung. Ferner sei $\underline{Y} := \{Y_1, \ldots, Y_k\} \subseteq \{X_1, \ldots, X_n\}$. Dann heißt $\{Y_1, \ldots, Y_k\}$* unabhängig *modulo I, falls*

$$I \cap \mathbb{K}[Y_1, \ldots, Y_k] = \{0\}.$$

$\{Y_1, \ldots, Y_k\}$ heißt stark unabhängig *modulo I bzgl. $<$, falls*

$$T(\{Y_1, \ldots, Y_k\}) \subseteq RT(I).$$

Falls $\underline{Y}$ (stark) unabhängig modulo I ist und es keine echte Obermenge von $\underline{Y}$ gibt, die (stark) unabhängig modulo I ist, so heißt $\underline{Y}$ maximal (stark) unabhängig *modulo I.*
Die Dimension *eines Ideals I mit $I \neq \mathbb{K}[\underline{X}]$ ist definiert als*

$$dim(I) := max\big\{|\underline{Y}| \mid \underline{Y} \subseteq \{X_1, \ldots, X_n\} \text{ unabhängig modulo } I\big\}.$$

Ferner ist $dim(\mathbb{K}[\underline{X}]) := -1$.

Bemerkungen: Es gilt für die Dimension eines Ideals I stets $-1 \leq dim(I) \leq n$. Außerdem hat ein Ideal I genau dann Dimension n, wenn $I = \{0\}$. Die Menge der maximal unabhängigen Variablenmengen sowie die Dimension eines Ideals können algorithmisch bestimmt werden(z.B. mit Hilfe [2], Corollary 6.17 und Table 9.6).
Aus der starken Unabhängigkeit einer Variablenmenge folgt die Unabhängigkeit der Variablenmenge(siehe [2], Lemma 9.23). Man kann zeigen, daß die Gleichung

$$dim(I) = max\big\{|\underline{Y}| \mid \underline{Y} \subseteq \{X_1, \ldots, X_n\} \text{ stark unabhängig modulo } I \text{ bzgl. } <\big\}$$

gilt(siehe [2], Kapitel 9.3). Die Menge der maximal stark unabhängigen Variablenmengen kann ebenfalls algorithmisch berechnet werden(siehe [2], Table 9.6).

Nicht jede maximal (stark) unabhängige Variablenmenge hat dieselbe Kardinalität wie die Dimension des zugehörigen Ideals:

Beispiel 1.9 *Sei* $I = Id(\{X_1X_3 + X_3, X_2X_3 + X_3\}) \subseteq \mathbb{K}[X_1, X_2, X_3]$. *Dann sind* $\{X_1, X_2\}$ *und* $\{X_3\}$ *maximal stark unabhängige Variablenmengen bzgl. der lexikographischen Termordnung* $<$ *mit* $X_3 < X_2 < X_1$. *Die Dimension von* I *ist* 2. $\diamond$

Definition 1.10 (Nulldimensional) *Sei* $\emptyset \neq I \subset \mathbb{K}[\underline{X}]$ *ein echtes Ideal. Dann heißt* I *nulldimensional, wenn* I *die Dimension* 0 *hat.*

Bemerkung: Die Nulldimensionalität eines Ideals ist äquivalent zur Existenz höchstens endlich vieler Nullstellen des Ideals im algebraischen Abschluß von $\mathbb{K}$.

Folgende beide Sätze bilden wichtige Charakterisierungen der Nulldimensionalität eines Ideals.

Satz 1.11 *Sei* $I \subseteq \mathbb{K}[\underline{X}]$ *ein echtes Ideal und* $<$ *eine beliebige Termordnung. Dann ist* I *genau dann nulldimensional, wenn die Menge der reduzierten Terme* $RT(I)$ *von* I *endlich ist.*

Beweis: Siehe [2], Theorem 6.54 mit Proposition 6.52. $\Box$

Satz 1.12 *Sei* $I \subseteq \mathbb{K}[\underline{X}]$ *ein echtes Ideal,* G *eine Gröbnerbasis von* I *bzgl. einer beliebigen Termordnung* $<$. *Dann gilt folgende Äquivalenz:*
I ist genau dann nulldimensional, wenn für alle $i \in \{1, \ldots, n\}$ *ein* $k_i \in \mathbb{N}$ *existiert mit* $X_i^{k_i} \in HT(G)$.

Beweis: Siehe [2], Theorem 6.54. $\Box$

Im folgenden werden einige wichtige Aussagen in Zusammenhang mit Normalformen von Polynomen gemacht. Das nachfolgende technische Lemma wird zum Beweis des darauffolgenden Lemmas benötigt.

Lemma 1.13 (Verschiebungslemma) *Seien* $f, g, h, h_1 \in \mathbb{K}[\underline{X}]$ *und endliches* $G \subseteq \mathbb{K}[\underline{X}]$ *gegeben. Gelte, daß* $f + g = h$ *und daß sich* h *mit* G *auf* h_1 *reduzieren läßt.*
Dann existieren $f_1, g_1 \in \mathbb{K}[\underline{X}]$, *so daß* $f_1 + g_1 = h_1$ *und sich* f *bzw.* g *auf* f_1 *bzw.* g_1 *mit* G *reduzieren lassen.*

Beweis: Siehe [2], Lemma 5.25. □

Mit Hilfe dieses Lemmas kann man folgende Eigenschaft in Bezug auf Normalformen gewinnen.

Lemma 1.14 *Sei $I \subseteq \mathbb{K}[\underline{X}]$ ein Ideal, G eine Gröbnerbasis von I bzgl. einer beliebigen Termordnung $<$. Seien $f, f' \in \mathbb{K}[\underline{X}]$ mit $f - f' \in I$, g Normalform von f, g' Normalform von f' modulo G. Dann ist $g = g'$.*

Beweis: Nach Satz 1.3 sind g und g' eindeutig bestimmt und $f - f'$ läßt sich mit G auf das Nullpolymom reduzieren. Mit dem Verschiebungslemma 1.13 folgt, daß f_1 und f_1' existieren, so daß sich f auf f_1 und f' auf f_1' mit G reduzieren lassen, so daß $f_1 - f_1' = 0$. Da f_1 und f_1' somit wiederum wegen Satz 1.3 dieselbe Normalform modulo G besitzen, folgt die Behauptung des Lemmas. □

Das nächste Lemma zeigt, wie man zu einer Linearkombination von Polynomen die Normalform modulo einer Gröbnerbasis aus den Normalformen der einzelnen Polynome berechnen kann.

Lemma 1.15 *Sei $1 \leq m \in \mathbb{N}$ und $G \subseteq \mathbb{K}[\underline{X}]$ eine Gröbnerbasis von $Id(G)$ bzgl. einer beliebigen Termordnung $<$. Seien $f, f_1, \ldots, f_m \in \mathbb{K}[\underline{X}]$, $\alpha_1, \ldots, \alpha_m \in \mathbb{K}\backslash\{0\}$, und $f = \sum_{i=1}^m \alpha_i f_i$. Seien $f_1', \ldots, f_m' \in \mathbb{K}[\underline{X}]$ die Normalformen von $f_1, \ldots, f_m$ modulo G.*
Dann ist die eindeutige Normalform f' von f modulo G gegeben durch

$$f' = \sum_{i=1}^{m} \alpha_i f_i'.$$

Beweis: Da G eine Gröbnerbasis ist, sind Normalformen modulo G eindeutig nach Satz 1.3. Also gilt für $i \in \{1, \ldots, m\}$, daß eindeutig bestimmte $q_i \in Id(G)$ existieren mit $f_i - q_i = f_i'$. Somit ist $\sum_{i=1}^m \alpha_i q_i \in Id(G)$ und damit folgt aus

$$f - \sum_{i=1}^{m} \alpha_i q_i = \sum_{i=1}^{m} \alpha_i f_i - \sum_{i=1}^{m} \alpha_i q_i = \sum_{i=1}^{m} \alpha_i (f_i - q_i) = \sum_{i=1}^{m} \alpha_i f_i' = f',$$

daß f' die Normalform von f modulo G ist, da f' als Linearkombination von Normalformen in Normalform ist. □

1.2 Gröbnersysteme

Parametrisierte Polynome und Spezialisierungen:

Im weiteren Verlauf dieser Arbeit werden Polynome mit unbestimmten Koeffizienten betrachtet werden, d.h. Polynome, deren Variablen entweder als Hauptvariablen von Polynomringen auftreten oder als Parameter. Die Parameter dienen dabei als Platzhalter, für welche man letztendlich konkrete Zahlen aus einem Grundring einsetzen möchte. Diese Polynome mit unbestimmten Koeffizienten werden für einen Grundring R als Elemente von

$$R[U_1, \ldots, U_m][X_1, \ldots, X_n] \quad \text{bzw.} \quad R(U_1, \ldots, U_m)[X_1, \ldots, X_n]$$

repräsentiert werden. Solche Polynome bezeichnet man als *parametrisiert.* Es ist darauf hinzuweisen, daß im Ring $R[U_1, \ldots, U_m][X_1, \ldots, X_n]$ die Divison durch Polynome aus $R[U_1, \ldots, U_m]$ nur dann erlaubt ist, falls man sichert, daß diese nicht zu Null ausgewertet werden. Mit anderen Worten, man darf für die Parameter nur solche Werte aus R einsetzen, so daß jene Polynome, durch welche dividiert wird, an diesen Stellen nicht verschwinden. Dies führt zu Fallunterscheidungen nach dem Verschwinden von Parametern bzw. von Polynomen in $R[U_1, \ldots, U_m]$.

Seien R, S Ringe, $R[\underline{U}][\underline{X}] := R[U_1, \ldots, U_m][X_1, \ldots, X_n]$ ein multivariater Polynomring, wobei $U_1, \ldots, U_m$ als Parameter und $X_1, \ldots, X_n$ als Hauptvariablen des Polynomrings betrachtet werden. Jeder Ringhomomorphismus $\varphi : R[\underline{U}] \to S$ läßt sich eindeutig erweitern zu einem Homomorphismus

$$\psi : R[\underline{U}][\underline{X}] \to S[\underline{X}] \quad \text{mit } \psi \mid_{R[\underline{U}]} = \varphi \text{ und } \psi(X_i) = X_i.$$

Ein solcher Erweiterungshomomorphismus wird Spezialisierung der Parameter $U_1, \ldots, U_m$ genannt. Im Weiteren wird ausschließlich der Fall auftreten, daß S ein Erweiterungskörper von R ist und φ eine Erweiterung der natürlichen Einbettung von R in S. Damit ist dann φ eindeutig durch die Bilder der Parameter $U_1, \ldots, U_m$ festgelegt. Daher kann eine solche Spezialisierung durch ein Element aus S^m festgelegt werden. Mit einer Spezialisierung $(u_1, \ldots, u_m) \in S^m$ ist damit der eindeutig bestimmte Ringhomomorphismus

$$\psi : R[\underline{U}][\underline{X}] \to S[\underline{X}] \quad \text{mit } \psi|_R = id_R, \psi(X_i) = X_i \text{ und } \psi(U_i) = u_i$$

gemeint.

Für ein Polynom $f \in R[\underline{U}]$ und eine Spezialisierung $\underline{u} \in S^m$ bezeichnet $f(\underline{u})$ die Auswertung von f an der Stelle $\underline{u}$. Eine Anwendung einer Spezialisierung $\underline{u} \in S^m$ auf ein Polynom $f \in R[\underline{U}][\underline{X}]$ wird als $f(\underline{u}, \underline{X})$ notiert. Diese Notation wird dann verwendet, wenn die Auswertung eines Polynoms unter einer Spezialisierung im Vordergrund steht.

Für eine Spezialisierung $\underline{u} \in S^m$ ist die Abbildung

$$\sigma_{\underline{u}} : \wp(R[\underline{U}][\underline{X}]) \to \wp(S[\underline{X}]) : M \mapsto \{f(\underline{u}, \underline{X}) \mid f \in M\}$$

definiert. Diese Notation wird vor allem dann verwendet, wenn die Spezialisierung der Parameter für eine Polynommenge von Bedeutung ist.

Entstehen offensichtlich durch die Betrachtung von parametrisierten Polynomen keine Probleme, so werden Aussagen für nicht-parametrisierte Polynome im Polynomring $R[\underline{X}]$ in den parametrischen Fall übernommen.

Im Weiteren wird der Ring R meist der Ring $\mathbb{Z}$ der ganzen Zahlen bzw. der Körper $\mathbb{Q}$ der rationalen Zahlen sein, und S der Körper $\mathbb{R}$ der reellen Zahlen.

Gröbnersysteme und ihre Eigenschaften:

Es wird nun eine kurze Einführung in die Theorie der Gröbnersysteme gegeben, wobei auf einige Beweise wiederum verzichtet werden wird.

Definition 1.16 (Umfassende Gröbnerbasis) *Sei $F \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ endlich, $G \subseteq Id(F)$ endlich und $\mathbb{Q} \subseteq \mathbb{K}$ ein Oberkörper von $\mathbb{Q}$. Dann heißt G eine umfassende Gröbnerbasis von $Id(F)$ bzgl. einer Termordnung $<$, falls für jede Spezialisierung $\underline{u} \in \mathbb{K}^m$ die Menge $\sigma_{\underline{u}}(G)$ eine Gröbnerbasis des Ideals $Id(\sigma_{\underline{u}}(F)) \subseteq \mathbb{K}[\underline{X}]$ bzgl. $<$ ist.*

Beispiel 1.17 *Sei $F := \{UX^2 + Y + 1, Y\} \subseteq \mathbb{Q}[U][X, Y]$. Dann ist u. a. wegen dem ersten Buchberger-Kriterium*

$$G = \{UX^2 + Y + 1, UX^2 + 1, Y\}$$

eine umfassende Gröbnerbasis von $Id(F) \subseteq \mathbb{Q}[U][X, Y]$ bzgl. der lexikographischen Termordnung mit $X > Y$.
Man beachte, daß F unter der Spezialisierung $U = 0$ keine Gröbnerbasis von $Id(\sigma_0(F))$ ist. $\diamond$

Das Problem bei der Bestimmung von umfassenden Gröbnerbasen ist, daß man Fallunterscheidungen durchführen muß, ob gewisse höchste Terme unter Spezialisierungen verschwindende Koeffizienten besitzen. Die zentrale Idee zur Bestimmung von umfassenden Gröbnerbasen ist, eine Fallunterscheidung nach dem Verschwinden von Koeffizienten zu machen, so daß man entscheiden kann, welche Terme in Polynomen die höchsten Terme sind. Diese Fallunterscheidung wird mit Hilfe von Gröbnersystemen durchgeführt. Für den Rest dieses Kapitels versteht man unter einer Spezialisierung $\underline{u}$ stets eine Spezialisierung $\underline{u} \in \mathbb{K}^m$ für einen beliebigen Oberkörper $\mathbb{K}$ von $\mathbb{Q}$.

Definition 1.18 (Bedingung) *Seien $P_=, P_{\neq} \subseteq \mathbb{Q}[\underline{U}]$ endlich. Dann heißt das Paar $(P_=, P_{\neq})$ eine* Bedingung*.*
Eine Bedingung $(P_=, P_{\neq})$ heißt von einer Spezialisierung $\underline{u}$ erfüllt*, falls folgende Konjunktion gilt:*

$$\Big(\bigwedge_{p \in P_=} p(\underline{u}) = 0\Big) \wedge \Big(\bigwedge_{p' \in P_{\neq}} p'(\underline{u}) \neq 0\Big).$$

Eine Bedingung heißt erfüllbar*, wenn eine Spezialisierung existiert, welche die Bedingung erfüllt.*

Bemerkung: Wenn man zu $p_1, \ldots, p_k, p'_1, \ldots, p'_l \in \mathbb{Q}[\underline{U}]$ von

$$\bigwedge_{i=1}^{k} p_i = 0 \wedge \bigwedge_{j=1}^{l} p'_j \neq 0$$

als Bedingung spricht, dann meint man damit die Bedingung

$$\big(\{p_1, \ldots, p_k\}, \{p'_1, \ldots, p'_l\}\big).$$

Definition 1.19 (Gröbnersystem) *Sei $(P_=^{(init)}, P_{\neq}^{(init)})$ eine Bedingung, $F \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ endlich und $<$ eine beliebige Termordnung. Sei*

$$S := \big\{((P_=, P_{\neq}), G) \mid (P_=, P_{\neq}) \text{ ist eine Bedingung, } G \subseteq \mathbb{Q}[\underline{U}][\underline{X}] \text{ endlich}\big\}.$$

Dann heißt S ein Gröbnersystem über einer initialen Bedingung *$(P_=^{(init)}, P_{\neq}^{(init)})$* bzgl. $<$*, falls S endlich ist und falls für jede Spezialisierung $\underline{u}$ der Parameter $\underline{U}$, welche die initiale Bedingung $(P_=^{(init)}, P_{\neq}^{(init)})$ erfüllt, genau ein Element $((P_=, P_{\neq}), G) \in S$ existiert, so daß*

1. *$(P_=, P_{\neq})$ von $\underline{u}$ erfüllt ist,*
2. *$G \subseteq Id(F)$ und $\sigma_{\underline{u}}(G)$ Gröbnerbasis von $Id(\sigma_{\underline{u}}(F))$ bzgl. $<$ ist und*
3. *$(P_=, P_{\neq})$ eindeutig den höchsten Term von jedem $g \in G$ festlegt.*

Ein Gröbnersystem *ist ein Gröbnersystem über der initialen Bedingung $(\emptyset, \emptyset)$. Jedes Element eines Gröbnersystems wird als* Ast *des Gröbnersystems bezeichnet.*
Ist in jedem Ast des Gröbnersystems S die zugehörige Gröbnerbasis G unter allen Spezialisierungen, welche die Bedingungen des jeweiligen Astes erfüllen, eine reduzierte Gröbnerbasis, so heißt S reduziertes Gröbnersystem*.*

Bemerkung: Zu einem Ideal $Id(F) \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ kann ein Gröbnersystem S algorithmisch bestimmt werden: Man berechnet mit Hilfe des Buchberger-Algorithmus aus F eine Gröbnerbasis, wobei ggf. Fallunterscheidungen durchgeführt werden müssen, ob parametrische Koeffizienten unter Spezialisierungen verschwinden. Inkonsistente Bedingungen in Ästen stören hierbei nicht, da es in dem Fall keine Spezialisierung der Parameter gibt, welche sie erfüllen. Aussagen über Äste mit inkonsistenten Bedingungen werden im folgenden offensichtlich bedeutungslos. Die Menge

$$\bigcup_{((P_=,P_{\neq}),G)\in S} G$$

bildet dann eine umfassende Gröbnerbasis von I.
Ist man im Sonderfall, daß $F = \{f\} \subseteq \mathbb{Q}[\underline{U}][X]$ einelementig ist mit $f = \sum_{i=0}^{d} a_i X^i \neq 0$, $a_d \in \mathbb{Q}[\underline{U}] \setminus \mathbb{Q}$, und ferner die initiale Bedingung mit $a_d \neq 0$ gegeben ist, so wird keine Fallunterscheidung nach dem höchsten Koeffizienten von f gemacht. In diesem Fall besitzt dann das zu $Id(F)$ gehörige Gröbnersystem über der initialen Bedingung $(\emptyset, \{a_d\})$ nur einen Ast, nämlich $((\emptyset, \{a_d\}), \{f\})$.

Definition 1.20 (Grüngefärbtes Gröbnersystem) *Werden alle Monome in allen Polynomen aller Gröbnerbasen eines Gröbnersystems S entfernt, die unter den jeweiligen Bedingungen der Fallunterscheidung des Gröbnersystems verschwinden, so heißt das entstehende System grüngefärbtes Gröbnersystem.*

Bemerkung: Man kann direkt bei der Berechnung von grüngefärbten Gröbnersystemen sofort in den einzelnen entstehenden Fallunterscheidungen auftretende verschwindende Monome aus Polynomen entfernen, d.h. man muß diese Monom-Entfernungen nicht erst nach der Berechnung des Gröbnersystems durchführen. Dies kann ggf. einen Performance-Gewinn darstellen, da „überflüssige" verschwindende Monome sofort entfernt werden und nicht mehr länger in den Berechnungen mitgeschleift werden.

Während zu einem Gröbnersystem S eines Ideals $I \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ für alle $((P_=, P_{\neq}), G) \in S$ die Inklusion $G \subseteq I$ erfüllt ist, ist dies für grüngefärbte Gröbnersysteme i. a. nicht mehr richtig. Folgendes Beispiel verdeutlicht dies:

Beispiel 1.21 *Sei $F := \{UX^2 + Y + 1, Y\} \subseteq \mathbb{Q}[U][X,Y]$. Sei*

$$S_{gruen} := \big\{((\emptyset, \{U\}), \{UX^2 + Y + 1, Y\}), ((\{U\}, \emptyset), \{1\})\big\}.$$

Dann ist S_{gruen} ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. der lexikographischen Termordnung mit $X > Y$. Also ist $((\{U\}, \emptyset), \{1\}) \in S_{gruen}$, aber $1 \notin Id(F) \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$, da F nach dem ersten Buchberger-Kriterium im Polynomring $\mathbb{Q}[U,X,Y]$ eine Gröbnerbasis ist bzgl. der lexikographischen Termordnung mit $U > X > Y$. $\diamond$

Im weiteren Verlauf dieser Arbeit werden ausschließlich grüngefärbte Gröbnersysteme benutzt werden, da man stets annehmen wird können, daß die Bedingung eines Astes erfüllt ist, wenn man mit der Gröbnerbasis dieses Astes rechnet. Deswegen wird man in den Polynomen der einzelnen Gröbnerbasen diejenigen Monome, die aufgrund der Bedingung eines Astes des Gröbnersystems verschwinden, sofort entfernen, um die Berechnungen einfacher gestalten zu können.

Im nächsten Lemma 1.22 sowie dem darauffolgenden Korollar 1.23 wird der Algorithmus `SMALLERGB` aus Abbildung 1.1 auf parametrisierte Polynommengen $G \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ angewandt. Dabei fasst man G als Teilmenge des Polynomrings $\mathbb{Q}(\underline{U})[\underline{X}]$ auf und arbeitet dann auf dieser Betrachtungsebene den Algorithmus ab. Das Ergebnis kann dann als Teilmenge von $\mathbb{Q}[\underline{U}][\underline{X}]$ bzw. $\mathbb{Q}(\underline{U})[\underline{X}]$ interpretiert werden.

Zum Beweis der folgenden Aussagen sei darauf hingewiesen, daß wiederum für $f \in \mathbb{Q}(\underline{U})[\underline{X}]$ analog zu Kapitel 1.1 $T(f)$ bzw. $HT(f)$ die Menge *aller* in f enthaltenen Terme bzw. der höchste Term unter *allen* auftretenden Termen in f ist. Man ignoriert also dabei, daß Koeffizienten aus $\mathbb{Q}[\underline{U}]$ unter Spezialisierungen verschwinden können.

Lemma 1.22 *Sei $F \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ endlich, S ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. einer beliebigen Termordnung $<$. Sei $((P_=, P_{\neq}), G) \in S$. Ferner sei $G' :=$ `SMALLERGB`$(G) \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$.*
Dann ist für jede Spezialisierung $\underline{u}$, welche die Bedingung $(P_=, P_{\neq})$ erfüllt, $\sigma_{\underline{u}}(G')$ eine Gröbnerbasis des Ideals $Id(\sigma_{\underline{u}}(F))$. Insbesondere ist jedes Element $g' \in \sigma_{\underline{u}}(G')$ nicht mehr topreduzibel modulo $\sigma_{\underline{u}}(G')\backslash\{g'\}$.

Beweis: Ist $(P_=, P_{\neq})$ inkonsistent, so ist die Aussage des Lemmas trivial. Sei also jetzt $(P_=, P_{\neq})$ erfüllbar und sei $\underline{u}$ eine Spezialisierung der Parameter, welche die Bedingung $(P_=, P_{\neq})$ erfüllt. Sei $G = \{g_1, \ldots, g_k\}$ und werden bei der Berechnung von G' im Algorithmus `SMALLERGB` in der *while*-Schleife die Elemente von G o.B.d.A. in der Reihenfolge $g_1, \ldots, g_k$ abgearbeitet. Ansonsten nummeriert man die Elemente in G so um, daß die Elemente von G in dieser Reihenfolge abgearbeitet werden. Sei das resultierende Ergebnis

$$G' = \texttt{SMALLERGB}(G) = \{g'_1, \ldots, g'_\ell\} \subseteq G \quad \text{mit } \ell \leq k.$$

Da S grüngefärbt ist, verschwindet der höchste Koeffizient in jedem $g \in G$ unter allen Spezialisierungen, welche die Bedingung $(P_=, P_{\neq})$ erfüllen, nicht. Somit gilt $HT(g_i) = HT(\sigma_{\underline{u}}(g_i))$ für $i \in \{1, \ldots, k\}$, sofern man auf der linken Seite der Gleichung g_i als Element von $\mathbb{Q}(\underline{U})[\underline{X}]$ auffasst. Wendet man nun `SMALLERGB` auf die Eingabemenge $\sigma_{\underline{u}}(G)$ an und arbeitet die Elemente dieser Menge in der Reihenfolge $\sigma_{\underline{u}}(g_1), \ldots, \sigma_{\underline{u}}(g_k)$ in der *while*-Schleife ab, wobei wiederholt auftretende Elemente in dieser Folge gestrichen werden können, so ist offensichtlich

$$\texttt{SMALLERGB}(\sigma_{\underline{u}}(G)) = \{\sigma_{\underline{u}}(g'_1), \ldots, \sigma_{\underline{u}}(g'_\ell)\} = \sigma_{\underline{u}}(G').$$

Nach Lemma 1.4 und der Definition eines Gröbnersystems ist $\sigma_{\underline{u}}(G')$ eine Gröbnerbasis von $Id(\sigma_{\underline{u}}(G)) = Id(\sigma_{\underline{u}}(F))$ bzgl. $<$.
Ebenso folgt direkt aus Lemma 1.4, daß jedes $g' \in \sigma_{\underline{u}}(G')$ nicht mehr topreduzibel modulo $\sigma_{\underline{u}}(G')\backslash\{g'\}$ ist. □

Korollar 1.23 *Sei $F \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ endlich. Sei S ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. einer beliebigen Termordnung $<$. Dann ist*

$$S' := \big\{((P_=, P_{\neq}), \texttt{SMALLERGB}(G)) \mid ((P_=, P_{\neq}), G) \in S\big\}$$

ebenfalls ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. $<$.

Beweis: Die Behauptung folgt sofort mit Lemma 1.22, welches man auf jeden Ast von S anwendet. Somit ist S' ein Gröbnersystem; auch grüngefärbt, da für alle $((P_=, P_{\neq}), G) \in S$ gilt, daß $\texttt{SMALLERGB}(G) \subseteq G$. □

Mit Hilfe von Korollar 1.23 kann man folgende Aussage zur Berechnung von reduzierten Gröbnersystem machen:

Satz 1.24 *Sei $F \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ endlich. Sei S ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. einer beliebigen Termordnung $<$.*
Dann kann aus S ein grüngefärbtes, reduziertes Gröbnersystem S_{red} von $Id(F)$ bzgl. $<$ berechnet werden, ohne daß dadurch neue Fallunterscheidungen nötig werden und neue Äste entstehen.
Genauer: Zu jedem Ast $((P_=, P_{\neq}), G) \in S$ kann eine Menge $G_{red} \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ bestimmt werden, so daß für jede Spezialisierung $\underline{u}$, welche $(P_=, P_{\neq})$ erfüllt, $\sigma_{\underline{u}}(G_{red})$ eine reduzierte Gröbnerbasis des Ideals $Id(\sigma_{\underline{u}}(F))$ bzgl. $<$ ist.

Beweis: Vorgreifende Bemerkung zum Beweis: Wenn man im folgenden von den (höchsten) Termen von Elementen $g \in \mathbb{Q}[\underline{U}][\underline{X}]$ spricht, dann fasst man g als Element von $\mathbb{Q}(\underline{U})[\underline{X}]$ auf. Analog für Teilmengen von $\mathbb{Q}[\underline{U}][\underline{X}]$.

Nach Korollar 1.23 ist

$$S' := \big\{((P_=, P_{\neq}), \texttt{SMALLERGB}(G)) \mid ((P_=, P_{\neq}), G) \in S\big\}$$

ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. $<$.

Sei nun ein Ast $((P_=, P_{\neq}), \texttt{SMALLERGB}(G)) \in S'$ gegeben und man definiere dazu $G' := \texttt{SMALLERGB}(G)$. Es genügt, zu zeigen, daß aus G' eine Menge $G_{red} \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ berechnet werden kann, so daß für jede Spezialisierung $\underline{u}$, welche $(P_=, P_{\neq})$ erfüllt, $\sigma_{\underline{u}}(G_{red})$ eine reduzierte Gröbnerbasis für $Id(\sigma_{\underline{u}}(F))$ ist.

Die Idee zur Bestimmung der gesuchten Polynommenge $G_{red} \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ ist, daß man allgemein in den Parametern $\underline{U}$ eine reduzierte Gröbnerbasis für diesen Ast aus G' berechnet. G_{red} wird dann folgendermaßen bestimmt:

Man betrachtet G' als Teilmenge von $\mathbb{Q}(\underline{U})[\underline{X}]$. Dann kann man aus G' eine reduzierte Menge $G'_{red} \subseteq \mathbb{Q}(\underline{U})[\underline{X}]$ berechnen. $G_{red} \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ entsteht dann aus G'_{red}, indem man jedes Element $g \in G'_{red}$ mit dem Hauptnenner der Koeffizienten von g multipliziert. G_{red} ist ebenfalls – gesehen als Teilmenge von $\mathbb{Q}(\underline{U})[\underline{X}]$ – eine reduzierte Menge.

Sei $\underline{u}$ eine Spezialisierung, welche $(P_=, P_{\neq})$ erfüllt. Es muß nun gezeigt werden, daß $\sigma_{\underline{u}}(G_{red})$ eine Gröbnerbasis von $Id(\sigma_{\underline{u}}(F))$ und eine reduzierte Menge ist. Zu Ästen mit inkonsistenter Bedingung $(P_=, P_{\neq})$ muß also nichts gezeigt werden.

Wenn man G' als Teilmenge von $\mathbb{Q}(\underline{U})[\underline{X}]$ auffasst, so ist nach Lemma 1.4 jedes $g' \in G'$ nicht mehr topreduzibel modulo $G' \backslash \{g'\}$. Also ist

$$HT(G'_{red}) = HT(G') \text{ und damit } HT(G_{red}) = HT(G'_{red}) = HT(G').$$

Da S grüngefärbt ist und $G' \subseteq G$, gilt für jedes Element $g' \in G'$, daß $HT(g') = HT(\sigma_{\underline{u}}(g'))$; also insbesondere $HT(G') = HT(\sigma_{\underline{u}}(G'))$. Da jedes $g' \in G' \subseteq \mathbb{Q}(\underline{U})[\underline{X}]$ nicht mehr topreduzibel ist, wird bei der Berechnung von G'_{red} nur durch Elemente $p \in \mathbb{Q}[\underline{U}]$ dividiert, welche höchste Koeffizienten von Elementen von G' sind, so daß für die Ungleichung $p(\underline{u}) \neq 0$ erfüllt ist. Also multipliziert man bei der Berechnung von G_{red} aus G'_{red} die einzelnen Elemente von G'_{red} nur mit Polynomen $q \in \mathbb{Q}[\underline{U}]$, für welche $q(\underline{u}) \neq 0$ gilt. Daraus folgt die Gleichheit

$$HT(\sigma_{\underline{u}}(G')) = HT(G') = HT(G_{red}) = HT(\sigma_{\underline{u}}(G_{red})).$$

Da $\sigma_{\underline{u}}(G')$ eine Gröbnerbasis von $Id(\sigma_{\underline{u}}(F))$ ist, folgt mit Satz 1.3 dann

$$HT\big(Id(\sigma_{\underline{u}}(F))\big) \subseteq mult\big(HT(\sigma_{\underline{u}}(G'))\big) = mult\big(HT(\sigma_{\underline{u}}(G_{red}))\big).$$

Also ist insbesondere $\sigma_{\underline{u}}(G_{red})$ eine Gröbnerbasis von $Id(\sigma_{\underline{u}}(F))$ bzgl. $<$.

Es bleibt noch zu zeigen übrig, daß $\sigma_{\underline{u}}(G_{red})$ eine reduzierte Menge ist. Dies folgt wegen

$$HT(\sigma_{\underline{u}}(G_{red})) = HT(G_{red}) \quad \text{und} \quad T(\sigma_{\underline{u}}(g)) \subseteq T(g) \quad \forall g \in G_{red}.$$

Wäre also ein $\sigma_{\underline{u}}(g) \in \sigma_{\underline{u}}(G_{red})$ reduzibel, so wäre $g \in G_{red} \subseteq \mathbb{Q}(\underline{U})[\underline{X}]$ reduzibel, was ein Widerspruch dazu wäre, daß G_{red} eine reduzierte Menge ist. Mit anderen Worten, G_{red} ist die allgemeine Form der reduzierten Gröbnerbasis. Dabei ist für jede Spezialisierung der Parameter, welche die Bedingung $(P_=, P_{\neq})$ erfüllt, gewährleistet, daß der höchste Koeffizient eines jeden

Elements von G_{red} unter der Spezialisierung $\underline{u}$ nicht verschwindet. Daher überträgt sich die Eigenschaft „reduzierte Menge" von G_{red} auf $\sigma_{\underline{u}}(G_{red})$. Somit ist mit diesen Notationen

$$S'' := \left\{((P_=, P_{\neq}), G_{red}) \mid ((P_=, P_{\neq}), G') \in S'\right\}$$

ein grüngefärbtes, reduziertes Gröbnersystem mit $|S''| = |S|$. □

Folgender Satz stellt eine wichtige Eigenschaft der Dimension und der Menge der maximal stark unabhängigen Variablenmengen von parametrischen Idealen vor.

Satz 1.25 *Sei $F \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ endlich. Sei S ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. einer beliebigen Termordnung $<$.*
Dann ist in jedem Ast des Gröbnersystems S die Dimension und die Menge der maximal stark unabhängigen Variablenmengen eindeutig bestimmt.

Beweis: Dieser Satz wird nur für den Spezialfall bewiesen, daß S ein Gröbnersystem bzgl. einer Totalgradtermordnung ist. Die Verallgemeinerung auf beliebige Termordnungen ist aber auch gültig.

In [2], Proposition 9.29 und Table 9.6, befindet sich ein Algorithmus, welcher zu einem Ideal $I \subseteq \mathbb{K}[\underline{X}]$, $\mathbb{K}$ Oberkörper von $\mathbb{Q}$, mit einer Gröbnerbasis G bzgl. einer Totalgradtermordnung aus $HT(G)$ die Dimension und die Menge der maximal stark unabhängigen Variablenmengen berechnet.

Sei nun $((P_=, P_{\neq}), G) \in S$. Dann ist die Menge der höchsten Terme der Gröbnerbasis G in diesem Ast eindeutig bestimmt, da S grüngefärbt ist. Für jede Spezialisierung $\underline{u}$, welche $(P_=, P_{\neq})$ erfüllt, ist also $HT(\sigma_{\underline{u}}(G))$ stets gleich. Somit liefert der eben zitierte Algorithmus für das Ideal $Id(\sigma_{\underline{u}}(F))$ mit Gröbnerbasis $\sigma_{\underline{u}}(G)$ für jede Spezialisierung $\underline{u}$, welche die Bedingung $(P_=, P_{\neq})$ erfüllt, dasselbe Ergebnis. Falls $(P_=, P_{\neq})$ inkonsistent ist, kann der eben zitierte Algorithmus zwar angewandt werden, jedoch ist das Ergebnis dann ohne Bedeutung. □

Bemerkung: Die Menge der maximal unabhängigen Variablenmengen ist in einem Ast eines Gröbnersystems i. a. <u>nicht</u> eindeutig bestimmt:

Beispiel 1.26 *Sei $F := \{X + UY\} \subseteq \mathbb{Q}[U][X, Y]$. Dann ist*

$$S := \left\{((\emptyset, \emptyset), \{X + UY\})\right\}$$

ein grüngefärbtes Gröbnersystem bzgl. der lexikographischen Termordnung $<$ mit $X > Y$.
Für $U \neq 0$ sind $\{X\}$ und $\{Y\}$ maximal unabhängige Variablenmengen. Für $U = 0$ ist allerdings $\{Y\}$ die einzige maximal unabhängige Variablenmenge. ◇

Satz 1.25 besagt also, daß in einem Ast zumindest jede maximal stark unabhängige Variablenmenge unter allen Spezialisierungen, die die Bedingung des Astes erfüllen, eine maximal unabhängige Variablenmenge ist.

Die folgenden Aussagen werden später benötigt werden, um die Nulldimensionalität eines Ideals in Verbindung zu bringen mit einer Menge maximal unabhängiger Variablen.

Lemma 1.27 *Sei $\mathbb{K}$ ein Körper der Charakteristik 0. Sei $F \subseteq \mathbb{K}[X_1, \ldots, X_n]$ endlich, $\{X_1, \ldots, X_d\}$ maximal unabhängig modulo $Id(F)$, $d < n$. Dann ist das von der Menge F im Ring $\mathbb{Q}(X_1, \ldots, X_d)[X_{d+1}, \ldots, X_n]$ erzeugte Ideal nulldimensional.*

Beweis: Siehe [2], Lemma 7.47 mit entsprechenden dortigen Definitionen. □

Lemma 1.28 *Sei $F \subseteq \mathbb{Q}[X_1, \ldots, X_n]$ endlich, $\{X_1, \ldots, X_d\}$ maximal unabhängig modulo $Id(F)$, $d < n$. Man betrachte ab jetzt F als Teilmenge von $\mathbb{Q}[X_1, \ldots, X_d][X_{d+1}, \ldots, X_n]$. Sei S ein grüngefärbtes Gröbnersystem von $Id(F)$ bzgl. einer beliebigen Termordnung $<$.*
Dann besitzt S mindestens einen nulldimensionalen Ast, dessen Bedingung von mindestens einer Spezialisierung $\underline{u} \in \mathbb{Q}^d$ der Parameter erfüllt wird.

Beweis: Die Behauptung folgt mit Lemma 1.27. Man berechnet zunächst für das nulldimensionale Ideal $Id(F) \subseteq \mathbb{Q}(X_1, \ldots, X_d)[X_{d+1}, \ldots, X_n]$ eine Gröbnerbasis G bzgl. $<$. Wegen der Nulldimensionalität existiert somit nach Satz 1.12 für jedes $i \in \{d+1, \ldots, n\}$ ein $k_i \in \mathbb{N}$ und ein $f_i \in G$ mit $HT(f_i) = X_i^{k_i}$.
Bei der Berechnung von G aus F sammelt man alle irgendwann auftretenden höchsten Koeffizienten, welche man stets als Elemente der Form $\frac{p}{q} \in \mathbb{Q}(X_1, \ldots, X_d)$ mit $p, q \in \mathbb{Q}[X_1, \ldots, X_d]$ darstellen kann. Die gesammelten Koeffizienten bilden eine endliche Menge, da höchstens endlich viele höchste Koeffizienten betrachtet werden. Sei P' diese Menge. Man definiert

$$P := \left\{ p \in \mathbb{Q}[X_1, \ldots, X_d] \;\middle|\; \frac{p}{q} \in P' \text{ oder } \frac{q}{p} \in P' \right\}.$$

Dann ist P ebenfalls endlich. Nun betrachtet man die daraus resultierende Konjunktion $\bigwedge_{p \in P} p \neq 0$. Dann ist diese erfüllbar, da es zu jeder endlichen Menge von Ungleichungen mindestens eine Belegung der Variablen in $\mathbb{Q}^d$ gibt, so daß alle Ungleichungen erfüllt sind.

Sei nun $\underline{u} \in \mathbb{Q}^d$ eine Spezialisierung der Parameter $X_1, \ldots, X_d$, welche diese Konjunktion erfüllt. Für diese Spezialisierung kann man nun eine Gröbnerbasis $G_{\underline{u}}$ von $Id(\sigma_{\underline{u}}(F))$ mit genau denselben Berechnungsschritten bestimmen wie bei der Berechnung von G, abgesehen von der Tatsache, daß man nun konkret $\underline{u}$ für die Parameter in den einzelnen Berechnungsschritten einsetzt. Dieses zur Berechnung von G analoge Bestimmen von $G_{\underline{u}}$ ist möglich, da alle auftretenden höchsten Koeffizienten nach Definition von P unter der Spezialisierung $\underline{u}$ nicht verschwinden und damit nicht durch Null geteilt wird. Mit anderen Worten, G ist die allgemeine Darstellung der Gröbnerbasis $G_{\underline{u}}$ in den Parametern. Somit ist

$$G_{\underline{u}} = \{g(\underline{u}, X_{d+1}, \ldots, X_n) \mid g \in G\}$$

und nach Definition von P gilt ferner

$$HT(g) = HT(g(\underline{u}, X_{d+1}, \ldots, X_n)) \quad \text{für alle } g \in G,$$

sofern g auf der linken Seite als Element von $\mathbb{Q}(X_1, \ldots, X_d)[X_{d+1}, \ldots, X_n]$ aufgefasst wird. Insbesondere gilt die letzte Gleichung für alle $f_i \in G$, d.h.

$$HT(f_i(\underline{u}, X_{d+1}, \ldots, X_n)) = X_i^{k_i} \quad \text{für } i \in \{d+1, \ldots, n\}.$$

Wegen $f_i(\underline{u}, X_{d+1}, \ldots, X_n) \in G_{\underline{u}}$ ist somit nach Satz 1.12 das Ideal $Id(G_{\underline{u}})$ nulldimensional. Da die Dimension in jedem Ast eines grüngefärbten Gröbnersystems eindeutig bestimmt ist, hat der Ast, zu welchem $\underline{u}$ gehört, Dimension Null. □

Korollar 1.29 *Sei $F \subseteq \mathbb{Q}[\underline{U}][X_1, \ldots, X_n]$ endlich. Sei S ein Gröbnersystem von $Id(F)$ bzgl. einer beliebigen Termordnung $<$ über einer initialen Bedingung $(P_=^{(0)}, P_{\neq}^{(0)})$. Sei $((P_=, P_{\neq}), G) \in S$ ein Ast von S, in welchem $\{X_1, \ldots, X_d\}$ maximal stark unabhängig modulo $Id(F)$, $d < n$, ist. Sei die Bedingung $(P'_=, P'_{\neq}) := (P_=^{(0)} \cup P_=, P_{\neq}^{(0)} \cup P_{\neq})$ erfüllbar. Man betrachte F nun als Teilmenge des Polynomrings $\mathbb{Q}[\underline{U}, X_1, \ldots, X_d][X_{d+1}, \ldots, X_n]$. Sei S' ein Gröbnersystem des Ideals $Id(F)$ bzgl. $<$ über der initialen Bedingung $(P'_=, P'_{\neq})$.*
Dann besitzt S' einen nulldimensionalen Ast.

Beweis: Sei $\underline{u} := (u_1, \ldots, u_m)$ eine Spezialisierung, welche $(P'_=, P'_{\neq})$ erfüllt. Im Ideal $\sigma_{\underline{u}}(F) \subseteq \mathbb{Q}[X_1, \ldots, X_d]$ ist dann die Menge $\{X_1, \ldots, X_d\}$ maximal unabhängig modulo $Id(\sigma_{\underline{u}}(F))$. Man betrachte nun $\sigma_{\underline{u}}(F)$ als Teilmenge von

$$\mathbb{Q}[X_1, \ldots, X_d][X_{d+1}, \ldots, X_n].$$

Sei $\tilde{S}$ ein Gröbnersystem von $Id(\sigma_{\underline{u}}(F))$ bzgl. $<$. Nach Lemma 1.28 besitzt also $\tilde{S}$ einen nulldimensionalen Ast, dessen Bedingung von einer reellen Spezialisierung $\underline{x} := (x_1, \ldots, x_d) \in \mathbb{R}^d$ erfüllt wird. Also ist das Ideal $Id(\sigma_{\underline{x}}(\sigma_{\underline{u}}(F))) \subseteq \mathbb{K}[X_{d+1}, \ldots, X_n]$, $\mathbb{Q} \subseteq \mathbb{K}$ Oberkörper, nulldimensional. Sei nun

$$\underline{s} := (u_1, \ldots, u_m, x_1, \ldots, x_d) \in \mathbb{K}^{m+d},$$

Man beachte, daß $\underline{s}$ nach Wahl von $\underline{u}$ insbesondere die initiale Bedingung $(P'_=, P'_{\neq})$ von S' erfüllt. Nach Konstruktion ist also derjenige Ast von S' nulldimensional, in welchem $\underline{s}$ liegt. $\underline{s}$ ist insbesondere eine reelle Spezialisierung, falls $\underline{u} \in \mathbb{R}^m$. □

Kapitel 2

Strukturkonstanten

In diesem Kapitel wird eine effiziente Methode zum Rechnen in Restklassenalgebren vorgestellt. Im folgenden wird für einen Körper $\mathbb{K}$ der Charakteristik 0 der Polynomring $\mathbb{K}[\underline{X}] := \mathbb{K}[X_1, \ldots, X_n]$ betrachtet. Ferner sei ein echtes nulldimensionales Ideal $I \subseteq \mathbb{K}[\underline{X}]$ und eine Gröbnerbasis G von I bzgl. einer beliebigen Termordnung $<$ gegeben. Wenn im weiteren Verlauf von Normalformen von Polynomen die Rede ist, dann sind damit stets die Normalformen der Polynome modulo der Gröbnerbasis G gemeint. Ebenso versteht sich in diesem Kapitel der Begriff „(top-)reduzibel" stets als „(top-)reduzibel modulo G". Außerdem sei die endliche Menge $RT(I)$ der reduzierten Terme des Ideals I bzgl. $<$ gegeben durch $\{v_1, \ldots, v_d\}$.

Das Ideal I definiert offensichtlich eine Äquivalenzrelation $\equiv_I$ auf $\mathbb{K}[\underline{X}]$: $f \equiv_I g \Leftrightarrow f - g \in I$. Die Äquivalenzklassen sind dann von der Form

$$f + I := \{g \in \mathbb{K}[\underline{X}] \mid f \equiv_I g\} = \{f + g \mid g \in I\} \quad \text{für } f \in \mathbb{K}[\underline{X}].$$

Zu $f \in \mathbb{K}[\underline{X}]$ nennt man $f + I$ die Restklasse von f bzgl. I. Die Menge der Restklassen wird dann mit $\mathbb{K}[\underline{X}]/I$ bezeichnet.

Definition 2.1 ($\mathbb{K}$-Algebra) *Sei $\mathbb{K}$ ein Körper. Eine (kommutative) $\mathbb{K}$-Algebra A ist ein $\mathbb{K}$-Vektorraum, in welchem zusätzlich eine kommutative Multiplikation $\circ : A \times A \to A$ definiert ist, so daß*

1. *A mit der Vektorraumaddition und der Multiplikation einen Ring mit Einselement bildet und*

2. *für alle $a, b \in \mathbb{K}, v, w \in A$ die Gleichung $(av) \circ (bw) = (ab)(v \circ w)$ gilt.*

Im nächsten Abschnitt wird nun gezeigt, wie man Operationen auf $\mathbb{K}[\underline{X}]/I$ definiert, so daß man daraus eine $\mathbb{K}$-Algebra erhält.

2.1 Rechnen in $\mathbb{K}[X_1,\ldots,X_n]/I$

2.1.1 Grundlagen

Seien $\alpha \in \mathbb{K}$, $f+I, g+I \in \mathbb{K}[\underline{X}]/I$. Dann sind die Skalarmultiplikation und Addition in $\mathbb{K}[\underline{X}]/I$ wie folgt definiert:

$$\begin{aligned}
\alpha \cdot (f+I) &:= (\alpha \cdot f) + I,\\
(f+I)+(g+I) &:= (f+g)+I.
\end{aligned}$$

Für die Multiplikation zweier Elemente $f+I, g+I \in \mathbb{K}[\underline{X}]/I$ definiert man

$$(f+I)\cdot(g+I) := (f\cdot g)+I.$$

Diese Definitionen der Operationen auf $\mathbb{K}[\underline{X}]/I$ sind wohldefiniert, d.h. unabhängig von der Wahl der Repräsentanten der einzelnen Restklassen. Denn aus $f+I = f'+I$ und $g+I = g'+I$ folgt, daß $f'', g'' \in I$ existieren mit $f+f''=f'$ und $g+g''=g'$. Somit ergibt sich für $\alpha \in \mathbb{K}$

$$\begin{aligned}
\alpha\cdot(f'+I) &= (\alpha f + \underbrace{\alpha f''}_{\in I}) + I = (\alpha f) + I,\\
(f'+I)+(g'+I) &= (f'+g')+I = (f+f''+g+g'')+I\\
&= ((f+g)+\underbrace{f''+g''}_{\in I}) + I = (f+g)+I
\end{aligned}$$

und analog für die Multiplikation

$$\begin{aligned}
(f'+I)\cdot(g'+I) &= (f'\cdot g')+I = ((f+f'')\cdot(g+g''))+I\\
&= (fg+\underbrace{f''g+fg''+f''g''}_{\in I})+I = (f\cdot g)+I.
\end{aligned}$$

Mit der Restklasse I als neutrales Element der Addition und $1+I$ als neutrales Element der Multiplikation bildet nun $\mathbb{K}[\underline{X}]/I$ offensichtlich eine $\mathbb{K}$-Algebra, genauer gesagt eine sog. Restklassenalgebra.

Also ist $\mathbb{K}[\underline{X}]/I$ insbesondere ein $\mathbb{K}$-Vektorraum. Folgender Satz zeigt, wie man eine Basis dazu findet:

Satz 2.2 *Sei $I_0 \subseteq \mathbb{K}[\underline{X}]$ ein echtes Ideal und $<$ eine beliebige Termordnung. Dann bildet die Menge*

$$\overline{RT(I_0)} := \{t+I_0 \mid t \in RT(I_0)\}$$

eine Basis des $\mathbb{K}$-Vektorraumes $\mathbb{K}[\underline{X}]/I_0$. Außerdem folgt für $t_1, t_2 \in RT(I_0)$ mit $t_1 \neq t_2$, daß $t_1 + I_0 \neq t_2 + I_0$.

Beweis: Siehe [2], Proposition 6.52. $\square$

Eine Basis von $\mathbb{K}[\underline{X}]/I$ ist nach Satz 2.2 damit $\overline{RT(I)} = \{v_1+I,\ldots,v_d+I\}$.

Das Hauptziel vom Rest dieses Kapitels ist nun, zu untersuchen, wie man in $\mathbb{K}[\underline{X}]/I$ effizient rechnen kann, insbesondere effizient die Multiplikation durchführen kann.

2.1.2 Darstellung der Restklassen von $\mathbb{K}[X_1,\ldots,X_n]/I$

In einer Restklassenalgebra werden die einzelnen Elemente üblicherweise nicht als ganze Klassen dargestellt, sondern man wählt in jeder Klasse einen geeigneten Repräsentanten für diese. Im Falle der Restklassenalgebra $\mathbb{K}[\underline{X}]/I$ wird folgendermaßen ein Repräsentant für jede Klasse bestimmt:

Sei $f+I \in \mathbb{K}[\underline{X}]/I$. Diese Restklasse wird repräsentiert durch die eindeutige Normalform g von f modulo G, d.h. f läßt sich mit der Gröbnerbasis G auf g reduzieren, wobei g nicht mehr reduzibel modulo G ist. Diese Bestimmung von g ist unabhängig von der Wahl von f. Sei etwa $f+I = f'+I$. Dann ist die eindeutige Normalform von f' modulo G ebenfalls g nach Lemma 1.14.

Ab jetzt werden also die Restklassen $f+I \in \mathbb{K}[\underline{X}]/I$ repräsentiert durch eindeutig bestimmte Elemente $g \in f+I \subseteq \mathbb{K}[\underline{X}]$, wobei g in Normalform modulo G ist. Für die Addition und Skalarmultiplikation ergeben sich dadurch keinerlei Schwierigkeiten, denn mit $f_1, f_2 \in \mathbb{K}[\underline{X}]$ in Normalform ist auch f_1+f_2 in Normalform. Ansonsten wäre f_1 oder f_2 reduzibel, also nicht in Normalform. Analog folgt für $\alpha \in \mathbb{K}$ und $f \in \mathbb{K}[\underline{X}]$ in Normalform, daß $\alpha \cdot f$ in Normalform ist, da sonst bereits f reduzibel wäre, d.h. nicht in Normalform wäre. Das Produkt zweier Normalformen ist hingegen i. a. nicht in Normalform.

Die Darstellung der einzelnen Klassen mittels eben erwähnter Repräsentanten kann man nun genauer gestalten mit Hilfe der Menge $RT(I)$ der reduzierten Terme. Nach Satz 2.2 bildet $\overline{RT(I)}$ eine Basis von $\mathbb{K}[\underline{X}]/I$, d.h. jede Restklasse $f+I \in \mathbb{K}[\underline{X}]/I$ läßt sich eindeutig darstellen als Linearkombination von Elementen der Form $v+I$ mit $v \in RT(I)$. Sei also $f+I \in \mathbb{K}[\underline{X}]/I$, dann existieren $\alpha_1,\ldots,\alpha_d \in \mathbb{K}$ mit

$$f+I = \sum_{i=1}^{d} \alpha_i \cdot (v_i+I) = \underbrace{\left(\sum_{i=1}^{d} \alpha_i \cdot v_i\right)}_{h:=} + I.$$

$h \in f+I$ ist als Linearkombination reduzierter Terme in Normalform modulo G, also der eindeutig bestimmte Repräsentant der Restklasse $f+I$ in Normalform. Daraus folgt, daß man jede Restklasse $f+I \in \mathbb{K}[\underline{X}]/I$ mit einer eindeutig bestimmten Linearkombination der reduzierten Terme repräsentieren kann. Es genügt also im Weiteren nun, Linearkombinationen der Form $\sum_{i=1}^{d} \alpha_i \cdot v_i$ zu betrachten anstelle ganzer Restklassen. Wie bereits erwähnt wurde, verursacht diese Darstellung keine Probleme bei der Addition und Skalarmultiplikation, da man für $\alpha \in \mathbb{K}, \sum_{i=1}^{d} \alpha_i \cdot v_i, \sum_{i=1}^{d} \gamma_i \cdot v_i$, mit $\alpha_i, \gamma_i \in \mathbb{K}$ sofort die Linearkombination der Summe bzw. eines skalaren Vielfachen erhält:

$$\begin{aligned}\Big(\sum_{i=1}^{d} \alpha_i \cdot v_i\Big) + \Big(\sum_{i=1}^{d} \gamma_i \cdot v_i\Big) &= \sum_{i=1}^{d} (\alpha_i + \gamma_i) \cdot v_i \ , \\ \alpha \cdot \Big(\sum_{i=1}^{d} \alpha_i \cdot v_i\Big) &= \sum_{i=1}^{d} (\alpha \cdot \alpha_i) \cdot v_i .\end{aligned}$$

Bei der Multiplikation zweier Linearkombinationen der Form $\sum_{i=1}^{d} \alpha_i \cdot v_i$ klappt die Berechnung der Normalform und der Linearkombinationsdarstellung des Produktes i. a. nicht so einfach. Dies wird im nächsten Abschnitt behandelt.

2.1.3 Multiplikation in $\mathbb{K}[X_1, \ldots, X_n]/I$

Im Gegensatz zur Addition zweier in Normalform befindlichen Elemente $f, g \in \mathbb{K}[\underline{X}]$ ist das Produkt $f \cdot g$ i. a. nicht in Normalform modulo G. Um den eindeutig bestimmten Repräsentanten der Restklasse $(fg) + I$ zu finden, muß man also noch eine Normalform des Produkts fg durch Reduktion mittels G berechnen. Mit Hilfe dieser Normalform wird diese Restklasse mit einem Polynom identifiziert, welches eine Linearkombination reduzierter Terme ist. Diese dabei auftretenden Reduktionsschritte können aber unter Umständen sehr zahlreich und dadurch sehr aufwendig werden, und daher einen nicht unerheblichen Zeitaufwand bei der Berechnung bewirken. Dieses Problem führt nun zur Einführung von sog. *Strukturkonstanten*, um die Multiplikation effizienter durchzuführen.

Seien nun zwei Polynome $f, g \in \mathbb{K}[\underline{X}]$ in Normalform gegeben, etwa

$$f = \sum_{i=1}^{d} \alpha_i \cdot v_i, \quad g = \sum_{j=1}^{d} \beta_j \cdot v_j \quad \text{mit } \alpha_i, \beta_j \in \mathbb{K}.$$

Dann gilt für das Produkt von f und g

$$f \cdot g = \Big(\sum_{i=1}^{d} \alpha_i \cdot v_i\Big) \cdot \Big(\sum_{j=1}^{d} \beta_j \cdot v_j\Big) = \sum_{i=1}^{d} \sum_{j=1}^{d} \alpha_i \beta_j \cdot v_i v_j .$$

Wenn also nun für alle $i, j \in \{1, \ldots, d\}$ bekannt ist, welche Normalformen die Terme $v_i v_j$ modulo G besitzen, so ist nach Lemma 1.15 auch automatisch sofort bekannt, welche Normalform $f \cdot g$ besitzt, da dieses Produkt eine Linearkombination von Termen der Form $v_i v_j$ ist. Denn sei die eindeutige Normalform von $v_i v_j$ modulo G die Summe $\sum_{k=1}^{d} \gamma_{i,j,k} \cdot v_k$ für $i, j \in \{1, \ldots, d\}$ mit $\gamma_{i,j,k} \in \mathbb{K}$, dann ist die eindeutige Normalform von $f \cdot g$:

$$\sum_{i=1}^{d} \sum_{j=1}^{d} \left(\alpha_i \beta_j \sum_{k=1}^{d} \gamma_{i,j,k} \cdot v_k \right) = \sum_{k=1}^{d} \Big(\sum_{i=1}^{d} \sum_{j=1}^{d} \alpha_i \beta_j \gamma_{i,j,k} \Big) \cdot v_k.$$

Um nun *beliebig viele* Multiplikation ohne zusätzliche Reduktionsschritte in $\mathbb{K}[\underline{X}]/I$ durchführen zu können, genügt es, die Koeffizienten $\gamma_{i,j,k}$ für alle $i, j, k \in \{1, \ldots, d\}$ *einmalig* zu bestimmen. Tatsächlich muss nicht jedes Paar $(v_i, v_j) \in RT(I) \times RT(I)$ betrachtet werden. Denn für $v_i, v_j, v_{i_1}, v_{j_1} \in RT(I)$ mit $v_i \cdot v_j = v_{i_1} \cdot v_{j_1}$ stimmen die Normalformen von $v_i v_j$ und $v_{i_1} v_{j_1}$ überein, d.h. in diesem Falle gilt $\gamma_{i,j,k} = \gamma_{i_1,j_1,k}$. Daher muß nicht für jedes Paar $(v_i, v_j) \in RT(I) \times RT(I)$ die Normalform von $v_i v_j$ berechnet werden, sondern es genügt die Berechnungen der Normalformen aller Elemente von

$$RT(I)^2 := \big\{ v_i \cdot v_j \mid v_i, v_j \in RT(I) \big\}.$$

Definition 2.3 (Kombinierte Strukturkonstanten) *Sei $u \in RT(I)^2$. Die eindeutige Normalform von u sei $h_u = \sum_{i=1}^{d} \beta_{u,v_i} \cdot v_i$ mit $\beta_{u,v_i} \in \mathbb{K}$. Dann heißt*

$$\hat{\beta} := \big\{ \beta_{u,v_i} \mid u \in RT(I)^2, v_i \in RT(I) \big\}$$

die Familie der (kombinierten) Strukturkonstanten.

Es wird sich im weiteren Verlauf dieses Kapitels herausstellen, daß die Berechnung der kombinierten Strukturkonstanten sehr effizient durchgeführt werden kann. Somit können Produkte in $\mathbb{K}[\underline{X}]/I$ nun wirklich wesentlich effizienter mit Hilfe von kombinierten Strukturkonstanten bestimmt werden als im Vergleich zur Multiplikation der Repräsentantenpolynome mit anschließender Reduktion auf Normalform. Außerdem sei noch einmal auf den Vorteil der Strukturkonstanten hingewiesen, daß sie nur einmalig bestimmt werden müssen und dann für beliebig viele Multiplikationen in der Restklassenalgebra $\mathbb{K}[\underline{X}]/I$ verwendet werden können, so daß keine weiteren Reduktionen mehr notwendig werden. Daher ist der Aufwand zur Bestimmung der Strukturkonstanten gerechtfertigt.

2.2 Verallgemeinerte Strukturkonstanten

Wie im letzten Abschnitt festgestellt wurde, können kombinierte Strukturkonstanten verwendet werden, um effizient in der Restklassenalgebra $\mathbb{K}[\underline{X}]/I$

die Multiplikation zweier Elemente durchzuführen. Dieses vorgestellte Prinzip der kombinierten Strukturkonstanten kann jedoch noch weiter verallgemeinert werden, was speziell für das später aufgeführte Quantoreneliminationsverfahren neu entworfen wurde. Diese Verallgemeinerung wird zunächst an einem Beispiel ausführlicher erläutert:

Beispiel 2.4 *Sei $h \in \mathbb{K}[\underline{X}]$ in Normalform modulo G gegeben. Seien ferner $v_i, v_j \in RT(I)$ beliebig, aber fest gegeben. Betrachte folgende Abbildung:*

$$f_{h,i,j} : \mathbb{K}[\underline{X}]/I \to \mathbb{K}[\underline{X}]/I : f + I \mapsto (f \cdot h \cdot v_i \cdot v_j) + I$$

Offensichtlich ist nach Abschnitt 2.1 die Abbildung $f_{h,i,j}$ linear. Da $\overline{RT(I)}$ eine Basis von $\mathbb{K}[\underline{X}]/I$ und endlich ist, existiert eine Matrixdarstellung $M_{h,i,j}$ von $f_{h,i,j}$, wobei die Spalten und Zeilen dieser Matrix mit den Elementen von $RT(I)$ indiziert werden. Es sollen nun für alle $i, j \in \{1, \ldots, d\}$ die Matrizen $M_{h,i,j}$ bestimmt werden. Dazu genügt es, die Bilder $f_{h,i,j}(v_l + I)$, $v_l \in RT(I)$, zu betrachten. Sei $v_l \in RT(I)$ und $v'_{i,j,l} = \sum_{k=1}^{d} \alpha_{i,j,k,l} \cdot v_k$, $\alpha_{i,j,k,l} \in \mathbb{K}$, eine Normalform von $v_l \cdot h \cdot v_i \cdot v_j$. Dann sind

$$\begin{aligned} f_{h,i,j}(v_l + I) &= \left(\sum_{k=1}^{d} \alpha_{i,j,k,l} \cdot v_k\right) + I = \sum_{k=1}^{d} \alpha_{i,j,k,l} \cdot (v_k + I) \quad \textit{und damit} \\ M_{h,i,j} &= (\alpha_{i,j,k,l})_{k,l \in \{1,\ldots,d\}}. \end{aligned}$$

Um die Matrizen $M_{h,i,j}$ zu bestimmen, müssen also wiederum Normalformen berechnet werden, und zwar von allen Polynomen $h_{i,j,l} := h \cdot v_i \cdot v_j \cdot v_l$ mit $i, j, l \in \{1, \ldots, d\}$. Da h in Normalform ist, läßt sich h schreiben als

$$h = \sum_{r=1}^{d} a_r \cdot v_r \quad \textit{mit } a_r \in \mathbb{K}.$$

Es gilt daher

$$h_{i,j,l} = h \cdot v_i \cdot v_j \cdot v_l = \sum_{r=1}^{d} a_r \cdot v_i v_j v_l v_r.$$

Anstatt jedes einzelne $h_{i,j,l}$ separat auf Normalform zu reduzieren oder die Produktberechnung durch iterierte Anwendung der Strukturkonstanten zu machen, genügt es nun, analog zum vorherigen Abschnitt einmalig für alle möglichen vierfachen Produkte von reduzierten Termen

$$v_i \cdot v_j \cdot v_l \cdot v_r \quad \textit{mit } i, j, l, r \in \{1, \ldots, d\}$$

eine Normalform modulo G zu bestimmen. Daraus können direkt ohne weitere Reduktionsschritte mittels Satz 1.15 die Normalformen der $h_{i,j,l}$ bestimmt werden. Es wird sich zeigen, daß die Bestimmung der Normalformen vierfacher Produkte reduzierter Terme effizient durchgeführt werden kann, so daß deren Verwendung bei der Berechnung der Normalformen der $h_{i,j,l}$ vorzuziehen ist, anstatt jedes $h_{i,j,l}$ separat auf Normalform zu reduzieren. $\diamond$

Die in Beispiel 2.4 neu entworfene vorgestellte Verallgemeinerung der kombinierten Strukturkonstanten läßt sich auf m-fache Produktbildung, $m \geq 2$, von reduzierten Termen analog erweitern, d.h. es müssen dann zu allen möglichen m-fachen Produkten $\prod_{k=1}^{m} v_{i_k}$ mit $v_{i_1}, \ldots, v_{i_m} \in RT(I)$ deren Normalformen berechnet werden. Wiederum folgt, daß man tatsächlich nur Normalformen aller Terme von

$$RT(I)^m := \left\{ v_{i_1} \cdots v_{i_m} | v_{i_1}, \ldots, v_{i_m} \in RT(I) \right\}$$

bestimmen muß. Denn für $u = v_{i_1} \cdots v_{i_m}, w = v_{j_1} \cdots v_{j_m}$ mit $v_{i_1}, \ldots, v_{i_m}$, $v_{j_1}, \ldots, v_{j_m} \in RT(I)$ und $u = w$ gilt, daß die Normalformen von u und w identisch sind, beide Normalformen also dieselbe Linearkombination in den reduzierten Termen besitzen.

Definition 2.5 (Kombinierte Strukturkonstanten m-ter Ordnung) *Sei $m \in \mathbb{N}, m \geq 2$. Sei $u \in RT(I)^m$. Die eindeutige Normalform von u sei $h_u = \sum_{i=1}^{d} \beta_{u,v_i}^{(m)} \cdot v_i$. Dann heißt*

$$\hat{\beta}^{(m)} := \left\{ \beta_{u,v_i}^{(m)} \mid u \in RT(I)^m, v_i \in RT(I) \right\}$$

die Familie der (kombinierten) Strukturkonstanten m-ter Ordnung.

Bemerkung: Definition 2.5 ist verträglich mit Definition 2.3, welche sich lediglich auf den Spezialfall $m = 2$ bezieht. In diesem Spezialfall wird für die Familie $\hat{\beta}^{(2)}$ auch kurz $\hat{\beta}$ geschrieben. Die Verwendung von Strukturkonstanten m-ter Ordnung mit $m > 2$ empfiehlt sich vor allem dann, wenn in mehreren zu berechnenden Produkten stets die Anzahl der Faktoren gleich ist und mehrere Faktoren reduzierte Terme sind. Dadurch kann man vermeiden, daß Produkte durch iterierte Anwendung der Strukturkonstanten 2-ter Ordnung berechnet werden.

Wenn aus dem Zusammenhang offensichtlich ist, bzgl. welcher Ordnung man von Strukturkonstanten spricht, so wird die Angabe der Ordnung der besseren Lesbarkeit wegen vernachläßigt.

In den folgenden beiden Lemmata werden zwei einfache Eigenschaften der kombinierten Strukturkonstanten m-ter Ordnung gezeigt.

Lemma 2.6 *Sei $m \in \mathbb{N}, m \geq 2$, sei $t \in RT(I)^m \cap RT(I)$ und $v_i \in RT(I)$. Dann ist*

$$\beta_{t,v_i}^{(m)} = \begin{cases} 1 & \text{falls } t = v_i \\ 0 & \text{sonst} \end{cases} .$$

Beweis: Jedes Element $f \in \mathbb{K}[\underline{X}]$ besitzt eine eindeutige Normalform f_{NF} modulo der Gröbnerbasis G. Diese Normalform läßt sich eindeutig als Linearkombination in den reduzierten Termen darstellen:

$$f_{NF} = \sum_{i=1}^{d} a_i \cdot v_i \quad \text{ mit } a_i \in \mathbb{K}.$$

Da $t \in RT(I)$, folgt aus der Eindeutigkeit dieser Darstellung sofort die Behauptung des Lemmas. □

Lemma 2.7 *Sei $u \in RT(I)^m$ und $u \notin RT(I)$. Dann ist $\beta^{(m)}_{u,v_i} = 0$ für alle $v_i \in RT(I)$ mit $v_i \geq u$.*

Beweis: Da $u \notin RT(I)$, ist der Term u topreduzibel. Sei $h = \sum_{i=1}^{d} a_i \cdot v_i$, $a_i \in \mathbb{K}$, eine Normalform von u. Dann gilt, daß $u > HT(h)$. Insbesondere folgt daraus, daß $a_i = 0$ für alle $i \in \{1, \ldots, d\}$ mit $v_i \geq u > HT(h)$. □

Nach diesen beiden vorbereitenden Lemmata wird nun ein effizienter Algorithmus vorgestellt, welcher die kombinierten Strukturkonstanten m-ter Ordnung eines gegebenen nulldimensionalen Ideals $I \subseteq \mathbb{K}[X_1, \ldots, X_n]$ explizit bestimmt.

Satz 2.8 *Sei $2 \leq m \in \mathbb{N}$. Sei $I \subseteq \mathbb{K}[X_1, \ldots, X_n]$ ein echtes nulldimensionales Ideal, G eine Gröbnerbasis von I bzgl. einer beliebigen Termordnung $<$. Sei dazu $RT(I) = \{v_1, \ldots, v_d\}$ die Menge der reduzierten Terme von I. Dann konstruiert der Algorithmus COMBSTRCONST aus Abbildung 2.1 die Familie $\hat{\beta}^{(m)}$ der kombinierten Strukturkonstanten m-ter Ordnung.*

Beweis:
Termination: Die Termination des Algorithmus ist offensichtlich, da beim ersten Eintritt in die *while*-Schleife die Menge U endlich ist und deren Kardinalität bei jedem Durchlauf der Schleife um eins abnimmt, also letztendlich leer wird. Alle auftretenden *for all*-Schleifen sind über die endliche Menge der reduzierten Terme $RT(I)$ indiziert. Also terminiert der Algorithmus.

Korrektheit: Es ist offensichtlich, daß der Algorithmus alle relevanten Paare von $RT(I)^m \times RT(I)$ berechnet. Nach Lemma 2.6 folgt sofort die Korrektheit des *if*-Falles. Im *elsif*-Fall wird explizit eine Normalform des Elements $u \in RT(I)^m$ modulo der gegebenen Gröbnerbasis G berechnet und dann die Linearkombination in den reduzierten Termen betrachtet. Die jeweiligen Koeffizienten der reduzierten Terme $v_i \in RT(I)$ werden somit richtig den $\beta^{(m)}_{u,v_i}$ in Zeile 15 zugeordnet.
Zu zeigen bleibt noch die Korrektheit des *else*-Falles. In diesem Fall ist der Term u ein echtes Vielfaches eines Elementes von $HT(G)$. Daher existiert ein Term $u' \notin RT(I)$ und ein $j \in \{1, \ldots, n\}$, so daß $u = u' \cdot X_j$. Weiters

folgt aus $u \in RT(I)^m$ und $u = u' \cdot X_j$, daß $u' \in RT(I)^m$ und $X_j \in RT(I)$. Denn sei

$$u = \prod_{i=1}^{m} r_i \quad \text{mit } r_i \in RT(I)$$

und sei o.B.d.A. $r_1 = r_1' \cdot X_j$, dann gilt

$$u' = r_1' \cdot \prod_{i=2}^{m} r_i.$$

Da ein Teilfaktor eines reduzierten Termes wieder ein reduzierter Term ist, folgt daraus, daß $u' \in RT(I)^m$ und ebenso $X_j \in RT(I)$. Also ist die Strukturkonstante $\beta^{(m)}_{u',w}$, $w \in RT(I)$, wohldefiniert. Da I ein echtes Ideal ist, ist $1 \in RT(I)$ und somit $1 \in RT(I)^{m-2}$. Also ist für $w \in RT(I)$

$$w \cdot X_j = 1 \cdot w \cdot X_j \in RT(I)^m,$$

woraus die Wohldefiniertheit von $\beta^{(m)}_{w \cdot X_j, v_i}$ folgt. Aus $1 < X_j$ bzw. $w < u'$ folgt aus der Definition einer Termordnung, daß

$$u' < u \quad \text{bzw.} \quad w \cdot X_j < u' \cdot X_j = u.$$

Damit sind die Werte $\beta^{(m)}_{u',w}$ und $\beta^{(m)}_{w \cdot X_j, v_i}$ bei der Berechnung von $\beta^{(m)}_{u,v_i}$ bereits berechnet und sind somit direkt verfügbar, um die Summe in Zeile 20 berechnen zu können. Es bleibt somit nur noch zu zeigen übrig, daß die berechnete Summe wirklich die gewünschte Strukturkonstante $\beta^{(m)}_{u,v_i}$ ist:

$$\begin{aligned}
u &= u' \cdot X_j \\
&\overset{2.7}{\equiv}_I \Big(\sum_{w \in RT(I), w < u'} \beta^{(m)}_{u',w} \cdot w \Big) \cdot X_j \\
&= \sum_{w \in RT(I), w < u'} \beta^{(m)}_{u',w} \cdot w X_j \\
&\equiv_I \sum_{w \in RT(I), w < u'} \beta^{(m)}_{u',w} \cdot \Big(\sum_{i=1}^{d} \beta^{(m)}_{w \cdot X_j, v_i} \cdot v_i \Big) \\
&= \sum_{i=1}^{d} \Big(\sum_{w \in RT(I), w < u'} \beta^{(m)}_{u',w} \cdot \beta^{(m)}_{w \cdot X_j, v_i} \Big) \cdot v_i.
\end{aligned}$$

Aus obiger Gleichung folgt somit

$$\beta^{(m)}_{u,v_i} = \sum_{w \in RT(I), w < u'} \beta^{(m)}_{u',w} \cdot \beta^{(m)}_{w \cdot X_j, v_i}. \quad \square$$

Algorithmus COMBSTRCONST:

Beschreibung: Berechnung der kombinierten Strukturkonstanten m-ter Ordnung eines nulldimensionalen Ideals

Eingabe: G, $RT(I)$ und m mit:

- Gröbnerbasis G eines echten nulldimensionalen Ideals I bzgl. einer beliebigen Termordnung $<$
- $RT(I)$, die Menge der reduzierten Terme des Ideals I bzgl. $<$
- $m \in \mathbb{N}, m \geq 2$, die Ordnung der kombinierten Strukturkonstanten

Ausgabe: Familie $\hat{\beta}^{(m)}$ der kombinierten Strukturkonstanten der Ordnung m

begin
$U := RT(I)^m$
$V := RT(I)$
$B := (\beta_{u,v_i})_{u \in RT(I)^m, v_i \in RT(I)}$ (d.h. Erstellung einer Matrix, deren Zeilen mit Elementen aus $RT(I)^m$ und deren Spalten mit Elementen aus $RT(I)$ indiziert werden)
while $U \neq \emptyset$ **do**
    $u :=$ „Wähle das kleinste Element von U bzgl. $<$."
    $U := U \backslash \{u\}$
    if $u \in RT(I)$ **then**
      for all $v_i \in V$ **do**
        $\beta^{(m)}_{u,v_i} := \begin{cases} 1 & \text{falls } u = v_i \\ 0 & \text{sonst} \end{cases}$
      end
    elsif $u \in HT(G)$ **then**
      $h :=$ „Normalform von u modulo G"
      for all $v_i \in V$ **do**
        $\beta^{(m)}_{u,v_i} := \begin{cases} \text{„Koeffizient von } v_i \text{ in } h\text{"} & \text{falls } v_i \in T(h) \\ 0 & \text{sonst} \end{cases}$
      end
    else
      $u' :=$ „Wähle $u' \in T(\underline{X})$ derart, daß $u = u' \cdot X_j$ für ein $j \in \{1, \ldots, n\}$ und $u' \notin V$."
      for all $v_i \in V$ **do**
        $\beta^{(m)}_{u,v_i} := \sum_{w \in V, w < u'} \beta^{(m)}_{u',w} \cdot \beta^{(m)}_{w \cdot X_j, v_i}$
      end
    end
end
return B
end COMBSTRCONST

Abbildung 2.1: Algorithmus zur Berechnung der kombinierten Strukturkonstanten m-ter Ordnung

Bemerkungen:
Der Algorithmus COMBSTRCONST ist eine Verallgemeinerung des in [2], Table 9.5, aufgeführten Algorithmus STRCONST, da STRCONST lediglich Strukturkonstanten zweiter Ordnung berechnet und außerdem als Argument eine *reduzierte* Gröbnerbasis benötigt. Bei einer Anwendung von COMBSTRCONST ist es nicht notwendig, daß die Eingabegröbnerbasis reduziert ist. Deshalb müssen ggf. Reduktionen in Zeile 13 von COMBSTRCONST durchgeführt werden, bis das jeweilige Produkt der reduzierten Terme in Normalform modulo der Gröbnerbasis G ist. Wenn aber nun eine reduzierte Gröbnerbasis G_{red} vorliegt, mit welcher COMBSTRCONST aufgerufen wird, so entfallen diese Ketten von Reduktionsschritten. Die in Zeile 13 zu berechnenden Polynome h sind dann in diesem Falle von der Form

$$u - g \quad \text{mit } u \in RT(I)^m, g \in G_{red} \text{ und } HT(g) = u.$$

Da jetzt eine reduzierte Gröbnerbasis vorliegt, ist $u-g$ nicht mehr reduzibel, also in Normalform modulo G_{red}. Denn angenommen $u - g$ wäre nicht in Normalform modulo G_{red}, also $u - g$ reduzibel. Wegen $T(u - g) = T(g)\backslash\{u\}$ wäre auch g reduzibel modulo $G_{red}\backslash\{g\}$, ein Widerspruch. Somit arbeitet der Algorithmus effizienter, falls eine reduzierte Gröbnerbasis vorliegt, da nun keine Reduktionsketten mehr auftreten, deren Berechnung evtl. sehr zeitintensiv werden können.

Wenn jedoch zum Beispiel aus Kostengründen auf eine explizite Berechnung einer reduzierten Gröbnerbasis G_{red} aus einer gegebenen Gröbnerbasis G verzichtet wird, so kann man die Anzahl der Polynome, welche man in Zeile 13 bis auf Normalform reduzieren muß, dennoch verringern. Diese Anzahl ist zunächst durch $|HT(G)|$ nach oben beschränkt, da man höchstens $|HT(G)|$-mal in den *elsif*-Fall eintreten kann. Es gilt natürlich $|HT(G)| \leq |G|$. Lemma 1.4 zeigt, wie man mit Hilfe des Algorithmus SMALLERGB eine gegebene Gröbnerbasis G bzgl. der Inklusionsbeziehung verkleinern kann, damit man die Anzahl der Eintritte in den *elsif*-Fall weiter verringern kann.

Mit dem Algorithmus SMALLERGB aus Abbildung 1.1 wurde nun eine Möglichkeit vorgestellt, um die Kardinalität einer gegebenen Gröbnerbasis in einem gewissen Maße zu verkleinern, so daß Strukturkonstanten effizienter berechnet werden können, ohne daß man gleich möglicherweise aufwendig eine reduzierte Gröbnerbasis berechnet. Diese Strategie bietet sich als Mittelweg an, wenn sich die Frage stellt, ob man beim Aufruf von COMBSTRCONST eine nicht-reduzierte Gröbnerbasis verwenden soll oder zunächst daraus eine reduzierte Gröbnerbasis berechnen soll, mit welcher man dann COMBSTRCONST aufruft. In den durchgerechneten Beispielen der vorliegenden Arbeit wirkte sich die Tatsache, daß reduzierte Gröbnerbasen explizit berechnet wurden, bei der Berechnung der Strukturkonstanten weder positiv noch negativ aus.

Der Algorithmus aus Abbildung 2.1 zeigt, daß kombinierte Strukturkonstanten m-ter Ordnung sehr effizient bestimmt werden können und damit deren Verwendung, wie in Beispiel 2.4, absolut rechtfertigt. In Beispiel 2.4 verdeutlicht sich die Wirksamkeit deren Verwendung, wenn man jene Matrizen mit mehreren unterschiedlichen Polynomen $h \in \mathbb{K}[\underline{X}]$ bestimmen möchte. Denn ein sehr großer Vorteil der Verwendung von kombinierten Strukturkonstanten m-ter Ordnung ist, daß sie nur einmalig bestimmt werden müssen und dann beliebig oft benutzt werden können. Somit entfallen nach der Bestimmung der kombinierten Strukturkonstanten m-ter Ordnung bei den Normalformberechnungen von Polynomen die evtl. langen Reduktionsketten, wie sie auftreten würden, wenn man Strukturkonstanten m-ter Ordnung nicht verwenden würde.

2.3 Zusammenfassung und Vergleich

Es wurde nun eine Methode vorgestellt, um mit Hilfe von Strukturkonstanten m-ter Ordnung, die Multiplikation in der Restklassenalgebra $\mathbb{K}[\underline{X}]/I$ effizient durchführen zu können. Dabei stellte sich heraus, daß durch die Verwendung jener bei der Multiplikation nun keine Reduktionsschritte mehr nötig sind, welche sich in der Praxis zum Teil als sehr lang und aufwendig herausstellen. Außerdem ist ein sehr großer Vorteil der Strukturkonstanten, daß sie nur einmalig berechnet werden müssen und dann beliebig oft verwendet werden können.

Es wurde auch gezeigt, wie man die Berechnung der Strukturkonstanten verbessern kann, indem man die gegebene Gröbnerbasis bzgl. der Inklusion verkleinert, so daß man nicht mehr Reduktionsschritte bei der Bestimmung der Strukturkonstanten macht als nötig. Es wurde auch gezeigt, daß reduzierte Gröbnerbasen als Eingabeargument zu einer noch effizienteren Berechnung von Strukturkonstanten verhelfen.

Die Verwendung der Strukturkonstanten m-ter Ordnung rentiert sich hingegen nicht, wenn nur wenige zu bestimmende Produkte von Restklassen vorliegen, insbesondere wenn in diesem Fall die Menge $RT(I)$ sehr groß ist. Dann würde die Bestimmung der Strukturkonstanten mehr Aufwand bedeuten, als wenn man das Produkt mittels Produktbildung der Repräsentantenpolynome mit anschließender Reduktion auf Normalform berechnet. Deshalb empfiehlt sich die Verwendung von Strukturkonstanten vor allem dann, wenn viele Produkte zu berechnen sind, insbesondere wenn alle zu berechnenden Produkte dieselbe Anzahl von Faktoren besitzen und viele Faktoren reduzierte Terme sind.

An folgenden drei Beispielen erkennt man deutlich, welchen Effizienzgewinn man durch Verwendung von Strukturkonstanten erhalten kann. In allen Beispielen wird für gegebene Polynome h und für alle $i, j \in \{1, \ldots, d\}$ eine explizite Berechnung der Spuren aller Matrizen $M_{h,i,j}$ aus Beispiel 2.4 erfolgen.

Beispiel 2.3.1: Sei $G = \{X^5 + X^2 + X + 1, Y^5\} \subseteq \mathbb{Q}[X, Y]$. Dann ist G eine Gröbnerbasis bzgl. $<_{revgradlex}$ mit $X < Y$. Seien

$$\begin{array}{lll} h_1 = X^3 + X^2, & h_2 = Y^{13} + Y^5 + 10, & h_3 = X^{13} + X + Y^{12}, \\ h_4 = Y^{10} + Y^3, & h_5 = X^{10} + X^6 + X, & h_6 = X^{12}Y^2 + X^{10}Y^5, \\ & h = \prod_{i=1}^{6} h_i^2. & \end{array}$$

Für dieses Polynom h werden die Spuren der Matrizen $M_{h,i,j}$ mit Verwendung von Strukturkonstanten 4-ter Ordnung in 560 Millisekunden berechnet. Ohne Verwendung von Strukturkonstanten benötigt man hingegen 51,7 Sekunden, wenn man Produkte in der Restklassenalgebra $\mathbb{Q}[X, Y]/Id(G)$ durch Produktbildung der Repräsentantenpolynome und anschließender Reduktion auf Normalform berechnet. $\diamond$

Beispiel 2.3.2: Sei $G = \{X^5 + X + 1, Y^4, Z^3 + Z\} \subseteq \mathbb{Q}[X, Y, Z]$. Dann ist G eine Gröbnerbasis bzgl. $<_{revgradlex}$ mit $X < Y < Z$. Seien ferner

$$g = X^5 + Y^4 + Z^3, \quad h_1 = X^{10}Y^3 - YZ, \quad h_2 = X^4YZ^{14} - XYZ \text{ und}$$

$$h = g \cdot h_1^2 \cdot h_2^2 \quad , \qquad h' = g^2 \cdot h_1^2 \cdot h_2^2.$$

Es sollen nun für alle $i, j \in \{1, \ldots, d\}$ jeweils die Spuren der Matrizen $M_{h,i,j}$ und $M_{h',i,j}$ berechnet werden. Bei Verwendung von Strukturkonstanten erhält man das Ergebnis in 83 Sekunden, ohne Verwendung von Strukturkonstanten wird das Ergebnis in 159 Sekunden berechnet. Im ersten Fall werden dafür 18 Sekunden benötigt, um die Strukturkonstanten 4-ter Ordnung zu berechnen. $\diamond$

Beispiel 2.3.3: Eine kleine Abänderung des letzten Beispiels zeigt einen noch deutlicheren Zeitgewinn durch die Verwendung von Strukturkonstanten. Sei $G = \{X^5 + X + 1, Y^4 - 1, Z^3 + Z + 1\} \subseteq \mathbb{Q}[X, Y, Z]$. Dann ist G eine Gröbnerbasis bzgl. $<_{revgradlex}$ mit $X < Y < Z$. Seien weiter

$$g = X^5 + Y^4 + Z^3, \quad h_1 = X^{10}Y^3 - YZ,$$

$$h_2 = X^4YZ^{14} - XYZ - 2X - 3Y - 4XZ \text{ und}$$

$$h = g \cdot h_1^2 \cdot h_2^2 \quad , \qquad h' = g^2 \cdot h_1^2 \cdot h_2^2.$$

Es sollen nun wiederum die Spuren der Matrizen $M_{h,i,j}$ und $M_{h',i,j}$ berechnet werden. Bei Verwendung von Strukturkonstanten erhält man das Ergebnis in 110 Sekunden, ohne Verwendung von Strukturkonstanten wird das Ergebnis in ca. 212 Minuten berechnet. $\diamond$

Im Kapitel 7 finden sich weitere Zeitvergleiche von Beispielen mit und ohne Verwendung von Strukturkonstanten.

Kapitel 3

Hermitesche Quantorenelimination

In diesem Kapitel wird ein Verfahren zur Quantorenelimination von Formeln erster Stufe in der Theorie der reellen Zahlen durch parametrisches Zählen reeller Nullstellen vorgestellt. Dieses Verfahren wurde von V. Weispfenning in [15] veröffentlicht; eine Ausarbeitung davon mit der Herleitung aller relevanten mathematischen Grundlagen, insbesondere im Zusammenhang mit Typformeln, findet sich in der Diplomarbeit [4] von Andreas Dolzmann, auf welche im weiteren Verlauf desöfteren verwiesen werden wird und auf welcher aufbauend Optimierungsmöglichkeiten vorgestellt werden. Dieses Quantoreneliminationsverfahren stützt sich auf die Bestimmung der Anzahl reeller Nullstellen eines nulldimensionalen Ideals unter Berücksichtigung endlich vieler zu erfüllenden Nebenbedingungen. Das verwendete Verfahren zum Zählen von reellen Nullstellen mit Berücksichtigung von Nebenbedingungen wird in den Arbeiten von Becker und Wörmann [1] bzw. Pedersen, Roy und Szpirglas [10] beschrieben, wobei die Grundidee zu diesem Verfahren auf den frz. Mathematiker Hermite zurückgeht.

Im folgenden betrachtet man für die Sprache der Terme die Sprache

$$\mathcal{L} = (0, 1, +, -, \ \cdot\ ; \leq)$$

der angeordneten Ringe. Man nimmt ferner die ganzen Zahlen als zusätzliche Konstanten hinzu. Somit können die auftretenden Terme als Polynome in einem Polynomring über den ganzen Zahlen $\mathbb{Z}$ betrachtet werden, wobei die Variablen der Terme den Variablen des Polynomrings entsprechen. Außerdem läßt man zur Bildung atomarer Formeln alle Relationen aus $\{<, \leq, =, \neq, \geq, >\}$ zu. Hier kommt als Vereinbarung o.B.d.A. hinzu, daß in atomaren Formeln auf der rechten Seite immer 0 steht. Dies stellt keine Einschränkung dar, da jede atomare Formel in diese gewünschte Gestalt äquivalent umgeformt werden kann. So wird z.B. für Terme t_1, t_2 aus $t_1 \geq t_2$ die atomare Formel $t_1 + (-t_2) \geq 0$.

Für eine gegebene Eingabeformel φ, zu der die Menge der darin auftretenden Variablen $\{X_1, \ldots, X_n\}$ ist, werden zunächst die Terme als Elemente des Polynomrings $\mathbb{Z}[X_1, \ldots, X_n]$ betrachtet, wobei man bei den Variablen später noch nach Paramatern und „echten" Variablen unterscheiden wird.

Ziel ist die Berechnung einer quantorenfreien Formel ψ zu einer gegebenen Eingabeformel φ erster Stufe, so daß in der Theorie der reellen Zahlen gilt:

$$\varphi \Longleftrightarrow \psi,$$

d.h. daß ψ eine zu φ äquivalente quantorenfreie Formel ist.

Als abkürzende Notation steht ab jetzt $\underline{x}$ für ein n-Tupel von Variablen $(x_1, \ldots, x_n)$ und $\underline{U}$ für ein m-Tupel von Parametern $(U_1, \ldots, U_m)$. Für eine Formel φ ist dann als weitere abkürzende Schreibweise

$$\exists \underline{x}(\varphi) := \exists x_1 \ldots \exists x_n(\varphi)$$

definiert. Wenn $\underline{U}$ die frei auftretenden Variablen in φ sind und $\underline{u} \in \mathbb{R}^m$, dann bezeichnet $\varphi(\underline{u})$ diejenige Formel, welche aus der zu φ zugehörigen erweiterten Formel $\varphi(\underline{U})$ entsteht, in dem man $\underline{u}$ für die Variablen $\underline{U}$ einsetzt.

3.1 Reduktion auf Eliminationsnormalform

Das in diesem Kapitel beschriebene Hermitesche Quantoreneliminationsverfahren wird die Quantorenelimination einer beliebigen Formel erster Stufe auf die iterierte Quantorenelimination von Formeln in sog. *Eliminationsnormalform* zurückführen.

Definition 3.1 (Eliminationsnormalform) *Eine Formel erster Stufe ist in Eliminationsnormalform, wenn sie von folgender Form ist:*

$$\exists x_1 \ldots \exists x_n \left(\bigwedge_{i=1}^{r} f_i(\underline{U}, \underline{x}) = 0 \wedge \bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0 \right)$$

mit $n, r, s, t \in \mathbb{N}, n > 0, f_1, \ldots, f_r, g_1, \ldots, g_s, h_1, \ldots, h_t \in \mathbb{Z}[\underline{U}][X_1, \ldots, X_n] \backslash \{0\}$. *Dabei dürfen* r, s, t *auch* 0 *sein.*

Mit dieser Definition identifiziert man also alle auftretenden Terme in einer Formel in Eliminationsnormalform als Elemente von

$$\mathbb{Z}[\underline{U}][\underline{X}] := \mathbb{Z}[U_1, \ldots, U_m][X_1, \ldots, X_n],$$

wobei die gebundenen Variablen $x_1, \ldots, x_n$ mit den Hauptvariablen $X_1, \ldots, X_n$ des Polynomrings identifiziert werden und die freien Variablen $U_1, \ldots, U_m$ als Parameter in diesem Polynomring betrachtet werden. Es müssen ab jetzt für den Rest der vorliegenden Arbeit ausschließlich nur noch reelle Spezialisierungen $\underline{u} \in \mathbb{R}^m$ der Parameter $\underline{U}$ betrachtet werden, da die Quantorenelimination in der Theorie der reellen Zahlen durchgeführt wird.

Im Kapitel 3.4 wird beschrieben werden, wie man die Quantoren einer Formel in Eliminationsnormalform eliminieren kann. Dieser Abschnitt wird nun zeigen, wie man die Quantorenelimination einer beliebigen Eingabeformel erster Stufe auf die Quantorenelimination von Formeln in Eliminationsnormalform zurückführen kann. Dazu müssen einige logische Grundlagen erwähnt werden:

Definition 3.2 (Disjunktive Normalform) *Eine Formel erster Stufe ist in disjunktiver Normalform, falls sie von der Form*

$$\bigvee_{i=1}^{k} \Big(\bigwedge_{j=1}^{l_k} \varphi'_{i,j} \Big)$$

ist, mit $k, l_k \in \mathbb{N}$ und atomaren Formeln/negierten atomaren Formeln $\varphi'_{i,j}$.

Lemma 3.3 *Sei φ eine quantorenfreie Formel. Dann existiert eine zu φ äquivalente Formel φ' in disjunktiver Normalform.*
φ' kann algorithmisch bestimmt werden. $\square$

Definition 3.4 (Pränexe Normalform) *Eine Formel erster Stufe ist in pränexer Normalform, wenn sie folgende Form besitzt:*

$$Q_1 x_1 \; Q_2 x_2 \ldots Q_n x_n \; (\varphi),$$

wobei $Q_i \in \{\exists, \forall\}$ für $i \in \{1, \ldots, n\}$ und φ eine quantorenfreie Formel ist.

Lemma 3.5 *Sei φ eine Formel erster Stufe. Dann existiert eine Formel φ' in pränexer Normalform, so daß φ und φ' zueinander äquivalent sind.*
φ' kann algorithmisch bestimmt werden. $\square$

Definition 3.6 (Positive Normalform, normalisierte Relation) *Sei φ eine quantorenfreie Formel.*
a) φ ist in positiver Normalform, falls φ den logischen Operator „$\neg$" nicht enthält.
b) φ enthält nur normalisierte Relationen, falls die atomaren Formeln von φ nur aus den Relationen „$=$", „$>$" und „$\neq$" bestehen.
c) φ enthält bzgl. einer Variablenmenge V normalisierte Relationen, falls alle atomaren Formeln, die von V abhängen, nur aus den Relationen „$=$", „$>$" und „$\neq$" bestehen.

Bemerkungen: Jede quantorenfreie Formel über der gegebenen Sprache der angeordneten Ringe läßt sich äquivalent in positive Normalform überführen: Man zieht die Negationen mit Hilfe der De-Morgan'schen Regel bis vor die atomaren Formeln rein und ersetzt dann die jeweiligen atomaren Formeln je nach Relation durch äquivalente Formeln ohne „$\neg$", wie es in Tabelle 3.1 aufgeführt wird.
Ebenso läßt sich jede quantorenfreie Formel in eine äquivalente Formel, welche nur normalisierte Relationen enthält bzw. für eine beliebige Variablenmenge nur bzgl. dieser Menge normalisierte Relationen enthält, umformen. Tabelle 3.2 verdeutlicht dies hier: Jede auftretende atomare Formel, die eine der Relationen „$\geq$", „$\leq$" oder „$<$" enthält, wird durch eine äquivalente Formel ersetzt, welche nur normalisierte Relationen enthält. Insbesondere sieht man aus Tabelle 3.2, daß eine Formel in positiver Normalform nach der Normalisierung der Relationen in positiver Normalform bleibt.

Für einen beliebigen Term t gilt:

Eingabeformel	Äquivalente positive Normalform
$\neg(t < 0)$	$t \geq 0$
$\neg(t \leq 0)$	$t > 0$
$\neg(t = 0)$	$t \neq 0$
$\neg(t \neq 0)$	$t = 0$
$\neg(t \geq 0)$	$t < 0$
$\neg(t > 0)$	$t \leq 0$

Abbildung 3.1: Bildung der positiven Normalform

Für einen beliebigen Term t gilt:

Eingabeformel	Äquivalente Formel mit normalisierten Relationen
$t < 0$	$-t > 0$
$t \leq 0$	$-t > 0 \vee t = 0$
$t \geq 0$	$t > 0 \vee t = 0$

Abbildung 3.2: Normalisierung der Relationen

Im folgenden Satz wird gezeigt, wie man die Quantorenelimination einer beliebigen Formel erster Stufe auf iterierte Quantorenelimination von Formeln in Eliminationsnormalform reduziert.

Satz 3.7 *Sei φ eine Formel erster Stufe, deren Quantoren eliminiert werden sollen. Sei* **QENF**(β,φ') *ein Algorithmus, der für eine Formel erster Stufe φ' in Eliminationsnormalform und eine Konjunktion atomarer Formeln*

$$\beta = \bigwedge_{l=1}^{q} p_l \ \varrho_l \ 0 \ , \quad \varrho_l \in \{<, \leq, =, \neq, \geq, >\}, \ q \in \mathbb{N},$$

wobei die Terme p_l keine in φ' gebunden auftretenden Variablen enthalten, eine quantorenfreie Formel ψ zurückgibt, so daß gilt: $\beta \wedge \varphi' \Leftrightarrow \beta \wedge \psi$. Dann eliminiert der Algorithmus in Abbildung 3.3 die Quantoren von φ.

Beweis:
Termination: Die Termination des Algorithmus ist offensichtlich, da alle *for*-Schleifen über endliche Mengen laufen, nämlich über die endliche Anzahl von Quantorenblöcken und von Disjunktionsglieder in einer Disjunktion.

Korrektheit: Die Zuweisungen von φ' in den Zeilen 3-6 sind einfache Äquivalenzumformungen der Eingabeformel φ mit Hilfe der Aussagenlogik und Lemma 3.5. Weitere Berechnungen können in Zeile 11/12 abgebrochen werden, wenn in Zeile 10 die Matrix von φ' gleich *true* oder *false* ist, da in diesem Fall die Matrix von φ' bereits das Eliminationsergebnis von φ' ist. Nun wird mit Hilfe einer Schleifeninvariante gezeigt, daß die *for*-Schleife aus Zeile 14 korrekt arbeitet.

Behauptung: Nach dem k-ten Schleifendurchlauf mit $0 \leq k < b$ gilt:

$$\varphi \Leftrightarrow Q_b \ldots Q_{k+1}(\psi),$$

wobei jedes Q_i ein Block von Existenz- bzw. Allquantoren der Form $\exists \underline{x_i}$ bzw. $\forall \underline{x_i}$ in den gebundenen Variablen $\underline{x_i} := (x_{i_1}, \ldots, x_{i_{n_i}})$ ist.

Beweis der Behauptung: Zu Beginn des ersten Schleifendurchlaufs gilt diese Äquivalenz, da beim ersten Eintritt in die Schleife ψ genau die Matrix von φ' ist, und φ äquivalent zu φ' ist. Gelte nun

$$\varphi \Leftrightarrow Q_b \ldots Q_k(\psi) \quad , \quad k < b,$$

vor dem k-ten Schleifendurchlauf. Da man letzlich mit `QENF` einen Existenzquantorblock eliminieren will, muß man zunächst ψ negieren, falls Q_k ein Block von Allquantoren ist. Dies bezieht sich auf die Äquivalenz, welche allgemein für eine Formel ϕ gilt:

$$\forall x(\phi) \Leftrightarrow \neg \exists x(\neg \phi).$$

D.h. falls Q_k von der Form $\forall \underline{x_k}$ ist, so werden zunächst die Existenzquantoren der Formel $\exists \underline{x_k}(\neg \psi)$ eliminiert und dann dieses Eliminationsergebnis in Zeile 31/32 negiert. Also genügt es, nur die Elimination von Existenzquantorenblöcken zu betrachten, was in den Zeilen 20-30 geschieht und nun genauer untersucht wird.

In den Zeilen 20-22 finden erneut einfache Äquivalenzumformulierungen, wie sie bereits in den vorausgegangenen Bemerkungen und Lemma 3.3 erwähnt wurden, statt, so daß man nun die Existenzquantoren einer Formel $\exists \underline{x_k}(\gamma)$ zu eliminieren hat, wobei γ in disjunktiver Normalform ist.
Nach der Berechnung in Zeile 22 ist jedes Disjunktionsglied von γ nun in

Algorithmus RRCQE:

Beschreibung: Hermitesche Quantorenelimination(Real Root Counting Quantifier Elimination)

Eingabe: φ, eine Formel erster Stufe über der Sprache der angeordneten Ringe

Ausgabe: ψ, eine zu φ äquivalente quantorenfreie Formel erster Stufe über der Sprache der angeordneten Ringe

procedure RRCQE(φ)
begin
$\varphi' :=$ „Zu φ äquivalente Formel mit Auswertung von evtl. auftretenden variablenfreien Termen, *true* und *false*"
$\varphi' :=$ „Zu φ' äquivalente Formel mit Quantoren und ausschließlich den Operatoren $\wedge$, $\vee$ und $\neg$"
$\varphi' :=$ „Zu φ' äquivalente Formel, so daß in deren atomaren Formeln alle rechten Seiten der Relationen gleich 0 sind."
$\varphi' :=$ „Pränexe Normalform von φ' mit b Quantorenblöcken"
for $k := 1$ **to** b **do**
  $Q_k :=$ „Der k-te, von innen gezählte Quantorenblock von φ'"
end
$\psi :=$ „Matrix von φ'"
if $\psi = \mathit{true}$ **or** $\psi = \mathit{false}$ **then**
  return ψ
end
for $k := 1$ **to** b **do**
  $\underline{x_k} :=$ „Die von Q_k gebundenen Variablen"
  $\underline{u} :=$ „Menge der in ψ auftretenden Variablen ohne die Variablen $\underline{x_k}$"
  if „Q_k ist Allquantorenblock" **then**
    $\psi := \neg\psi$
  end
  $\psi :=$ „Zu ψ äquivalente Formel in positiver Normalform"
  $\psi :=$ „Normalisierung der Relationen der atomaren Formeln von ψ bzgl. $\underline{x_k}$"
  $\gamma :=$ „Disjunktive Normalform von ψ ohne Veränderung der Relationen"
  $\psi := \mathit{false}$
  for each „Disjunktionsglied γ' von γ" **do**
    $\beta :=$ „Konjunktion der atomaren Formeln von γ', die nur von $\underline{u}$ abhängig sind."
    $\alpha :=$ „Konjunktion der von $\underline{x_k}$ abhängigen atomaren Formeln von γ'"
    $\alpha := \exists \underline{x_k}(\alpha)$
    $\gamma'' := \beta \wedge \mathtt{QENF}(\beta, \alpha)$
    $\psi := \psi \vee \gamma''$
  end
  if „Q_k ist Allquantorenblock" **then**
    $\psi := \neg\psi$
  end
  $\psi :=$ „Zu φ äquivalente Formel mit Auswertung von evtl. auftretenden Wahrheitswerten *true* und *false*"
  if $\psi = \mathit{true}$ **or** $\psi = \mathit{false}$ **then**
    return ψ
  end
end
return ψ
end RRCQE

Abbildung 3.3: Algorithmus zur Quantorenelimination einer beliebigen Formel erster Stufe durch Reduktion auf Elimination von Formeln in Eliminationsnormalform positiver Normalform und hat normierte Relationen bzgl. $\underline{x_k}$. In den Zeilen 24-30 wird nun der Quantorenblock $\exists \underline{x_k}$ für jedes einzelne Disjunktionsglied γ' von γ separat eliminiert. Da γ eine disjunktive Normalform ist, ist jedes γ' eine Konjunktion atomarer Formeln. Da zuvor alle Relationen bzgl. $\underline{x_k}$ normalisiert wurden, ist somit in Zeile 28 α in Eliminationsnormalform. Ferner ist β eine Konjunktion atomarer Formeln, welche keine Variablen aus $\underline{x_k}$ enthalten; somit ist `QENF`(β, α) wohldefiniert und liefert nach Voraussetzung eine quantorenfreie Formel α', so daß

$$\beta \wedge \alpha \quad \Leftrightarrow \quad \beta \wedge \alpha'.$$

In Zeile 29 werden die einzelnen Eliminationsergebnisse aller Disjunktionsglieder von γ somit korrekt zum Gesamteliminationsergebnis ψ von $\exists \underline{x_k}(\gamma)$ disjunktiv verbunden. Somit wurde nach dem k-ten Schleifendurchlauf der Block Q_k eliminiert, so daß

$$\varphi \Leftrightarrow Q_b \dots Q_{k+1}(\psi).$$

Man beachte, daß der $k+1$-te Schleifendurchlauf nicht mehr stattfindet, falls ψ gleich *true* oder *false* ist, da ψ dann bereits das Eliminationsergebnis von φ ist. Also gilt die behauptete Schleifeninvariante.

Es gilt damit insbesondere $\varphi \Leftrightarrow Q_b(\psi)$ vor dem letzten Schleifendurchlauf, in welchem dann der letzte Quantorenblock Q_b analog wie in den Schleifendurchläufen davor korrekt eliminiert wird. Somit gilt nach dem letzten Schleifendurchlauf:

$$\varphi \Leftrightarrow \psi,$$

wobei nun ψ quantorenfrei ist, also das gewünschte Ergebnis ist. □

Somit genügt es nun, Quantorenelimination für Formeln erster Stufe in Eliminationsnormalform zu behandeln.

3.2 Zählen reeller Nullstellen von Idealen

Als Vorbereitung zur Quantorenelimination von Formeln in Eliminationsnormalform wird nun gezeigt, wie man charakterisieren kann, wann echte nulldimensionale Ideale in $\mathbb{Q}[X_1, \dots, X_n]$ reelle Nullstellen besitzen, die gewissen endlich vielen Nebenbedingungen genügen. Das Ergebnis wird dann erweitert auf parametrische Ideale. Um diese Aufgabenstellung behandeln zu können, müssen nun erst einige Begriffe und zugehörige Eigenschaften eingeführt werden:

Definition 3.8 (Typ eines Polynoms) *Sei $f \in \mathbb{R}[X]$ mit ausschließlich reellen Nullstellen. Dann ist der* Typ *$\tau(f)$ des Polynoms f definiert als die Differenz der Anzahlen der positiven und negativen Nullstellen von f. Die Nullstellen werden dabei mit ihren Vielfachheiten gezählt.*

Den Typ eines Polynomprodukts kann man als Summe der Typen der einzelnen Faktoren des Produkts berechnen:

Lemma 3.9 *Sei $2 \leq k \in \mathbb{N}$. Seien $f_1, \ldots, f_k \in \mathbb{R}[X]$ mit ausschließlich reellen Nullstellen. Dann ist*

$$\tau\Big(\prod_{i=1}^{k} f_i\Big) = \sum_{i=1}^{k} \tau(f_i).$$

Beweis: durch vollständige Induktion nach k. Sei $k = 2$. Haben f_1 bzw. f_2 p_1 bzw. p_2 positive Nullstellen und n_1 bzw. n_2 negative Nullstellen, jeweils mit Vielfachheiten gezählt. Dann ist

$$\tau(f_1 \cdot f_2) = (p_1 + p_2) - (n_1 + n_2) = (p_1 - n_1) + (p_2 - n_2) = \tau(f_1) + \tau(f_2),$$

was sofort aus der Tatsache folgt, daß eine Nullstelle von f_i, $i \in \{1,2\}$, auch Nullstelle des Produkts $f_1 \cdot f_2$ ist und umgekehrt jede Nullstelle dieses Produkts eine Nullstelle eines $f_i, i \in \{1,2\}$, ist, da der Körper der reellen Zahlen keine Nullteiler besitzt.

Induktionsschritt von k nach $k+1$: Gelte die Behauptung für ein $k \geq 2$. Seien $f_1, \ldots, f_{k+1} \in \mathbb{R}[X]$ mit ausschließlich reellen Nullstellen. Dann ist

$$\tau\Big(\prod_{i=1}^{k+1} f_i\Big) = \tau\Big(\Big(\prod_{i=1}^{k} f_i\Big) \cdot f_{k+1}\Big) \overset{Fall\ k=2}{=} \tau\Big(\prod_{i=1}^{k} f_i\Big) + \tau(f_{k+1}) \overset{I.-A.}{=} \sum_{i=1}^{k+1} \tau(f_i).$$

□

Nun will man mit Hilfe von quantorenfreien Formeln charakterisieren, wann ein Polynom vom Typ 0 ist.

Definition 3.10 (Typformel) *Sei $d \in \mathbb{N}$ mit $d > 0$. Sei*

$$\chi = X^d + \sum_{i=0}^{d-1} C_i X^i \in \mathbb{Q}[C_0, \ldots, C_{d-1}][X]$$

ein parametrisiertes, normiertes, univariates Polynom. Ferner sei $A \subseteq \mathbb{R}^d$ die Menge aller Spezialisierungen von $C_0, \ldots, C_{d-1}$, so daß für jedes $\underline{a} \in A$ das univariate Polynom $\chi(\underline{a}, X)$ nur reelle Nullstellen besitzt. Dann heißt eine erweiterte quantorenfreie Formel $T_d^{(0)}(C_0, \ldots, C_{d-1})$ eine Typformel (für Polynome) vom Grad d (bzgl. des Typs 0), falls für jede Spezialisierung $\underline{a} := (a_0, \ldots, a_{d-1}) \in A$ gilt:

$$T_d^{(0)}(a_0, \ldots, a_{d-1}) \Leftrightarrow \tau(\chi(\underline{a}, X)) = 0 \ .$$

$T_d^{(0)}(C_0, \ldots, C_{d-1})$ heißt strikte Typformel vom Grad d, falls sogar gilt:

$$T_d^{(0)}(a_0, \ldots, a_{d-1}) \Leftrightarrow \Big(\tau(\chi(\underline{a}, X)) = 0 \wedge a_0 \neq 0\Big) \ .$$

Bemerkung: Diese Definition einer Typformel kann man dazu verwenden, falls man ein parametrisiertes Polynom

$$f = X^d + \sum_{i=0}^{d-1} u_i X^i \in \mathbb{Q}[U_1, \ldots, U_m][X]$$

mit ausschließlich reellen Nullstellen für alle Spezialisierungen von $U_1, \ldots, U_m$ betrachtet und in Abhängigkeit der Parameter charakterisieren will, wann dieses Polynom vom Typ 0 ist. Sei $T_d(C_0, \ldots, C_{d-1})$ eine Typformel für Polynome vom Grad d. Dann gilt somit für eine Spezialisierung $\underline{u} \in \mathbb{R}^m$ von $U_1, \ldots, U_m$, daß

$$\tau(\chi(\underline{u}, X)) = 0 \quad \Leftrightarrow \quad T_d(u_0(\underline{u}), \ldots, u_{d-1}(\underline{u})).$$

Analoges gilt, wenn Polynome $p = X^d + \sum_{i=0}^{d-1} \frac{q_{i,1}}{q_{i,2}} X^i \in \mathbb{Q}(U_1, \ldots, U_m)[X]$ mit $q_{i,1}, q_{i,2} \in \mathbb{Z}[U_1, \ldots, U_m]$ betrachtet werden, wobei man eine Anwendung einer Spezialisierung $\underline{u} \in \mathbb{R}^m$ auf p hier definiert als $X^d + \sum_{i=0}^{d-1} \frac{q_{i,1}(\underline{u})}{q_{i,2}(\underline{u})} X^i$. Man beachte, daß man dabei nur Spezialisierungen $\underline{u}$ betrachten darf, so daß kein Polynom im Nenner eines Koeffizienten unter diesen Spezialisierungen verschwindet, d.h. daß $q_{i,2}(\underline{u}) \neq 0$ für alle $i \in \{0, \ldots, d-1\}$.

Satz 3.11 *Sei $0 < d \in \mathbb{N}$. Dann existiert eine Typformel $T_d^{(0)}(C_0, \ldots, C_{d-1})$ für Polynome vom Grad d. Diese Formel läßt sich algorithmisch konstruieren: Für $d \in \mathbb{N}, d \geq 2$, gilt:*

$$\begin{aligned}
T_1^{(0)}(C_0) &:= C_0 = 0 \ , \\
T_2^{(0)}(C_0, C_1) &:= C_1 = 0 \vee C_0 < 0 \ , \\
T_{2d-1}^{(0)}(C_0, \ldots, C_{2d-2}) &:= C_0 = 0 \wedge T_{2d-2}^{(0)}(C_1, \ldots, C_{2d-2}) \ , \\
T_{2d}^{(0)}(C_0, \ldots, C_{2d-1}) &:= \Big(C_0 = 0 \wedge C_1 = 0 \wedge T_{2d-2}^{(0)}(C_2, \ldots, C_{2d-1})\Big) \vee \\
&\quad\ \Big(C_0 \neq 0 \wedge T_{2d}^{(0)'}(C_0, \ldots, C_{2d-1})\Big),
\end{aligned}$$

wobei $T_{2d}^{(0)'}(C_0, \ldots, C_{2d-1})$ eine strikte Typformel für Polynome vom Grad 2d mit Typ 0 ist, welche sich ebenfalls algorithmisch konstruieren läßt.

Beweis: Siehe [4], Proposition 3.18 und 3.19. und 3.22. □

Der folgende Satz bildet nun die Grundlage dafür, bestimmen zu können, ob ein echtes nulldimensionales Ideal in $\mathbb{Q}[X_1, \ldots, X_n]$ mindestens eine reelle Nullstelle besitzt, welche vorerst eine einzige Nebenbedingung der Form $g > 0$ oder $h \neq 0$, $g, h \in \mathbb{Q}[X_1, \ldots, X_n]$, erfüllt. Insbesondere ist damit auch der Fall abgedeckt, falls keine Nebenbedingung vorliegt, denn man kann die künstliche triviale Nebenbedingung $1 \neq 0$ wählen.

Satz 3.12 (Pedersen, Roy, Szpirglas) *Sei $\mathbb{K}$ ein angeordneter Körper der Charakteristik 0, $\mathbb{K}'$ reell abgeschlossener Erweiterungskörper von $\mathbb{K}$ und $\overline{\mathbb{K}'}$ der algebraische Abschluß von $\mathbb{K}'$. Sei $I \subseteq \mathbb{K}[X_1, \ldots, X_n] = \mathbb{K}[\underline{X}]$ ein echtes nulldimensionales Ideal und $B = \{v_1, \ldots, v_{d_0}\}$ eine Basis des $\mathbb{K}$-Vektorraums $\mathbb{K}[\underline{X}]/I$. Bezeichne $V_{\mathbb{K}'}(I)$ bzw. $V_{\overline{\mathbb{K}'}}(I)$ die Menge der Nullstellen von I in $\mathbb{K}'$ bzw. in $\overline{\mathbb{K}'}$. Sei $h \in \mathbb{K}[\underline{X}]$. Für $v_i, v_j \in B$ seien folgende lineare Abbildungen definiert:*

$$f_{h,i,j} : \mathbb{K}[\underline{X}]/I \to \mathbb{K}[\underline{X}]/I : g + I \mapsto (g \cdot h \cdot v_i \cdot v_j) + I.$$

Ferner sei

$$Q_h := (Spur(f_{h,i,j}))_{i,j \in \{1,\ldots,d_0\}}.$$

Dann gilt:

1. *Q_h ist eine reelle, symmetrische, quadratische Matrix, besitzt also nur reelle Eigenwerte. Insbesondere besitzt das charakteristische Polynom von Q_h nur reelle Nullstellen.*
2. *$rang(Q_h) = |\{\alpha \in V_{\overline{\mathbb{K}'}}(I) \mid h(\alpha) \neq 0\}|$*
3. *$sig(Q_h) = |\{\alpha \in V_{\mathbb{K}'}(I) \mid h(\alpha) > 0\}| - |\{\alpha \in V_{\mathbb{K}'}(I) \mid h(\alpha) < 0\}|$*

Beweis: Siehe [10]. □

Mit Hilfe der Signaturen könnte man nun die Anzahl der reellen Nullstellen eines nulldimensionalen Ideals in $\mathbb{Q}[X_1, \ldots, X_n]$ unter einer Nebenbedingung der Form $h > 0$ oder $h \neq 0$ durch Aufstellen und Lösen eines linearen Gleichungssystems feststellen. Diese Bestimmung der Anzahl der Nullstellen geht auf Hermite zurück. Das Ergebnis dieses Satzes wird aber jetzt gleich dazu verwendet, um die Aufgabenstellung auf endlich viele zu erfüllende Nebenbedingungen zu erweitern, so daß man überprüfen kann, ob ein gegebenes, echtes, nulldimensionales Ideal $I \subseteq \mathbb{Q}[X_1, \ldots, X_n]$ mindestens eine reelle Nullstelle $\alpha \in \mathbb{R}^n$ besitzt, so daß

$$(*) \quad \bigwedge_{j=1}^{s} g_j(\alpha) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\alpha) \neq 0 \quad \text{mit } g_j, h_k \in \mathbb{Q}[X_1, \ldots, X_n]$$

erfüllt ist. Ferner sei für $s, t \in \mathbb{N}$

$$E_{s,t} := \underbrace{\{1,2\} \times \cdots \times \{1,2\}}_{s-mal} \times \underbrace{\{2\} \times \cdots \times \{2\}}_{t-mal}.$$

Für $e := (e_1, \ldots, e_{s+t}) \in E_{s,t}$ sei

$$h^e := \prod_{j=1}^{s} g_j^{e_j} \cdot \prod_{k=1}^{t} h_k^{e_{s+k}}.$$

Mit diesen Notationen kann man nun über den Typ eines Polynoms charakterisieren, ob eine solche reelle Nullstelle α von I existiert.

Satz 3.13 *Für $e \in E_{s,t}$ seien $\chi_e \in \mathbb{Q}[Y_{char}]$ die charakteristischen Polynome von Q_{h^e} aus Satz 3.12. Dann gilt:*
Es existiert eine reelle Nullstelle von I, welche die Nebenbedingung in $(*)$ erfüllt, genau dann, wenn der Typ des Polynoms $\chi := \prod_{e \in E_{s,t}} \chi_e \in \mathbb{Q}[Y_{char}]$ ungleich 0 ist.

Beweis: Siehe [4], Korollar 2.18. □

Bemerkung: Das Polynom χ in Satz 3.13 besitzt ausschließlich reelle Nullstellen, da es ein Produkt von Polynomen mit ausschließlich reellen Nullstellen ist.

Somit kann man für ein echtes nulldimensionales Ideal $I \subseteq \mathbb{Q}[X_1, \ldots, X_n]$ folgendermaßen bestimmen, ob eine reelle Nullstelle α existiert, welche $(*)$ erfüllt: Man berechne das Produkt der charakteristischen Polynome

$$\chi = \prod_{e \in E_{s,t}} \chi_e = Y_{char}^d + \sum_{i=0}^{d-1} a_i Y_{char}^i \in \mathbb{Q}[Y_{char}]$$

aus Satz 3.13. Sei $T_d^{(0)}(C_0, \ldots, C_{d-1})$ eine Typformel vom Grad d. Dann existiert eine solche Nullstelle α genau dann, wenn $\neg T_d^{(0)}(a_0, \ldots, a_{d-1})$ gilt.

Anwendung auf parametrischen Fall:

Satz 3.13 kann man verallgemeinern, um ihn auf ein parametrisches Ideal $I \subseteq \mathbb{Q}[\underline{U}][\underline{X}]$ anzuwenden. Dazu muß ein Gröbnersystem von I einen nulldimensionalen Ast besitzen und man muß zur Anwendung des Satzes im Polynomring $\mathbb{Q}(\underline{U})[\underline{X}]$ rechnen. Außerdem muß man dafür vorübergehend gebrochen-rationale Ausdrücke in den Variablen $\underline{U}$ als Terme in den atomaren Formeln zulassen. Mit anderen Worten, man läßt hier vorübergehend einen Divisionsoperator $\div$ als zusätzliche Operation in den atomaren Formeln zu mit folgender semantischer Definition:

$$a \div b := \begin{cases} \frac{a}{b} & \text{falls } b \neq 0 \\ 0 & \text{falls } b = 0 \end{cases} \qquad \text{für } a, b \in \mathbb{R}.$$

Jedes Element in $\mathbb{Q}(\underline{U})$ wird nun in der Form $\frac{p}{q} \in \mathbb{Q}(\underline{U})$ mit $p, q \in \mathbb{Z}[\underline{U}]$ dargestellt und kann damit als Term $p \div q$ interpretiert werden, und umgekehrt.

Sei jetzt ein nulldimensionaler Ast eines Gröbnersystems von I mit erfüllbarer Bedingung ζ und Gröbnerbasis G gegeben. Es werden nun ausschließlich Spezialisierungen $\underline{u} \in \mathbb{R}^m$ der Parameter $\underline{U}$ betrachtet, welche die Bedingung ζ des nulldimensionalen Astes erfüllen. In allen auftretenden Termen der Form $a \div b$ wird sichergestellt sein, daß $b \in \mathbb{Z}[\underline{U}]$ und unter all diesen betrachteten Spezialisierungen $\underline{u}$ die Ungleichung $b(\underline{u}) \neq 0$ gilt. Da alle übrigen Spezialisierungen, welche ζ nicht erfüllen, nicht betrachtet werden, ist damit die Operation $\div$ im Wesentlichen die übliche Division auf den reellen Zahlen. Letztlich werden aber die hier auftretenden atomaren Formeln äquivalent so umgeformt, daß wieder gültige atomare Formeln über der zu Beginn dieses Kapitels eingeführten Sprache ohne den Operator $\div$ entstehen.

Ziel ist es nun, im gegebenen nulldimensionalen Ast zu untersuchen, für welche ζ-erfüllenden Spezialisierungen $\underline{u} \in \mathbb{R}^m$ der Parameter $\underline{U}$ eine reelle Nullstelle $\alpha \in \mathbb{R}^n$ von $\sigma_{\underline{u}}(I)$ existiert, die die nun parametrisierte Nebenbedingung erfüllt:

$$\xi(\underline{u}) := \bigwedge_{j=1}^{s} g_j(\underline{u}, \alpha) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{u}, \alpha) \neq 0 \quad \text{mit } g_j, h_k \in \mathbb{Q}[\underline{U}][\underline{X}].$$

Sei $E_{s,t}$ wie bisher definiert. Für $e := (e_1, \ldots, e_{s+t}) \in E_{s,t}$ und eine Spezialisierung $\underline{u}$ sei wiederum

$$h^e := \prod_{j=1}^{s} g_j^{e_j} \cdot \prod_{k=1}^{t} h_k^{e_{s+k}} \quad \text{und} \quad h^e(\underline{u}, \underline{X}) := \prod_{j=1}^{s} g_j^{e_j}(\underline{u}, \underline{X}) \cdot \prod_{k=1}^{t} h_k^{e_{s+k}}(\underline{u}, \underline{X}).$$

Für $e \in E_{s,t}$ seien $\chi_e \in \mathbb{Q}(\underline{U})[Y_{char}]$ die charakteristischen Polynome von Q_{h^e} aus Satz 3.12, wobei man zu deren Berechnung die Arithmetik von $\mathbb{Q}(\underline{U})[\underline{X}]$ verwendet. Daraus kann man nun das Produkt

$$\chi := \prod_{e \in E_{s,t}} \chi_e = Y_{char}^d + \sum_{i=0}^{d-1} a_i Y_{char}^i \in \mathbb{Q}(\underline{U})[Y_{char}]$$

berechnen. Sei $T_d^{(0)}(C_0, \ldots, C_{d-1})$ eine Typformel vom Grad d und $\underline{u}$ eine Spezialisierung, die ζ erfüllt. Dann existiert genau dann eine Nullstelle des Ideals $\sigma_{\underline{u}}(I)$, welche $\xi(\underline{u})$ erfüllt, wenn

$$\neg T_d^{(0)}(a_0(\underline{u}), \ldots, a_{d-1}(\underline{u})) \ ,$$

denn: Seien zu einer Spezialisierung $\underline{u}$, die ζ erfüllt, $\chi_{e,\underline{u}} \in \mathbb{Q}[Y_{char}]$ die charakteristischen Polynome von $Q_{h^e(\underline{u},\underline{X})}$ aus Satz 3.12 und sei das zugehörige Produkt dieser charakteristischen Polynome aus Satz 3.13

$$\chi_{\underline{u}} := \prod_{e \in E_{s,t}} \chi_{e,\underline{u}} = Y_{char}^d + \sum_{i=0}^{d-1} a_{i,\underline{u}} Y_{char}^i \in \mathbb{Q}[Y_{char}].$$

Offensichtlich ist χ die allgemeine Darstellung des Polynoms $\chi_{\underline{u}}$ in den Parametern $\underline{U}$, d.h. $\chi(\underline{u}, Y_{char}) = \chi_{\underline{u}}(Y_{char})$. Dies folgt aus der Tatsache, daß jedes charakteristische Polynom $\chi_{e,\underline{u}}$ mit derselben Folge von Berechnungsschritten bestimmt werden kann wie die in den Parametern allgemeine Darstellung χ_e. Dies ist möglich, da durch die Gültigkeit von ζ in diesem Ast nun gewährleistet ist, daß die höchsten Terme von G eindeutig bestimmt sind und somit bei der Berechnung der charakteristischen Polynome χ_e nur durch höchste Koeffizienten $p \in \mathbb{Q}[\underline{U}]$ von G geteilt wird, wobei für alle erlaubten Spezialisierungen $\underline{u}$ aufgrund von ζ in diesem Ast die Ungleichung $p(\underline{u}) \neq 0$ erfüllt ist.

Somit ist also $a_i(\underline{u}) = a_{i,\underline{u}}$ für alle $i \in \{0, \ldots, d-1\}$. Daraus folgt mit Satz 3.13, daß genau dann eine reelle Nullstelle des Ideals $\sigma_{\underline{u}}(I)$ existiert, welche $\xi(\underline{u})$ erfüllt, wenn die Formel $\neg T_d^{(0)}(a_{0,\underline{u}}, \ldots, a_{d-1,\underline{u}})$ gilt, oder dazu äquivalent, wenn $\neg T_d^{(0)}(a_i(\underline{u}), \ldots, a_{d-1}(\underline{u}))$ erfüllt ist.

Die Terme der Formel $\neg T_d^{(0)}(a_0, \ldots, a_{d-1})$ sind nun Elemente von $\mathbb{Q}(\underline{U})$. Um wieder gültige Formeln über der zu Beginn dieses Kapitels eingeführten Sprache zu erhalten, muß man ggf. zunächst durch Hauptnennerbildung jeden Term in die Form $t_1 \div t_2$ mit $t_1, t_2 \in \mathbb{Z}[\underline{U}]$ äquivalent umwandeln. Danach muß jede auftretende atomare Formel der Form „$t_1 \div t_2 \ \varrho\ 0$", mit $t_1, t_2 \in \mathbb{Z}[\underline{U}]$, durch Multiplikation mit t_2 bzw. t_2^2 äquivalent umgeformt werden, so daß alle Terme Elemente von $\mathbb{Z}[\underline{U}]$ sind:

$$\begin{array}{llll} \text{Aus } t_1 \div t_2 \ \varrho\ 0 & \text{wird} & t_1 \ \varrho\ 0\ , & \text{falls } \varrho \in \{=, \neq\}. \\ \text{Aus } t_1 \div t_2 \ \varrho\ 0 & \text{wird} & t_1 \cdot t_2 \ \varrho\ 0\ , & \text{falls } \varrho \in \{<, \leq, \geq, >\}. \end{array}$$

Somit wurde gezeigt, daß man Satz 3.13 auch im parametrischen Falle anwenden kann.

3.3 Auffinden nicht-trivialer Gleichungen

Der Algorithmus zur Quantorenelimination von Formeln in Eliminationsnormalform setzt eigentlich voraus, daß in einer Eingabeformel in Eliminationsnormalform mindestens eine nicht-triviale Gleichung enthalten ist. Ist dies nicht der Fall, so müssen nicht-triviale Gleichungen gesucht werden und die Eingabeformel entsprechend äquivalent umgeformt werden. Dieser Abschnitt beschäftigt sich mit den theoretischen Grundlagen zum Auffinden dieser nicht-trivialen Gleichungen. Ziel ist es, eine Eingabeformel der Form

$$\exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U}, x) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, x) \neq 0 \Big) \quad \text{mit } g_j, h_k \in \mathbb{Z}[\underline{U}][X]$$

derart äquivalent umzuformen, daß man zur Elimination der Variablen x nur Formeln in Eliminationsnormalform mit nicht-trivialen Gleichungen eliminieren muß. Zunächst werden einige dafür notwendige Definitionen und abkürzende Schreibweisen eingeführt:

Definition 3.14 (Formale Ableitung) *Sei $f = \sum_{i=0}^{k} a_i X^i \in \mathbb{R}[\underline{U}][X]$. Dann ist die formale Ableitung von f nach der Variablen X definiert als*

$$\frac{\partial f}{\partial X} := \sum_{i=1}^{k} i a_i X^{i-1}.$$

Definition 3.15 (Partieller Grad bzgl. einer Variablen) *Sei $f \in \mathbb{R}[\underline{U}][X_1, \ldots, X_n]$ und sei $i \in \{1, \ldots, n\}$. Man betrachte f als Element von $\mathbb{R}[\underline{U}, X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n][X_i]$, etwa $f = \sum_{j=0}^{k} a_j X_i^j$ mit $a_k \neq 0$. Dann ist der partielle Grad $Grad_{X_i}(f)$ von f bzgl. X_i gleich k.*

Als abkürzende Schreibweisen werden folgende Notationen eingeführt:

Definition 3.16 (Notationen) *Sei $f = \sum_{i=0}^{k} a_i X^i \in \mathbb{Z}[\underline{U}][X]$ mit $a_k \neq 0$ und $\ell \in \{1, \ldots, k\}$. Sei $T(f)$ die Menge aller Terme in f, insbesondere aller Terme mit parametrischen Koeffizienten. Dann definiert man*

$$\begin{aligned}
f(\underline{U}, \infty) > 0 \quad &:= \quad \bigvee_{i=0}^{k} \Big(a_i > 0 \wedge \bigwedge_{j=i+1}^{k} a_j = 0 \Big) \ , \\
f(\underline{U}, -\infty) > 0 \quad &:= \quad \bigvee_{i=0}^{k} \Big((-1)^i a_i > 0 \wedge \bigwedge_{j=i+1}^{k} a_j = 0 \Big), \\
\omega(f, \ell) \quad &:= \quad a_\ell \neq 0 \wedge \bigwedge_{q=\ell+1}^{k} a_q = 0 \\
f \neq_0 0 \quad &:= \quad \bigvee_{X^i \in T(f)} \omega(f, i) \ , \\
f \neq_1 0 \quad &:= \quad \bigvee_{X^i \in T(f), i \neq 0} \omega(f, i) \ , \\
f \neq_{koeff} 0 \quad &:= \quad \bigvee_{i=0}^{k} a_i \neq 0 \ .
\end{aligned}$$

Bemerkungen:

1. $f(\underline{U}, \infty) > 0$ bzw. $f(\underline{U}, -\infty) > 0$ stellen quantorenfreie Formeln in den Parametern $\underline{U}$ dar, so daß für jede Spezialisierung $\underline{u} \in \mathbb{R}^m$ diese Formeln genau dann erfüllt sind, wenn

   $$\lim_{x \to \infty} f(\underline{u}, x) > 0 \quad \text{bzw.} \quad \lim_{x \to -\infty} f(\underline{u}, x) > 0.$$

   Dies folgt unmittelbar aufgrund der Eigenschaften eines univariaten Polynoms $p \in \mathbb{R}[X]$, daß es gegen $+\infty$ konvergiert für $x \to \infty$ genau dann, wenn $HK(p) > 0$, bzw. daß das Polynom p gegen $+\infty$ konvergiert für $x \to -\infty$ genau dann, wenn $(-1)^{Grad_X(p)} \cdot HK(p) > 0$.

2. $\omega(f,\ell)$ ist eine quantorenfreie Formel in den Parametern $\underline{U}$, so daß eine beliebige Spezialisierung $\underline{u} \in \mathbb{R}^m$ diese Formel genau dann erfüllt, wenn $Grad_X(f(\underline{u},X)) = \ell$. Man beachte, daß $X^\ell \notin T(f)$ impliziert, daß $\omega(f,\ell) \equiv \mathit{false}$, und daß die Erfüllungsmengen von $\omega(f,\ell_1)$ und $\omega(f,\ell_2)$, $\ell_1 \neq \ell_2$, disjunkt sind.

3. $f \neq_0 0$ und $f \neq_{koeff} 0$ sind quantorenfreie Formeln in den Parametern $\underline{U}$, so daß eine Spezialisierung $\underline{u} \in \mathbb{R}^m$ diese beiden Formeln genau dann erfüllt, wenn $f(\underline{u},X)$ nicht das Nullpolynom ist. Dies folgt aus der Tatsache, daß ein univariates Polynom genau dann nicht das Nullpolynom ist, wenn mindestens ein Koeffizient nicht verschwindet. $f \neq_{koeff} 0$ ist die kürzeste Charakterisierung dieser Eigenschaft. Die einzelnen Disjunktionsglieder $\omega(f,i)$ in $f \neq_0 0$ stellen eine disjunkte Fallunterscheidung nach dem Grad von f unter einer Spezialisierung $\underline{u}$ dar.

4. $f \neq_1 0$ ist eine quantorenfreie Formel in den Parametern $\underline{U}$, so daß eine Spezialisierung $\underline{u} \in \mathbb{R}^m$ diese Formel genau dann erfüllt, wenn $f(\underline{u},X)$ nicht konstant bzgl. X ist. Wiederum stellen die einzelnen Disjunktionsglieder $\omega(f,i)$ eine disjunkte Fallunterscheidung nach dem Grad von f unter einer Spezialisierung $\underline{u}$ dar.

Mit folgendem Satz kann man nun aus den Ungleichungen und $>$-Relationen einer Eingabeformel nicht-triviale Gleichungen erzeugen, wobei die in 3.16 eingeführten Notationen als abkürzende Schreibweisen verwendet werden.

Satz 3.17 *Seien $g_1,\dots,g_s,h_1,\dots,h_t \in \mathbb{Z}[\underline{U}][X]$ parametrisierte Polynome. Ferner seien die Formeln erster Stufe $\varphi_1,\varphi_2,\varphi_3,\varphi_3',\varphi_4,\varphi_4'$ wie folgt definiert:*

$$\begin{aligned}
\varphi_1 &:= \bigwedge_{j=1}^{s} g_j(\underline{U},\infty) > 0 \ , \\
\varphi_2 &:= \bigwedge_{j=1}^{s} g_j(\underline{U},-\infty) > 0 \ , \\
\varphi_3 &:= \bigvee_{\ell=1}^{s} \exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \frac{\partial g_\ell}{\partial X} \neq_1 0 \wedge \frac{\partial g_\ell}{\partial X}(\underline{U},x) = 0 \Big) \ , \\
\varphi_3' &:= \bigvee_{\ell=1}^{s} \exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \frac{\partial g_\ell}{\partial X} \neq_0 0 \wedge \frac{\partial g_\ell}{\partial X}(\underline{U},x) = 0 \Big) \ , \\
\varphi_4 &:= \bigvee_{1 \le k < \ell \le s} \exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge (g_k - g_\ell) \neq_1 0 \wedge (g_k - g_\ell)(\underline{U},x) = 0 \Big) \ , \\
\varphi_4' &:= \bigvee_{1 \le k < \ell \le s} \exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge (g_k - g_\ell) \neq_0 0 \wedge (g_k - g_\ell)(\underline{U},x) = 0 \Big) \ .
\end{aligned}$$

Dann gelten folgende Äquivalenzen:

$$\begin{aligned}
&\exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U},x) \neq 0\Big)\\
\overset{(1)}{\Longleftrightarrow}\;& \Big(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0\Big) \wedge \big(\varphi_1 \vee \varphi_2 \vee \varphi_3' \vee \varphi_4'\big)\\
\overset{(2)}{\Longleftrightarrow}\;& \Big(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0\Big) \wedge \big(\varphi_1 \vee \varphi_2 \vee \varphi_3 \vee \varphi_4\big).
\end{aligned}$$

Beweis: Sei $\underline{u}$ eine Spezialisierung. Die Äquivalenz

$$\begin{aligned}
&\exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{u},x) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{u},x) \neq 0\Big)\\
\Longleftrightarrow\;& \Big(\bigwedge_{k=1}^{t} (h_k \neq_{koeff} 0)(\underline{u})\Big) \wedge \big(\varphi_1(\underline{u}) \vee \varphi_2(\underline{u}) \vee \varphi_3'(\underline{u}) \vee \varphi_4'(\underline{u})\big)
\end{aligned}$$

wurde in [4], Satz 4.6, bewiesen. Daraus folgt somit direkt Äquivalenz (1).

Um Äquivalenz (2) zu beweisen, genügt der Beweis der Äquivalenz

$$\big(\varphi_3' \vee \varphi_4'\big) \Leftrightarrow (\varphi_3 \vee \varphi_4)\,.$$

Sei wieder $\underline{u}$ eine beliebige Spezialisierung, zu welcher auf der linken Seite ein Disjunktionsglied von φ_3' bzw. φ_4' gilt, etwa

$$\exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{u},x) > 0 \wedge (f \neq_0 0)(\underline{u}) \wedge f(\underline{u},x) = 0\Big),$$

wobei f eine Ableitung eines g_ℓ bzgl. X bzw. eine Differenz von g_k und g_ℓ ist. Die in diesem Glied auftretende Ableitung $f(\underline{u},X)$ bzw. Differenz $f(\underline{u},X)$ ist also nicht das Nullpolynom. Es genügt zu zeigen, daß diese Ableitung bzw. Differenz nicht konstant bzgl. X ist. Denn angenommen diese wäre konstant, so hätte sie keine Nullstelle, also würde dieses Disjunktionsglied nicht gelten. Also gilt mit den Notationen 3.16 sogar

$$\exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{u},x) > 0 \wedge (f \neq_1 0)(\underline{u}) \wedge f(\underline{u},x) = 0\Big).$$

Somit gilt die Implikation von Äquivalenz (2).

Die Umkehrung dieser eben gezeigten Implikation ist trivial, da auf der rechten Seite eine stärker formulierte Aussage als auf der linken Seite steht, da per Definition $f \neq_1 0$ die Aussage $f \neq_0 0$ impliziert. $\square$

Für ein parametrisiertes, univariates Polynom $f = \sum_{i=0}^{k} a_i X^i \in \mathbb{Z}[\underline{U}][X]$ mit $a_k \in \mathbb{Z}[\underline{U}] \setminus \{0\}$, d.h. $a_k \neq 0$, und $\ell \in \{1, \ldots, k\}$ definiert man

$$f^{(\ell)} := \sum_{i=0}^{\ell} a_i X^i \ .$$

Offensichtlich gilt dann folgende Äquivalenz für eine Konjunktion ϕ atomarer Formeln:

$$\omega(f,\ell) \wedge \exists x\big(\phi \wedge f(\underline{U},x) = 0\big) \quad \Longleftrightarrow \quad \omega(f,\ell) \wedge \exists x\big(\phi \wedge f^{(\ell)}(\underline{U},x) = 0\big) \ .$$

$T(f)$ bezeichnet wieder die Menge *aller* Terme in f, welche also insbesondere auch diejenigen Terme enthält, deren Koeffizienten unter gewissen, aber nicht allen Spezialisierungen verschwinden. Das nächste Lemma ist eine technische Umformulierung der Formel $\varphi_3 \vee \varphi_4$ aus Satz 3.17, so daß die darin enthaltenen existenzquantifizierten Teilformeln in Eliminationsnormalform sind:

Lemma 3.18 *Seien φ_3, φ_4 wie in Satz 3.17.*
Dann kann $\varphi_3 \vee \varphi_4$ äquivalent als Disjunktion γ' dargestellt werden, wobei alle darin enthaltenen existenzquantifizierten Teilformeln in Eliminationsnormalform sind:

$$\gamma' := \bigvee_{\ell=1}^{s} \bigvee_{\substack{X^i \in T(\frac{\partial g_\ell}{\partial X}),\\ i\neq 0}} \left(\omega\Big(\frac{\partial g_\ell}{\partial X}, i\Big) \wedge \exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \Big(\frac{\partial g_\ell}{\partial X}\Big)^{(i)}(\underline{U},x) = 0\Big)\right)$$

$$\vee \bigvee_{1\leq k<\ell\leq s} \bigvee_{\substack{X^i \in T(g_k - g_\ell),\\ i\neq 0}} \left(\omega(g_k - g_\ell, i) \wedge \exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge (g_k - g_\ell)^{(i)}(\underline{U},x) = 0\Big)\right).$$

Beweis: Diese Äquivalenz folgt unmittelbar aus der Tatsache, daß die Formeln $\frac{\partial g_\ell}{\partial X} \neq_1 0$ bzw. $(g_k - g_\ell) \neq_1 0$ unabhängig von der Variablen x sind. Es gilt für jedes Disjunktionsglied in φ_3:

$$\begin{aligned}
& \exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \frac{\partial g_\ell}{\partial X} \neq_1 0 \wedge \frac{\partial g_\ell}{\partial X}(\underline{U},x) = 0\Big) \\
\equiv \quad & \frac{\partial g_\ell}{\partial X} \neq_1 0 \wedge \exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \frac{\partial g_\ell}{\partial X}(\underline{U},x) = 0\Big) \\
\equiv \quad & \bigvee_{X^i \in T(\frac{\partial g_\ell}{\partial X}), i\neq 0} \left(\omega\Big(\frac{\partial g_\ell}{\partial X}, i\Big) \wedge \exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \frac{\partial g_\ell}{\partial X}(\underline{U},x) = 0\Big)\right) \\
\equiv \quad & \bigvee_{X^i \in T(\frac{\partial g_\ell}{\partial X}), i\neq 0} \left(\omega\Big(\frac{\partial g_\ell}{\partial X}, i\Big) \wedge \exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \Big(\frac{\partial g_\ell}{\partial X}\Big)^{(i)}(\underline{U},x) = 0\Big)\right)
\end{aligned}$$

Analoge Umformung ist für die Disjunktionsglieder von φ_4 möglich. Damit folgt die Behauptung des Lemmas. □

Das nächste Lemma zeigt, wie man γ' verkleinern kann, indem man zu den gegebenen Polynomen $g_1, \ldots, g_s$ Disjunktionsglieder in γ' entfernt, welche offensichtlich äquivalent zu *false* sind. Dazu betrachtet man folgende Menge von Formeln zu einem Polynom $f = \sum_{i=0}^{k} a_i X^i \in \mathbb{Z}[\underline{U}][X]$ mit $a_k \neq 0$:

$$\Omega_f := \big\{\omega(f,i) \,\big|\, i \in \{1, \ldots, k\} \text{ mit } a_i \neq 0 \text{ und für } i < j \leq k \text{ ist } a_j \notin \mathbb{Z}\backslash\{0\}\big\}.$$

Für $k = 0$ ist $\Omega_f = \emptyset$. Die Menge Ω_f sortiert also gewisse Formeln $\omega(f,i)$ aus, welche offensichtlich zu f nicht erfüllbar sind. Folgendes Beispiel verdeutlicht diesen einfachen Zusammenhang.

Beispiel 3.19 *Sei* $f = UX^3 + 2X^2 + VX + 1 \in \mathbb{Z}[U,V][X]$. *Dann sind*

$$\begin{aligned} \omega(f,3) &\equiv U \neq 0\ , \\ \omega(f,2) &\equiv U = 0 \wedge 2 \neq 0\ , \\ \omega(f,1) &\equiv U = 0 \wedge 2 = 0 \wedge V \neq 0. \end{aligned}$$

Offensichtlich ist $\omega(f,1)$ *äquivalent zu false. Die Menge* Ω_f *besteht nur aus den Elementen* $\omega(f,3)$ *und* $\omega(f,2)$. ◇

Also kann man diejenigen Disjunktionsglieder in γ' entfernen, deren Teilformeln $\omega(\frac{\partial g_\ell}{\partial X}, i)$ bzw. $\omega(g_k - g_\ell, i)$ nicht Element von $\Omega_{\frac{\partial g_\ell}{\partial X}}$ bzw. $\Omega_{g_k - g_\ell}$ sind.

Lemma 3.20 *Sei* γ' *wie in Lemma 3.18 und*

$$\gamma := \bigvee_{\ell=1}^{s} \bigvee_{\omega(\frac{\partial g_\ell}{\partial X},i)\in\Omega_{\frac{\partial g_\ell}{\partial X}}} \Big(\omega\Big(\frac{\partial g_\ell}{\partial X}, i\Big) \wedge \exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \Big(\frac{\partial g_\ell}{\partial X}\Big)^{(i)} (\underline{U},x) = 0\Big)\Big)$$

$$\vee \bigvee_{1\leq k<\ell\leq s} \bigvee_{\omega(g_k-g_\ell,i)\in\Omega_{g_k-g_\ell}} \Big(\omega(g_k - g_\ell, i) \wedge \exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge (g_k - g_\ell)^{(i)}(\underline{U},x) = 0\Big)\Big).$$

Dann ist γ *zu* γ' *äquivalent.*

Beweis: Die Behauptung folgt sofort aus der Tatsache, daß nur diejenigen Disjunktionsglieder in γ' weggelassen werden, die aufgrund der Teilformeln $\omega(f,i)$ äquivalent zu *false* sind, wobei f eine Ableitung eines g_ℓ bzw. Differenz von g_k und g_ℓ ist . □

Somit hat man nun folgende Äquivalenz gezeigt:

$$\underbrace{\exists x \Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U},x) \neq 0\Big)}_{\varphi_0:=} \Leftrightarrow \underbrace{\Big(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0\Big) \wedge \big(\varphi_1 \vee \varphi_2 \vee \gamma\big)}_{\hat{\varphi}_0:=}.$$

Um x in der Formel φ_0 auf der linken Seite dieser Äquivalenz zu eliminieren, genügt es also, x in der Formel $\hat{\varphi}_0$ zu eliminieren. Dazu muß x in jedem Disjunktionsglied von γ eliminiert werden. Dabei ist jedes dieser Disjunktionsglieder von γ in der Form $\omega(f,i) \wedge \exists x(\phi)$, wobei ϕ in Eliminationsnormalform mit genau einer nicht-trivialen Gleichung ist. Man beachte, daß die Teilformeln $(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0) \wedge (\varphi_1 \vee \varphi_2)$ sowie $\omega(f,i)$ quantorenfrei sind und die Variable x nicht enthalten. Somit genügt es zu jedem Disjunktionsglied von γ obiger Form die Formel $\exists x(\phi)$ zu eliminieren. Also wurde gezeigt, wie man eine Eingabeformel φ_0 in Eliminationsnormalform mit einer gebundenen Variablen und ohne nicht-triviale Gleichungen geeignet äquivalent umformen kann, so daß man zur Quantorenelimination dieser Eingabeformel ausschließlich Formeln in Eliminationsnormalform mit genau einer nicht-trivialen Gleichung eliminieren muß.

3.4 QE von Formeln in Eliminationsnormalform

In diesem Abschnitt wird erläutert, wie man die Quantoren einer Formel erster Stufe φ in Eliminationsnormalform eliminiert. Sei ab jetzt für den gesamten Abschnitt 3.4 die Formel φ von der Form

$$\exists x_1 \ldots \exists x_n \Big(\bigwedge_{i=1}^{r} f_i(\underline{U},\underline{x}) = 0 \wedge \bigwedge_{j=1}^{s} g_j(\underline{U},\underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U},\underline{x}) \neq 0 \Big)$$

mit $n,r,s,t \in \mathbb{N}, n > 0$ und $f_1,\ldots,f_r,g_1,\ldots,g_s,h_1,\ldots,h_t \in \mathbb{Z}[\underline{U}][\underline{X}] \backslash \{0\}$. Ferner wird eine Konjunktion ξ atomarer Formeln der Form

$$\xi := \bigwedge_{l=1}^{q} p_l \varrho_l 0 \quad \text{mit } p_l \in \mathbb{Z}[\underline{U}],\ \varrho_l \in \{<,\leq,=,\neq,\geq,>\},\ q \in \mathbb{N},$$

berücksichtigt, deren Gültigkeit bei der Elimination von φ angenommen werden darf. ξ wird auch als Prämisse bezeichnet. Ziel ist es nun, eine quantorenfreie Formel ψ zu berechnen, so daß

$$\xi \wedge \varphi \quad \Leftrightarrow \quad \xi \wedge \psi\ .$$

Man beachte, daß bei Inkonsistenz von ξ das Ergebnis ψ unbedeutend ist und beliebig gewählt werden kann.

Es wird nun der eigentliche Quantoreneliminationsalgorithmus aus Abbildung 3.4 für Formeln in Eliminationsnormalform ausführlich anhand der Eingabeformel φ und der Konjunktion ξ erläutert. Dazu werden alle auftretenden Terme jetzt als Polynome in $\mathbb{Z}[\underline{U}][\underline{X}]$ betrachtet, wobei die Variable x_i der Variablen X_i im Polynomring entspricht. Diese beiden Betrachtungsebenen der Variablen dienen dazu, um expliziter anzugeben, ob man jeweils aktuell von Polynomen oder Termen spricht.

Die Erläuterung dieses Algorithmus teilt sich auf folgende Unterkapitel auf. Zunächst wird in Unterkapitel 3.4.1 gezeigt, wie man den Fall $r = 0$ behandelt, d. h. wie eine Eingabeformel in Eliminationsnormalform ohne nicht-triviale Gleichungen mit Hilfe von Kapitel 3.3 durch Auffinden von nicht-trivialen Gleichungen eliminiert wird. Dadurch kann man ab dann o.B.d.A. vom Fall ausgehen, daß Gleichungen in φ vorhanden sind, d.h. $r > 0$ gilt. Im Unterkapitel 3.4.2 wird die Elimination einer Formel in Eliminationsnormalform mit mindestens einer enthaltenen nicht-trivialen Gleichung beschrieben: In 3.4.2.1 wird gezeigt, wie man aus den Gleichungen von φ ein Gröbnersystem bestimmen kann, dessen Äste dann – wie in 3.4.2.2 erläutert – separat eliminiert werden. In 3.4.2.3 wird beschrieben, wie aus den Eliminationsergebnissen der einzelnen Äste das Gesamteliminationsergebnis bestimmt wird. Abschließend folgt in 3.4.3 eine Zusammenfassung des Algorithmus mit Beweis der Termination und Korrektheit sowie einigen Anmerkungen.

3.4.1 Elimination im Fall $\mathbf{r = 0}$

In diesem Abschnitt wird gezeigt, wie man die Quantoren von φ eliminiert, falls keine nicht-trivialen Gleichungen in φ vorhanden sind, d. h. falls $r = 0$. Zur Elimination werden jedoch nicht-triviale Gleichungen benötigt. Falls in φ keine nicht-trivialen Gleichungen vorkommen, werden aus den Ungleichungen und $>$-Relationen von φ mit Hilfe von Kapitel 3.3 Eliminationsnormalformen mit nicht-trivialen Gleichungen konstruiert, so daß man den Fall $r = 0$ gewissermaßen auf den Fall $r > 0$ zurückführen kann. Man geht nun im Fall $r = 0$ folgendermaßen zur Elimination der Quantoren von φ vor:

Man vernachläßigt zunächst alle Quantoren von φ bis auf den innersten „$\exists x_n$". Unter Anwendung von Satz 3.17 mit zusätzlicher Anwendung von Lemma 3.20 wandelt man die Formel

$$\exists x_n \Big(\bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0 \Big)$$

äquivalent um in die Form

$$\Big(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0 \Big) \wedge \big(\varphi_1 \vee \varphi_2 \vee \gamma \big).$$

Die Polynome g_j und h_k müssen dabei als parametrisierte, univariate Polynome in X_n aufgefasst werden.

Jedes Disjunktionsglied von γ besitzt die Form

$$\omega(f, i) \wedge \exists x_n(\phi) \quad \text{ mit } \phi := \bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge f^{(i)}(\underline{U}, \underline{x}) = 0 \ ,$$

wobei $\exists x_n(\phi)$ in Eliminationsnormalform ist und f eine Ableitung eines g_ℓ bzgl. X_n ist bzw. eine Differenz $g_k - g_\ell$. Bezeichne $\tilde{\omega}(f,i)$ die einzige in $\omega(f,i)$ enthaltene Ungleichung. Die Variable x_n ist in den Formeln

$$\left(\bigwedge_{k=1}^{t} h \neq_{koeff} 0\right) \wedge (\varphi_1 \vee \varphi_2) \quad \text{und} \quad \omega(f,i)$$

nicht enthalten. Daher kann nun zu jedem Disjunktionsglied von γ separat die Variable x_n in der jeweiligen Teilformel $\exists x_n(\phi)$ unter Berücksichtung von $\tilde{\omega}(f,i)$ mittels rekursiven Aufrufen eliminiert werden, wobei jetzt die Formel $\exists x_n(\phi)$ in Eliminationsnormalform mit genau einer nicht-trivialen Gleichung ist. D. h. in diesen rekursiven Aufrufen tritt der Fall $r > 0$ ein. Sei das Eliminationsergebnis von $\exists x_n(\phi)$ die Formel ψ_ϕ. Dann werden die einzelnen Eliminationsergebnisse unter Berücksichtigung der jeweiligen Formel $\omega(f,i)$ disjunktiv verknüpft zu

$$\psi := \bigvee_{\substack{\text{„}\omega(f,i)\wedge\exists x_n(\phi)\text{"} \\ \text{ist Disjunktionsglied von } \gamma}} \big(\omega(f,i) \wedge \psi_\phi\big).$$

Somit wurde x_n eliminiert, und zur Elimination der restlichen Variablen wird `RRCQE` unter weiterer Berücksichtigung von ξ mit der Eingabeformel

$$\xi \wedge \exists x_1 \ldots \exists x_{n-1}\Big(\Big(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0\Big) \wedge (\varphi_1 \vee \varphi_2 \vee \psi)\Big)$$

aufgerufen. Damit wurde gezeigt, wie der Fall $r = 0$ auf den Fall $r > 0$ zurückgeführt werden kann.

3.4.2 Elimination im Fall r > 0

Es wird ab jetzt der Fall betrachtet, daß $r > 0$ gilt, d.h. daß in φ nicht-triviale Gleichungen auftreten. Es wird für diesen Fall nun in den Unterkapiteln 3.4.2.1 bis 3.4.2.3 ausführlich erklärt, wie man die Quantoren von φ unter Berücksichtigung der Konjunktion ξ eliminiert: Man berechnet aus den Gleichungen in φ ein Gröbnersystem(siehe 3.4.2.1), eliminiert separat die Äste dieses Gröbnersystems(siehe 3.4.2.2) und setzt die Eliminationsergebnisse der einzelnen Äste zum Gesamteliminationsergebnis von φ zusammen(siehe 3.4.2.3).

3.4.2.1 Berechnung eines Gröbnersystems aus den Gleichungen

Man bestimmt zunächst aus den in φ enthaltenen Gleichungen ein grüngefärbtes Gröbnersystem, was in diesem Unterkapitel genauer erläutert wird. Dabei geht man folgendermaßen vor:

Man berechnet bzgl. einer beliebigen Termordnung $<$ ein grüngefärbtes Gröbnersystem S für das Ideal

$$I := Id\big(\{f_i \mid i \in \{1,\dots,r\}\}\big)$$

über einer initialen Bedingung $\big(P_=^{(init)}, P_{\neq}^{(init)}\big)$, welche man aus ξ bestimmt. Dazu definiert man

$$\begin{aligned} P_=^{(init)} &:= \big\{p_l \mid l \in \{1,\dots,q\}, \varrho_l \in \{=\}\big\} \quad \text{und} \\ P_{\neq}^{(init)} &:= \big\{p_l \mid l \in \{1,\dots,q\}, \varrho_l \in \{<,\neq,>\}\big\}. \end{aligned}$$

Durch diese Definition von $P_{\neq}^{(init)}$ werden die in ξ auftretenden $<$- und $>$-Relationen zu Ungleichungen abgeschwächt. Durch die Verwendung von initialen Bedingungen werden also nur solche Äste im folgenden betrachtet, deren Bedingungen unter der Annahme der Gültigkeit von ξ erfüllbar sind. Man berechnet dadurch keine aufgrund möglicher Inkonsistenz zu ξ offensichtlich überflüssigen Äste.

Das nächste Ziel ist, jeden einzelnen Ast $((P_=, P_{\neq}), G) \in S$ des Gröbnersystems separat zu eliminieren, d.h. für jeden Ast eliminiert man φ unter der Annahme, daß die jeweilige Bedingung des Astes gilt. Dieses Aufteilen der Elimination auf die verschiedenen Äste ist möglich, da jede Spezialisierung der Parameter genau die Bedingung eines einzelnen Astes des Gröbnersystems erfüllt. Die Elimination eines Astes wird im folgenden Unterkapitel 3.4.2.2 beschrieben.

Es wird nun erläutert, warum man ein grüngefärbtes Gröbnersystem berechnet und warum dessen Berechnung zur separaten Elimination der Äste ausreicht. Es werden nämlich bei dessen Berechnung Fallunterscheidungen durchgeführt, ob die Koeffizienten gewisser Terme in den auftretenden Polynomen verschwinden. D.h. falls $a \in \mathbb{Q}[\underline{U}]$ als höchster Koeffizient eines Polynoms auftritt, muß ggf. eine Fallunterscheidung durchgeführt werden, ob für Spezialisierungen $\underline{u}$ gilt:

$$a(\underline{u}) \neq 0 \qquad \text{oder} \qquad a(\underline{u}) = 0.$$

Im Falle des Verschwindens dieser Koeffizienten werden die zugehörigen Terme mit ihren Koeffizienten aus dem jeweiligen Polynom entfernt. Dies vereinfacht weitere Berechnungen, da nun gewisse, aufgrund der Fallunterscheidung verschwindende Monome nicht mehr weiter betrachtet werden.

Dadurch kann man aus dem grüngefärbten Gröbnersystem keine umfassende Gröbnerbasis mehr durch Vereinigung der einzelnen Gröbnerbasen konstruieren, was aber für die Quantorenelimination in diesem Fall nicht nötig ist. Bei der Veröffentlichung dieses Quantoreneliminationsverfahrens wurden noch umfassende Gröbnerbasen benutzt, um die quantorenfreien Ergebnisformeln zu bestimmen.

Für die separate Quantorenelimination jedes einzelnen Astes genügt deswegen die Berechnung eines grüngefärbten Gröbnersystems, da dadurch lediglich im jeweiligen Ast direkt die Informationen aus der zu erfüllenden Bedingung $(P_=, P_{\neq})$ des Astes bereits benutzt werden, um gewisse Terme mit verschwindenden Koeffizienten aus den Polynomen zu entfernen und somit diese Polynome teilweise zu vereinfachen. Wenn im weiteren Verlauf dieses Kapitels von Gröbnersystemen die Rede ist, so meint man damit ausschließlich grüngefärbte Gröbnersysteme.

Außerdem genügt es bei der Berechnung des Gröbnersystems S, nur Äste $((P_=, P_{\neq}), G) \in S$ zu bilden, so daß mindestens eine reelle Spezialisierung der Parameter existiert, die $(P_=, P_{\neq})$ erfüllt. Dies wird an folgendem Beispiel erläutert:

Beispiel 3.21 *Sei $f = (U^2 + 1) \cdot X + Y \in \mathbb{Q}[U][X, Y]$. Berechnet man nun ein grüngefärbtes Gröbnersystem von $Id(\{f\})$ bzgl. der lexikographischen Termordnung mit $X > Y$, so besitzt dieses zwei Äste, nämlich:*

Ast 1:	*Gröbnerbasis ist $\{(U^2 + 1) \cdot X + Y\}$,*	*falls $U^2 + 1 \neq 0$.*
Ast 2:	*Gröbnerbasis ist $\{Y\}$,*	*falls $U^2 + 1 = 0$.*

Zu Ast 2 existiert aber keine reelle Spezialisierung von U, so daß die Bedingung von Ast 2 erfüllt ist. Da aber die hier vorgestellte Quantorenelimination in der Theorie der reellen Zahlen durchgeführt wird, braucht man diesen Ast nicht mehr weiter zu untersuchen, da nur reelle Spezialisierungen für alle Parameter betrachtet werden müssen. Somit ist man für jede reelle Spezialisierung von U in Ast 1. $\diamond$

Dieses Beispiel kann natürlich beliebig verallgemeinert werden. Also muß man nur Äste $((P_=, P_{\neq}), G) \in S$ eliminieren, so daß

$$\exists \underline{u} \in \mathbb{R}^m \ : \ \text{„}\underline{u} \text{ erfüllt } (P_=, P_{\neq})\text{."}$$

gilt. Dieses Ausschließen von Ästen kann ggf. zu einem Performance-Gewinn führen. Denn wenn man bei der Berechnung von grüngefärbten Gröbnersystemen bereits solche Äste ausschließt, deren jeweilige Bedingung von keiner reellen Spezialisierungen der Parameter erfüllt wird, müssen zunächst bei der Berechnung weniger Fallunterscheidungen durchgeführt werden und damit letztlich weniger Äste eliminiert werden. Letztendlich kann damit auch ggf. die Komplexität der Ergebnisformel gesenkt werden.

Es führt jedoch nicht zu einem falschen Endergebnis, wenn trotzdem Äste eliminiert werden, deren Bedingungen von keiner reellen Spezialisierung der Parameter erfüllt werden. Weil jeder Ast unter der Annahme der Gültigkeit seiner Bedingung eliminiert wird, ist dann das Eliminationsergebnis von einem Ast, dessen Bedingung von keiner reellen Spezialisierung erfüllt wird, bedeutungslos, da dann bereits die Bedingung jenes Astes äquivalent zu *false* ist. In Kapitel 7 findet sich eine Beispielformel, deren Quantoren einmal mit und einmal ohne Berücksichtigung der Existenz reeller Spezialisierungen von Ästen eliminiert werden. Dort wird die Berechnung beschleunigt, indem man gewisse Äste, deren Bedingungen von keiner reellen Spezialisierung der Parameter erfüllt werden, streicht und damit nicht mehr weiter betrachtet.

Man darf also auch dann eine Fallunterscheidung nach dem Verschwinden eines parametrischen Koeffizienten machen, wenn dieser Koeffizient unter reellen Spezialisierungen nie verschwindet. Somit muß nicht explizit in jedem Ast untersucht werden, ob die Bedingung eines Astes von einer reellen Spezialisierung der Parameter erfüllt wird. In der Praxis ist es jedoch sinnvoll, nur die notwendigen Äste zu eliminieren, d.h. nur die Äste, deren jeweilige Bedingungen von jeweils einer reellen Spezialisierung erfüllt werden.

Als Sonderfall sei dazu erwähnt, daß ein Gröbnersystem über einer initialen Bedingung, welche von keiner reellen Spezialisierung der Parameter erfüllt wird, keine Äste besitzen muß. D.h. in diesem Fall kann das zugehörige Gröbnersystem die leere Menge sein. Damit müssen dann keine Äste eliminiert werden und das Eliminationsergebnis ist dann *false*, da bereits die initiale Bedingung in der Theorie der reellen Zahlen äquivalent zu *false* ist.

Es wird ab jetzt aus Vereinfachungsgründen für den Rest dieses Kapitels folgende Konvention getroffen, so daß die mathematische Theorie einfacher erläutert werden kann: Eine Inkonsistenz von initialen Bedingungen wird nicht berücksichtigt. Ist eine initiale Bedingung inkonsistent, so kann ein beliebiges Gröbnersystem berechnet werden und eine beliebige Formel als Eliminationsergebnis zurückgegeben werden. Diese Freizügigkeit kann man sich deshalb erlauben, weil die Elimination unter der Annahme der Gültigkeit von ξ durchgeführt wird, aus welcher sich die initiale Bedingung ableitet. Wenn die initiale Bedingung $\left(P_=^{(init)}, P_{\neq}^{(init)}\right)$ inkonsistent ist, ist somit das Gesamteliminationsergebnis ψ unbedeutend, denn $\xi \wedge \psi$ ist aufgrund der Inkonsistenz von $\left(P_=^{(init)}, P_{\neq}^{(init)}\right)$ stets äquivalent zu *false*. Somit spielt es mathematisch gesehen keine Rolle, ob in initialen Bedingungen eine Inkonsistenz auftritt bzw. eine solche erkannt wird, da im Falle von Inkonsistenz das Eliminationsergebnis irrelevant ist. Somit muß eine evtl. vorliegende Inkonsistenz von $\left(P_=^{(init)}, P_{\neq}^{(init)}\right)$ nicht gesondert beachtet werden.

Mit dieser Konvention können alle auftretenden Gröbnersysteme als nichtleer angenommen werden. Dies vereinfacht Beweise und erlaubt es, auf eine Fallunterscheidung zu verzichten, ob auftretende Gröbnersysteme leer sind.

In der Praxis ist es natürlich äußerst sinnvoll, die Berechnung von Gröbnersystemen mit leicht erkennbaren Inkonsistenzen in initialen Bedingungen zu vermeiden. Im Fall von Inkonsistenz kann die Berechnung des Gröbnersystems abgebrochen werden und direkt *false* als Eliminationsergebnis ψ ausgegeben werden. Jedoch ist es in der Praxis ein großes Problem, Inkonsistenzen „schnell" zu erkennen.

Außerdem soll an dieser Stelle das Ergebnis aus Satz 1.24 erwähnt werden. Man kann im Hinblick auf weitere Berechungen, insbesondere der Berechnung von Strukturkonstanten, in jedem Ast $((P_=, P_{\neq}), G) \in S$ die Gröbnerbasis G sowohl mit Algorithmus `SMALLERGB` aus Abbildung 1.1 bzgl. der Inklusion verkleinern als auch daraus direkt eine reduzierte Gröbnerbasis berechnen. In beiden Fällen muß man keine weiteren Fallunterscheidungen nach den Parametern $\underline{U}$ machen und somit entstehen in beiden Fällen keine neuen Äste. Dies ist allerdings nun eine abzuwägende Frage, ob man durch die Berechnung von reduzierten Gröbnerbasen mehr Effizienz gewinnt, anstatt durch deren Berechnung an Effizienz zu verlieren. Bisher zeigte sich in der Praxis, daß eine Berechnung von reduzierten Gröbnersystemen auf eben aufgeführte Art und Weise weder einen Performance-Verlust noch einen Effizienzgewinn mit sich brachte.

Nach der eben beschriebenen Berechnung eines grüngefärbten Gröbnersystems S aus den Gleichungen von φ wird nun – wie bereits erwähnt – jeder einzelne Ast separat eliminiert, was im folgenden Unterkapitel erläutert wird.

3.4.2.2 Elimination eines Astes

In diesem Unterkapitel wird gezeigt, wie man einen beliebigen Ast des berechneten Gröbnersystems S eliminiert. Dazu betrachtet man jetzt einen einzelnen Ast mit Bedingung $(P_=, P_{\neq})$, daraus resultierender Konjunktion

$$\zeta := \bigwedge_{p \in P_=} p = 0 \wedge \bigwedge_{p' \in P_{\neq}} p' \neq 0$$

und zugehöriger Gröbnerbasis G. Ziel ist die Bestimmung einer quantorenfreien Formel ψ_{Ast}, so daß gilt:

$$\xi \wedge \zeta \wedge \varphi \iff \xi \wedge \zeta \wedge \psi_{Ast}.$$

Eine mögliche Inkonsistenz von ζ bzw. $\xi \wedge \zeta$ muß im folgenden nicht gesondert berücksichtigt werden, da dann das Eliminationsergebnis ψ_{Ast} bedeutungslos ist. Somit kann man sich darauf beschränken, eine korrekte quantorenfreie Formel ψ_{Ast} für den Fall zu bestimmen, daß ζ bzw. $\xi \wedge \zeta$ erfüllbar sind. Man geht zur Elimination des Astes nun wie folgt vor:

Aus den höchsten Termen von G bestimmt man zunächst die Dimension *dim(I)* des Ideals I im aktuell betrachteten Ast, sowie eine maximal stark unabhängige Menge von Variablen von minimaler Kardinalität in diesem Ast. Nach Satz 1.25 sind die Dimension und die Menge der maximal stark unabhängigen Variablenmengen eindeutig in jedem Ast des Gröbnersystems S bestimmt und lassen sich in jedem Ast algorithmisch bestimmen(siehe [2], Table 9.6). Es wird nun eine Fallunterscheidung nach vier Fällen von *dim(I)* gemacht:

dim(I) = −1 : In diesem Fall ist

$$\sigma_{\underline{u}}(I) = \mathbb{Q}[\underline{U}][\underline{X}]$$

für jede Spezialisierung $\underline{u} \in \mathbb{R}^m$, welche die Bedingung ζ des Astes erfüllt. Somit ist $1 \in \sigma_{\underline{u}}(I)$ und damit hat I in diesem Ast weder reelle noch komplexe Nullstellen. Also besitzen die f_i keine gemeinsamen Nullstellen in diesem Ast, d.h. die Existenzaussage von φ in diesem Ast ist äquivalent zu *false*.

dim(I) = 0 : In diesem Fall wird untersucht, für welche ζ-erfüllenden Spezialisierungen $\underline{u} \in \mathbb{R}^m$ das Ideal $\sigma_{\underline{u}}(I)$ eine reelle Nullstelle $\alpha \in \mathbb{R}^n$ besitzt, welche die Nebenbedingung

$$\bigwedge_{j=1}^{s} g_j(\underline{u}, \alpha) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{u}, \alpha) \neq 0$$

erfüllt. Es sei darauf hingewiesen, daß eine Nullstelle von $\sigma_{\underline{u}}(I)$ insbesondere eine gemeinsame Nullstelle der $f_i(\underline{u}, \underline{X})$,$i \in \{1, \ldots, r\}$, ist und umgekehrt, da

$$\sigma_{\underline{u}}(I) = Id\big(\{f_i(\underline{u}, \underline{X}) \mid i \in \{1, \ldots, r\}\}\big).$$

Somit ist zu $\underline{u}$ die Existenz eines solchen α äquivalent zur Gültigkeit der Existenzaussage von $\varphi(\underline{u})$. Es ist nun im folgenden die Aufgabe, mit den Parametern $\underline{U}$ zu beschreiben, wann eine Nullstelle α von I in diesem Ast existiert, die der Nebenbedingung genügt. Dazu werden die Ergebnisse aus Abschnitt 3.2 verwendet, wobei man ab jetzt die Arithmetik des Polynomrings $\mathbb{Q}(\underline{U})[\underline{X}]$ verwendet.

Man bestimmt also in diesem Ast gemäß Satz 3.13 und dessen anschließenden Bemerkungen für das Ideal I, die Gröbnerbasis G und den Nebenbedingungen

$$g_j(\underline{U}, \underline{x}) > 0, \quad h_k(\underline{U}, \underline{x}) \neq 0, \quad j \in \{1, \ldots, s\}, k \in \{1, \ldots, t\},$$

zunächst das charakteristische Polynom

$$\chi = \prod_{e \in E_{s,t}} \chi_e = Y_{char}^d + \sum_{i=0}^{d-1} a_i Y_{char}^i \in \mathbb{Q}(\underline{U})[Y_{char}].$$

Sei $T_d^{(0)}(C_0, \ldots, C_{d-1})$ eine Typformel für Polynome vom Grad d mit Typ 0. Dann ist also in diesem Ast die Existenzaussage von φ äquivalent zur quantorenfreien Formel

$$\neg T_d^{(0)}(a_0, \ldots, a_{d-1}).$$

Die Terme dieser Formel sind aber Elemente von $\mathbb{Q}(\underline{U})$, so daß – um syntaktisch korrekte Formeln über der zu Beginn dieses Kapitels eingeführten Sprache zu erhalten – man nun noch jede auftretende atomare Formel geeignet äquivalent umformen muß, wie es am Ende von Abschnitt 3.2 beschrieben wurde. D. h. alle Terme werden zunächst u. a. durch Hauptnennerbildung umgewandelt in die Form $t_1 \div t_2$ mit $t_1, t_2 \in \mathbb{Z}[U_1, \ldots, U_m]$ und danach werden alle atomaren Formeln der Form „$t_1 \div t_2 \ \varrho \ 0$" äquivalent umgeformt zu „$t_1 \ \varrho \ 0$", falls $\varrho \in \{=, \neq\}$, bzw. zu „$t_1 \cdot t_2 \ \varrho \ 0$", falls $\varrho \in \{<, \leq, \geq, >\}$. Somit erhält man nun mit der Annahme der Gültigkeit von ξ für die Existenzaussage von φ in diesem Ast eine äquivalente quantorenfreie Formel ψ_{Ast} über der Sprache der angeordneten Ringe in den Variablen $\underline{U}$, so daß gilt:

$$\xi \wedge \zeta \wedge \varphi \quad \Leftrightarrow \quad \xi \wedge \zeta \wedge \psi_{Ast}.$$

Es wurde nun gezeigt, wie man für den Fall $dim(I) = 0$ die Quantoren von φ in diesem Ast eliminieren kann. Dabei greift man auf die Bestimmung der Matrizen aus Satz 3.12 zurück. Dort rechnet man im Restklassenpolynomring $\mathbb{Q}(\underline{U})[\underline{X}]/I$ und führt i. a. sehr viele Multiplikationsrechnungen durch. In Kapitel 2 wurde mit der Einführung von Strukturkonstanten gezeigt, wie man derartige Berechnungen effizienter gestalten kann. Beispiel 2.4 berechnet genau dieselben Matrizen, wie sie in Satz 3.12 auftreten. Zu diesem Beispiel wurden bereits in Abschnitt 2.3 Zeitvergleiche durchgeführt, falls man die Spuren der Matrizen aus Satz 3.12 mit oder ohne Zuhilfenahme von Strukturkonstanten 4-ter Ordnung berechnet, und es zeigte sich, daß große Zeitgewinne durch die Verwendung von Strukturkonstanten möglich sind. Weitere Zeitvergleiche bei Eliminationsbeispielen in Abhängigkeit davon, ob man Strukturkonstanten an dieser Stelle des Algorithmus verwendet, finden sich in Kapitel 7.2, wo erneut die positive Wirkung der Strukturkonstanten unterstrichen wird.

Da also an dieser Stelle sehr viele Produkte in $\mathbb{Q}(\underline{U})[\underline{X}]/I$ zu bilden sind, empfiehlt es sich, Strukturkonstanten 4-ter Ordnung zu verwenden. Insbesondere kann man zunächst G mittels Algorithmus `SMALLERGB` aus Abbildung 1.1 bzgl. der Inklusion verkleinern oder sogar daraus eine reduzierte Gröbnerbasis berechnen, um die Strukturkonstanten noch effizienter bestimmen zu können. Beide Berechnungen führen, wie eingangs erwähnt, zu keinen neuen Fallunterscheidungen oder Ästen des Gröbnersystems. In der Praxis zeigte sich, daß die Berechnung von reduzierten Gröbnerbasen zur effizienteren Bestimmung der Strukturkonstanten weder positiv noch negativ bei den Berechnungszeiten der Eliminationsergebnisse zu Buche schlugen.

$\mathbf{dim(I)} \in \{\mathbf{1}, \ldots, \mathbf{n-1}\}$: In diesem Fall wird die eingangs berechnete maximale Menge stark unabhängiger Variablen von minimaler Kardinalität d verwendet. Sei nach eventueller Umnummerierung nun o.B.d.A.

$$\{x_{n-d+1}, \ldots, x_n\}$$

diese Menge maximal stark unabhängiger Variablen und $\{x_1, \ldots, x_{n-d}\}$ ihr Komplement bzgl. den gebundenen Variablen von φ. Dann wird zunächst in einem rekursiven Aufruf die Formel in Eliminationsnormalform

$$\text{(Schritt 1)} \quad \exists x_1 \ldots \exists x_{n-d}\Big(\bigwedge_{i=1}^{r} f_i(\underline{U}, \underline{x}) = 0 \wedge \bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0\Big)$$

eliminiert, wobei bei dieser Elimination weiterhin die Gültigkeit der Konjunktion ξ und zusätzlich die Gültigkeit von ζ angenommen werden darf. Es werden bei diesem rekursiven Eliminationsaufruf also die stark unabhängigen Variablen als zusätzliche Parameter betrachtet.

Sei nun ψ das Ergebnis der Elimination in Schritt 1. Dann eliminiert man unter weiterer Berücksichtigung von ξ und ζ mittels eines zweiten rekursiven Aufrufs von `RRCQE` die nun i. a. nicht mehr in Eliminationsnormalform befindliche Formel

$$\text{(Schritt 2)} \quad \xi \wedge \zeta \wedge \exists x_{n-d+1} \ldots \exists x_n(\psi).$$

Ziel ist es in Schritt 1, die Elimination – evtl. mit Hilfe erneuter rekursiver Aufrufe – auf den Fall eines nulldimensionalen Ideals zurückzuführen. Daß dies wirklich gelingt und auch keine endlosen Eliminationsschleifen auftreten können, wird später gezeigt.

Es wird nun erläutert, warum man in Schritt 1 das Komplement einer maximal stark unabhängigen Variablenmenge mit minimaler Kardinalität eliminiert. Die Komplementbildung ist hier sowie im folgenden bzgl. der in φ gebundenen Variablen $\{x_1, \ldots, x_n\}$ zu verstehen. Die Auswahl solch einer Menge zur Elimination in Schritt 1 erweist sich aus heuristischer Sicht positiv. Man möchte bei dieser Elimination letztlich wieder auf den nulldimensionalen Fall zurückkommen. Deshalb will man nur so viele Variablen eliminieren, so daß bei der Elimination der Quantoren in Schritt 1 in mindestens einem Ast des zugehörigen grüngefärbten Gröbnersystems das Ideal nulldimensional ist, d.h. der Fall „$dim(I) = 0$" eintritt. Andererseits möchte man in einem Eliminationsschritt möglichst viele Variablen eliminieren. Um diese beiden Kriterien möglichst gut zu erfüllen, eignet sich als zu eliminierende Variablenmenge in Schritt 1 das Komplement einer bzgl. der Kardinalität kleinsten maximal stark unabhängigen Variablenteilmenge von $\{x_1, \ldots, x_n\}$. Bei dieser Wahl als zu eliminierende Variablen in Schritt 1 hat das zugehörige Gröbnersystem nach Korollar 1.29 mindestens einen nulldimensionalen Ast, falls $\xi \wedge \zeta$ erfüllbar ist.

Somit kann man bei der Quantorenelimination in Schritt 1 in mindestens einem Ast die Elimination auf den Fall „$dim(I) = 0$" zurückführen, sofern die freien Variablen $\underline{U}$ nicht zu „bösartigen" Fallunterscheidungen führen, welche eine Inkonsistenz von $\xi \wedge \zeta$ erzeugen. Andererseits werden so möglichst viele Variablen in einem Schritt eliminiert, dadurch daß man das Komplement einer möglichst kleinen maximal stark unabhängigen Variablenmenge zur Elimination auswählt. Umgekehrt werden dadurch möglichst wenig Variablen als zusätzliche Parameter betrachtet, so daß damit die Anzahl der Äste im auftretenden Gröbnersystem bei der Elimination in Schritt 1 als möglichst klein zu erwarten ist. Denn man kann i. a. davon ausgehen, daß ein Gröbnersystem umso mehr Äste besitzt, je mehr Parameter existieren.

Wenn die in Schritt 1 zu eliminierenden Variablen kein Komplement einer maximal stark unabhängigen Variablenmenge bilden, muß es bei der Elimination in Schritt 1 im zugehörigen Gröbnersystem keinen nulldimensionalen Ast geben, so daß es durchaus möglich ist, daß in jenem Gröbnersystem jeder Ast neue rekursive Aufrufe impliziert. Ein kleines Beispiel zeigt diesen Sachverhalt anhand eines Ideals, wenn man eine Variable zum Parameter macht und dann aus dieser Sichtweise ein Gröbnersystem berechnet:

Beispiel 3.22 *Man betrachte das Ideal*

$$Id(\{XYZ - X\}) \subseteq \mathbb{Q}[X, Y, Z].$$

Dann ist bzgl. der lexikographischen Ordnung mit $X > Y > Z$ *die Menge* $\{X, Y\}$ *eine maximal stark unabhängige Teilmenge der Variablenmenge* $\{X, Y, Z\}$. *Also ist* $\{X\}$ *nicht maximal stark unabhängig. Betrachtet man nun das Ideal als Teilmenge von* $\mathbb{Q}[X][Y, Z]$ *so besitzt das zugehörige Gröbnersystem bzgl. der lexikographischen Ordnung mit* $Y > Z$ *zwei nicht-nulldimensionale Äste:*

Ast 1: *Bedingung:* $X \neq 0$, *Dimension des Astes:* 1.
Ast 2: *Bedingung:* $X = 0$, *Dimension des Astes:* 2.

◇

An dieser Stelle sei der Vollständigkeit halber noch erwähnt, daß es auch möglich wäre, in Schritt 1 das Komplement einer *dim(I)*-elementigen maximal stark unabhängigen Variablenmenge zu eliminieren. Dies hätte zur Folge, daß die im Schritt 1 auftretenden Dimensionen der Ideale möglichst klein sind und somit möglichst schnell der Fall „$dim(I) = 0$" in den weiteren rekursiven Aufrufen eintritt, wobei auch bei dieser Wahl der in Schritt 1 zu eliminierenden Variablen mindestens ein Ast des auftretenden Gröbnersystems Dimension 0 besitzt, falls $\xi \wedge \zeta$ nicht inkonsistent ist. Jedoch widerspricht diese Strategie dem Kriterium, möglichst viele Variablen in einem Eliminationsschritt zu eliminieren. Außerdem werden durch diese Wahl der zu eliminierenden Variablen evtl. mehr Variablen als notwendig als Parameter betrachtet, was die Anzahl der Äste des in Schritt 1 auftretenden Gröbnersystems ggf. mehr erhöht, als wenn man weniger Variablen als zusätzliche Parameter betrachtet. Somit empfiehlt es sich, in Schritt 1 das Komplement einer maximal stark unabhängigen Variablenmenge *minimaler* Kardinalität zu eliminieren. In der Praxis zeigte sich aber bisher kein Unterschied, ob maximal stark unabhängige Variablenmengen minimaler oder maximaler Kardinalität ausgewählt werden.

$\mathbf{dim(I) = n}$: In diesem Fall ist

$$\sigma_{\underline{u}}(I) = \{0\}$$

für jede Spezialisierung $\underline{u}$, welche die Bedingung ζ des Astes des Gröbnersystems erfüllt. Es sind also alle auftretenden Gleichungen in φ trivial. Daher müssen in diesem Fall nicht-triviale Gleichungen analog zum Fall $r = 0$ gefunden werden. Dazu werden wiederum die Ergebnisse aus Kapitel 3.3 herangezogen. Man geht im Fall $dim(I) = n$ wie folgt vor:

Da alle Gleichungen in φ trivial sind, eliminiert man in einem rekursiven Aufruf die folgende Formel in Eliminationsnormalform:

$$\exists x_1 \ldots \exists x_n \Big(\bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0 \Big),$$

wobei wiederum die Gültigkeit der Konjunktionen ξ und ζ weiter angenommen werden darf. Somit kommt man beim rekursiven Aufruf sofort in den Fall $r = 0$, wo nicht-triviale Gleichungen aus den $>$-Relationen und den Ungleichungen von φ konstruiert werden.

Hiermit ist die Fallunterscheidung nach der Dimension *dim(I)* des Ideals I vollständig. Die Eliminationsergebnisse der einzelnen Äste von S müssen nun noch zum Gesamteliminationsergebnis von φ zusammengesetzt werden, was im nächsten Unterkapitel beschrieben wird.

3.4.2.3 Zusammensetzen der Eliminationsergebnisse der einzelnen Äste

Da die Äste von S eine disjunkte Fallunterscheidung nach allen reellen Spezialisierungen der Parameter bilden, kann man die Eliminationsergebnisse der einzelnen Äste wie folgt zum Gesamteliminationsergebnis von φ zusammensetzen. Sei in einem Ast A die zugehörige Bedingung $(P^A_=, P^A_{\neq})$ und sei das Eliminationsergebnis in diesem Ast die quantorenfreie Formel ψ_A. Dann hat das Eliminationsergebnis ψ von φ folgende Form:

$$\psi := \bigvee_{A \in S} \Big(\Big(\bigwedge_{p \in P^A_=} p = 0 \Big) \wedge \Big(\bigwedge_{p' \in P^A_{\neq}} p' \neq 0 \Big) \wedge \psi_A \Big).$$

Es wird dabei immer noch die Gültigkeit der Konjunktion ξ vorausgesetzt. Jedoch kann es möglich sein, daß insbesondere in der Bedingung $(P^A_=, P^A_{\neq})$

eines Astes des Gröbnersystems Polynome $p \in P_{=}^{A}$ bzw. $p' \in P_{\neq}^{A}$ auftauchen, die in ξ als Konjunktionsglied $p = 0$ bzw. $p' \neq 0$ enthalten sind, da aus ξ die initiale Bedingung für das grüngefärbte Gröbnersystem abgeleitet wurde. Im eben aufgeführten Eliminationsergebnis ψ können somit atomare Formeln, die aus der initialen Bedingung abgeleitet wurden und deren Auftreten in ψ aufgrund der Bedingung eines Astes verursacht wird, wegen der Gültigkeit von ξ gestrichen werden.

Für diese quantorenfreie Formel ψ gilt nun die gewünschte Äquivalenz:

$$\xi \wedge \varphi \quad \Longleftrightarrow \quad \xi \wedge \psi.$$

Somit wurde nun gezeigt, wie man eine Formel φ in Eliminationsnormalform unter Berücksichtigung von ξ eliminieren kann. Im nächsten Unterkapitel werden Termination und Korrektheit des Algorithmus nachgereicht, sowie zusätzliche Anmerkungen gemacht.

3.4.3 Zusammenfassung des Algorithmus

Der Eliminationsalgorithmus für Formeln erster Stufe in Eliminationsnormalform läßt sich wie folgt zusammenfassen:

Satz 3.23 *Sei φ eine Formel erster Stufe in Eliminationsnormalform. Ferner sei ξ eine Konjunktion atomarer Formeln der Form*

$$\bigwedge_{i=1}^{q} p_i \ \varrho_i \ 0 \qquad \textit{mit } \varrho_i \in \{<, \leq, =, \neq, \geq, >\}, \ q \in \mathbb{N} \ \ \textit{und} \ \ p_i \in \mathbb{Q}[\underline{U}],$$

wobei die Variablen $\underline{U}$ disjunkt zu den in φ gebunden auftretenden Variablen sind.
Dann berechnet der Algorithmus `QENF` in Abbildung 3.4 zu ξ und φ eine quantorenfreie Formel ψ, so daß gilt:

$$\xi \wedge \varphi \quad \Longleftrightarrow \quad \xi \wedge \psi \ .$$

Beweis:

Termination: Es wird gezeigt, daß die Anzahl der gebundenen Variablen antiton ist und spätestens nach einem zweiten rekursiven Aufruf echt abnimmt. Zunächst sei angemerkt, daß die *for*-Schleifen in Zeile 8 und Zeile 16 endlich sind, da eine Disjunktion nur endlich viele Disjunktionsglieder besitzt bzw. ein grüngefärbtes Gröbnersystem nur endlich viele Äste besitzt. Es wird eine Fallunterscheidung nach $r = 0$ bzw. $r \neq 0$ und der Dimension $dim(I)$ des zugehörigen Ideals in einem Ast gemacht:

$\mathbf{r \neq 0}$ und $\mathbf{dim(I) \in \{-1, 0\}}$: In diesen Fällen gibt es keine rekursiven Aufrufe von `QENF` bzw. `RRCQE`, also terminiert der Algorithmus.

Algorithmus QENF:

Eingabe: φ und ξ mit:

- φ, eine Formel erster Stufe über der Sprache der angeordneten Ringe in Eliminationsnormalform
- ξ, eine Konjunktion atomarer Formeln über der Sprache der angeordneten Ringe, deren Variablen unabhängig von den gebundenen Variablen von φ sind

Ausgabe: ψ, eine quantorenfreie Formel erster Stufe über der Sprache der angeordneten Ringe, so daß gilt: $\xi \wedge \varphi \Leftrightarrow \xi \wedge \psi$

procedure QENF(ξ,φ)
begin
if $\varphi =$ *true* **or** $\varphi =$ *false* **then return** φ **end**
if $r = 0$ **then**
  $\hat{\varphi}$:=„Sei $\tilde{\varphi}$ die Matrix von φ. Forme $\exists x_n(\tilde{\varphi})$ nach Satz 3.17 um in die Form $\left(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0\right) \wedge \left(\varphi_1 \vee \varphi_2 \vee \varphi_3 \vee \varphi_4\right)$."
  $\psi := \varphi_1 \vee \varphi_2$
  $\gamma :=$ „Umformung von $\varphi_3 \vee \varphi_4$ nach Lemma 3.20 zu γ"
  for each „Disjunktionsglied der Form $\omega(f,i) \wedge \exists x_n(\phi)$ in γ" **do**
    $\psi := \psi \vee \left(\omega(f,i) \wedge \mathtt{QENF}(\tilde{\omega}(f,i), \exists x_n(\phi))\right)$
  end
  $\psi := \mathtt{RRCQE}\left(\xi \wedge \exists x_1 \ldots \exists x_{n-1}\left(\left(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0\right) \wedge \psi\right)\right)$
  return ψ
end
$\psi :=$ *false*
$S :=$ „Grüngefärbtes (reell spezifiziertes) Gröbnersystem für die Polynommenge $\{f_i \mid i \in \{1, \ldots, r\}\}$ über der initialen Bedingung ξ"
for each $((P_=, P_{\neq}), G) \in S$ **do**
  $\zeta := \left(\bigwedge_{p \in P_=} p = 0\right) \wedge \left(\bigwedge_{p' \in P_{\neq}} p' \neq 0\right)$
  dim := „Dimension des Astes $((P_=, P_{\neq}), G)$"
  if *dim* $= -1$ **then**
    $\psi := \psi \vee (\zeta \wedge \mathit{false})$
  elsif *dim* $= 0$ **then**
    χ :=„Bestimme $\prod_{e \in E_{s,t}} \chi_e$ nach Satz 3.13; sei $\chi = Y_{char}^d + \sum_{\ell=0}^{d-1} a_\ell Y_{char}^\ell$."
    $\psi' := \neg T_d^{(0)}(a_0, \ldots, a_{d-1})$
    ψ' :=„Äquivalente Umformung von ψ', so daß alle Terme Elemente von $\mathbb{Z}[\underline{U}]$, wobei $\underline{U}$ die ungebundenen Variablen von φ sind."
    $\psi := \psi \vee (\zeta \wedge \psi')$
  elsif *dim* $\in \{1, \ldots, n-1\}$ **do**
    $\underline{x_u}$:=„Wähle eine maximale Menge stark unabhängiger Variablen von $\{x_1, \ldots, x_n\}$ kleinster Kardinalität."
    $\underline{x_a}$:=„Komplement von $\underline{x_u}$ bzgl. $\{x_1, \ldots, x_n\}$"
    ψ' :=„Matrix von φ"
    $\psi' := \mathtt{QENF}(\xi \wedge \zeta, \exists \underline{x_a}(\psi'))$
    $\psi' := \mathtt{RRCQE}(\xi \wedge \zeta \wedge \exists \underline{x_u}(\psi'))$
    $\psi := \psi \vee (\zeta \wedge \psi')$
  else
    $\psi := \psi \vee \left(\zeta \wedge \mathtt{QENF}\left(\xi \wedge \zeta, \exists \underline{x}\left(\bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0\right)\right)\right)$
  end
end
return ψ
end QENF

Abbildung 3.4: Algorithmus zur Hermiteschen Quantorenelimination einer beliebigen Formel erster Stufe in Eliminationsnormalform

$\mathbf{r \neq 0}$ und $\mathbf{dim(I) \in \{1, \ldots, n-1\}}$: In diesen Fällen nimmt beim rekursiven Aufruf von `QENF` in Zeile 30 die Anzahl der gebundenen Variablen echt ab, ebenso beim Aufruf von `RRCQE` in Zeile 31: In Zeile 30 müssen $n - dim(I)$ Variablen eliminiert werden, und in Zeile 31 $dim(I)$ Variablen. Die auftretenden `QENF`-Aufrufe innerhalb des `RRCQE`-Aufrufs von Zeile 31 müssen daher maximal $dim(I)$ Variablen eliminieren. Da die Anzahl der gebundenen Variablen nach unten durch 0 beschränkt ist, kann also dieser Fall nicht unendlich oft auftreten.

$\mathbf{r = 0}$: Es wird zunächst x_n in den endlich vielen Disjunktionsgliedern von γ separat eliminiert: Nach Konstruktion von γ ist jedes Disjunktionsglied von der Form $\omega(f,i) \wedge \exists x_n(\phi)$(siehe Lemma 3.20), wobei ϕ genau eine nicht-triviale Gleichung der Form $f^{(i)}(\underline{U}, \underline{x}) = 0$ enthält. Es wird nun untersucht, wenn ein Disjunktionsglied dieser Form in Zeile 9 rekursiv eliminiert wird:

Die Prämisse $\tilde{\omega}(f,i)$ sichert beim Aufruf in Zeile 9, daß $Grad_{X_n}(f^{(i)}) = i > 0$ ist und der höchste Koeffizient von $f^{(i)}$ bzgl. X_n unter Spezialisierungen nicht verschwindet. Also besitzt das beim rekursiven Aufruf in Zeile 9 entstehende Gröbnersystem zur einzigen Gleichung in ϕ mit initialer Bedingung $\tilde{\omega}(f,i)$ nur einen Ast, welcher Dimension 0 besitzt. Vgl. Bemerkungen zu Definition 1.19.

Somit tritt bei den rekursiven Aufrufen von `QENF` in Zeile 9 stets der Fall ein, daß

$$r \neq 0 \quad \text{und} \quad dim(I) = 0 \; .$$

In diesem Fall wurde aber bereits die Termination von `QENF` bewiesen. Im Aufruf von `RRCQE` in Zeile 11 ist nun eine Variable weniger zu eliminieren, so daß alle rekursiven Aufrufe von `QENF` innerhalb des `RRCQE`-Aufrufs mindestens eine gebundene Variable weniger eliminieren müssen. Also kann auch dieser Fall nicht unendlich oft auftreten, da die Anzahl der gebundenen Variablen nach unten durch 0 beschränkt ist.

$\mathbf{r \neq 0}$ und $\mathbf{dim(I) = n}$: Es werden in diesem Fall alle nun trivialen Gleichungen in φ weggelassen und mit dieser „verkürzten" Formel der Algorithmus `QENF` rekursiv aufgerufen. Damit tritt im rekursiven Aufruf der Fall $r = 0$ ein, von welchem bereits dessen Termination gezeigt wurde. Insbesondere kann dadurch der Fall $dim(I) = n$ nicht erneut auftreten.

Also terminiert jeder einzelne Ast der *for*-Schleife in Zeile 16 und damit aufgrund der endlichen Anzahl von Ästen in einem Gröbnersystem auch der gesamte Algorithmus.

Korrektheit: Zunächst sei nochmals erwähnt, daß man durch die Betrachtung eines grüngefärbten Gröbnersystems mit einer initialen Bedingung ξ eine vollständige Fallunterscheidung nach den (reellen) Spezialisierungen der ungebundenen Variablen von φ – den Parametern – macht, welche ξ erfüllen. Somit ist es korrekt, die Eliminiationsergebnisse der einzelnen Äste unter Berücksichtigung der Bedingungen der Äste disjunktiv zusammenzusetzen, wie es in den Zeilen 20, 25, 32 und 34 geschieht.

Ferner sind alle auftretenden Aufrufe von `RRCQE` und `QENF` wohldefiniert.

Der Korrektheitsbeweis wird durch vollständige Induktion nach der Anzahl n der gebundenen Variablen von φ geführt. Dabei wird eine Fallunterscheidung nach $dim(I)$ und $r = 0$ bzw. $r \neq 0$ gemacht und die Korrektheit jedes Falles gezeigt.

Sei $n = 1$. Somit kann im Fall $r \neq 0$ nur $dim(I) \in \{-1, 0, 1\}$ gelten.

Falls $\mathbf{r \neq 0}$ und $\mathbf{dim(I) = -1}$, so ist das Ideal I im zugehörigen Ast des Gröbnersystems der ganze Polynomring in der gebundenen Variablen, enthält also inbesondere die Konstante 1 und daher hat das Ideal in diesem Fall keine Nullstelle. Also ist die Existenzaussage von φ in diesem Ast äquivalent zu *false*.

Falls $\mathbf{r \neq 0}$ und $\mathbf{dim(I) = 0}$, so wird nach Abschnitt 3.2 korrekt eine zu φ äquivalente quantorenfreie Formel für diesen Ast konstruiert.

Falls $\mathbf{r = 0}$, so finden zuerst nach Satz 3.17 und Lemma 3.20 Äquivalenzumformungen statt und es wird dann in Zeile 9 jedes einzelne Disjunktionsglied von γ der Form $\omega(f, i) \wedge \exists x_n(\phi)$ separat eliminiert, wobei $\exists x_n(\phi)$ in Eliminationsnormalform ist. Dabei gilt folgendes:
In den rekursiven Aufrufen von Zeile 9 wird nur eine gebundene Variable eliminiert. Man beachte, daß $\tilde{\omega}(f, i)$ als atomare Formel in $\omega(f, i)$ enthalten ist, so daß $\tilde{\omega}(f, i)$ als Prämisse an den rekursiven Aufruf in Zeile 9 übergeben werden darf. Also tritt in diesen Aufrufen nach dem Auffinden nicht-trivialer Gleichungen, wie bereits im Terminationsbeweis zum Fall $r = 0$ gezeigt, stets der bereits als korrekt gezeigte Fall

$$r \neq 0 \quad \text{und} \quad dim(I) = 0$$

ein. Ferner werden die einzelnen Eliminationsergebnisse aus Zeile 9 disjunktiv korrekt verknüpft. Der Aufruf von `RRCQE` findet hier nicht statt, da $n = 1$ ist und somit keine weitere Variable außer x_1 zu eliminieren ist, so daß auch der Fall $r = 0$ korrekt behandelt wird.

Falls $\mathbf{r \neq 0}$ und $\mathbf{dim(I) = 1}$, so wird der Algorithmus in Zeile 34 rekursiv aufgerufen, wobei bei diesem Aufruf alle nun trivialen Gleichungen in φ korrekt weggelassen werden dürfen, somit der Fall $r = 0$ eintritt und ebenfalls nur eine Variable zu eliminieren ist. Die Korrektheit des Falles $r = 0$ wurde bereits gezeigt.

Somit werden für $n = 1$ alle auftretenden Fälle korrekt behandelt.

Man betrachtet nun den Induktionsschritt von n nach $n+1$. Die Korrektheit des Algorithmus sei für $1, \ldots, n$ bewiesen.

Die Aufrufe von `RRCQE` in Zeile 11 und 31 liefern nach Induktionsannahme ein korrektes Ergebnis: Dies liegt zum einen an der bereits bewiesenen Korrektheit von `RRCQE`. Zum anderen wird `RRCQE` in beiden Zeilen bereits mit einer Formel in pränexer Normalform mit echt weniger als $n + 1$ mittels Existenzquantoren gebundenen Variablen aufgerufen, so daß bei der Abarbeitung dieser `RRCQE`-Aufrufe `QENF` nur mit Eingabeformeln mit höchstens n gebundenen Variablen aufgerufen wird. Nach Induktionsannahme sind diese `QENF`-Aufrufe korrekt.

Wiederum wird zum Korrektheitsbeweis im Induktionsschritt eine Fallunterscheidung durchgeführt:

Falls $\mathbf{r \neq 0}$ und $\mathbf{dim(I) \in \{-1, 0\}}$, finden keine rekursiven Aufrufe statt. Die Korrektheit folgt völlig identisch wie im Fall $n = 1$.

Falls $\mathbf{r \neq 0}$ und $\mathbf{dim(I) \in \{1, \ldots, n\}}$, so werden die gebundenen Variablen von φ in zwei Schritten eliminiert. Beim rekursiven Aufruf des Algorithmus in Zeile 30 wird eine Formel in Eliminationsnormalform mit echt weniger als $n + 1$, nämlich $n - dim(I)$ gebundenden Variablen an `QENF` übergeben. Somit liefert nach Induktionsannahme Zeile 30 ein korrektes Ergebnis.
Der Aufruf von `RRCQE` liefert, wie bereits gezeigt, ebenso ein korrektes Ergebnis. Daraus folgt die Korrektheit des Falles.

Falls $\mathbf{r = 0}$, wird zunächst in Zeile 9 `QENF` rekursiv aufgerufen und die Variable x_n eliminiert, wobei bei diesen Aufrufen in Zeile 9 der Fall einer einzigen gebundenen Variablen wie im Induktionsbeginn vorliegt, also korrekt ist.
Nach der Elimination von x_n werden in einem Aufruf von `RRCQE` in Zeile 11 die restlichen in φ gebundenen Variablen eliminiert. Dessen Korrektheit wurde bereits gezeigt, also ist der Fall $r = 0$ korrekt.

Falls $\mathbf{r \neq 0}$ und $\mathbf{dim(I) = n + 1}$, kann man alle nun trivialen Gleichungen in φ weglassen und macht einen rekursiven Aufruf von `QENF`, in dem dann automatisch der Fall $r = 0$ eintritt, dessen Korrektheit bereits gezeigt wurde.

Damit folgt die Korrektheit des Algorithmus. □

Somit wurde ein Quantoreneliminationsverfahren vorgestellt, das hauptsächlich mit der Charakterisierung arbeitet, wann parametrische Ideale reelle Nullstellen besitzen, die gewissen Nebenbedingungen genügen. Da das hier verwendete Zählen dieser reellen Nullstellen von Idealen auf den frz. Mathematiker Hermite zurückgeht, nennt man dieses vorgestellte Eliminationsverfahren *Hermitesche Quantorenelimination.*

Anmerkungen zum Fall $\mathbf{r = 0}$:
Es wird jetzt erläutert, welchen Vorteil im Algorithmus `QENF` in Zeile 5 die Äquivalenzumformungen aus Satz 3.17 und Lemma 3.20 mit sich bringen, so daß gesichert ist, daß gewisse Polynome f nicht konstant sind. Es werden nämlich dadurch die rekursiven Aufrufe von `QENF` in Zeile 9 möglichst leicht und unkompliziert gestaltet. Nach Umformungen mittels Lemma 3.20 sind also in Zeile 9 Formeln der Form

$$(*) \quad \exists x\Big(\bigwedge_{j=1}^{s} g_j(\underline{U},x) > 0 \wedge f^{(i)}(\underline{U},x) = 0\Big) \qquad \text{mit } g_j, f \in \mathbb{Z}[\underline{U}][X]$$

unter der Annahme der Gültigkeit der Ungleichung $\tilde{\omega}(f,i)$ zu eliminieren, wobei f die formale Ableitung eines g_ℓ bzgl. der Variablen X bzw. die Differenz zweier Elemente g_k, g_ℓ ist und $\omega(f,i) \in \Omega_f$. Im Beweis von Satz 3.23 wurde gezeigt, daß bei diesen Eliminationen mittels rekursiven Aufrufen von `QENF` die auftretenden grüngefärbten Gröbnersysteme nur einen Ast besitzen, da $\tilde{\omega}(f,i)$ eindeutig den höchsten Term in $f^{(i)}$ festlegt und sichert, daß $f^{(i)}$ nicht konstant bzgl. der Hauptvariablen X in diesem Ast ist. Also tritt bei jeder dieser Eliminationen in dem jeweiligen einzigen Ast des Gröbnersystems der Fall

$$„r \neq 0 \quad \text{und} \quad dim(I) = 0“$$

ein. Dieser wird dann wie üblich behandelt. Somit kann man die Rekursionen in Zeile 9 *auflösen*, da man hier bei den Aufrufen von `QENF` mit den Eingabeformeln der Form $(*)$ sofort in den nulldimensionalen Fall kommt. Außerdem ist ein sehr wichtiger Punkt hierbei, daß die Formeln $\omega(f,q) \in \Omega_f$ eine disjunkte Fallunterscheidung nach dem höchsten Koeffizienten von f bzgl. X bilden. Diese Tatsache wirkt sich sowohl auf die Größe der Ergebnisformel als auch auf die Berechnungszeit positiv aus.

Bei der Implementierung des Verfahrens kann diese Erkenntnis gleich direkt dazu benutzt werden, um jene rekursiven Aufrufe des Algorithmus zu ersparen und um direkt für die erzeugte Gleichung mit nicht-verschwindendem höchsten Koeffizienten und alle $>$-Relationen zu jedem einzelnen Disjunktionsglied von γ die äquivalente quantorenfreie Formel nach Abschnitt 3.2 zu bestimmen. Außerdem sollte bei der Implementierung die Elimination derjenigen Disjunktionsglieder der Form $\omega(f,i) \wedge \exists x_n(\phi)$ verhindert werden, zu welchen die Teilformel $\omega(f,i)$ offensichtlich nicht erfüllbar ist. Denn in diesem Fall ist das Ergebnis der Elimination von $\exists x_n(\phi)$ ohne Bedeutung, da $\omega(f,i)$ bereits äquivalent zu *false* ist. Dies kann unnötige Eliminationen ersparen. Z. B. sollte erkannt werden, daß die Formel $U = 0 \wedge U \neq 0$ nicht erfüllbar ist. In der Praxis ist also darauf zu achten, daß leicht erkennbare Inkonsistenzen der atomaren Formeln von $\omega(f,q)$ festgestellt werden, insbesondere Inkonsistenz der Formeln $\omega(f,q)$ zu ξ, und somit unnötige Eliminationen in Zeile 9 verhindert werden.

Im Fall $r = 0$ kann man außerdem eine weitere interessante Beobachtung feststellen: Man braucht die Wahl von x_n als erste zu eliminierende Variable in den Aufrufen in Zeile 9 nicht dem Zufall, welcher in Form der Reihenfolge der gebundenen Variablen in der Eingabeformel auftritt, zu überlassen. Es ist zunächst grundsätzlich möglich, statt x_n irgendeine beliebige Variable $x_i \in \{x_1, \dots, x_n\}$ zu verwenden, bzgl. welcher die Umformung in Zeile 5 gemacht wird. Dadurch muß man logischerweise in Zeile 9 Formeln der Form $\exists x_i(\tilde{\phi})$ eliminieren, und dann in Zeile 11 die Formel $\xi \wedge \exists x_1 \dots \exists x_{i-1} \exists x_{i+1} \dots \exists x_n(\psi)$. Im Kapitel 5.2 wird dazu eine Möglichkeit vorgestellt werden, wie man x_i sinnvoll wählen kann.

3.5 Stärken und Schwächen des Verfahrens

Es wird hier kurz zusammengefasst, in welchen Fällen dieses vorgestellte Quantoreneliminationsverfahren mehr oder weniger gut zur Anwendung geeignet ist. Zunächst werden hierzu Faktoren erläutert, die die Komplexität der Berechnungen bei einer Anwendung des Verfahrens rapide ansteigen lassen.

Darunter fällt zum einen die Größe der auftretenden Restklassenringe der Form $\mathbb{Q}(\underline{U})[\underline{X}]/I$, deren Dimensionen durch die Anzahl $|RT(I)|$ der reduzierten Terme gegeben ist. Falls diese Dimensionen sehr groß sind, werden die zu berechnenden Typformeln dementsprechend groß, da die Größe jener Typformeln von diesen Dimensionen abhängen und mit ihnen steigen. Diese großen Typformeln werden nach dem Einsetzen gewisser Koeffizienten dann ggf. als Teilformel wieder an einen rekursiven Aufruf übergeben, so daß der rekursive Aufruf bereits sehr große Eingabeformeln handhaben muß. Dies führt letzlich dazu, daß damit entweder der rekursive Aufruf in der Praxis nicht mehr durchführbar ist oder die Ergebnisformeln sehr groß sind gemessen an der Anzahl der atomaren Formeln.

Ein weiterer störender Faktor zur effizienten Anwendung dieses Verfahrens ist die Existenz einer größeren Anzahl von Nebenbedingungen der Form $g_j > 0$ beim Nullstellenzählen. Für jede zusätzliche Bedingung dieser Form verdoppelt sich der Grad, zu dem eine Typformel zu bestimmen ist, wenn man bei der Elimination in `QENF` in den nulldimensionalen Fall trifft. Denn bei der Berechnung der einzelnen charakteristischen Polynome der Matrizen

Q_{h^e} tritt solch eine Bedingung zum einen in h^e als g_j auf, zum anderen auch als g_j^2(siehe Satz 3.12 und 3.13). Wegen

$$| E_{s+1,t} |= 2\cdot | E_{s,t} | \qquad \text{und} \qquad \chi = \prod_{e \in E_{s+1,t}} \chi_e \quad \text{(Notationen aus 3.13)}$$

müssen deshalb für eine zusätzliche $>$-Relation der Form $g_j > 0$ doppelt so viele charakteristische Polynome berechnet werden. Da die einzelnen charakteristischen Polynome χ_e, $e \in E_{s+1,t}$, denselben Grad besitzen, muß damit bei der Frage nach der Existenz von Nullstellen im nulldimensionalen Fall eine Typformel von doppelt so hohem Grade berechnet werden als im Falle ohne diese zusätzliche Bedingung. Da die Größe der Typformeln gemessen an der Anzahl der atomaren Formeln exponentiell mit dem Grad steigt, ist offensichtlich, daß viele Bedingungen der Form $g_j > 0$ die praktikable Anwendung dieses Verfahrens sehr stark einschränken bzw. unmöglich machen. In der Praxis zeigte sich, daß bereits Eingabeformeln in Eliminationsnormalform mit wenigen $>$-Relationen, aber vielen Parametern, eine effiziente Anwendung des Eliminationsverfahrens verhindern.

Wenn keine Gleichungen in einer Formel in Eliminationsnormalform enthalten sind und die Anzahl der darin enthaltenen $>$-Relationen groß ist, so wird dieser Effekt noch deutlicher. In diesem Fall kommt man im Algorithmus `QENF` in den Fall $r = 0$. Dort müssen dann in Zeile 9 viele einzelne Eliminationen durchgeführt werden, bei denen der nulldimensionale Fall auftritt mit denselben $>$-Relationen als Nebenbedingungen.

Generell kann man die Aussage machen, daß die Berechnungszeiten umso größer werden, je mehr Parameter vorhanden sind. Bereits kleine Abänderungen von Eingabeformeln durch Hinzufügen eines Parameters in einer Eingabeformel können die Berechnungszeiten drastisch ansteigen lassen.

Positiv anzumerken sind die global nulldimensionalen Fälle, in denen jedes auftretende Ideal Dimension Null besitzt. Das spezielle Problem der parametrischen Bestimmung der Existenz von reellen Nullstellen von nulldimensionalen Idealen unter Berücksichtung von Nebenbedingungen ist gut durchführbar, da man beim Aufruf von `QENF` sofort im nulldimensionalen Fall landet. Dadurch muß man innerhalb von `QENF` keine rekursiven Aufrufe starten. Zusätzliche Nebenbedingungen der Form $h \neq 0$ führen erfreulicherweise nicht dazu, daß größere Typformneln berechnet werden müssen, als im Fall ohne solche zusätzlichen Bedingungen. Deshalb machen sich zusätzliche Ungleichungen nicht sofort derart schlecht bemerkbar wie $>$-Relationen. Diese positiven Eigenschaften führen darauf zurück, daß dieses Verfahren auf dem parametrischen Zählen reeller Nullstellen von nulldimensionalen Idealen unter Nebenbedingungen aufbaut und andere, nicht-nulldimensionale Fälle darauf zurückführt. Bei diesen Fragestellungen steigt der Aufwand zur Berechnung des Ergebnisses durch zusätzliche gebunden auftretende Variablen in Formeln in Eliminationsnormalform nicht gleich drastisch im Vergleich zu anderen Eliminationsverfahren. Allerdings wird bei zusätzlichen gebunden auftretenden Variablen dann die Berechnung des Gröbnersystems zeitintensiver.

Umso mehr Äste der Gröbnersysteme der auftretenden Ideale in einem Problem existieren, die eine Dimension größer Null besitzen, desto schneller stößt man an die Grenzen des Verfahrens. Wenn Ideale mit einer Dimension größer Null vorliegen, müssen nun manche Variablen vorerst als zusätzliche Parameter betrachtet werden. In einem rekursiven Aufruf wirken diese zusätzlichen Parameter bei der Berechnung von Gröbnersystemen störend und können noch mehr Fallunterscheidungen und Äste erzeugen.

Für komplexere Eingabeformeln zeigt sich also der Algorithmus als weniger gut anwendbar, jedoch zeigt das Verfahren die Existenz dazu äquivalenter quantorenfreier Formeln.

Kapitel 4

Generische Quantorenelimination

4.1 Einführung

In diesem Abschnitt wird eine eingeschränkte Form der Quantorenelimination einer Formel erster Stufe vorgestellt, die sog. *generische Quantorenelimination*. Die Idee dabei ist, die Quantoren einer Formel erster Stufe zu eliminieren, wobei man Annahmen über die frei vorkommenden Variablen machen darf.
Sei nun

$$\varphi := Q_1 x_1 \dots Q_n x_n (\varphi'(x_1, \dots, x_n, U_1, \dots, U_m)) \quad \text{mit } Q_1, \dots, Q_n \in \{\exists, \forall\}$$

und mit einer quantorenfreien Formel φ'. Die Menge $\{x_1, \dots, x_n\}$ der in φ gebunden auftretenden Variablen wird jetzt mit $\underline{x}$ notiert, analog die Menge $\{U_1, \dots, U_m\}$ der freien Variablen als $\underline{U}$, deren Elemente jetzt als Parameter bezeichnet werden

Eine `Theorie` Θ ist in diesem Fall eine Konjunktion von Ungleichungen in den Parametern $\underline{U}$.

Dann besteht das Ergebnis der generischen Quantorenelimination von φ aus einer Theorie Θ und aus einer quantorenfreien Formel ψ in den Variablen $\underline{U}$, so daß in der Theorie der reellen Zahlen gilt:

$$\forall U_1 \dots \forall U_m \big(\Theta \Longrightarrow (\varphi \Longleftrightarrow \psi)\big)$$

Mit anderen Worten: Die Äquivalenz zwischen φ und ψ gilt mindestens für diejenigen reellen Parameterwerte, welche Θ erfüllen.

Dabei entsteht Θ so: Wann immer man bei der Elimination von φ eine Fallunterscheidung machen muß, ob ein Polynom $t \in \mathbb{Z}[\underline{U}] \backslash \mathbb{Z}$ unter gewissen

Spezialisierungen verschwindet, so fügt man die atomare Formel $t \neq 0$ als zusätzliches Konjunktionsglied zu Θ hinzu. Für alle weiteren Berechnungen kann dann stets angenommen werden, daß $t \neq 0$ erfüllt ist.

Bemerkung: Man beachte, daß es zu Θ stets mindestens eine reelle Belegung der Parameter gibt, welche die Konjunktion Θ erfüllt. Mit anderen Worten, Θ ist nicht inkonsistent.
Denn zu

$$\Theta := \bigwedge_{i=1}^{k} t_i \neq 0 \quad \text{mit } t_i \in \mathbb{Z}[\underline{U}]\backslash\mathbb{Z} \quad \text{und } p := \prod_{i=1}^{k} t_k$$

wäre ansonsten jedes $\underline{u} \in \mathbb{R}^m$ eine Nullstelle des Produkts p. Also wäre p das Nullpolynom. Da $\mathbb{Z}[\underline{U}]$ ein Integritätsbereich ist, wäre also mindestens ein solches Polynom $t_i \in \mathbb{Z}[\underline{U}]\backslash\mathbb{Z}$ das Nullpolynom, ein Widerspruch.

Es werden also Annahmen über gewisse Polynome $t \in \mathbb{Z}[\underline{U}]$ gemacht, die in den Berechnungen verwendet werden dürfen. Nun wird im nächsten Abschnitt gezeigt, wie man bei der Hermiteschen Quantorenelimination solche geeignete Annahmen machen kann. Da man bei der Berechnung von Gröbnersystemen Fallunterscheidungen in Form der verschiedenen Äste machen muß, bietet es sich an, immer anzunehmen, daß gewisse Koeffizienten, deren Variablen ausschließlich aus Parametern bestehen, unter Spezialisierungen nicht verschwinden. Man beachte aber, daß nur Annahmen über Polynome, die rein aus den Variablen $\underline{U}$ bestehen, gemacht werden dürfen.

4.2 Generische Hermitesche QE

Wenn man die Hermitesche Quantorenelimination generisch auf eine Eingabeformel φ anwenden will, so berechnet man die Ergebnisformel ganz normal wie in den Algorithmen `RRCQE` und `QENF` aufgeführt, bis auf folgende zwei Unterschiede:

1. Die Bestimmung der Formel γ in Zeile 7 von `QENF` wird generisch durchgeführt

2. Die Bestimmung von Gröbnersystemen in Zeile 15 von `QENF` wird generisch durchgeführt.

Zur genaueren Erläuterung dieser beiden Punkte sei φ' die äquivalente pränexe Normalform von φ, welche man in Zeile 6 von `RRCQE` berechnet. Sei etwa φ' von der Form

$$Q_1 x_1 \dots Q_n x_n (\varphi''(x_1, \dots, x_n, U_1, \dots, U_m)) \quad \text{mit } Q_1, \dots, Q_n \in \{\exists, \forall\}$$

und mit einer quantorenfreien Formel φ'', deren Terme wieder als Elemente von $\mathbb{Z}[\underline{U}][\underline{X}]$ aufgefasst werden. In den beiden oben genannten Fällen macht man Annahmen über Polynome in den frei auftretenden Variablen $U_1, \dots, U_m$ von φ'. Man beachte, daß man auch bei rekursiven Aufrufen von `RRCQE` bei der Elimination von φ trotzdem nur Annahmen über $U_1, \dots, U_m$ machen darf, und über keine sonstigen Variablen.

Zu Punkt 1.: Wenn man bei der Elimination von φ bei einem Aufruf von `QENF` in den Fall $r = 0$ kommt, so muß eine Formel der Form

$$\exists x_{i_1} \dots \exists x_{i_q} \Big(\bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0 \Big) \quad \text{mit } g_j, h_k \in \mathbb{Z}[\underline{U}, \underline{X}]$$

und mit $1 \leq i_1 < \dots < i_q \leq n$ und $1 \leq q \leq n$ eliminiert werden. Es müssen also zu den Polynomen g_j die Ableitungen bzgl. der Variablen X_{i_q} bzw. die Differenzen der Form $g_k - g_\ell$ berechnet werden und sichergestellt werden, daß die Ableitungen bzw. Differenzen aufgefasst als univariate Polynome in X_{i_q} keine Nullpolynome sind. Dazu werden Mengen der Form Ω_f verwendet, welche Fallunterscheidungen nach den höchsten Termen von parametrisierten, univariaten Polynomen f unter Spezialisierungen bilden. f soll nun im folgenden stets eine Ableitung eines g_j bzgl. X_{i_q} bzw. eine Differenz der Form $g_k - g_\ell$ sein, aufgefasst als univariates Polynom in X_{i_q}. Die Menge Ω_f kann man verkleinern, wenn man Annahmen über Parameter zuläßt. Sei $f = \sum_{i=0}^{k} a_i X_{i_q}^i \in \mathbb{Z}[\underline{U}, X_1, \dots, X_{i_q-1}, X_{i_q+1}, \dots, X_n][X_{i_q}]$ mit $a_k \neq 0$ und

$$\Omega'_f := \big\{ \omega(f, i) \bigm| i \in \{1, \dots, k\} \text{ mit } a_i \neq 0, \text{ für } i < j \leq k \text{ ist } a_j \notin \mathbb{Z}[\underline{U}] \backslash \{0\} \big\}$$

Für $k = 0$ ist $\Omega'_f := \emptyset$. In Zeile 7 von `QENF` ersetzt man nun die Formel γ durch folgende Formel:

$$\tilde{\gamma} := \bigvee_{\ell=1}^{s} \bigvee_{\omega\left(\frac{\partial g_\ell}{\partial X_{i_q}}, i\right) \in \Omega'_{\frac{\partial g_\ell}{\partial X_{i_q}}}} \Bigg(\omega\Big(\frac{\partial g_\ell}{\partial X_{i_q}}, i\Big) \wedge \exists x_{i_q} \Big(\bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \Big(\frac{\partial g_\ell}{\partial X_{i_q}}\Big)^{(i)} (\underline{U}, \underline{x}) = 0 \Big) \Bigg)$$

$$\vee \bigvee_{1 \leq k < \ell \leq s} \bigvee_{\omega(g_k - g_\ell, i) \in \Omega'_{g_k - g_\ell}} \Big(\omega(g_k - g_\ell, i) \wedge \exists x_{i_q} \Big(\bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge (g_k - g_\ell)^{(i)} (\underline{U}, \underline{x}) = 0 \Big) \Big).$$

Da $\Omega'_f \subseteq \Omega_f$, folgt aus der Gültigkeit von $\tilde{\gamma}$ die Gültigkeit von γ aus Lemma 3.20. Man kann nun mit $\tilde{\gamma}$ als neues γ die Berechnungen in den Zeilen 8-10 von `QENF` wie üblich fortführen. Jedoch wurden jetzt evtl. Annahmen gemacht, welche man zu Θ als zusätzliche Konjunktionsglieder hinzufügen muß:

Zu jedem auftretenden nicht-leeren Ω'_f existiert ein kleinster Index i, so daß $\omega(f, i) \in \Omega'_f$. $\omega(f, i)$ besteht nach Definition aus Gleichungen und genau einer Ungleichung der Form $t \neq 0$ mit $t \in \mathbb{Z}[\underline{U}, X_1, \dots, X_{i_q-1}, X_{i_q+1}, \dots, X_n]$.

Ist $t \in \mathbb{Z}[\underline{U}]\backslash\mathbb{Z}$, so fügt man die Ungleichung $t \neq 0$ als zusätzliches Konjunktionsglied zu Θ hinzu. Für jedes auftretende Ω'_f wird dieser Schritt durchgeführt.

Zu Punkt 2.: Muß ein Gröbnersystem in Zeile 15 von `QENF` bestimmt werden, so geht man wie folgt vor:

Man beginnt zunächst die Berechnung eines grüngefärbten Gröbnersystems aus den Gleichungen einer Formel in Eliminationsnormalform ganz normal. Muß eine Fallunterscheidung gemacht werden, ob ein Polynom t unter Spezialisierungen verschwindet oder nicht, so muß untersucht werden, ob

$$t \in \mathbb{Z}[\underline{U}]\backslash\mathbb{Z} \quad \text{oder} \quad t \notin \mathbb{Z}[\underline{U}]\backslash\mathbb{Z}.$$

Gilt Ersteres, d.h. $t \in \mathbb{Z}[\underline{U}]\backslash\mathbb{Z}$, so kann man die Annahme machen, daß t unter Spezialisierungen nicht verschwindet, und nur für diesen Fall die Berechnung des Gröbnersystems fortsetzen. Dafür muß aber zu Θ die atomare Formel $t \neq 0$ als zusätzliches Konjunktionsglied hinzugefügt werden. Also entsteht bei der Berechnung des Gröbnersystems an dieser Stelle keine neue Fallunterscheidung und somit kein neuer Ast.

Gilt hingegen $t \notin \mathbb{Z}[\underline{U}]\backslash\mathbb{Z}$, so enthält t Variablen aus $\underline{X}$. Da die Variablen $\underline{X}$ im Polynomring den logischen Variablen $\underline{x}$ entsprechen, dürfen über den logischen Term $t(\underline{U}, \underline{x})$ keine Annahmen gemacht werden. Somit muß wie im nicht-generischen Fall zur Berechnung eines Gröbnersystems ganz normal eine Fallunterscheidung durchgeführt werden, ob $t = 0$ oder $t \neq 0$ gilt.

Da die einzelnen Äste eines Gröbnersystems eine vollständige, disjunkte Fallunterscheidung bilden nach den reellen Spezialisierungen der Parameter, folgt sofort, daß die „weggelassenen Äste" unter der Annahme, daß die Formeln in Θ gültig sind, äquivalent zu *false* eliminiert und ausgewertet werden würden. Somit wird offensichtlich auf diese eben beschrieben Weise die Hermitesche Quantorenelimination generisch korrekt angewandt.

An einem ersten Beispiel wird die generische Hermitesche Quantorenelimination anschaulich demonstriert:

Beispiel 4.1 *Es ist der Quantor der folgenden Formel mittels generischer Hermitescher Quantorenelimination zu eliminieren:*

$$\varphi := \exists x(ax^2 + bx + c = 0).$$

Das zur Menge $\{ax^2 + bx + c\}$ gehörige grüngefärbte Gröbnersystem hat 4 Äste mit folgenden Gröbnerbasen:

Ast 1: $\{ax^2 + bx + c\}$, *falls* $a \neq 0$.

Ast 2: $\{bx + c\}$, *falls* $a = 0 \wedge b \neq 0$.

Ast 3: $\{1\}$, *falls* $a = 0 \wedge b = 0 \wedge c \neq 0$.

Ast 4: $\{0\}$, *falls* $a = 0 \wedge b = 0 \wedge c = 0$.

Bei nicht-generischer Hermitescher Quantorenelimination müßten alle vier Äste separat eliminiert werden, während man hingegen bei Generischer nur Ast 1 eliminieren muß und die atomare Formel $a \neq 0$ zur Annahme macht. Somit ist im generischen Fall das Ergebnis:

$$\Theta \equiv a \neq 0 \quad , \quad \psi \equiv \neg(2a^2 - 2ac + b^2 = 0 \vee 4ac - b^2 > 0). \ \diamond$$

4.3 Beispiele aus der Geometrie

In diesem Abschnitt werden einige Anwendungsbeispiele der generischen Hermiteschen Quantorenelimination aus der Geometrie vorgestellt, anhand welchen gezeigt wird, daß die entstehenden Theorien durchaus anschaulich interpretierbar sind. Diese Theorien schließen prinzipiell Sonderfälle aus. Alle hier aufgeführten Beispiele wurden [14] entnommen. Auf eine Herleitung der Problemformulierungen als Formel erster Stufe wird dabei größtenteils verzichtet und dazu auf [14] verwiesen. Alle Ergebnisse konnten in 0–10 Millisekunden berechnet werden.

Beispiel 4.3.1: Schnittpunkt von Geraden im $\mathbb{R}^2$

Problemformulierung: Gilt folgende Aussage: „Zwei Geraden im $\mathbb{R}^2$ haben genau einen Schnittpunkt oder sind parallel."?

Diese Behauptung wird nun mit Hilfe generischer Hermitescher Quantorenelimination automatisch bewiesen. Die erste Gerade kann man durch eine geeignete Bewegung in $\mathbb{R}^2$ o.B.d.A. als x-Achse modellieren, die zweite als affine Funktion der Form $m \cdot x + b$. Damit kann die Behauptung folgendermaßen als Formel erster Stufe formuliert werden:

$$\varphi_1 := \exists x(mx + b = 0 \wedge \forall y(y \neq x \rightarrow my + b \neq 0)).$$

Das Eliminationsergebnis der generischen Hermiteschen Quantorenelimination angewandt auf φ_1 ist $\psi \equiv \mathit{true}$ unter der Bedingung $\Theta \equiv m \neq 0$. Die Theorie Θ schließt also den Sonderfall aus, daß die beiden Geraden parallel sind. Also gilt die eingangs behauptete Aussage. $\diamond$

Beispiel 4.3.2: Schnittpunkt der Diagonalen eines Parallelogramms

Problemformulierung: Sei ABCD ein Parallelogramm und sei E der Schnittpunkt von dessen Diagonalen. Ist E der Mittelpunkt von beiden Diagonalen? Siehe Skizze 4.1.

Mit Hilfe generischer Quantorenelimination soll nun überprüft werden, wie ein Parallelogramm ABCD beschaffen sein muß, so daß E der Mittelpunkt beider Diagonalen ist. Durch eine geeignete Bewegung im $\mathbb{R}^2$ kann man o.B.d.A. die Koordinaten für die gegebenen Punkte wie folgt wählen:

$$A = (0, 0), \quad B = (u_1, 0), \quad C = (u_2, u_3), \quad D = (x_2, x_1), \quad E = (x_4, x_3).$$

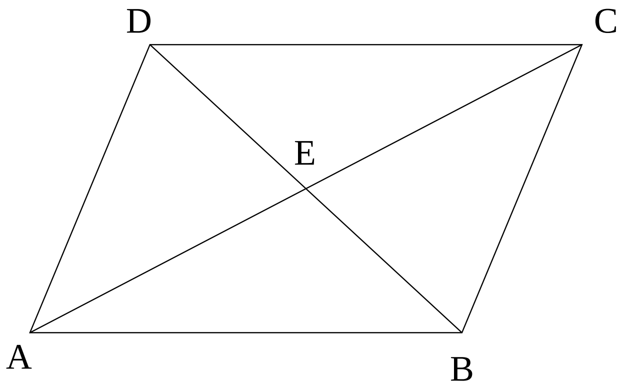

Abbildung 4.1: Schnittpunkt der Diagonalen eines Parallelogramms(Beispiel 4.3.2)

Dabei sind die Variablen u_1, u_2 und u_3 frei wählbar und bestimmen eindeutig das Parallelogramm; die übrigen Variablen sind dadurch automatisch festgelegt. Mit Hilfe folgender Gleichungen lassen sich sich nun einige geometrische Eigenschaften zu diesem Problem so formulieren:

$h_1 \equiv u_1x_1 - u_1u_3 = 0$	AB $\parallel$ DC
$h_2 \equiv u_3x_2 - (u_2 - u_1)x_1 = 0$	DA $\parallel$ CB
$h_3 \equiv x_1x_4 - (x_2 - u_1)x_3 - u_1x_1 = 0$	E $\in$ BD
$h_4 \equiv u_3x_4 - u_2x_3 = 0$	E $\in$ AC
$g \equiv 2u_2x_4 + 2u_3x_3 - u_3^2 - u_2^2 = 0$	Länge(AE) = Länge(CE)

Damit läßt sich dieses Problem nach [14], Example 3.8, wie folgt als Formel erster Stufe formulieren:

$$\varphi_2 \quad := \quad \forall x_1 \forall x_2 \forall x_3 \forall x_4 (h_1 \wedge h_2 \wedge h_3 \wedge h_4 \rightarrow g).$$

Das Eliminationsergebnis ist:

$$\Theta \equiv u_1 \neq 0 \wedge u_3 \neq 0 \quad , \quad \psi \equiv u_1 \neq 0 \vee u_3 = 0.$$

ψ ist offensichtlich unter der Theorie Θ äquivalent zu *true*. Die Theorie Θ macht die Voraussetzung, daß es sich bei ABCD um ein echtes Parallelogramm handelt, also keine gegenüberliegenden Seiten zusammenfallen. Es wurde somit gezeigt, daß E für echte Parallelogramme stets der Mittelpunkt beider Diagonalen ist. $\diamond$

Beispiel 4.3.3: Quadrate an den Seiten eines Dreiecks

Problemformulierung: Sei ABC ein Dreieck. Seien an den beiden Seiten AC und BC des Dreiecks zwei Quadrate ACDE und BCFG anliegend, wobei

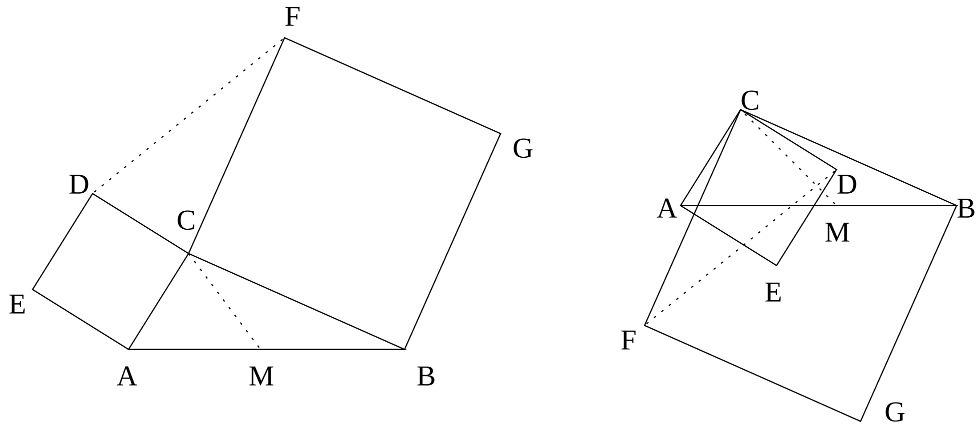

Abbildung 4.2: Zwei zulässige Möglichkeiten zum Anlegen von Quadraten an den Seiten eines Dreiecks(Beispiel 4.3.3)

entweder beide außerhalb des Dreiecks liegen oder beide das Dreieck schneiden. Ferner sei M der Mittelpunkt der Strecke AB. Ist die Länge der Strecke DF genau doppelt so groß wie die Länge der Strecke CM? Siehe Skizze 4.2.

Wiederum wird überprüft, wie ein Dreieck beschaffen sein muß, so daß die gewünschten Längenbeziehungen gelten. O.B.d.A. können die Koordinaten der Punkte wie folgt notiert werden:

$$\begin{array}{lll} A = (u_1, 0), & B = (u_2, u_3), & C = (0, 0), \\ D = (0, u_1), & F = (x_1, x_2), & M = (x_3, x_4). \end{array}$$

Durch die Punkte u_1, u_2 und u_3 ist das Dreieck ABC eindeutig bestimmt und die Koordinaten der anderen Punkte ergeben sich daraus automatisch. Damit läßt sich dieses Problem nach [14], Example 3.13, wie folgt als Formel erster Stufe formulieren, wobei auf eine Erläuterung der enthaltenen atomaren Formeln ab jetzt verzichtet wird:

$$\begin{aligned} \varphi_3 \quad := \quad & \forall x_1 \forall x_2 \forall x_3 \forall x_4 (x_2^2 + x_1^2 - u_3^2 - u_2^2 = 0 \wedge u_3 x_2 + u_2 x_1 = 0 \wedge \\ & 2x_3 - u_2 - u_1 = 0 \wedge 2x_4 - u_3 = 0 \wedge u_2 x_2 - u_3 x_1 < 0 \\ & \rightarrow 4x_4^2 + 4x_3^2 - x_2^2 + 2u_1 x_2 - x_1^2 - u_1^2 = 0). \end{aligned}$$

Das Ergebnis der generischen Hermiteschen Quantorenelimination angewandt auf φ_3 ist:

$$\Theta \equiv u_3 \neq 0 \quad , \quad \psi \equiv true.$$

Die Theorie Θ ist eine notwendige, aber nicht hinreichende Bedingung, daß es sich bei ABC um ein echtes Dreieck handelt. $u_3 \neq 0$ sichert, daß nicht alle 3 Eckpunkte des Dreiecks auf der x-Achse liegen. Die in der Problemformulierung gestellte Frage ist also zu bejahen, falls $u_3 \neq 0$. $\diamond$

Beispiel 4.3.4: Gergonne's Theorem

Problemformulierung: Man betrachte ein Dreieck ABC und einen beweglichen Punkt D. Man zeichne von D die Lote auf alle drei Seiten des Dreiecks mit Fußpunkten E, F und G. Was ist der geometrische Ort des Punktes D, so daß die Fläche des Dreiecks EFG konstant bleibt? Siehe Skizze 4.3.

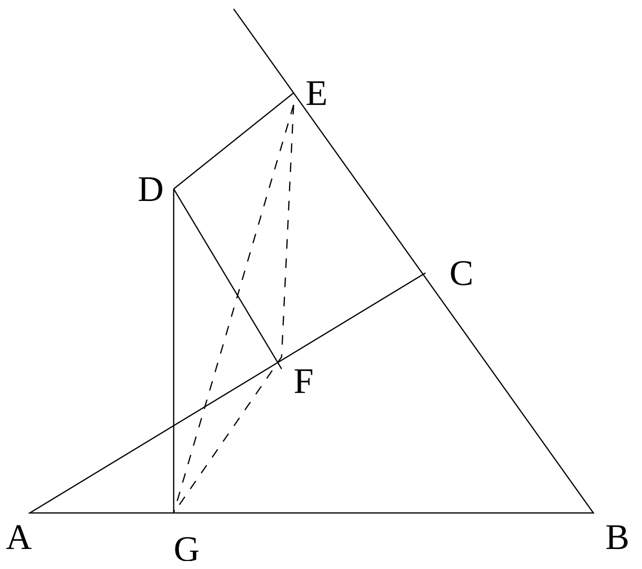

Abbildung 4.3: Gergonne's Theorem(Beispiel 4.3.4)

Sei die Fläche von EFG o.B.d.A. mit $a/2$ vorgegeben. O.B.d.A. können die Koordinaten der Punkte wie folgt notiert werden:

$$\begin{aligned} &A = (0,0), \quad B = (u_1, 0), \quad C = (u_2, u_3), \quad D = (x, y), \\ &E = (x_2, x_1), \quad F = (x_4, x_3), \quad G = (x, 0). \end{aligned}$$

Wiederum ist das Dreieck ABC durch u_1, u_2 und u_3 eindeutig bestimmt. Ferner sind x, y und a frei auftretende Variablen. Die Koordinaten der übrigen Punkte sind durch die Werte jener festgelegt. Damit läßt sich dieses Problem nach [14], Example 3.15, wie folgt als Formel erster Stufe formulieren:

$$\begin{aligned} \varphi_4 \quad := \quad & \forall x_1 \forall x_2 \forall x_3 \forall x_4 \big(u_3 x_2 + (-u_2 + u_1) x_1 - u_1 u_3 = 0 \wedge \\ & (u_2 - u_1) x_2 + u_3 x_1 - u_3 y + (-u_2 + u_1) x = 0 \wedge \\ & u_3 x_4 - u_2 x_3 = 0 \wedge u_2 x_4 + u_3 x_3 - u_3 y - u_2 x = 0) \\ & \rightarrow x_1 x_4 + (-x_2 + x) x_3 - x x_1 + a = 0 \big). \end{aligned}$$

Das Ergebnis der generischen Hermiteschen Quantorenelimination angewandt auf φ_4 ist:

$$\begin{aligned}\Theta &\equiv u_1^2 - 2u_1u_2 + u_2^2 + u_3^2 \neq 0 \wedge u_1 - u_2 \neq 0 \wedge u_2 \neq 0 \wedge u_3 \neq 0 \ , \\ \psi &\equiv au_1^2u_2^2 + au_1^2u_3^2 - 2au_1u_2^3 - 2au_1u_2u_3^2 + au_2^4 + 2au_2^2u_3^2 + au_3^4 + \\ &\quad u_1^2u_2u_3^2y - u_1^2u_3^3x - u_1u_2^2u_3^2y - u_1u_3^4y + u_1u_3^3x^2 + u_1u_3^3y^2 = 0 \ .\end{aligned}$$

Die erste Ungleichung von Θ ist äquivalent zu $(u_1 - u_2)^2 + u_3^2 \neq 0$ und wird offensichtlich von $u_3 \neq 0$ impliziert. Die Bedingungen $u_2 \neq 0$ bzw. $u_1 \neq u_2$ besagen, daß die Winkel $\angle$ BAC bzw. $\angle$ CBA keine rechten Winkel sind. Die Bedingung $u_3 \neq 0$ sichert, daß nicht alle Eckpunkte des Dreiecks auf der x-Achse liegen. $u_3 \neq 0$ ist wiederum eine notwendige, aber nicht hinreichende Bedingung, daß das Dreieck ABC echt ist, d.h. nicht degeneriert ist. Das Eliminationsergebnis ψ liefert eine Beschreibung des geometrischen Ortes unter der Voraussetzung der Gültigkeit von Θ. $\diamond$

Beispiel 4.3.5: Winkel und Umkreis

Problemformulierung: Sei O der Mittelpunkt des Umkreises eines Dreiecks ABC und liege O nicht außerhalb des Dreiecks. Gilt dann die Winkelbeziehung $\angle$ ACB = $\angle$ AOB/2? Siehe Skizze 4.4.

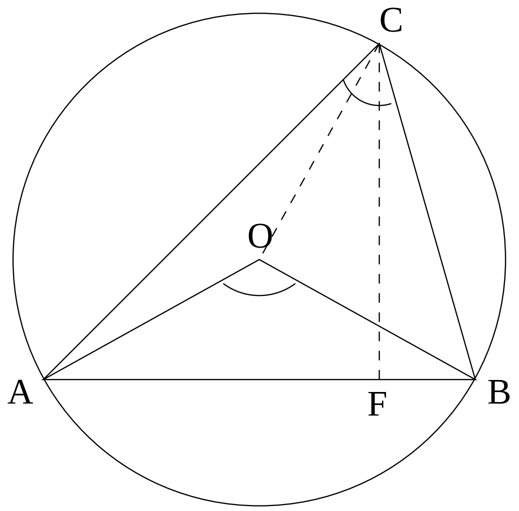

Abbildung 4.4: Winkel am Umkreismittelpunkt/Innenwinkel(Beispiel 4.3.5)

O.B.d.A. können die Koordinaten der Punkte wie folgt notiert werden:

$$A = (-u_1, 0), \quad B = (u_1, 0), \quad C = (u_2, u_3), \quad O = (0, x_1), \quad F = (0, u_3).$$

Wiederum legen u_1, u_2 und u_3 eindeutig das Dreieck ABC fest; die übrigen Koordinatenwerte resultieren daraus direkt. Sei r der Radius des Umkreises. Die Variablen t_1, t_2, t und t' stehen für gewisse Tangenswerte, auf deren genaue Interpretation hier verzichtet wird. Damit läßt sich dieses Problem nach [14], Example 3.16, wie folgt als Formel erster Stufe formulieren:

$$\begin{aligned}\varphi_5 \quad := \quad & \forall r \forall x_1 \forall t_1 \forall t_2 \forall t \forall t' \big(r^2 = u_1^2 + x_1^2 \wedge r^2 = u_2^2 + (u_3 - x_1)^2 \wedge \\ & \quad u_3 t_1 = u_1 + u_2 \wedge u_3 t_2 = u_1 - u_2 \wedge (1 - t_1 t_2) t = t_1 + t_2 \wedge \\ & \qquad x_1 t' = u_1 \rightarrow t = t' \big).\end{aligned}$$

Das Ergebnis der generischen Hermiteschen Quantorenelimination angewandt auf φ_5 ist:

$$\Theta \equiv u_1^2 - u_2^2 - u_3^2 \neq 0 \wedge u_3 \neq 0 \quad , \quad \psi \equiv \mathit{true}.$$

Die Bedingung $u_3 \neq 0$ sichert, daß die Eckpunkte des Dreiecks ABC nicht alle auf der x-Achse liegen, was eine notwendige, aber nicht hinreichende Bedingung dafür ist, daß das Dreieck ABC echt ist. Die erste Bedingung $u_1^2 \neq u_2^2 + u_3^2$ von Θ besagt, daß der Umkreismittelpunkt O nicht auf der Strecke AB liegen darf. Liegt also O nicht auf der Strecke AB und ist $u_3 \neq 0$, so gilt $\angle$ ACB = $\angle$ AOB/2. $\diamond$

Kapitel 5

Rund um Typformeln

In diesem Kapitel werden einige Themen rund um Typformeln behandelt. Zunächst wird eine vom Autor der vorliegenden Arbeit entworfene obere und untere Abschätzung für die Größenordnung von strikten Typformeln hergeleitet. Darauf aufbauend wird dann gezeigt, wie man im Algorithmus **QENF** einige Verbesserungen vornehmen kann, die ebenfalls vom Autor entworfen wurden. Dazu gehört zum einen die Wahl einer geeigneten gebundenen Variablen x_i, falls man beim Aufruf von **QENF** im Fall ist, daß keine nichttrivialen Gleichungen in einer Eingabeformel vorhanden sind. Zum anderen wird gezeigt, wie man Typformeln vom Typ 0 zurückführen kann auf Typformeln von anderen Typen. Beide Verbesserungen bedienen sich der zunächst gemachten Abschätzungen über die Größe der strikten Typformeln vom Typ 0.

5.1 Komplexität von strikten Typformeln

Zur Bestimmung von strikten Typformeln für Polynome vom Grad d mit Typ 0 bestimmt man zunächst die sog. *Menge der guten Tupel.* Dazu benötigt man einige vorgreifende Definitionen und Sätze.

Definition 5.1 (Vorzeichenwechsel) *Sei* $\underline{c} := (c_0, \ldots, c_d) \in \mathbb{R}^{d+1}$. *Sei*

$$M(\underline{c}) := \big\{(i,j) \in \mathbb{N}^2 \mid 0 \leq i < j \leq d, a_i \cdot a_j < 0, a_k = 0 \text{ für } i < k < j\big\}.$$

Dann ist der Vorzeichenwechsel von $\underline{c}$ *definiert als* $Vzw(\underline{c}) := |M(\underline{c})|$.

Es werden nun einige abkürzende Notationen eingeführt. Sei dazu ein Tupel $\underline{c} := (c_0, \ldots, c_d) \in \mathbb{R}^{d+1}$ gegeben. Dann definiert man

$$\begin{aligned}
\overline{\underline{c}} &:= (c_0, (-1) \cdot c_1, c_2, \ldots, (-1)^i \cdot c_i, \ldots, (-1)^d \cdot c_d), \\
\pi(\underline{c}) &:= Vzw(\underline{c}), \\
\nu(\underline{c}) &:= Vzw(\overline{\underline{c}}).
\end{aligned}$$

Das Hauptinstrument hier, um die Anzahl der positiven bzw. negativen Nullstellen eines Polynoms zu bestimmen, ist der folgende Satz von Descartes:

Satz 5.2 (Descartes) *Sei $f = \sum_{i=0}^{d} c_i X^i \in \mathbb{R}[X]$ ein Polynom mit ausschließlich reellen Nullstellen. Sei $\underline{c} := (c_0, \ldots, c_d)$. Dann ist die Anzahl der positiven Nullstellen von f gleich $Vzw(\underline{c})$ und die Anzahl der negativen Nullstellen von f gleich $Vzw(\overline{\underline{c}})$.*

Beweis: Siehe z.B. [12]. □

Bemerkung: Für ein Polynom $f = \sum_{i=0}^{d} c_i X^i \in \mathbb{R}[X]$ mit ausschließlich reellen Nullstellen ist der Typ eines Polynoms dann

$$\tau(f) = \pi(c_0, \ldots, c_d) - \nu(c_0, \ldots, c_d).$$

Es werden im weiteren Verlauf nun Tupel $(c_0, \ldots, c_d) \in \{-1, 0, 1\}^{d+1}$ betrachtet werden, welche bestimmten Bedingungen genügen. Dabei wird eine auftretende „0" in einem Tupel eine besondere Bedeutung zukommen, was später noch genauer erläutert wird. Man kann jetzt die Menge der guten Tupel für Polynome vom Grad d mit Typ 0 wie folgt definieren:

Definition 5.3 (Gute Tupel) *Sei $2 \leq d \in \mathbb{N}$, d gerade.*
$(c_0, \ldots, c_d) \in \{-1, 0, 1\}^{d+1}$ ist ein gutes Tupel (für Polynome) vom Grad d (mit Typ 0), falls es folgende Eigenschaften erfüllt:

1. $c_d = 1$
2. $c_0 = \begin{cases} 1 & \text{falls } d \bmod 4 = 0 \\ -1 & \text{falls } d \bmod 4 = 2 \end{cases}$
3. $\pi((c_0, \ldots, c_d)) = \frac{d}{2}$
4. $\nu((c_0, \ldots, c_d)) = \frac{d}{2}$
5. *Für alle $i \in \{1, \ldots, d-1\}$ gilt:* $c_i = 0 \Longleftrightarrow c_{i-1} \cdot c_{i+1} = -1$

Die Menge

$$M_d^{(0)} := \left\{(c_0, \ldots, c_d) \mid (c_0, \ldots, c_d) \text{ ist gutes Tupel}\right\} \subseteq \left\{-1, 0, 1\right\}^{d+1}$$

heißt die Menge der guten Tupel für Polynome vom Grad d mit Typ 0.

Diese Menge kann algorithmisch bestimmt werden. In der Diplomarbeit von Andreas Dolzmann [4], Abbildung 5, findet sich ein entsprechender Algorithmus. Es wird nun kurz anhand eines Beispiels gezeigt, wie man aus $M_d^{(0)}$

eine strikte Typformel für Polynome vom Grad d mit Typ 0 konstruieren kann. Dabei werden hierzu keine Beweise geführt. Alle zugehörigen Beweise finden sich ebenfalls in Kapitel 3 von [4]. Von einem guten Tupel kommt man folgendermaßen zu einer Konjunktion atomarer Formeln:

Sei $X^d + \sum_{i=0}^{d-1} C_i X^i \in \mathbb{R}[C_0, \ldots, C_{d-1}][X]$ ein parametrisiertes Polynom, $(c_0, \ldots, c_d) \in M_d^{(0)}$. Für $i \in \{0, \ldots, d-1\}$ definiere man

$$\xi_i := \begin{cases} C_i > 0 & \text{falls } c_i = 1 \\ C_i < 0 & \text{falls } c_i = -1 \\ \mathit{true} & \text{falls } c_i = 0 \end{cases}$$

Daraus kann man die Konjunktion $\bigwedge_{i=0}^{d-1} \xi_i$ bilden. Eine Erweiterung dazu ist $(\bigwedge_{i=0}^{d-1} \xi_i)(C_0, \ldots, C_{d-1})$. Jedes gute Tupel stellt also eine Bedingung an die Vorzeichen der Koeffizienten eines Polynoms. Die „0" in einem guten Tupel bedeutet also, daß das Vorzeichen des zugehörigen Koeffizienten keine Bedeutung spielt.

Wenn nun $f = X^d + \sum_{i=0}^{d-1} a_i X^i \in \mathbb{R}[\underline{U}][X]$ ein parametrisiertes Polynom ist, $\underline{u}$ eine Spezialisierung ist, so daß $f(\underline{u}, X)$ ausschließlich reelle Nullstellen besitzt und $\underline{u}$ die Formel $(\bigwedge_{i=0}^{d-1} \xi_i)(a_0(\underline{u}), \ldots, a_{d-1}(\underline{u}))$ erfüllt, so hat $f(\underline{u}, X)$ den Typ 0 und es ist $a_0(\underline{u}) \neq 0$.

Bildet man nun für alle guten Tupel in $M_d^{(0)}$ diese Konjunktionen $\bigwedge_{i=0}^{d-1} \xi_i$ und verknüpft diese disjunktiv miteinander, so erhält man eine strikte Typformel $T_d^{(0)'}(C_0, \ldots, C_{d-1})$ für Polynome vom Grad d mit Typ 0.

Beispiel 5.4 *Für $d = 4$ ist die Menge der guten Tupel*

$$M_4^{(0)} = \big\{(1,0,-1,0,1), (1,1,0,-1,1), (1,-1,0,1,1)\big\}.$$

Sei $T_4^{(0)'}(C_0, C_1, C_2, C_3)$ eine strikte Typformel für Polynome vom Grad 4. Dann gilt:

$$\begin{aligned} T_4^{(0)'}(C_0, C_1, C_2, C_3) \quad \equiv \quad & (C_2 < 0 \wedge C_0 > 0) \vee (C_3 < 0 \wedge C_1 > 0 \wedge C_0 > 0) \vee \\ & (C_3 > 0 \wedge C_1 < 0 \wedge C_0 > 0). \end{aligned}$$

◇

Die Größe von strikten Typformeln hängt somit direkt von der Größe der Menge der guten Tupel $M_d^{(0)}$ ab. Es ist intuitiv klar, daß mit steigendem Grad d deren Kardinalität mitansteigt. In [4], Lemma 3.45, wurde eine obere Schranke für die Kardinalität von $M_d^{(0)}$ angegeben:

$$\left| M_d^{(0)} \right| \leq 2^{d-1}.$$

Diese obere Schranke wird nun verbessert. Dazu wird folgende Menge betrachtet:

$$\begin{aligned} W_d \;:=\; \Big\{(w_1,\ldots,w_d)\in\{+,-\}^d \;\Big|\; & |\{i\;|w_i=+,i\in\{1,\ldots,d\}\}| \;= \\ & = \;|\{i\;|w_i=-,i\in\{1,\ldots,d\}\}| \;=d/2\Big\}. \end{aligned}$$

Es wird nun jedes Element von $M_d^{(0)}$ eindeutig mit einem Element aus W_d identifiziert. Dies geschieht mit folgender Abbildung Υ_d:

$$\Upsilon_d : M_d^{(0)} \to W_d : (c_0,\ldots,c_d) \mapsto (w_1,\ldots,w_d),$$

wobei

$$w_i := \begin{cases} + & \text{falls } (c_{i-1},c_i)\in\{(1,1),(1,0),(-1,-1),(-1,0)\} \\ - & \text{falls } (c_{i-1},c_i)\in\{(1,-1),(0,-1),(0,1),(-1,1)\} \end{cases}.$$

Die Idee von Υ_d ist, zwei aufeinanderfolgenden Elementen eines guten Tupels, etwa c_{i-1} und c_i, genau dann ein Minuszeichen zuzuweisen, wenn ein Vorzeichenwechsel stattfindet. Dabei interpretiert man den Übergang zu 0, d.h. im Fall $c_i = 0$, als keinen Vorzeichenwechsel und den Übergang von einer 0 zu einem beliebigen Element, d.h. im Fall $c_{i-1} = 0$, als Vorzeichenwechsel. Man beachte, daß in jedem guten Tupel vom Grad d genau $\frac{d}{2}$ Vorzeichenwechsel stattfinden.

Es muß nun gezeigt werden, daß tatsächlich jedes Minuszeichen im Υ_d-Bild eines guten Tupels genau einem Vorzeichenwechsel des guten Tupels entspricht und umgekehrt, so daß damit Υ_d in der Tat in die Menge W_d abbildet. Dieses Entsprechen von Minuszeichen und Vorzeichenwechseln liegt hier vor, da:

Falls $c_{i-1} \neq 0$ und $c_{i-1} \cdot c_i \neq -1$, findet kein Vorzeichenwechsel statt. Nach Konstruktion ist in diesem Fall stets $w_i = +$.
Im Fall $c_{i-1} \cdot c_i = -1$, findet zwischen c_{i-1} und c_i ein Vorzeichenwechsel statt. Nach Konstruktion ist $w_i = -$.
Ist hingegen $c_{i-1} = 0$, so ist nach Definition eines guten Tupels $c_{i-2} \cdot c_i = -1$, und somit tritt in der Folge (c_{i-2}, c_{i-1}, c_i) genau ein Vorzeichenwechsel auf. Nach Definition von Υ_d ist $w_{i-1} = +$ und $w_i = -$.

Also entspricht jeder Vorzeichenwechsel in $(c_0,\ldots,c_d)$ genau einem Minuszeichen in $(w_1,\ldots,w_d)$ und umgekehrt. Da in $(c_0,\ldots,c_d) \in M_d^{(0)}$ genau $\frac{d}{2}$ Vorzeichenwechsel stattfinden, treten damit in $(w_1,\ldots,w_d)$ genau $\frac{d}{2}$ Minuszeichen auf. Damit enthält jedes Bildelement von Υ_d genau $\frac{d}{2}$ Pluszeichen und genau $\frac{d}{2}$ Minuszeichen.

So ist z.B.

$$\begin{aligned} \Upsilon_4\big((1,-1,0,1,1)\big) &= (-,+,-,+) \qquad \text{und} \\ \Upsilon_4\big((1,0,-1,0,1)\big) &= (+,-,+,-). \end{aligned}$$

Nun ist die Injektivität von Υ_d zu zeigen, so daß gewährleistet ist, daß die Elemente von $M_d^{(0)}$ mit Elementen von W_d identifiziert werden können.

Lemma 5.5 *Die Abbildung Υ_d ist injektiv.*

Beweis: Seien $\underline{c} := (c_0, \ldots, c_d), \underline{c'} := (c'_0, \ldots, c'_d) \in M_d^{(0)}$ mit $\Upsilon_d(\underline{c}) = \Upsilon_d(\underline{c'}) = (\Upsilon_d(\underline{c})_1, \ldots, \Upsilon_d(\underline{c})_d)$. Angenommen $\underline{c} \neq \underline{c'}$. Da $c_0 = c'_0$ und $c_d = c'_d = 1$ nach Konstruktion der Menge der guten Tupel gilt, existiert also ein kleinstes $i \in \{1, \ldots, d-1\}$ mit $c_i \neq c'_i$. Es wird eine Fallunterscheidung nach c_{i-1} und $\Upsilon_d(\underline{c})_i$ gemacht:

$\mathbf{c_{i-1} = 1}$ **und** $\mathbf{\Upsilon_d(\underline{c})_i = +}$**:** Dann sind $c_i, c'_i \in \{0, 1\}$ und $c_i \neq c'_i$. Seien o.B.d.A. $c_i = 0$ und $c'_i = 1$. Dann ist nach Definition von $M_d^{(0)}$

$$c_{i+1} = -1, \quad \text{also } \Upsilon_d(\underline{c})_{i+1} = -.$$

Wegen $\Upsilon_d(\underline{c}) = \Upsilon_d(\underline{c'})$ muß also $c'_{i+1} = -1$ gelten. Dies ist ein Widerspruch zur Eigenschaft 5. in der Definition eines guten Tupels, da $\underline{c'} \in M_d^{(0)}$, aber $c'_{i-1} \cdot c'_{i+1} = -1$ und $c'_i \neq 0$.

$\mathbf{c_{i-1} = 1}$ **und** $\mathbf{\Upsilon_d(\underline{c})_i = -}$**:** Also ist $c_i = -1$. Wegen $\Upsilon_d(\underline{c}) = \Upsilon_d(\underline{c'})$ und $c'_{i-1} = c_{i-1} = 1$ ist $c'_i = -1 = c_i$, ein Widerspruch zur Wahl von i.

$\mathbf{c_{i-1} = 0}$**:** c_i und c'_i sind durch $c_{i-2} = c'_{i-2}$ festgelegt aufgrund der Definition eines guten Tupels. Also gilt $c_i = c'_i$, ein Widerspruch zur Wahl von i.

$\mathbf{c_{i-1} = -1}$ **und** $\mathbf{\Upsilon_d(\underline{c})_i = +}$**:** Analog zum Fall „$c_{i-1} = 1$ und $\Upsilon_d(\underline{c})_i = +$".

$\mathbf{c_{i-1} = -1}$ **und** $\mathbf{\Upsilon_d(\underline{c})_i = -}$**:** Analog zum Fall „$c_{i-1} = 1$ und $\Upsilon_d(\underline{c})_i = -$".

Also ist $\underline{c} = \underline{c'}$ und damit Υ_d injektiv. □

Somit kann nun folgende obere Schranke für $M_d^{(0)}$ bewiesen werden:

Lemma 5.6 *Sei $2 \leq d \in \mathbb{N}$ gerade. Dann gilt*

$$O_d := \frac{2^d}{\sqrt{\pi \cdot \frac{d}{2}}} \geq \left| M_d^{(0)} \right| .$$

Beweis: Da Υ_d eine injektive Abbildung von $M_d^{(0)}$ nach W_d ist, gilt somit

$$\left| M_d^{(0)} \right| \leq \left| W_d \right| = \binom{d}{\frac{d}{2}} .$$

Dieser Binomialkoeffizient wird mit Hilfe der Stirling-Formel(siehe [11], Bemerkung 1.2.6) nun nach oben abgeschätzt. Für $n \in \mathbb{N}$ liefert die Stirling-Formel folgende Abschätzung für $n!$:

$$\sqrt{2\pi n} \cdot n^n \cdot e^{-n+\frac{1}{12n+1}} \leq n! \leq \sqrt{2\pi n} \cdot n^n \cdot e^{-n+\frac{1}{12n}}.$$

Weiters gilt folgende Äquivalenz:

$$\frac{1}{12d} - \frac{2}{6d+1} \leq 0 \Leftrightarrow 1 \leq \frac{24d}{6d+1} \Leftrightarrow 6d+1 \leq 24d \Leftrightarrow 1 \leq 18d \Leftrightarrow 1 \leq d$$

Hieraus folgt nun mit der Stirlingformel die Abschätzung

$$\begin{aligned}
\binom{d}{\frac{d}{2}} &= \frac{d!}{\frac{d}{2}! \cdot \frac{d}{2}!} \\
&\leq \frac{\sqrt{2\pi d} \cdot d^d \cdot e^{-d+\frac{1}{12d}}}{\sqrt{\pi d} \cdot (\frac{d}{2})^{\frac{d}{2}} \cdot e^{-\frac{d}{2}+\frac{1}{6d+1}} \cdot \sqrt{\pi d} \cdot (\frac{d}{2})^{\frac{d}{2}} \cdot e^{-\frac{d}{2}+\frac{1}{6d+1}}} \\
&= \frac{d^d \cdot 2^d}{\sqrt{\pi \frac{d}{2}} \cdot d^d} \cdot \underbrace{e^{\frac{1}{12d}-\frac{2}{6d+1}}}_{\leq 1} \\
&\leq \frac{2^d}{\sqrt{\pi \frac{d}{2}}} \quad .
\end{aligned}$$

Daraus folgt die Behauptung des Lemmas. □

Es ist offensichtlich, daß die im letzten Lemma erhaltene obere Schranke O_d besser ist als die eingangs erwähnte Schranke 2^{d-1}, da im Nenner von O_d der Grad d auftritt, und somit mit steigendem Grad d der Nenner umso größer wird.

Nun wird auch eine untere Schranke für $M_d^{(0)}$ angegeben:

Lemma 5.7 *Für $12 \leq d \in \mathbb{N}$, d gerade, gilt:*

$$U_d := 2^{\frac{2}{3}d} \leq \left| M_d^{(0)} \right|,$$

d.h. U_d ist eine untere Abschätzung für die Kardinalität von $M_d^{(0)}$.

Beweis: durch vollständige Induktion nach d mit Rückgriff auf alle Vorgänger. Für $d \in \{12, 14, 16, 18, 20, 22\}$ gilt die Aussage nach Tabelle 5.1. Für gerade $d \in \mathbb{N}$ mit $d < 12$ gilt die Aussage nicht.

Der Induktionsschritt wird hier von d nach $d+2$ geführt. Sei $22 \leq d \in \mathbb{N}$, d gerade, und gelte die Abschätzung für alle geraden $d' \in \mathbb{N}$ mit $12 \leq d' \leq d$. Dann ist $d+2-12 = d-10 \geq 12$, d.h. für $d-10$ gilt nach Induktionsannahme die Behauptung auch. Es wird jetzt gezeigt, wie man aus einem guten Tupel vom Grad 12 und einem guten Tupel vom Grad $d-10$ ein gutes Tupel vom Grad $d+2$ konstruieren kann:

Sei $(c_0,\ldots,c_{12}) \in M_{12}^{(0)}$ und $(c'_0,\ldots,c'_{d-10}) \in M_{d-10}^{(0)}$. Nach Definition von guten Tupeln gilt somit $c_{12}=c_0=c'_{d-10}=1$. Es wird behauptet, daß

$$\underline{c} := (c'_0,\ldots,c'_{d-9},c,c_1,\ldots,c_{12})$$

ein gutes Tupel vom Grad $d+2$ ist, wobei

$$c = \begin{cases} 0 & \text{falls } c'_{d-9}\cdot c_1 = -1 \\ 1 & \text{sonst} \end{cases}.$$

Offensichtlich hat $\underline{c}$ die richtige Länge und das letzte Element von $\underline{c}$ ist $c_{12}=1$. Man beachte, daß im Fall $c=0$ wegen $c'_{d-9}\cdot c_1 = -1$ gilt:

$$\begin{aligned}&Vzw(c'_{d-9},c,c_1) = 1 = Vzw(c'_{d-9},1,c_1) \quad \text{und}\\ &Vzw((-1)^{d-9}\cdot c'_{d-9},c,(-1)\cdot c_1) = 1 = Vzw((-1)^{d-9}\cdot c'_{d-9},1,(-1)\cdot c_1).\end{aligned}$$

Damit kann man in diesem Fall zum Zählen der Vorzeichenwechsel von $\underline{c}$ den Eintrag c durch 1 ersetzen. Für die Vorzeichenwechsel gilt damit:

$$\begin{aligned}\pi(\underline{c}) &= \begin{cases} \pi(c'_0,\ldots,c'_{d-9},c'_{d-10}) + \pi(c_0,c_1,\ldots,c_{12}) & \text{falls } c=1 \\ \pi(c'_0,\ldots,c'_{d-9},1) + \pi(1,c_1,\ldots,c_{12}) & \text{falls } c=0 \end{cases}\\ &= \begin{cases} \frac{d-10}{2}+\frac{12}{2} = \frac{d+2}{2} & \text{falls } c=1 \\ \pi(c'_0,\ldots,c'_{d-9},c'_{d-10}) + \pi(c_0,c_1,\ldots,c_{12}) = \frac{d+2}{2} & \text{falls } c=0 \end{cases}\end{aligned}$$

und

$$\begin{aligned}\nu(\overline{\underline{c}}) &= \begin{cases} Vzw(c'_0,\ldots,(-1)^{d-9}c'_{d-9},c'_{d-10}) + Vzw(c_0,(-1)\cdot c_1,\ldots,c_{12}) & \text{f. } c=1 \\ Vzw(c'_0,\ldots,(-1)^{d-9}c'_{d-9},1) + Vzw(1,(-1)\cdot c_1,\ldots,c_{12}) & \text{f. } c=0 \end{cases}\\ &= \begin{cases} \frac{d-10}{2}+\frac{12}{2} = \frac{d+2}{2} & \text{falls } c=1 \\ \frac{d-10}{2}+\frac{12}{2} = \frac{d+2}{2} & \text{falls } c=0 \end{cases}.\end{aligned}$$

Ferner ist $(d+2) \bmod 4 = (d-10) \bmod 4$, d.h. das erste Element von $\underline{c}$ ist mit c'_0 richtig besetzt. Offensichtlich erfüllt $\underline{c}$ auch Eigenschaft 5. in der Definition eines guten Tupels. Somit ist $\underline{c}$ ein gutes Tupel vom Grad $d+2$. Zu jedem Element aus $M_{12}^{(0)}$ und jedem Element aus $M_{d-10}^{(0)}$ kann man also ein gutes Tupel vom Grad $d+2$ konstruieren, so daß offensichtlich alle diese entstehenden Tupel allesamt verschieden voneinander sind. Nach Induktionsanfang und mit der Induktionsannahme gilt

$$\left|M_{12}^{(0)}\right| \geq 2^{\frac{2}{3}12} = 2^8 \quad \text{und} \quad \left|M_{d-10}^{(0)}\right| \geq 2^{\frac{2}{3}(d-10)}.$$

Es ergibt sich nun folgende Abschätzung:

$$\left|M_{d+2}^{(0)}\right| \geq \left|M_{d-10}^{(0)}\right| \cdot \left|M_{12}^{(0)}\right| \overset{I.-A.}{>} 2^{\frac{2}{3}(d-10)} \cdot 2^{8} = 2^{\frac{2d-20+24}{3}} = 2^{\frac{2}{3}(d+2)}.$$

Somit gilt die im Lemma behauptete untere Schranke. □

Tabelle 5.1 vergleicht die Anzahl der guten Tupel für gerades $d \in \mathbb{N}$, $d \leq 24$, mit den beiden erhaltenen Schranken O_d und U_d, wobei die berechneten Schrankenwerte gerundet wurden.

d	U_d	$\lvert M_d^{(0)}\rvert$	O_d	2^{d-1}
2	2,52	1	2,26	2
4	6,35	3	6,38	8
6	16	9	20,85	32
8	40,32	27	72,22	128
10	101,59	83	258,37	512
12	256	259	943,43	2.048
14	645,08	817	3.493,78	8.192
16	1.625,50	2.599	13.072,54	32.768
18	4.096	8.323	49.299,64	131.072
20	10.321,27	26.797	187.078,97	524.288
22	26.007,98	86.659	713.491,21	2.097.152
24	65.536	281.287	2.732.463,29	8.388.608

Abbildung 5.1: Größenordnung der Menge der guten Tupel $M_d^{(0)}$

5.2 Geeignete Wahl von Variablen

In diesem Abschnitt wird – wie bereits im Kapitel 3.4 angedeutet – eine vom Autor der vorliegenden Arbeit entworfene Möglichkeit vorgestellt, wie man beim Aufruf von `QENF` eine geeignete gebundene Variable $x_i \in \{x_1, \ldots, x_n\}$ wählt, falls man nicht-triviale Gleichungen aus der Eingabeformel konstruieren muß. In Zeile 5 von `QENF` wird dazu eine der gebundenen Variablen benötigt, um die Matrix der Eingabeformel nach Satz 3.17 bzgl. dieser Variablen dementsprechend umzuformen und um dann zunächst nur diese ausgewählte Variable zu eliminieren. Je nachdem welche Variable man in Zeile 5 zur Umformung benutzt, desto kleiner oder größer kann das Ergebnis ψ der Elimination dieser Variablen in Zeile 9 nach dem letzten Schleifendurchlauf werden. Die Größe von ψ hängt im wesentlichen von den Größen der einzelnen Ergebnisformeln der Aufrufe von `QENF` in Zeile 9 ab, da bei diesen einzelnen Aufrufen ggf. sehr komplexe Typformeln berechnet werden müssen.

Als vorgreifende Bemerkung sei erwähnt, daß bei den Eliminationen in Zeile 9 stets der Fall „$r \neq 0$ und $dim(I) = 0$" eintritt. Also sind die einzelnen Ergebnisse der Eliminationen in Zeile 9 im wesentlichen Typformeln, in welche die Koeffizienten gewisser Polynome eingesetzt werden.

Ziel ist es nun im folgenden, x_i ad-hoc vor der Anwendung des Satzes 3.17 derart zu bestimmen, daß bei den Eliminationen von x_i in Zeile 9 die auftretenden Restklassenringe möglichst kleine Dimensionen besitzen, somit die dort zu berechnenden Typformeln von möglichst kleinem Grad und dadurch von möglichst geringer Komplexität sind und damit die Disjunktion dieser Typformeln möglichst klein wird. Dadurch soll die Ergebnisformel

$$\tilde{\psi} := \Big(\bigwedge_{k=1}^{t} h_k \neq_{koeff} 0\Big) \wedge \Big(\varphi_1 \vee \varphi_2 \vee \bigvee_{\substack{\text{„}\omega(f,j) \wedge \exists x_i(\phi)\text{"} \\ \text{in } \gamma}} \big(\omega(f,j) \wedge \mathtt{QENF}(\tilde{\omega}(f,j), \exists x_i(\phi))\big)\Big)$$

der Elimination von x_i möglichst klein werden. Die große Disjunktion in $\tilde{\psi}$ geht dabei über alle Disjunktionsglieder von γ. Da das Eliminationsergebnis $\tilde{\psi}$ existenzqantorifiziert als Argument an den Aufruf von `RRCQE` in Zeile 11 übergeben wird, ist es somit sicher auch wünschenswert, daß dieser Aufruf eine möglichst einfache Eingabeformel, deren Quantoren eliminiert werden sollen, bekommen soll. Denn mit steigender Komplexität der Eingabeformel, steigt auch die Komplexität der Berechnungen innerhalb eines Aufrufs von `RRCQE`. Die Größen der Teilformeln $h_k \neq_{koeff} 0$, φ_1 und φ_2 werden im weiteren nicht betrachtet, da dort keine Typformeln berechnet werden müssen.

Es sei darauf hingewiesen, daß es sich hierbei nur um eine worst-case-Strategie zur Wahl einer Variablen x_i handelt. Denn in allen auftretenden Typformeln müssen die Koeffizienten von Produkten charakteristischer Polynome eingesetzt werden und die daraus entstehenden Formeln können evtl. noch weiter vereinfacht werden, je nachdem wie es die zu eliminierende Eingabeformel zuläßt. Man kann aber davon ausgehen, daß eine Formel umso weniger vereinfacht werden kann, je größer die Formel ist. Außerdem müssen alle notwendigen Typformeln zunächst erst einmal berechnet und dann die jeweiligen Substitutionen durchgeführt werden, bevor eventuelle Simplifizierungen vorgenommen werden können. Ferner sind Typformeln von größerem Grad aufwendiger zu berechnen als Typformeln von kleinerem Grad. Somit ist die oben aufgeführte Strategie zur Wahl eines geeigneten x_i gerechtfertigt.

Es wird nun eine worst-case-Analyse durchgeführt, d.h. ein Vergleich gemacht mit der Annahme, daß in allen auftretenden Typformeln nach dem Einsetzen gewisser Koeffizienten nichts mehr weiter simplifiziert werden kann. Es wird dabei vorausgesetzt, daß alle Typformeln rekursiv nach Satz 3.11 berechnet werden.

Nachfolgendes Lemma ist ein technisches Hilfsmittel zum Beweis der Hauptaussage dieses Unterkapitels, welche angibt, welche Variable man für die Transformationen in Zeile 5 von `QENF` verwenden soll. Das Lemma greift dabei auf die bereits eingeführten Mengen $M_d^{(0)}$ der guten Tupel zurück, da man die Variable x_i später u. a. aufgrund der zu berechnenden Mengen $M_d^{(0)}$ auswählen wird.

Lemma 5.8 *Sei* $1 \leq m, s \in \mathbb{N}$. *Dann gilt*

$$\left(s + \binom{s}{2}\right) \left| \bigcup_{i=1}^{m \cdot 2^{s-1}} M_{2i}^{(0)} \right| \leq \left| \bigcup_{i=1}^{(m+1) \cdot 2^{s-1}} M_{2i}^{(0)} \right|.$$

Beweis: Für $s = 1$ ist diese Behauptung trivial, da

$$\binom{1}{2} = 0 \quad \text{und} \quad \bigcup_{i=1}^{m} M_{2i}^{(0)} \subseteq \bigcup_{i=1}^{m+1} M_{2i}^{(0)}.$$

Die Behauptung wird nun für $s \geq 2$ bewiesen. Da

$$(m+1) \cdot 2^{s-1} - m \cdot 2^{s-1} = 2^{s-1}$$

und wegen $s \geq 2$, $m \geq 1$ folgt, daß $(m \cdot 2^{s-1}) \geq 2$, genügt es für $2 \leq n \in \mathbb{N}$, n gerade, die Aussage

$$(1) \qquad \left(s + \binom{s}{2}\right) \left| \bigcup_{i=1}^{n} M_{2i}^{(0)} \right| \leq \left| \bigcup_{i=1}^{n+2^{s-1}} M_{2i}^{(0)} \right|$$

für alle $s \geq 2$ zu zeigen. Diese Ungleichung wird durch vollständige Induktion nach s bewiesen. Sei jetzt $s = 2$ und $2 \leq n \in \mathbb{N}$ beliebig mit n gerade. Zu zeigen ist hier:

$$3 \cdot \left| \bigcup_{i=1}^{n} M_{2i}^{(0)} \right| \leq \left| \bigcup_{i=1}^{n+2} M_{2i}^{(0)} \right|.$$

Die Beweisidee ist wie folgt: Man nimmt ein beliebiges gutes Tupel $\underline{c} \in \bigcup_{i=1}^{n} M_{2i}^{(0)}$ und zeigt, daß dieses gute Tupel ein Anfangsstück von mindestens drei verschiedenen guten Tupeln in $\bigcup_{i=1}^{n+2} M_{2i}^{(0)}$ ist.

Sei nun $\underline{c} = (c_0, \ldots, c_{2\kappa}) \in M_{2\kappa}^{(0)}$ für ein $\kappa \in \{1, \ldots, n\}$. Dann kann man aus $\underline{c}$ wie folgt 3 verschiedene Elemente $\underline{c_1}, \underline{c_2}, \underline{c_3}$ von $M_{2\kappa+4}^{(0)}$ erzeugen:

$$\begin{aligned}
\underline{c_1} &:= (c_0, \ldots, c_{2\kappa-1}, c'_{2\kappa}, 1, 0, -1, 1),\\
\underline{c_2} &:= (c_0, \ldots, c_{2\kappa-1}, c_{2\kappa}, 0, -1, 0, 1),\\
\underline{c_3} &:= (c_0, \ldots, c_{2\kappa-1}, c''_{2\kappa}, -1, 0, 1, 1),
\end{aligned}$$

wobei

$$c'_{2\kappa} := \begin{cases} 0 & \text{falls } c_{2\kappa-1} = -1 \\ 1 & \text{sonst} \end{cases} \quad \text{und} \quad c''_{2\kappa} := \begin{cases} 0 & \text{falls } c_{2\kappa-1} = 1 \\ 1 & \text{sonst} \end{cases} .$$

Zum Zählen der Vorzeichenwechsel von $\underline{c_1}, \underline{c_2}, \underline{c_3}$ kann man wieder $c'_{2\kappa}$ bzw. $c''_{2\kappa}$ durch 1 ersetzen, da im Fall $c'_{2\kappa} = 0$ bzw. $c''_{2\kappa} = 0$ für die Vorzeichenwechsel gilt:

$$\begin{aligned} Vzw(c_{2\kappa-1}, c'_{2\kappa}, 1) &= 1 = & Vzw(c_{2\kappa-1}, 1, 1) \quad & \text{bzw.} \\ Vzw(c_{2\kappa-1}, c''_{2\kappa}, -1) &= 1 = & Vzw(c_{2\kappa-1}, 1, -1) \quad & \text{und} \\ Vzw((-1)^{2\kappa-1} \cdot c_{2\kappa-1}, c'_{2\kappa}, -1) &= 1 = & Vzw((-1)^{2\kappa-1} \cdot c_{2\kappa-1}, 1, -1) \quad & \text{bzw.} \\ Vzw((-1)^{2\kappa-1} \cdot c_{2\kappa-1}, c''_{2\kappa}, 1) &= 1 = & Vzw((-1)^{2\kappa-1} \cdot c_{2\kappa-1}, 1, 1). & \end{aligned}$$

Daraus folgt

$$\begin{aligned} \pi(\underline{c_1}) &= Vzw((c_0, \ldots, c_{2\kappa-1}, c'_{2\kappa}, 1, 0, -1, 1)) \\ &= Vzw((c_0, \ldots, c_{2\kappa-1}, 1)) + Vzw((1, 1, 0, -1, 1)) = \frac{2\kappa}{2} + 2 = \frac{2\kappa + 4}{2} , \\ \nu(\underline{c_1}) &= Vzw((c_0, \ldots, (-1)^{2\kappa-1} \cdot c_{2\kappa-1}, c'_{2\kappa}, -1, 0, 1, 1)) \\ &= Vzw((c_0, \ldots, (-1)^{2\kappa-1} \cdot c_{2\kappa-1}, 1)) + Vzw((1, -1, 0, 1, 1)) \\ &= \frac{2\kappa}{2} + 2 = \frac{2\kappa + 4}{2} . \end{aligned}$$

Analog gelten diese Gleichungen für $\underline{c_2}, \underline{c_3}$. Offensichtlich erfüllen $\underline{c_1}, \underline{c_2}, \underline{c_3}$ nach Konstruktion Eigenschaft 5. in der Definition eines guten Tupels. Ferner sind nach Konstruktion in $\underline{c_1}, \underline{c_2}, \underline{c_3}$ die letzten Einträge jeweils „1" und die ersten Einträge in Abhängigkeit von κ richtig besetzt, da $\kappa \bmod 4 = (\kappa + 4) \bmod 4$. Somit sind $\underline{c_1}, \underline{c_2}, \underline{c_3}$ Elemente von $M^{(0)}_{2\kappa+4}$.

Seien zu $\underline{c'} \in M^{(0)}_{2\kappa}$, $\underline{c} \neq \underline{c'}$, die nach obigem Prinzip verlängerten Tupel $\underline{c'_1}, \underline{c'_2}, \underline{c'_3} \in M^{(0)}_{2\kappa+4}$. Dann gilt offensichtlich $\{\underline{c_1}, \underline{c_2}, \underline{c_3}\} \cap \{\underline{c'_1}, \underline{c'_2}, \underline{c'_3}\} = \emptyset$. Sei nun

$$\rho : \bigcup_{i=1}^{n} M^{(0)}_{2i} \to \bigcup_{i=1}^{n+2} M^{(0)}_{2i} : \underline{c} \mapsto \{\underline{c_1}, \underline{c_2}, \underline{c_3}\}$$

die injektive Abbildung, die jedem $\underline{c} \in \bigcup_{i=0}^{n} M^{(0)}_{2i}$, entsprechend wie oben beschrieben, die Menge der drei verlängerten guten Tupel konstruiert und zuweist. Dann gilt also

$$3 \cdot \left| \bigcup_{i=1}^{n} M^{(0)}_{2i} \right| = \left| \bigcup_{\underline{c} \in \bigcup_{i=1}^{n} M^{(0)}_{2i}} \rho(\underline{c}) \right| \leq \left| \bigcup_{i=1}^{n+2} M^{(0)}_{2i} \right| .$$

Also gilt die gewünschte Abschätzung (1) im Fall $s = 2$.

Induktionsschritt von s auf $s+1$: Gelte (1) für ein $s \in \mathbb{N}$, $s \geq 2$, und sei $2 \leq n \in \mathbb{N}$ mit n gerade. Zu zeigen ist:

$$(2) \qquad \left((s+1) + \binom{s+1}{2}\right) \left| \bigcup_{i=1}^{n} M_{2i}^{(0)} \right| \leq \left| \bigcup_{i=1}^{n+2^{s+1}} M_{2i}^{(0)} \right|$$

Nach Induktionsannahme gelten folgende beiden Ungleichungen:

$$\begin{aligned} \left(s + \binom{s}{2}\right) \left| \bigcup_{i=1}^{n} M_{2i}^{(0)} \right| &\leq \left| \bigcup_{i=1}^{n+2^{s}} M_{2i}^{(0)} \right| \qquad \text{und} \\ \left(s + \binom{s}{2}\right) \left| \bigcup_{i=1}^{n+2^{s}} M_{2i}^{(0)} \right| &\leq \left| \bigcup_{i=1}^{n+2^{s+1}} M_{2i}^{(0)} \right| . \end{aligned}$$

Also gilt folgende Ungleichung:

$$\left(s + \binom{s}{2}\right)^2 \left| \bigcup_{i=1}^{n} M_{2i}^{(0)} \right| \leq \left| \bigcup_{i=1}^{n+2^{s+1}} M_{2i}^{(0)} \right| .$$

Es bleibt nun noch übrig, die Ungleichung

$$(3) \qquad \left((s+1) + \binom{s+1}{2}\right) \leq \left(s + \binom{s}{2}\right)^2$$

zu zeigen, woraus dann direkt die zu beweisende Behauptung (2) folgt. Man beachte, daß für $2 \leq n \in \mathbb{N}$ die Gleichung $\binom{n}{2} = \frac{n(n-1)}{2}$ gilt. Man formt nun die Ungleichung (3) äquivalent wie folgt um:

$$\begin{aligned} (3) \quad &\Leftrightarrow \quad \left(s + 1 + \frac{(s+1)s}{2}\right) \leq \left(s + \frac{s(s-1)}{2}\right)^2 \\ &\Leftrightarrow \quad s + 1 + \frac{s^2 + s}{2} \leq s^2 + s^3 - s^2 + \frac{s^2(s^2 - 2s + 1)}{4} \\ &\Leftrightarrow \quad 4s + 4 + 2s^2 + 2s \leq 4s^3 + s^4 + s^2 - 2s^3 \\ &\Leftrightarrow \quad 0 \leq \underbrace{s^4 + 2s^3 - s^2 - 6s - 4}_{p :=} \end{aligned}$$

Das Polynom p in der Variable s hat 2 reelle und 2 echt komplexe Nullstellen. Die reellen Nullstellen sind bei –1 und bei ungefähr 1,66. Da der höchste Koeffizient von p positiv ist, ist also für $s \geq 2$: $p(s) \geq 0$. Also gilt somit die Ungleichung (3) für $s \geq 2$ und damit (2). Damit wurde die Behauptung des Lemmas bewiesen. □

Mit dem eben bewiesenen Lemma kann nun folgender Satz bewiesen werden:

Satz 5.9 *Seien $1 \leq m, s \in \mathbb{N}$. Dann gilt:*

$$\sum_{i=1}^{m} \left(s + \binom{s}{2} \right) \left| \bigcup_{j=1}^{i \cdot 2^{s-1}} M_{2j}^{(0)} \right| \leq \sum_{i=1}^{m+1} \left| \bigcup_{j=1}^{i \cdot 2^{s-1}} M_{2j}^{(0)} \right| .$$

Beweis: Wegen $|\bigcup_{j=1}^{2^{s-1}} M_{2j}^{(0)}| \geq 0$ und mit Lemma 5.8 gilt:

$$\begin{aligned}
\sum_{i=1}^{m} \left(s + \binom{s}{2} \right) \left| \bigcup_{j=1}^{i \cdot 2^{s-1}} M_{2j}^{(0)} \right| &\overset{5.8}{\leq} \sum_{i=1}^{m} \left| \bigcup_{j=1}^{(i+1) \cdot 2^{s-1}} M_{2j}^{(0)} \right| \\
&= \sum_{i=2}^{m+1} \left| \bigcup_{j=1}^{i \cdot 2^{s-1}} M_{2j}^{(0)} \right| \\
&\leq \sum_{i=1}^{m+1} \left| \bigcup_{j=1}^{i \cdot 2^{s-1}} M_{2j}^{(0)} \right| .
\end{aligned}$$

□

Es werden jetzt die Notationen aus 3.4 übernommen: Sei also für den Rest dieses Unterkapitels 5.2 φ eine Formel in Eliminationsnormalform ohne nichttriviale Gleichungen und mit mindestens einer $>$-Relation, d.h.

$$\varphi := \exists x_1 \ldots \exists x_n \left(\bigwedge_{j=1}^{s} g_j(\underline{U}, \underline{x}) > 0 \wedge \bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0 \right) \text{ mit } g_j, h_k \in \mathbb{Z}[\underline{U}][\underline{X}]$$

und $s \geq 1$. Ziel ist es, beim Aufruf von `QENF` mit Eingabeformel φ die Transformation in Zeile 5 bzgl. einer möglichst geeigneten Variablen durchzuführen.

Wenn man im folgenden von allen auftretenden/zu berechnenden guten Tupeln bei der Bestimmung einer Typformel für Polynome vom Grad d, d gerade, mit Typ 0 spricht, so ist damit die Menge

$$\bigcup_{i=1}^{d/2} M_{2i}^{(0)}$$

gemeint. Diese Vereinigungsmenge beinhaltet somit genau diejenigen guten Tupel, welche zur rekursiven Berechnung einer Typformel für Polynome vom Grad d mit Typ 0 benötigt werden.

Spricht man im weiteren von der Wahl x_i als Hauptvariable, dann ist damit zu verstehen, daß man in Zeile 5 die Umformung bzgl. x_i durchführt, so daß im Ergebnis $\hat{\varphi}$ dieser Umformung in Zeile 5 x_i als einzige gebundene Variable auftritt.

Als weitere vorgreifende Bemerkung wird festgehalten, welche Typformeln bei den einzelnen Eliminationen in Zeile 9 von `QENF` berechnet werden müssen. Es müssen dort Formeln der Form

$$\exists x \Big(\bigwedge_{j=1}^{s} p_j(\underline{U}, x) > 0 \wedge f^{(q)}(\underline{U}, x) = 0 \Big) \qquad \text{mit } p_j, f \in \mathbb{Z}[\underline{U}][X]$$

unter der Annahme der Gültigkeit von $\tilde{\omega}(f, q)$ eliminiert werden, wobei f die formale Ableitung eines p_j bzgl. der Variablen X bzw. die Differenz zweier Elemente p_j, p_k ist und $q \in \{1, \ldots, Grad_X(f)\}$, so daß $\omega(f, q) \in \Omega_f$. Wie bereits am Ende des letzten Kapitels gezeigt wurde, existiert bei der Elimination einer Formel obiger Form in Zeile 9 von `QENF` im zugehörigen grüngefärbten Gröbnersystem nur ein Ast, der stets nulldimensional ist. Dabei haben die auftretenden Restklassenringe eine Dimension $q \leq Grad_X(f)$. Für festes $f, \tilde{\omega}(f, q)$ müssen nach Satz 3.13 2^s charakteristische Polynome berechnet werden. Also muß zur Elimination einer Formel obiger Form eine Typformel für Polynome vom Grad $q \cdot 2^s$ mit Typ 0 berechnet werden.

Wenn man vom worst-case ausgeht, dann ist zu einem solchen gegebenen f die Menge Ω_f von der Kardinalität $Grad_X(f)$ und jedes Element darin in der Theorie der reellen Zahlen erfüllbar. In diesem Fall ist $T(f) = \{X^1, \ldots, X^{Grad_X(f)}\}$ und alle Koeffizienten dieser Terme sind echt parametrisch. Somit müssen zu solch einem f im worst-case Typformeln vom Grad $Grad_X(f) \cdot 2^s, (Grad_X(f) - 1) \cdot 2^s, \ldots, 1 \cdot 2^s$ berechnet werden. Zur weiteren Analyse wird daher ab jetzt für den Rest dieses Unterkapitels stets davon ausgegangen, daß dieser worst-case vorliegt, d.h. daß zu f gilt: $|\Omega_f| = Grad_X(f)$ und jedes $\omega(f, q) \in \Omega_f$ erfüllbar ist.

Nun wird eine Antwort auf das eingangs formulierte Problem beim Aufruf von `QENF` mit zu eliminierender Eingabeformel φ gegeben, bzgl. welcher Variable $x_i \in \{x_1, \ldots, x_n\}$ die Umformung in Zeile 5 vorgenommen werden soll, so daß x_i in den Zeilen 6-10 von `QENF` eliminiert wird. Dazu wird der in Abbildung 5.2 aufgeführte Algorithmus `SELECTX` verwendet, dessen Arbeitsweise jetzt erläutert wird. Die Termination des Algorithmus ist dabei offensichtlich, da alle *for*-Schleifen über endliche Mengen laufen.

Zur Anwendung des Algorithmus `SELECTX` muß zunächst noch folgende Ordnung auf $\mathbb{N}^q$ eingeführt werden. Für $1 \leq q \in \mathbb{N}$ ist folgendermaßen eine Ordnung $<_{lex}$ auf $\mathbb{N}^q$ definiert:

$$(n_1, \ldots, n_q) <_{lex} (n'_1, \ldots, n'_q) \quad \Leftrightarrow \quad \exists i \in \{1, \ldots, q-1\} : n_\ell = n'_\ell \text{ für } \ell < i \text{ und } n_i < n'_i.$$

Wie bereits erwähnt, tritt bei den Eliminationen in Zeile 9 von `QENF` stets der nulldimensionale Fall ein. Die Idee von `SELECTX` ist, daß für jede Variable $x_i \in \{x_1, \ldots, x_n\}$ die Dimensionen aller auftretenden Restklassenringe bei der Elimination von x_i in Zeile 9 gesammelt werden, falls x_i als Hauptvariable in Zeile 5 gewählt wird.

Zur Formel φ bildet man nun folgende zwei Mengen:

$$\begin{aligned} G_1 &:= \left\{g_j \mid j \in \{1,\ldots,s\}\right\} \backslash \mathbb{Z}[\underline{U}] \subseteq \mathbb{Z}[\underline{U}][\underline{X}], \\ G_2 &:= \left\{g \mid g = g_j - g_k \text{ für } 1 \leq j < k \leq s\right\} \backslash \mathbb{Z}[\underline{U}] \subseteq \mathbb{Z}[\underline{U}][\underline{X}]. \end{aligned}$$

Daraus berechnet man jetzt `SELECTX`($\{x_1,\ldots,x_n\}$,G_1,G_2). Im Algorithmus `SELECTX` werden also zunächst in den Tupeln M_{x_i} alle höchsten Potenzen der Variablen X_i in den einzelnen Ableitungen der g_j und den Differenzen $g_j - g_k$ gesammelt, sofern man alle Ableitungen und Differenzen als univariate parametrisierte Polynome in der Hauptvariable X_i betrachtet. Diese höchsten Potenzen entsprechen den Graden $Grad_X(f)$ von oben. Also ist jeder Eintrag $m \neq 0$ von M_{x_i} ein Verweis auf zu berechnende Typformeln für Polynome von den Graden

$$m \cdot 2^s, (m-1) \cdot 2^s, \ldots, 1 \cdot 2^s$$

mit Typ 0 aufgrund der worst-case-Sichtweise. Man beachte, daß nur zu denjenigen Ableitungen der g_j bzgl. X_i bzw. Differenzen $g_j - g_k$ Eliminationen in Zeile 9 durchgeführt werden müssen, so daß

$$\Omega_{\frac{\partial g_j}{\partial X_i}} \neq \emptyset \quad \text{bzw.} \quad \Omega_{g_j - g_k} \neq \emptyset,$$

sofern man alle Ableitungen und Differenzen als univariate Polynome in X_i betrachtet und x_i als Hauptvariable in der Transformation von Zeile 5 gewählt wird. Deswegen sind in G_1 und G_2 Elemente von $\mathbb{Z}[\underline{U}]$ überflüssig, da $\Omega_f = \emptyset$ für $f \in \mathbb{Z}[\underline{U}]$, egal welche Variable $x_i \in \{x_1,\ldots,x_n\}$ man als Hauptvariable wählt.

Also bedeutet ein Eintrag 0 in M_{x_i}, daß die von diesem Eintrag stammende Ableitung $\hat{g}_{j,i}$ eines g_j bzgl. der Variablen X_i bzw. Differenz von g_j und g_k konstant bzgl. X_i ist. Wenn man für die Transformation in Zeile 5 x_i als Hauptvariable auswählt, so ist in diesem Fall $\Omega_{\hat{g}_{j,i}} = \emptyset$ bzw. $\Omega_{g_j - g_k} = \emptyset$. Mit anderen Worten, zum Eintrag 0 müssen keine Typformeln berechnet werden, wenn x_i als Hauptvariable gewählt wird.

Nach Konstruktion kodiert M_{x_i} also die Dimensionen aller auftretenden Restklassenringe bei den Eliminationen in Zeile 9, wenn man x_i als Haupvariable bei der Transformation in Zeile 5 wählt.
Jedes M_{x_i} wird dann absteigend sortiert bzgl. der natürlichen Ordnung auf den natürlichen Zahlen. Danach wählt man bzgl. der lexikographischen Ordnung $<_{lex}$ das kleinste Element aus $\left\{M_{x_j} \mid j \in \{1,\ldots,n\}\right\}$, etwa M_{x_i}. Somit ist dann $x_i =$ `SELECTX`$(\{x_1,\ldots,x_n\}, G_1, G_2)$.

Es muß nun gezeigt werden, daß aufgrund eines geeigneten Vergleichsmaßes x_i den anderen Variablen als zu benutzende Hauptvariable für die Transformation in Zeile 5 von `QENF` vorzuziehen ist.

Als Vergleichskriterium wird die Gesamtanzahl aller auftretenden guten Tupel bei der Berechnung aller Typformeln, die bei den Eliminationen mittels `QENF` in Zeile 9 bestimmt werden müssen, betrachtet. Dieses Kriterium eignet sich dafür, da die Komplexität von Typformeln insbesondere von der Anzahl der zu deren Berechnung notwendigen zu bestimmenden guten Tupel abhängt. Falls man also x_i als Hauptvariable bei der Umformung in Zeile 5 von `QENF` wählt und dadurch in Zeile 9 bei den Eliminationen insgesamt q_i Typformeln für Polynome vom Grad $d_{i,1}, \ldots, d_{i,q_i}$, mit $d_{i,r}$ gerade für $r \in \{1, \ldots, q_i\}$, bestimmt werden müssen, so ist der zugehörige Vergleichswert

$$\lambda_i := \sum_{r=1}^{q_i} \sum_{\kappa=1}^{d_{i,r}/2} \left| M_{2\kappa}^{(0)} \right|.$$

Aus mathematischer Sicht ist also die Funktion

$$\Lambda : \{x_1, \ldots, x_n\} \to \mathbb{N} \quad \text{mit} \quad \Lambda(x_i) = \lambda_i$$

zu minimieren. Sei jetzt $q := |G_1| + |G_2|$. Dann definiert man die Abbildung

$$\hat{\Lambda} : \mathbb{N}^q \to \mathbb{N} : (c_1, \ldots, c_q) \mapsto \sum_{i=1}^{q} \sum_{j=1}^{c_i} \sum_{\kappa=1}^{j \cdot 2^{s-1}} \left| M_{2\kappa}^{(0)} \right|.$$

Offensichtlich gilt für $k \in \{1, \ldots, n\}$ aufgrund der Vorbemerkungen und nach Konstruktion von M_{x_k} nach Abarbeitung des Algorithmus `SELECTX`:

$$\Lambda(x_k) = \lambda_k = \hat{\Lambda}(M_{x_k}).$$

Die nächsten beiden Lemmata sind technische Hilfsmittel für den Beweis des darauffolgenden Satzes.

Lemma 5.10 *Sei $k \in \{1, \ldots, n\}$ und sei nach Abarbeitung des Algorithmus* `SELECTX` *$M_{x_k} = (m_1, \ldots, m_q)$; also ist M_{x_k} insbesondere absteigend sortiert. Sei ferner $m \in \mathbb{N}$ mit $m \leq m_1$. Dann ist*

$$\lambda_k \geq \hat{\Lambda}\big((m, 0, \ldots, 0)\big).$$

Beweis:

$$\begin{aligned} \lambda_k = \hat{\Lambda}(M_{x_k}) &= \sum_{i=1}^{q} \sum_{j=1}^{m_i} \sum_{\kappa=1}^{j \cdot 2^{s-1}} \left| M_{2\kappa}^{(0)} \right| \\ &\geq \sum_{j=1}^{m_1} \sum_{\kappa=1}^{j \cdot 2^{s-1}} \left| M_{2\kappa}^{(0)} \right| \geq \sum_{j=1}^{m} \sum_{\kappa=1}^{j \cdot 2^{s-1}} \left| M_{2\kappa}^{(0)} \right| = \hat{\Lambda}\big((m, 0, \ldots, 0)\big) \end{aligned}$$

□

Lemma 5.11 *Sei $k \in \{1,\ldots,n\}$ und sei nach Abarbeitung des Algorithmus* `SELECTX` *$M_{x_k} = (m_1,\ldots,m_q)$; also ist M_{x_k} insbesondere absteigend sortiert. Sei ferner $m \in \mathbb{N}$ mit $m \geq m_1$. Dann ist*

$$\lambda_k \leq \hat{\Lambda}\big((m,\ldots,m)\big).$$

Beweis:

$$\lambda_k = \hat{\Lambda}(M_{x_k}) = \sum_{i=1}^{q}\sum_{j=1}^{m_i}\sum_{\kappa=1}^{j\cdot 2^{s-1}} \left|M_{2\kappa}^{(0)}\right| \leq \sum_{i=1}^{q}\sum_{j=1}^{m}\sum_{\kappa=1}^{j\cdot 2^{s-1}} \left|M_{2\kappa}^{(0)}\right| = \hat{\Lambda}\big((m,\ldots,m)\big)$$

□

Somit kann man zeigen:

Satz 5.12 *Sei x_i = `SELECTX`$(\{x_1,\ldots,x_n\},G_1,G_2)$. Für die Wahl x_i als Hauptvariable bei den Umformungen in Zeile 5 von* `QENF` *wird die Gesamtanzahl aller zu berechnenden guten Tupel zur Bestimmung aller notwendigen Typformeln bei den Aufrufen von* `QENF` *in Zeile 9 minimal, d.h. $\Lambda(x_i)$ ist minimal in $\big\{\Lambda(x_j) \mid j \in \{1,\ldots,s\}\big\}$.*

Beweis:
Zum Beweis dieser Aussage muß man nach Abarbeitung des Algorithmus `SELECTX` zwei Tupel M_{x_i}, M_{x_j}, $i \neq j$, vergleichen. Sei mit $q := |G_1| + |G_2|$ nach Ablauf des Algorithmus $M_{x_j} >_{lex} M_{x_i}$, etwa

$$M_{x_j} = (m_1,\ldots,m_q) \quad \text{und} \quad M_{x_i} = (m'_1,\ldots,m'_q)$$

mit $\ell \in \{1,\ldots,q-1\}$ und $m_{q'} = m_{q'}$ für $q' < \ell$, und $m_\ell > m'_\ell$. Es ist also zu zeigen:

$$\Lambda(x_j) \geq \Lambda(x_i).$$

Da die Summe

$$\sum_{r=1}^{\ell-1}\sum_{j=1}^{m_r}\sum_{\kappa=1}^{j\cdot 2^{s-1}} \left|M_{2\kappa}^{(0)}\right|$$

als Teilsumme sowohl in $\Lambda(x_j)$ als auch in $\Lambda(x_i)$ auftritt, kann man in M_{x_i} und M_{x_j} die ersten $\ell-1$ Stellen vernachlässigen. Also kann man zum weiteren Vergleich o.B.d.A. $\ell = 1$ und somit $m_1 > m'_1$ annehmen.

Für $m'_1 = 0$ müssen bei der Wahl von x_i keine Formeln in Zeile 9 eliminiert werden, so daß in diesem Falle x_i der Variablen x_j in Zeile 5 bei den Umformungen als zu benützende Hauptvariable vorzuziehen ist, da $\Lambda(x_i) = 0$

Beschreibung: Auswahl einer Variablen x_i in `QENF` zur Anwendung der Transformation in Zeile 5

Eingabe: Eine Menge $\underline{x} := \{x_1, \ldots, x_n\}$ von Variablen und zwei Mengen von Polynomen $M_1, M_2 \subseteq \mathbb{Z}[U_1, \ldots, U_m][X_1, \ldots, X_n]$, wobei $M_1 \neq \emptyset$.

Output Ein Element $x \in \{x_1, \ldots, x_n\}$, so daß $\Lambda(x)$ minimal ist.

procedure SELECTX($\underline{x}$**,**M_1**,**M_2**)**
begin
for $i := 1$ **to** n **do**
    $M_{x_i} := ()$
od
for $i := 1$ **to** n **do**
    for each $g \in M_1$ **do**
        $M_{x_i} := (Grad_{X_i}(g) - 1) \circ M_{x_i}$
    od
    for each $g \in M_2$ **do**
        $M_{x_i} := (Grad_{X_i}(g)) \circ M_{x_i}$
    od
od
for $i := 1$ **to** n **do**
    $M_{x_i} :=$ „Sortiere M_{x_i} absteigend bzgl. der natürlichen Ordnung auf $\mathbb{N}$."
od
$x :=$„Wähle $x \in \{x_1, \ldots, x_n\}$, so daß M_x kleinstes Element in
    $\{M_{x_i} \mid i \in \{1, \ldots, n\}\}$ bzgl. der Ordnung $<_{lex}$ ist."
return x
end SELECTX

Abbildung 5.2: Algorithmus `SELECTX` zur Wahl von x_i

minimal ist.
Sei nun $m_1' > 0$, und damit $m_1 > 1$. Also gilt wegen $q \leq s + \binom{s}{2}$:

$$\begin{aligned}
\Lambda(x_j) &= \hat{\Lambda}(M_{x_j}) \\
&\overset{5.10}{\geq} \hat{\Lambda}\big((m_1, 0, \ldots, 0)\big) \\
&= \sum_{j=1}^{m_1} \sum_{\kappa=1}^{j\cdot 2^{s-1}} \left| M_{2\kappa}^{(0)} \right| \\
&= \sum_{j=1}^{m_1} \left| \bigcup_{i=1}^{j\cdot 2^{s-1}} M_{2i}^{(0)} \right| \\
&\overset{5.9}{\geq} \sum_{j=1}^{m_1-1} \left(s + \binom{s}{2} \right) \left| \bigcup_{i=1}^{j\cdot 2^{s-1}} M_{2i}^{(0)} \right| \\
&\geq \sum_{i=1}^{q} \sum_{j=1}^{m_1-1} \sum_{\kappa=1}^{j\cdot 2^{s-1}} \left| M_{2\kappa}^{(0)} \right| \\
&\geq \sum_{i=1}^{q} \sum_{j=1}^{m_1'} \sum_{\kappa=1}^{j\cdot 2^{s-1}} \left| M_{2\kappa}^{(0)} \right| \\
&= \hat{\Lambda}\big((m_1', \ldots, m_1')\big) \\
&\overset{5.11}{\geq} \Lambda(x_i).
\end{aligned}$$

Daraus folgt die Behauptung des Satzes. □

Man geht also folgendermaßen vor, wenn man den Algorithmus `SELECTX` verwendet, um eine geeignete Variable für die Umformungen und Eliminationen in den Zeilen 5-9 von `QENF` zu bestimmen:

Man berechnet aus der zu eliminierenden Eingabeformel mit den gebundenen Variablen $\{x_1, \ldots, x_n\}$ zunächst die Mengen G_1 und G_2, berechnet daraus das Ergebnis x_i von `SELECTX`($\{x_1, \ldots, x_n\}$,G_1,G_2), führt mit x_i als Hauptvariablen die Umformung in Zeile 5 durch, und eliminiert in Zeile 9 die Formeln der Form $\exists x_i(\phi)$. In Zeile 11 wird dann die Formel $\exists x_1 \ldots \exists x_{i-1} \exists x_{i+1} \ldots \exists x_n(\psi)$ eliminiert.

Man beachte, daß die Menge $G_1^{(i)}$ der Ableitungen der Elemente von G_1 bzgl. X_i und die Menge G_2 nicht disjunkt sein müssen. Bei der Implementierung ist darauf zu achten, daß man in Zeile 9 von `QENF` keine gleichen Formeln doppelt eliminiert, deren Gleichungen von verschiedenen Ableitungen bzw. Differenzen stammen. Aufgrund der worst-case-Strategie kann man aber annehmen, daß für alle $i \in \{1, \ldots, n\}$ gilt: $G_1^{(i)} \cap G_2 = \emptyset$.

Umso größer die Anzahl s der $>$-Relationen in einer Eingabeformel und die

Anzahl der in Zeile 9 zu eliminierenden Formeln ist, desto positiver kann sich die Anwendung von `SELECTX` auswirken.

Folgendes Beispiel verdeutlicht die Verwendung von `SELECTX`:

Beispiel 5.13 *Es soll folgende Formel eliminiert werden:*

$$\exists x \exists y \left(ay + bx > 0 \wedge cy^2 + d > 0 \right).$$

Es liegt also der Fall vor, daß bei der Quantorenelimination dieser Formel nicht-triviale Gleichungen hinzugefügt werden müssen. Wenn man nun die Elimination mit Anwendungen von ***SELECTX*** *durchführt, so werden die Quantoren dieser Formel in 3,7 Sekunden eliminiert. Dabei ist das Ergebnis des allerersten* ***SELECTX****-Aufrufes die Variable*

$$x = \textit{SELECTX}(\{x, y\}, \{ay + bx, cy^2 + d\}, \{ay + bx - cy^2 - d\}),$$

wobei nach diesem ersten Ablauf des ***SELECTX****-Algorithmus gilt:*

$$M_x = (1, 1, 0) \quad \text{und} \quad M_y = (2, 2, 1).$$

Ohne Anwendung des ***SELECTX****-Algorithmus wird y als Hauptvariable für die allererste Umformung in Zeile 5 verwendet und in Zeile 6-9 eliminiert. Die Quantoren dieser Formel konnten ohne Anwendung von* ***SELECTX*** *in akzeptabler Zeit nicht mehr eliminiert werden.* $\diamond$

Bis jetzt wurde angenommen, daß in der Eingabeformel φ mindestens eine $>$-Relation auftritt. Ist dies nicht der Fall, so vereinfacht sich die entsprechende Elimination, und es wird nicht mehr nötig sein, die Umformung in Zeile 5 von `QENF` durchzuführen und in Zeile 11 rekursive Aufrufe von `RRCQE` zu machen. Falls also

$$\varphi := \exists x_1 \ldots \exists x_n \left(\bigwedge_{k=1}^{t} h_k(\underline{U}, \underline{x}) \neq 0 \right) \qquad \text{mit } h_k \in \mathbb{Z}[\underline{U}][\underline{X}]$$

und mit $t \geq 1$, so können die Quantoren dieser Formel φ direkt eliminiert werden. Denn hier liegt eine Existenzaussage für endlich viele zu erfüllende Ungleichungen vor. In der Theorie der reellen Zahlen ist jede endliche Menge von nicht-trivialen Ungleichungen stets erfüllbar. Es muß also nur gesichert sein, daß die einzelnen h_k in den Parametern $\underline{U}$ und den Variablen $\underline{X}$ keine Nullpolynome sind. Dies ist genau dann der Fall, wenn in jedem h_k mindestens ein Koeffizient nicht verschwindet. Falls also $T(h_k)$ die Menge aller auftretenden Terme in h_k ist und

$$h_k = \sum_{v \in T(h_k)} a_v \cdot v \quad \text{mit } a_v \in \mathbb{Z}[\underline{U}] \text{ für alle } k \in \{1, \ldots, t\},$$

dann ist φ äquivalent zu folgender quantorenfreien Formel:

$$\bigwedge_{k=1}^{t} \left(\bigvee_{v \in T(h_k)} a_v \neq 0 \right).$$

Somit kann man bei der Elimination von φ direkt die dazu äquivalente quantorenfreie Formel bestimmen, ohne weitere rekursive Aufrufe durchführen zu müssen.

5.3 Aufsplitten von Typformeln des Typs 0

In diesem Abschnitt wird eine vom Autor der vorliegenden Arbeit entworfene Möglichkeit vorgestellt, wie man für ein Produkt $f_1 \cdot f_2$ von zwei parametrisierten Polynomen $f_1, f_2 \in \mathbb{Z}[\underline{U}][X]$ mittels einer quantorenfreien Formel mit den Parametern $\underline{U}$ als Variablen charakterisieren kann, wann dieses Produkt vom Typ 0 ist. Dabei ist es nicht notwendig, die Koeffizienten des Produkts $f_1 \cdot f_2$ in eine Typ-0-Typformel für Polynome vom Grad $Grad_X(f_1 \cdot f_2)$ einzusetzen. Der zentrale Punkt ist, daß ein Produkt von Polynomen genau dann Typ 0 besitzt, wenn die Summe der Typen der einzelnen Faktoren 0 ist. Es wird gezeigt werden, daß man eine quantorenfreie Formel berechnen kann, die eine Fallunterscheidung nach den Typen der beiden Faktoren f_1, f_2 darstellt, so daß das Produktpolynom $f_1 \cdot f_2$ Typ 0 hat. Dabei wird sich herausstellen, daß die durch diese Fallunterscheidung resultierende, quantorenfreie Formel i. a. von geringerer Größe und Komplexität ist als eine Typ-0-Typformel für Polynome vom Grad des Produktpolynoms.

Die Ergebnisse dieses Kapitels werden dann insbesondere für den Algorithmus `QENF` interessant sein und können für diesen noch weiter verallgemeinert werden. Da dort im nulldimensionalen Fall genau diese Aufgabenstellung vorliegt, bringt es somit einen entscheidenden Effizienzgewinn mit sich, wenn man diese dort auftretenden, den Typ 0 charakterisierenden Formeln „kleiner" machen kann. Dazu müssen Typformeln für beliebige Typen eingeführt werden.

Definition 5.14 (Typformeln vom Typ k) *Sei $k \in \mathbb{Z}$, $|k| \leq d \in \mathbb{N}$. Sei $f = X^d + \sum_{i=0}^{d-1} C_i X^i \in \mathbb{R}[C_0, \ldots, C_{d-1}][X]$. Sei $A \subseteq \mathbb{R}^d$ die Menge der Spezialisierungen der Parameter $C_0, \ldots, C_{d-1}$, so daß für jedes $\underline{a} \in A$ das Polynom $f(\underline{a}, X)$ ausschließlich reelle Nullstellen besitzt.*
Dann ist $T_d^{(k)}(C_0, \ldots, C_{d-1})$ eine Typformel (für Polynome) vom Grad d mit Typ k, falls für jede Spezialisierung $\underline{a} := (a_0, \ldots, a_{d-1}) \in A$ gilt:

$$T_d^{(k)}(a_0, \ldots, a_{d-1}) \quad \text{genau dann, wenn} \quad \tau\big(f(\underline{a}, X)\big) = k.$$

$T_d^{(k)}(C_0,\ldots,C_{d-1})$ *heißt strikte Typformel vom Grad d mit Typ k, falls*

$$T_d^{(k)}(a_0,\ldots,a_{d-1}) \quad \text{genau dann, wenn} \quad \big(\tau(f(\underline{a},X)) = k \text{ und } a_0 \neq 0\big).$$

Bemerkung: Man beachte, daß diese Definition von Typformeln für Typ $k = 0$ genau der Definition 3.10 entspricht.

Typformeln für Polynome vom Typ $k \neq 0$ können analog wie Typformeln zum Typ 0 berechnet werden:

Satz 5.15 *Sei $0 \neq k \in \mathbb{Z}, |k| \leq d \in \mathbb{N}$. Dann existiert eine Typformel $T_d^{(k)}(C_0,\ldots,C_{d-1})$ für Polynome vom Grad d mit Typ k. Diese Formel läßt sich algorithmisch rekursiv konstruieren: Für $0 \neq k \in \mathbb{Z}$ gilt:*

$$\text{Falls } k > 0: \qquad T_{|k|}^{(k)}(C_0,\ldots,C_{|k|-1}) \equiv \bigwedge_{i=1}^{|k|} (-1)^i \cdot C_{|k|-i} > 0$$

$$\text{Falls } k < 0: \qquad T_{|k|}^{(k)}(C_0,\ldots,C_{|k|-1}) \equiv \bigwedge_{i=0}^{|k|-1} C_i > 0$$

Für $0 \neq k \in \mathbb{Z}$, $d \in \mathbb{N}, d > |k|$, gilt:

$$\begin{aligned} \text{Falls } d-|k| \text{ gerade}: \quad & T_d^{(k)}(C_0,\ldots,C_{d-1}) \equiv \Big(C_0 \neq 0 \wedge T_d^{(k)'}(C_0,\ldots,C_{d-1})\Big) \vee \\ & \qquad \Big(C_0 = 0 \wedge T_{d-1}^{(k)}(C_1,\ldots,C_{d-1})\Big) \\ \text{Falls } d-|k| \text{ ungerade}: \quad & T_d^{(k)}(C_0,\ldots,C_{d-1}) \equiv C_0 = 0 \wedge T_{d-1}^{(k)}(C_1,\ldots,C_{d-1}), \end{aligned}$$

wobei $T_d^{(k)'}(C_0,\ldots,C_{d-1})$ eine strikte Typformel für Polynome vom Grad d mit Typ k ist, welche ebenfalls algorithmisch bestimmt werden kann.

Beweis: Sei $f = X^d + \sum_{i=0}^{d-1} a_i X^i \in \mathbb{R}[X]$ ein Polynom mit ausschließlich reellen Nullstellen.

Gelte zunächst $\tau(f) = k$. Zu zeigen ist, daß für die Belegung $a_0,\ldots,a_{d-1}$ der Variablen $C_0,\ldots,C_{d-1}$ obige Formeln erfüllt sind.

<u>Fall $k > 0$ und $d = k$:</u> Es müssen in diesem Fall sämtliche Nullstellen positiv sein. Nach Satz 5.2 ist somit

$$Vzw(a_0,\ldots,a_{d-1},1) = k = d,$$

also muß zwischen jeweils zwei aufeinanderfolgenden Koeffizienten a_i und a_{i+1}, $i \in \{0,\ldots,d-1\}$, ein Vorzeichenwechsel stattfinden. Da der höchste Koeffizient von f fest 1 ist, muß dadurch für $i \in \{1,\ldots,d\}$ gelten, daß $(-1)^i a_{d-i} > 0$, also ist die Formel

$$\bigwedge_{i=1}^{|k|} (-1)^i \cdot a_{|k|-i} > 0$$

erfüllt.

Fall $k < 0$ und $d = |k|$: Es müssen in diesem Fall also sämtliche Nullstellen negativ sein. Nach Satz 5.2 ist also

$$Vzw(a_0, \ldots, a_{d-1}, 1) = 0 \quad \text{und} \quad Vzw(a_0, \ldots, (-1)^i \cdot a_i, \ldots, (-1)^d) = |k|.$$

Also dürfen zwischen zwei aufeinanderfolgenden Koeffizienten a_i und a_{i+1}, $i \in \{0, \ldots, d-1\}$, keine Vorzeichenwechsel stattfinden. Dabei kann kein a_i gleich 0 sein, da sonst

$$Vzw(a_0, \ldots, (-1)^i \cdot a_i, \ldots, (-1)^d) < d = |k|$$

wäre, d.h. f hätte weniger als k negative Nullstellen, ein Widerspruch. Da der höchste Koeffizient von f fest 1 ist, muß somit für $i \in \{0, \ldots, d-1\}$ gelten, daß $a_i > 0$, also ist die Formel

$$\bigwedge_{i=0}^{|k|-1} a_i > 0$$

erfüllt.

Fall $d > |k|$ und $d - |k|$ ungerade: Dann muß f für $k > 0$ mindestens k positive Nullstellen haben bzw. mindestens $|k|$ negative Nullstellen haben, falls $k < 0$. Sei etwa $f = \left(\prod_{j=1}^{|k|}(X - z_j)\right) \cdot g$, wobei $g = \sum_{\ell=0}^{d-|k|} b_\ell X^\ell$ und $z_j > 0$ für $k > 0$ und $z_j < 0$ für $k < 0$ für alle $j \in \{1, \ldots, |k|\}$. g besitzt als Faktor von f ebenso ausschließlich reelle Nullstellen und hat Typ 0:

$$0 = \tau(f) - k = \tau\left(\prod_{j=1}^{|k|}(X - z_j)\right) + \tau(g) - k = k + \tau(g) - k = \tau(g).$$

Seien P, N die Anzahl der positiven bzw. negativen Nullstellen von g und N_0 die Vielfachheit der Nullstelle 0 in g. Da $\tau(g) = P - N = 0$, ist $P = N$. Daraus folgt nun mit Hilfe von $P + N_0 + N = d - |k| = Grad_X(g)$, daß $N_0 = d - |k| - 2P$. Da nach Voraussetzung in diesem Fall $d - |k|$ ungerade ist und $d - |k| \geq 2P$, ist somit $N_0 > 0$, also ist $b_0 = 0$ und daraus folgt $a_0 = b_0 \cdot \prod_{j=1}^{|k|} z_j = 0$. Somit ist

$$f = X \cdot f_1 \quad \text{für } f_1 = X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}.$$

Wegen $k = \tau(f) = \tau(X) + \tau(f_1) = \tau(f_1)$, ist $T_{d-1}^{(k)}(a_1, \ldots, a_{d-1})$ erfüllt.

Fall $d > |k|$ und $d - |k|$ gerade: In diesem Fall wird eine Fallunterscheidung gemacht, ob $a_0 = 0$. Ist $a_0 \neq 0$, so ist 0 keine Nullstelle von f, also ist $T_d^{(k)'}(a_0, \ldots, a_{d-1})$ erfüllt. Ist hingegen $a_0 = 0$, so ist wiederum

$$f = X \cdot f_1 \quad \text{für } f_1 = X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}$$

und somit wiederum $\tau(f_1) = k$, also ist $T^{(k)}_{d-1}(a_1, \ldots, a_{d-1})$ erfüllt.

Gelten nun umgekehrt die im Satz aufgeführten Formeln für die Belegung $a_0, \ldots, a_{d-1}$ der Variablen $C_0, \ldots, C_{d-1}$. Zu zeigen ist: $\tau(f) = k$.

<u>Fall $k > 0$ und $d = k$:</u> Gelte $\bigwedge_{i=1}^{|k|}(-1)^i a_{|k|-i} > 0$. Dann ist

$$Vzw(a_0, \ldots, a_{d-1}, 1) = k,$$

also besitzt f nach Satz 5.2 genau k positive Nullstellen. Da $Grad_X(f) = k$ kann f somit keine negativen Nullstellen besitzen, also ist $\tau(f) = k$.

<u>Fall $k < 0$ und $d = |k|$:</u> Gelte $\bigwedge_{i=0}^{|k|-1} a_i > 0$. In diesem Fall ist

$$Vzw(a_0, \ldots, a_{d-1}, 1) = 0,$$

also besitzt f nach Satz 5.2 keine einzige positive Nullstelle. Wegen $a_0 > 0$ ist 0 keine Nullstelle von f. Da $Grad_X(f) = k$ kann f somit nur negative Nullstellen besitzen, nämlich $|k|$ Stück, also ist $\tau(f) = k$.

<u>Fall $d > |k|$ und $d - |k|$ ungerade:</u> Also ist $a_0 = 0$ und es gilt $T^{(k)}_{d-1}(a_1, \ldots, a_{d-1})$, d.h. $\tau(X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}) = k$. Es folgt:

$$\begin{aligned} k &= \tau\Big(X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}\Big) = \tau(X) + \tau\Big(X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}\Big) \\ &= \tau\Big(X^d + \sum_{i=1}^{d-1} a_i X^i + a_0\Big) = \tau(f). \end{aligned}$$

<u>Fall $d > |k|$ und $d - |k|$ gerade:</u> Wiederum wird eine Fallunterscheidung gemacht, ob $a_0 = 0$. Gelte $a_0 \neq 0$ und $T^{(k)'}_{d}(a_0, \ldots, a_{d-1})$. Dann hat nach Definition einer strikten Typformel f den Typ k. Gelte andererseits $a_0 = 0$ und $T^{(k)}_{d-1}(a_1, \ldots, a_{d-1})$. Dann ist wiederum $f = X \cdot (X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1})$ und $\tau(X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}) = k$, also

$$\tau(f) = \tau\bigg(X \cdot \Big(X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}\Big)\bigg) = \underbrace{\tau(X)}_{=0} + \tau\Big(X^{d-1} + \sum_{i=1}^{d-1} a_i X^{i-1}\Big) = k.$$

□

Strikte Typformeln für Polynome vom Grad d mit Typ $k \neq 0$ können analog wie im Fall $k = 0$ berechnet werden. Man benötigt dazu die Menge aller guten Tupel vom Grad d mit Typ k:

Definition 5.16 (Gute Tupel vom Typ k) *Seien $d \in \mathbb{N}$, $k \in \mathbb{Z}$, mit $|k| \leq d$, und $d - |k|$ gerade.*
$(c_0, \ldots, c_d) \in \{-1, 0, 1\}^{d+1}$ ist ein gutes Tupel (für Polynome) vom Grad d des Typs k, falls es folgende Eigenschaften erfüllt:

1. $c_d = 1$

2. $\pi((c_0, \ldots, c_d)) + \nu((c_0, \ldots, c_d)) = d$

3. $\pi((c_0, \ldots, c_d)) - \nu((c_0, \ldots, c_d)) = k$

4. Für alle $i \in \{1, \ldots, d-1\}$ *gilt:* $c_i = 0 \Longleftrightarrow c_{i-1} \cdot c_{i+1} = -1$

Die Menge

$M_d^{(k)} := \big\{(c_0, \ldots, c_d) \mid (c_0, \ldots, c_d)$ *ist gutes Tupel des Typs* $k\big\} \subseteq \big\{-1, 0, 1\big\}^{d+1}$

heißt die Menge der guten Tupel für Polynome vom Grad d *des Typs* k.

Bemerkungen:

1. Die Mengen $M_d^{(k)}$ können algorithmisch analog wie die Mengen $M_d^{(0)}$ bestimmt werden. Der in [4], Abbildung 5, vorgestellte Algorithmus zur Berechnung von $M_d^{(0)}$ kann leicht abgewandelt werden, um $M_d^{(k)}$, $k \neq 0$, zu bestimmen, da alle relevanten Resultate zur Berechnung von $M_d^{(0)}$ sich nicht auf den Typ 0 stützen. Auf nähere Einzelheiten wird hierzu verzichtet.

2. Man beachte, daß die Definition von $M_d^{(k)}$ für $k = 0$ kompatibel ist zur Definition 5.3. Man kann zeigen, daß zu festem d, k in einem guten Tupel $(c_0, \ldots, c_d) \in M_d^{(k)}$ das Element c_0 eindeutig durch d und k festgelegt ist. Für die weiteren Rechnungen spielt dies aber keine Rolle, so daß verzichtet wird, hier näher darauf einzugehen.

Aus $M_d^{(k)}$, $k \neq 0$, kann genau wie im Fall $k = 0$ eine strikte Typformel für Polynome vom Grad d mit Typ k konstruiert werden. Man bildet aus den einzelnen guten Tupeln wiederum analog zum Fall $k = 0$ jeweils eine Konjunktion, und verknüpft diese Konjunktionen disjunktiv miteinander. Die enstehende disjunktive Normalform ist dann eine strikte Typformel vom Grad d bzgl. des Typs k. Ein Beispiel verdeutlicht dies:

Beispiel 5.17 *Für* $d = 5$ *und* $k = 3$ *ist*

$M_5^{(3)} = \big\{(1, -1, 1, 0, -1, 1), (1, -1, 0, 1, -1, 1), (1, 0, -1, 1, -1, 1), (1, -1, 1, -1, 0, 1)\big\}.$

Sei $T_5^{(3)'}(C_0, \ldots, C_4)$ *eine strikte Typformel für Polynome vom Grad 5 mit Typ 3. Dann gilt somit:*

$$\begin{aligned} T_5^{(3)'} \equiv\ & (C_0 > 0 \wedge C_1 < 0 \wedge C_2 > 0 \wedge C_4 < 0) \vee \\ & (C_0 > 0 \wedge C_1 < 0 \wedge C_3 > 0 \wedge C_4 < 0) \vee \\ & (C_0 > 0 \wedge C_2 < 0 \wedge C_3 > 0 \wedge C_4 < 0) \vee \\ & (C_0 > 0 \wedge C_1 < 0 \wedge C_2 > 0 \wedge C_3 < 0). \end{aligned}$$

◇

Für die weiteren Berechnungen wird nun eine vereinfachte Schreibweise im Zusammenhang mit Typformeln eingeführt: Für ein Polynom

$$f = X^d + \sum_{i=0}^{d-1} a_i X^i \in \mathbb{Z}[U_1, \dots, U_m][X] \ , \ k \in \mathbb{Z} \ , \ |k| \le d \in \mathbb{N}$$

und eine Typformel $T_d^{(k)}(C_0, \dots, C_{d-1})$ für Polynome vom Grad d mit Typ k wird ab jetzt statt $T_d^{(k)}(a_0, \dots, a_{d-1})$ kurz $T_d^{(k)}(f)$ notiert.

Nach diesen vorbereitenden Definitionen und zugehörigen Bemerkungen kann man nun zum wesentlichen in diesem Abschnitt übergehen. Ziel ist es, anders als über eine Typformel für Polynome mit Typ 0 zu charakterisieren, wann ein Produkt von zwei parametrisierten Polynomen vom Typ 0 ist.

Definition 5.18 (Aufgesplittete Typ-0-Typformel) *Sei $1 \le d \in \mathbb{N}$. Seien $a_0, \dots, a_{d-1}, b_0, \dots, b_{d-1}$ paarweise verschiedene Variablen und seien für alle $k \in \mathbb{Z}$, $|k| \le d$, $T_d^{(k)}(a_0, \dots, a_{d-1})$ bzw. $T_d^{(k)}(b_0, \dots, b_{d-1})$ Typformeln für Polynome vom Grad d mit Typ k. Dann heißt die Formel*

$$T_{split}^{(d)}(a_0, \dots, a_{d-1}, b_0, \dots, b_{d-1}) := \bigvee_{i=-d}^{d} \Big(T_d^{(i)}(a_0, \dots, a_{d-1}) \wedge T_d^{(-i)}(b_0, \dots, b_{d-1}) \Big)$$

aufgesplittete Typ-0-Typformel vom Grad d.

Bemerkung: Für $T_{split}^{(d)}(a_0, \dots, a_{d-1}, b_0, \dots, b_{d-1})$ wird auch die Kurzschreibweise $T_{split}^{(d)}(\underline{a}, \underline{b})$ verwendet.

Nachfolgender Satz zeigt, warum der Name „Aufgesplittete Typ-0-Typformel" für $T_{split}^{(d)}(\underline{a}, \underline{b})$ gerechtfertigt ist. Dazu betrachtet man zwei parametrisierte Polynome

$$f_1 = X^d + \sum_{i=0}^{d-1} a_i X^i \ , \ f_2 = X^d + \sum_{j=0}^{d-1} b_j X^j \in \mathbb{R}[a_0, \dots, a_{d-1}, b_0, \dots, b_{d-1}][X].$$

Sei $S \subseteq \mathbb{R}^{2d}$ die Menge aller Spezialisierungen der Parameter $a_0, \dots, a_{d-1}$, $b_0, \dots, b_{d-1}$, so daß für alle $\underline{s} \in S$ die Polynome $f_1(\underline{s}, X)$ und $f_2(\underline{s}, X)$ ausschließlich reelle Nullstellen besitzen. Mit diesen Notationen kann man nun zeigen:

Satz 5.19 *Für jede Spezialisierung $\underline{s} \in S$ gilt:*

$$T_{split}^{(d)}\big(a_0(\underline{s}), \dots, a_{d-1}(\underline{s}), b_0(\underline{s}), \dots, b_{d-1}(\underline{s})\big) \quad \Longleftrightarrow \quad \tau\big((f_1 \cdot f_2)(\underline{s}, X)\big) = 0 \ .$$

Beweis: Sei $\underline{s} \in S$ gegeben. Gelte zunächst $\tau\big((f_1 \cdot f_2)(\underline{s}, X)\big) = 0$. Es folgt mit Lemma 3.9, daß

$$0 = \tau\big((f_1 \cdot f_2)(\underline{s}, X)\big) = \tau\big(f_1(\underline{s}, X)\big) + \tau\big(f_2(\underline{s}, X)\big) \ .$$

Daher ist $\tau(f_1(\underline{s}, X)) = -\tau(f_2(\underline{s}, X))$. Falls $\tau(f_1(\underline{s}, X)) = i \in \{-d, \ldots, d\}$, so ist auf der linken Seite der Äquivalenz die Formel

$$T_d^{(i)}\big(f_1(\underline{s}, X)\big) \wedge T_d^{(-i)}\big(f_2(\underline{s}, X)\big)$$

erfüllt, also ist

$$T_{split}^{(d)}\big(a_0(\underline{s}), \ldots, a_{d-1}(\underline{s}), b_0(\underline{s}), \ldots, b_{d-1}(\underline{s})\big)$$

erfüllt.

Gelte umgekehrt die linke Seite. Dann existiert ein wahres Disjunktionsglied von $T_{split}^{(d)}(a_0(\underline{s}), \ldots, a_{d-1}(\underline{s}), b_0(\underline{s}), \ldots, b_{d-1}(\underline{s}))$ auf der linken Seite, etwa

$$T_d^{(i)}\big(f_1(\underline{s}, X)\big) \wedge T_d^{(-i)}\big(f_2(\underline{s}, X)\big)$$

für ein $i \in \{-d, \ldots, -1, 0, 1, \ldots, d\}$. D.h. $\tau(f_1(\underline{s}, X)) = i$ und $\tau(f_2(\underline{s}, X)) = -i$. Es folgt wiederum mit Satz 3.9, daß

$$\tau\big((f_1 \cdot f_2)(\underline{s}, X)\big) = \tau\big(f_1(\underline{s}, X)\big) + \tau\big(f_2(\underline{s}, X)\big) = i + (-i) = 0 \ .$$

Daraus folgt die Behauptung des Satzes. □

Es wurde also gezeigt, daß die einzelnen Disjunktionen der aufgesplitteten Typ-0-Typformel $T_{split}^{(d)}(\underline{a}, \underline{b})$ eine Fallunterscheidung nach den Typen der Faktoren f_1 bzw. f_2 darstellen. Somit stehen nun zu folgender Aufgabenstellung zwei Lösungsmöglichkeiten zur Verfügung:

Aufgabenstellung 5.20 *Seien*

$$f_1 = X^d + \sum_{i=0}^{d-1} a'_i X^i \quad , \quad f_2 = X^d + \sum_{j=0}^{d-1} b'_j X^j \in \mathbb{Z}[\underline{U}][X],$$

so daß f_1 und f_2 ausschließlich reelle Nullstellen für jede Spezialisierung der Parameter $\underline{U}$ besitzen. Ferner ist

$$f_1 \cdot f_2 = X^{2d} + \sum_{k=0}^{2d-1} c_k X^k \qquad mit \ \ c_k := \sum_{\ell=0}^{k} a'_\ell b'_{k-\ell}.$$

Gesucht ist eine quantorenfreie Formel φ in den Variablen $\underline{U}$, so daß für jede Spezialisierung $\underline{u} \in \mathbb{R}^m$ von $\underline{U}$ gilt: Das Polynom $(f_1 \cdot f_2)(\underline{u}, X)$ hat genau dann Typ 0, wenn $\varphi(\underline{u})$ gilt.

Sei $T_{2d}^{(0)}(C_0, \ldots, C_{2d-1})$ eine Typformel für Polynome vom Grad $2d$ mit Typ 0 und $T_{split}^{(d)}(a_0, \ldots, a_{d-1}, b_0, \ldots, b_{d-1})$ eine aufgesplittete Typ-0-Typformel vom Grad d. Mit dem Ergebnis aus Satz 5.19 gibt es dann folgende zwei Lösungsmöglichkeiten φ_1, φ_2 zu obiger Aufgabenstellung 5.20:

- $\varphi_1 := T_{2d}^{(0)}(c_0, \ldots, c_{2d-1})$
- $\varphi_2 := T_{split}^{(d)}(a'_0, \ldots, a'_{d-1}, b'_0, \ldots, b'_{d-1})$

Berechnet man die Lösung über φ_2, so spricht man von einer Anwendung des *Splitting-Verfahrens* oder auch von *Aufsplitten* der Typ-0-Typformel.

Nun taucht hierbei die Frage auf, die Berechnung welcher Lösungsmöglichkeit aus Effizienz- und Komplexitätsgründen vorzuziehen ist. Dazu führt man wieder eine worst-case-Analyse durch, d.h. man vergleicht die beiden den Lösungen φ_1 und φ_2 zugrunde liegenden Formeln

$$\hat{\varphi}_1 := T_{2d}^{(0)}(C_0, \ldots, C_{2d-1}) \quad \text{bzw.} \quad \hat{\varphi}_2 := T_{split}^{(d)}(a_0, \ldots, a_{d-1}, b_0, \ldots, b_{d-1})$$

und nimmt an, daß man durch das Einsetzen der Koeffizienten a'_i, b'_j, c_k keine Simplifikationen mehr an den entstehenden Formeln durchführen kann. Also genügt es, eine Typformel für Polynome vom Grad $2d$ mit Typ 0 mittels eines geeigneten Maßes zu vergleichen mit einer aufgesplitteten Typ-0-Typformel vom Grad d. Dabei erachtet man eine Formel als umso komplexer, je mehr atomare Formeln sie enthält.

Es ist aber hierbei zu erwähnen, daß ein Komplexitätsvergleich dieser beiden Formeln nur eine Heuristik liefert, welche Formel besser geeignet ist zur Bestimmung einer Lösungsformel zu obiger Aufgabenstellung 5.20. Da die Größe und Komplexität von φ_1 und φ_2 letzlich immer von den konkreten Werten $a'_0, \ldots, a'_{d-1}, b'_0, \ldots, b'_{d-1}$ abhängen, können φ_1 und φ_2 mehr oder weniger stark simplifiziert werden. Dennoch ist es sinnvoll, anhand eines Komplexitätsvergleichs zwischen den Formeln $\hat{\varphi}_1$ und $\hat{\varphi}_2$ zu entscheiden, ob man aus Effizienzgründen die Formel φ_1 oder φ_2 als Lösung berechnen soll. Denn zuerst müssen die Formeln $\hat{\varphi}_1$ bzw. $\hat{\varphi}_2$ berechnet werden und erst daraus können die Formeln φ_1 bzw. φ_2 bestimmt werden. Außerdem kann man i. a. davon ausgehen, daß Formeln umso weniger simplifiziert werden können, je komplexer sie sind. Kann also gezeigt werden, daß $\hat{\varphi}_1$ von größerer Komplexität ist als $\hat{\varphi}_2$, so ist i. a. davon auszugehen, daß auch nach eventuellen Simplifikationen die Formel φ_1 von größerer Komplexität ist als φ_2. Somit ist der beabsichtigte worst-case-Komplexitätsvergleich gerechtfertigt, was das nächste Ziel in diesem Abschnitt sein wird.

Man geht im folgenden davon aus, daß alle Typformeln rekursiv nach Satz 3.11 bzw. Satz 5.15 berechnet werden. Für eine Typformel $T_d^{(k)}(C_0, \ldots, C_{d-1})$ wird ab jetzt auch die abkürzende Notation $T_d^{(k)}$ verwendet, analog für strikte Typformeln die Notation $T_d^{(k)'}$.

Um die beiden Formeln $T_{2d}^{(k)}$ und $T_{split}^{(d)}(\underline{a}, \underline{b})$ nun geeignet vergleichen zu können, wird hierzu eine *Penalty-Funktion* γ eingeführt, welche die Komplexität von (strikten) Typformeln mißt, indem ,,Strafpunkte" vergeben werden. Diese Strafpunkte von Typformeln entstehen, wenn die Komplexität der Typformeln aufgrund rekursiver Berechnungsschritte steigt oder Mengen guter Tupel betrachtet werden müssen. Sei

$$\begin{aligned} \Theta &:= \left\{T_d^{(k)} \mid k \in \mathbb{Z}, 1 \leq d \in \mathbb{N}, |k| \leq d\right\} \quad \text{bzw.} \\ \Theta' &:= \left\{T_d^{(k)'} \mid k \in \mathbb{Z}, 1 \leq d \in \mathbb{N}, |k| \leq d, d-|k| \text{ gerade}\right\} \end{aligned}$$

die Menge aller Typformeln bzw. aller strikten Typformeln für Polynome von beliebigem Grad und beliebigem Typ. Grundgedanke bei der folgenden Definition von γ ist, in einer Typformel zum einen die Anzahl aller zu deren Berechnung notwendigen guten Tupel als Kriterium einzubeziehen und zum anderen die Anzahl der aufgrund der rekursiven Berechnung der Typformeln auftretenden atomaren Formeln. Umso größer die Komplexität einer Typformel ist, umso höher soll der zugehörige Wert von γ sein. Man definiert γ folgendermaßen:

$$\gamma : \Theta \cup \Theta' \to \mathbb{N} \quad \text{mit folgenden Zuweisungen:}$$

(i) Strikte Typformeln: Seien $1 \leq d \in \mathbb{N}$, $k \in \mathbb{Z}$, $|k| \leq d$, $d - |k|$ gerade. Die Größe und Komplexität von strikten Typformeln hängen nach Konstruktion im wesentlichen vom Grad d und von der Menge der guten Tupel $M_d^{(k)}$ ab. Ein gutes Tupel von $M_d^{(k)}$ hat Länge $d+1$, wobei der letzte Eintrag eines jeden guten Tupels nicht zur Bildung der strikten Typformel verwendet wird. Also sind in jedem guten Tupel nur die ersten d Einträge für die Berechnung einer strikten Typformel $T_d^{(k)'}$ relevant; aus jedem guten Tupel werden bis zu d atomare Formeln erzeugt. Somit wird jedes gute Tupel in $M_d(k)$ mit d Strafpunkten gewertet. Es ergibt sich folgende Definition von γ für strikte Typformeln:
$$\gamma\left(T_d^{(k)'}\right) := d \cdot \left|M_d^{(k)}\right|.$$

(ii) Nach Satz 3.11 besitzt eine Typformel für Polynome vom Grad 1 bzw. 2 mit Typ 0 eine bzw. zwei atomare Formeln. Jede in diesen beiden Formeln auftretende atomare Formel wird mit einem Strafpunkt gewertet, d.h.
$$\gamma\big(T_1^{(0)}\big) := 1 \quad \text{und} \quad \gamma\big(T_2^{(0)}\big) := 2.$$

(iii) Für $k \neq 0$, $d = |k|$: Nach Satz 5.15 ist eine Typformel für Polynome vom Grad d mit Typ k eine Konjunktion von d atomaren Formeln. Wiederum wird jede darin auftretende atomare Formel mit einem Strafpunkt gewertet, d.h.
$$\gamma\big(T_{|k|}^{(k)}\big) := d.$$

(iv) Falls „$d > 2$ und $k = 0$ und d ungerade" bzw. „$d > |k| \neq 0$ und $d - |k|$ ungerade": Da nach Satz 3.11 bzw. 5.15

$$T_d^{(k)}(C_0, \dots, C_{d-1}) \equiv C_0 = 0 \wedge T_{d-1}^{(k)}(C_1, \dots, C_{d-1})$$

gilt, erbt $T_d^{(k)}$ die Strafpunkte von $T_{d-1}^{(k)}$. Außerdem verursacht die Formel $C_0 = 0$ einen weiteren Strafpunkt, also

$$\gamma\big(T_d^{(k)}\big) := \gamma\big(T_{d-1}^{(k)}\big) + 1.$$

(v) Falls „$d > 2$ und $k = 0$ und d gerade" bzw. „$d > |k| \neq 0$ und $d - |k|$ gerade": Da nach Satz 3.11 bzw. 5.15

$$\begin{aligned} T_d^{(k)}(C_0, \dots, C_{d-1}) \quad \equiv \quad & \Big(C_0 = 0 \wedge T_{d-1}^{(k)}(C_1, \dots, C_{d-1})\Big) \vee \\ & \Big(C_0 \neq 0 \wedge T_d^{(k)'}(C_0, \dots, C_{d-1}\Big) \end{aligned}$$

gilt, erbt wiederum $T_d^{(k)}$ die Strafpunkte von $T_{d-1}^{(k)}$. Die atomaren Formeln $C_0 = 0$ bzw. $C_0 \neq 0$. verursachen jeweils einen Strafpunkt. Außerdem verursacht die strikte Typformel $T_d^{(k)'}$ Strafpunkte. Somit ist

$$\gamma\big(T_d^{(k)}\big) := 2 + \gamma\big(T_{d-1}^{(k)}\big) + \gamma\big(T_d^{(k)'}\big).$$

Mit dieser Definition ist γ ein sinnvolles Maß für die Komplexität von Typformeln. Mit Hilfe von γ soll nun gezeigt werden, daß – falls man d groß genug wählt – $T_{2d}^{(0)}$ von größerer Komplexität ist als $T_{split}^{(d)}(\underline{a}, \underline{b})$. Da letztere Formel aus insgesamt jeweils zwei Typformeln für Polynome vom Grad d mit den Typen $k \in \{-d, \dots, -1, 0, 1, \dots, d\}$ aufgebaut ist, ist folgende Ungleichung für genügend großes d zu zeigen:

$$(*) \quad \gamma\Big(T_{2d}^{(0)}\Big) > 2 \cdot \sum_{i=-d}^{d} \gamma\Big(T_d^{(i)}\Big).$$

Wenn dies gezeigt werden kann, so ist $T_{2d}^{(0)}$ von größerer Komplexität als $T_{split}^{(d)}(\underline{a}, \underline{b})$ und damit ist zu obiger Aufgabenstellung 5.20 aus Effizienzgründen die Berechnung der Formel φ_2 der Formel φ_1 als Lösung vorzuziehen.

Zuerst wird $\gamma\big(T_{2d}^{(0)}\big)$ nach unten abgeschätzt:

Lemma 5.21 *Sei $16 \leq d \in \mathbb{N}$, d gerade. Dann gilt folgende Ungleichung:*

$$\gamma\Big(T_d^{(0)}\Big) > \left(\frac{d}{2} + 2\right) 2^{\frac{2}{3}(d-4)} \left(1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}\right).$$

Beweis: Sei $d \in \mathbb{N}$ mit $d \geq 16$ und d gerade gegeben. Mit der Definition von γ, Lemma 5.7 und der Äquivalenz

$$\frac{d}{2} + 2 \leq d - 4 \Leftrightarrow d \geq 12$$

folgt die Ungleichungskette

$$\begin{aligned}
\gamma\big(T_d^{(0)}\big) &= 2 + \gamma\big(T_d^{(0)'}\big) + \gamma\big(T_{d-1}^{(0)}\big) \\
&= 2 + \gamma\big(T_d^{(0)'}\big) + 1 + \gamma\big(T_{d-2}^{(0)}\big) \\
&= 3 + d \cdot \big|M_d^{(0)}\big| + 2 + \gamma\big(T_{d-3}^{(0)}\big) + \gamma\big(T_{d-2}^{(0)'}\big) \\
&= 5 + d \cdot \big|M_d^{(0)}\big| + (d-2) \cdot \big|M_{d-2}^{(0)}\big| + 1 + \gamma\big(T_{d-4}^{(0)}\big) \\
&= 6 + d \cdot \big|M_d^{(0)}\big| + (d-2) \cdot \big|M_{d-2}^{(0)}\big| + 2 + \underbrace{\gamma\big(T_{d-5}^{(0)}\big)}_{\geq 0} + \gamma\big(T_{d-4}^{(0)'}\big) \\
&> d \cdot \big|M_d^{(0)}\big| + (d-2) \cdot \big|M_{d-2}^{(0)}\big| + (d-4) \cdot \big|M_{d-4}^{(0)}\big| \\
&\overset{5.7}{\geq} d \cdot 2^{\frac{2}{3}d} + (d-2) \cdot 2^{\frac{2}{3}(d-2)} + (d-4) \cdot 2^{\frac{2}{3}(d-4)} \\
&\geq (d-4) \cdot \Big(2^{\frac{2}{3}d} + 2^{\frac{2}{3}(d-2)} + 2^{\frac{2}{3}(d-4)}\Big) \\
&\geq \left(\frac{d}{2} + 2\right) \cdot 2^{\frac{2}{3}(d-4)} \cdot \Big(1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}\Big).
\end{aligned}$$

□

Nun wird $\gamma\big(T_d^{(k)}\big)$ nach oben wie folgt abgeschätzt:

Lemma 5.22 *Seien $k \in \mathbb{Z}$ und $|k| \leq d \in \mathbb{N}$, $d \geq 1$. Dann gilt:*

$$\gamma\big(T_d^{(k)}\big) < 2^d \cdot (d+2).$$

Beweis: durch vollständige Induktion nach d. Sei $k \in \mathbb{Z}$ beliebig, aber fest gegeben.

<u>Basisfälle:</u>

$k = 0$ **und** $d = 1$**:** $\gamma\big(T_1^{(0)}\big) = 1 < 6 = 2^1 \cdot (1+2) = 2^d \cdot (d+2)$

$k = 0$ **und** $d = 2$**:** $\gamma\big(T_2^{(0)}\big) = 2 < 16 = 2^2 \cdot (2+2) = 2^d \cdot (d+2)$

$k \neq 0$ **und** $d = |k|$**:** $\gamma\big(T_d^{(k)}\big) = d \overset{d \geq 1}{\leq} 2d < 2d + 4 = 2 \cdot (d+2) \overset{d \geq 1}{\leq} 2^d \cdot (d+2)$

<u>Induktionsschritt von d auf $d+1$:</u> Sei die Abschätzung für ein $1 \leq d \in \mathbb{N}$ mit $d \geq |k| \neq 0$ bzw. für ein $d \in \mathbb{N}$, $d \geq 2$, im Fall $k = 0$ gültig. Zu zeigen ist:

$$\gamma\big(T_{d+1}^{(k)}\big) < 2^{d+1} \cdot (d+3).$$

Es wird eine Fallunterscheidung gemacht, ob $d+1-|k|$ gerade ist.
Sei $d+1-|k|$ ungerade. Daraus folgt folgende Ungleichungskette:

$$\gamma\big(T_{d+1}^{(k)}\big) = 1+\gamma\big(T_d^{(k)}\big) \overset{I.-A.}{<} 1+2^d\cdot(d+2) \overset{d\geq 1}{\leq} 2^{d+1}+2^{d+1}\cdot(d+2) = 2^{d+1}\cdot(d+3).$$

Sei nun $d+1-|k|$ gerade. Dann folgt:

$$\begin{aligned}
\gamma\big(T_{d+1}^{(k)}\big) &= 2+\gamma\big(T_d^{(k)}\big)+\gamma\big(T_{d+1}^{(k)'}\big) \\
&\overset{I.-A.}{<} 2+2^d\cdot(d+2)+(d+1)\cdot\underbrace{\big|M_{d+1}^{(k)}\big|}_{\leq 2^d} \\
&\leq 2+d\cdot 2^d+2^{d+1}+(d+1)\cdot 2^d \\
&= 2+d\cdot 2^{d+1}+3\cdot 2^d \\
&= 2+d\cdot 2^{d+1}+1,5\cdot 2^{d+1} \\
&= 2+(d+1,5)\cdot 2^{d+1} \\
&\overset{d\geq 1}{\leq} 1,5\cdot 2^{d+1}+(d+1,5)\cdot 2^{d+1} \\
&= 2^{d+1}\cdot(d+3)\ .
\end{aligned}$$

Insgesamt folgt daraus die Behauptung des Lemmas. □

Daraus kann man nun die angestrebte Ungleichung $(*)$ zeigen:

Lemma 5.23 *Sei $4 \leq d \in \mathbb{N}$. Dann gilt:*

$$2\cdot\sum_{i=-d}^{d}\gamma\Big(T_d^{(i)}\Big) < \gamma\Big(T_{2d}^{(0)}\Big).$$

Für $d < 4$ gilt diese Ungleichung nicht.

Beweis: Abbildung 5.3 stellt für $d \leq 16$ die Werte der linken und rechten Seite der Ungleichung gegenüber, wobei $\Sigma_{split}^{(d)} := 2\cdot\sum_{i=-d}^{d}\gamma(T_d^{(i)})$. Die im Lemma behauptete Aussage gilt also für $4 \leq d \leq 16$. Für $d < 4$ gilt sie nicht. Sei nun $d \geq 17$. Zunächst werden zwei Hilfsabschätzungen benötigt:

<u>Beh.1:</u>

$$d'+2 < (d'+1)\cdot 2^{\frac{1}{6}} \qquad \text{für } 34 \leq d' \in \mathbb{N}.$$

<u>Beweis:</u> durch vollständige Induktion nach d'. Mit numerischer Abschätzung ergibt sich für $d' = 34$:

$$34+2 = 36 < (34+1)\cdot 2^{\frac{1}{6}} \approx 39,29\ .$$

d	$\Sigma_{split}^{(d)}$	$\gamma(T_{2d}^{(0)})$	Splitting besser?
1	6	2	nein
2	28	18	nein
3	86	76	nein
4	204	296	ja
5	442	1.130	ja
6	932	4.242	ja
7	1.938	15.684	ja
8	4.000	57.272	ja
9	8.210	207.090	ja
10	16.748	743.034	ja
11	33.950	2.649.536	ja
12	68.388	9.400.428	ja
13	136.922	33.214.014	ja
14	272.580	116.944.742	ja
15	539.818	410.535.816	ja
16	1.064.008	4.518.434.568	ja

Abbildung 5.3: Vergleich der Strafpunkte beim Aufsplitten

Gelte Beh. 1 für ein $d' \geq 34$. Im Induktionsschritt von d' auf $d'+1$ ist die Abschätzung $(d'+3) < (d'+2) \cdot 2^{\frac{1}{6}}$ zu zeigen:

$$((d'+1)+2) < (d'+2) + 2^{\frac{1}{6}} \overset{I.-A.}{<} (d'+1) \cdot 2^{\frac{1}{6}} + 2^{\frac{1}{6}} = (d'+2) \cdot 2^{\frac{1}{6}}.$$

Damit ist die Beh.1 bewiesen.

Außerdem benötigt man folgende Hilfsabschätzung:

<u>Beh.2:</u>

$$d'+1 < 2^{\frac{1}{6}d' - \frac{11}{3}}(1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}) \quad \text{für } 34 \leq d' \in \mathbb{N}.$$

<u>Beweis:</u> durch vollständige Induktion nach d'. Sei zunächst $d' = 34$. Mit numerischer Abschätzung ergibt sich:

$$34 + 1 = 35 < 2^{\frac{34}{6} - \frac{11}{3}}(1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}) \approx 39,48.$$

Gelte nun die Behauptung für ein $d' \geq 34$. Somit ergibt sich der Induktionsschritt von d' auf $d'+1$ wie folgt:

$$(d'+1)+1 = d'+2 \overset{Beh.1}{<} (d'+1)2^{\frac{1}{6}} \overset{I.-A.}{<} 2^{\frac{1}{6}(d'+1) - \frac{11}{3}}(1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}).$$

Somit gilt Beh.2.

Nun kann die Behauptung des Lemmas für $d \geq 17$ gezeigt werden:

$$\begin{aligned}
2 \cdot \sum_{i=-d}^{d} \gamma\Big(T_d^{(i)}\Big) &\overset{5.22}{<} 2 \cdot \sum_{i=-d}^{d} \big(2^d \cdot (d+2)\big) \\
&= (2d+1) \cdot 2^{d+1} \cdot (d+2) \\
&\overset{Beh.2, d\geq 17}{<} 2^{\frac{1}{6}2d - \frac{11}{3}} \cdot (1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}) \cdot 2^{d+1} \cdot (d+2) \\
&= (d+2) \cdot 2^{\frac{4}{3}d - \frac{8}{3}} \cdot (1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}) \\
&= (d+2) \cdot 2^{\frac{2}{3}(2d-4)} \cdot \Big(1 + 2^{\frac{4}{3}} + 2^{\frac{8}{3}}\Big) \\
&\overset{5.21, d\geq 17}{<} \gamma\Big(T_{2d}^{(0)}\Big).
\end{aligned}$$

Somit folgt die Behauptung des Lemmas. □

Also wurde gezeigt, daß die Formel $T_{2d}^{(0)}$ von größerer Komplexität ist als die aufgesplittete Typ-0-Typformel $T_{split}^{(d)}$, falls $d \geq 4$. Deshalb ist aus Effiziengründen in Aufgabenstellung 5.20 die Berechnung der Lösung φ_2 genau dann der Lösung φ_1 vorzuziehen, wenn $d \geq 4$.

Die in 5.20 formulierte Aufgabe kann man nun erweitern, falls man nicht zwei parametrisierte Polynome gegeben hat, sondern 2^s Polynome vom Grad d mit ausschließlich reellen Nullstellen für ein $s \in \mathbb{N} \setminus \{0\}$. Es soll analog eine Charakterisierung gefunden werden, so daß das Produkt dieser Polynome vom Typ 0 ist. Wiederum kann man dazu aufgesplittete Typ-0-Typformeln verwenden, wobei man die darin auftretenden Typformeln vom Typ 0 erneut durch aufgesplittete Typ-0-Typformeln ersetzen kann. Genau dies macht der in Abbildung 5.3 vorgestellte neu-entworfene Algorithmus `SPLITTING`.

Lemma 5.24 *Der Algorithmus `SPLITTING` in Abbildung 5.3 terminiert und arbeitet korrekt.*

Beweis:

Termination: Beweis durch vollständige Induktion nach s. Im Fall $s = 0$ ist die Termination des Algorithmus trivial.

Terminiere nun der Algorithmus für ein $s \in \mathbb{N}$. Sei $F = (f_1, \ldots, f_{2^{s+1}})$. Falls man beim Aufruf von `SPLITTING` mit diesem F als Eingabeargument in den *else*-Fall trifft, so finden keine rekursiven Aufrufe statt. Tritt man aber in den *if*-Fall ein, so finden dort rekursive Aufrufe von `SPLITTING` statt, nämlich

$$SPLITTING\big((f_1, \ldots, f_{2^s})\big) \quad \text{und} \quad SPLITTING\big((f_{2^s+1}, \ldots, f_{2^{s+1}})\big).$$

Beide Eingabeargumente sind Tupel der Länge 2^s. Nach Induktionsannahme terminieren diese Aufrufe. Somit terminiert der Aufruf mit Eingabetupel F auch.

Beschreibung: Iterierte Anwendung des Splittingverfahrens

Eingabe: Ein Tupel $F := (f_1, \ldots, f_{2^s}) \in \mathbb{Z}[\underline{U}][X]^{2^s}$, $s \in \mathbb{N}$ beliebig, so daß $Grad_X(f_1) = \cdots = Grad_X(f_{2^s})$ und $\prod_{i=1}^{2^s} f_i$ ausschließlich reelle Nullstellen für alle Spezialisierungen der Parameter besitzt.

Output Eine Formel ψ in den Variablen $\underline{U}$, so daß für jede reelle Spezialisierung $\underline{u}$ der Parameter $\underline{U}$ gilt: $\psi(\underline{u}) \Leftrightarrow \tau(\prod_{i=1}^{2^s} f_i(\underline{u}, X)) = 0$.

procedure SPLITTING(F)
begin
$d := Grad_X(f_1)$
if $s \neq 0$ **and** $d \cdot 2^{s-1} \geq 4$ **do**
$\quad \psi := \bigvee_{i=-d2^{s-1}, i\neq 0}^{d2^{s-1}} \Big(T_{d2^{s-1}}^{(i)}\big(\prod_{j=1}^{2^{s-1}} f_j\big) \wedge T_{d2^{s-1}}^{(-i)}\big(\prod_{j=2^{s-1}+1}^{2^s} f_j\big)\Big) \vee$
$\qquad \big(SPLITTING((f_1, \ldots, f_{2^{s-1}})) \wedge SPLITTING((f_{2^{s-1}+1}, \ldots, f_{2^s}))\big)$
else
$\quad \psi := T_{d2^s}^{(0)}\big(\prod_{j=1}^{2^s} f_j\big)$
end
return ψ
end SPLITTING

Abbildung 5.4: Splitting-Algorithmus

<u>Korrektheit:</u> Der Algorithmus verwendet iteriert Satz 5.19. Beweis der Korrektheit durch vollständige Induktion nach s: Im Fall $s = 0$ ist die Korrektheit trivial, da für den einzigen Eintrag in F eine Typformel für Polynome mit Typ 0 von passendem Grad berechnet wird, in welche die Koeffizienten des Elements von F eingesetzt werden. Analog dazu folgt die Korrektheit im Fall $s = 1$, wenn man bei der Abarbeitung des Algorithmus in den *else*-Fall gelangt. Im Fall $s = 1$ mit Eintreten in den *if*-Teil folgt die Behauptung direkt aus Satz 5.19, da man bei den dann auftretenden rekursiven Aufrufen in den *else*-Teil kommt und somit direkt über eine aufgesplittete Typ-0-Typformel charakterisiert, wann $f_1 \cdot f_2$ vom Typ 0 ist.

Liefere nun für ein $s \in \mathbb{N}$ der Algorithmus ein korrektes Ergebnis. Sei nun das Eingabetupel

$$F = (f_1, \ldots, f_{2^{s+1}}).$$

Falls man beim Aufruf von `SPLITTING` mit Eingabeargument F in den *else*-Fall trifft, so ist die Korrektheit des Algorithmus offensichtlich. Trifft man in den *if*-Fall, so wird Satz 5.19 verwendet und im wesentlichen bis auf Einsetzen von gewissen Koeffizienten eine aufgesplittete Typ-0-Typformel berechnet, wobei die darin auftretenden Typ-0-Typformeln durch rekursive `SPLITTING`-Aufrufe ersetzt werden. Nach Induktionsannahme gelten aber für diese rekursiven Aufrufe mit

$$\begin{aligned}\psi_1 &:= SPLITTING\big((f_1,\ldots,f_{2^s})\big) \quad \text{und}\\ \psi_2 &:= SPLITTING\big((f_{2^s+1},\ldots,f_{2^{s+1}})\big),\end{aligned}$$

daß für jede Spezialisierung $\underline{u}$ der Parameter gilt:

$$\psi_1(\underline{u}) \overset{I.-A.}{\Leftrightarrow} \tau\Big(\prod_{i=1}^{2^s} f_i(\underline{u},X)\Big) = 0 \Leftrightarrow T^{(0)}_{d2^s}\Big(\prod_{i=1}^{2^s} f_i(\underline{u},X)\Big) \quad \text{und}$$

$$\psi_2(\underline{u}) \overset{I.-A.}{\Leftrightarrow} \tau\Big(\prod_{i=2^s+1}^{2^{s+1}} f_i(\underline{u},X)\Big) = 0 \Leftrightarrow T^{(0)}_{d2^s}\Big(\prod_{i=2^s+1}^{2^{s+1}} f_i(\underline{u},X)\Big)\ .$$

Mit Satz 5.19 folgt daraus sofort die Korrektheit des Algorithmus. □

Bemerkungen: Der Algorithmus `SPLITTING` berechnet im wesentlichen eine aufgesplittete Typ-0-Typformel, in welcher die auftretenden Typformeln für Polynome mit Typ 0 ggf. erneut durch aufgesplittete Typformeln ersetzt werden und in welche anschließend dann die Koeffizienten der auftretenden Polynome bzw. Polynomprodukte eingesetzt werden.
Die Abfrage, ob $d2^{s-1} \geq 4$, bezieht sich dabei auf den Komplexitätsvergleich von Lemma 5.23, in welchem festgestellt wurde, daß sich Splitten dann rechtfertigt, wenn

$$Grad_X\Big(\prod_{i=1}^{2^{s-1}} f_i\Big) = Grad_X\Big(\prod_{i=2^{s-1}+1}^{2^s} f_i\Big) = d\cdot 2^{s-1} \geq 4.$$

Also werden nur dann iteriert Typformeln vom Typ 0 durch aufgesplittete Typ-0-Typformeln ersetzt, wenn die auftretenden Grade groß genug sind. Somit wird das Ergebnis aus Lemma 5.23 so oft wie möglich ausgenutzt.

Aus Lemma 5.23 folgt also offensichtlich per Induktion, daß `SPLITTING` besser geeignet ist, um eine Formel zu bestimmen, welche charakterisiert, wann ein Produkt von 2^s parametrisierten Polynomen mit Grad d vom Typ 0 ist, als wenn man diese charakterisierende Formel mittels einer Typ-0-Typformel für Polynome vom Grad $d \cdot 2^s$ bestimmt.

Nun wird noch eine konkrete Anwendung des Algorithmus `SPLITTING` gezeigt. Im Algorithmus `QENF` aus Kapitel 3.4 müssen, falls die Gleichungen von zu eliminierenden Eingabeformeln nulldimensionale Ideale bilden, Typformeln für Polynome mit Typ 0 berechnet werden. Dabei wird zur Quantorenelimination in diesem Fall Satz 3.13 angewandt. Mit den dortigen Notationen wird das Produkt χ von $|E_{s,t}| = 2^s$ charakteristischen Polynomen $\chi_e \in \mathbb{Q}(\underline{U})[Y_{char}]$, $e \in E_{s,t}$, berechnet und daraus eine quantorenfreie Formel ψ', so daß ψ' genau dann erfüllt ist, wenn χ nicht den Typ 0 besitzt. Dabei werden nur Spezialisierungen betrachtet, so daß unter diesen keine Nenner in den einzelnen Koeffizienten der Polynome χ_e verschwinden. Bisher wurde ψ' so bestimmt, daß man zu

$$\chi = \prod_{e \in E_{s,t}} \chi_e = Y_{char}^d + \sum_{i=0}^{d-1} a_i Y_{char}^i \in \mathbb{Q}(\underline{U})[Y_{char}]$$

eine Typformel $T_d^{(0)}(C_0, \ldots, C_{d-1})$ für Polynome vom Grad d mit Typ 0 bestimmt, und daraus die Formel $\psi' := \neg T_d^{(0)}(a_0, \ldots, a_{d-1})$ berechnet, wobei darin noch alle atomaren Formeln derart äquivalent umgeformt werden müssen, so daß alle Terme Elemente von $\mathbb{Z}[\underline{U}]$ sind. Es liegt jedoch hier genau der Fall vor, daß man charakterisieren will, wann das Produkt der χ_e Typ 0 besitzt. Man beachte, daß die Polynome χ_e jeweils den selben Grad besitzen. Sei $E_{s,t} = \{e_1, \ldots, e_{2^s}\}$. Mit den in diesem Abschnitt gezeigten Ergebnissen ist es somit besser, die einzelnen charakteristischen Polynome χ_e zu berechnen und daraus das Ergebnis ψ von `SPLITTING`$((\chi_{e_1}, \ldots, \chi_{e_{2^s}}))$ zu bestimmen, wobei man hier erlauben darf, daß das Eingabeargument von `SPLITTING` ein Tupel aus $\mathbb{Q}(\underline{U})[X]^{2^s}$ ist. Denn bei der Berechnung von ψ läßt man wiederum vorerst die Division $\div$ als Operation in den atomaren Formeln zu, hierbei aber gesichert ist, daß alle auftretenden Nenner unter den betrachteten, zulässigen Spezialisierungen nicht verschwinden. Somit sind ψ und $T_d^{(0)}(a_0, \ldots, a_{d-1})$ zueinander äquivalent. Nach der Berechnung von ψ wird wieder jede darin enthaltene atomare Formel äquivalent umgeformt, so daß jeder Term in einer atomaren Formel ein Element von $\mathbb{Z}[\underline{U}]$ ist. Sei das Ergebnis dieser Äquivalenzumformung ψ_0. Damit stellt die Formel $\neg\psi_0$ ebenfalls eine Charakterisierung dar, wann das Produkt $\prod_{e \in E_{s,t}} \chi_e$ nicht vom Typ 0 ist.

Folgendes Beispiel zeigt, daß sich in der Praxis die Verwendung von `SPLITTING` zur Berechnung dieser den Typ 0 charakterisierenden Formeln bei Aufrufen von `QENF` als sehr gut erweist:

Beispiel 5.25 *Sei*

$$\varphi := \exists x \left(x^2 - 1 > 0 \wedge x^3 - 1 > 0 \wedge x^2 - x + a = 0\right).$$

Das Ergebnis der Quantorenelimination von φ mittels `RRCQE` erhält man unter Verwendung des Algorithmus `SPLITTING` in ca. 0,25 Sekunden. Dabei besteht das Eliminationsergebnis aus 113 atomaren Formeln.
Ohne Verwendung von `SPLITTING` wird der Quantor „$\exists x$" von φ in 15,4 Sekunden eliminiert. Hier besteht das Eliminationsergebnis aus sogar 201 atomaren Formeln. $\diamond$

An diesem Beispiel erkennt man deutlich den Effizienzgewinn bei der Berechnungszeit als auch an der Komplexität des Ergebnisses, wenn man aufgesplittete Typ-0-Typformeln verwendet. Weitere Beispiele mit und ohne deren Verwendung sowie weitere Untersuchungen dazu finden sich in Kapitel 7.2.

Es gibt noch einen anderen wesentlichen Aspekt, welcher die Verwendung von aufgesplitteten Typ-0-Typformeln rechtfertigt. Die Ergebnisformeln bei der Elimination mittels `RRCQE` besitzen atomare Formeln mit z.T. sehr großen Termen, d.h. Terme mit hohem Totalgrad und großen Koeffizienten. Dieser Effekt tritt bereits bei Eingabeformeln auf, deren Terme sehr einfach gestaltet sind. Da die Eliminationsergebnisse z.T. als Argumente an rekursive Aufrufe übergeben werden, ist es somit wünschenswert, die Ergebnisformeln von geringer Komplexität zu halten. Dieser Aspekt wird hier folgendermaßen berücksichtigt:

Wenn man das Produkt von parametrisierten Polynomen mit einer Hauptvariablen bildet, so werden in der Praxis i. a. die parametrischen Koeffizienten des Produktpolynoms größer und umfangreicher als die der Faktoren. Deswegen sind die Koeffizienten des Produkts χ der charakteristischen Polynome χ_e aus Satz 3.13 i. a. von größerer Komplexität als die Koeffizienten der einzelnen charakteristischen Polynome χ_e. Somit ist es sicher sinnvoll, möglichst wenig charakteristische Polynome χ_e miteinander zu multiplizieren und die Koeffizienten eines solchen Produkts in eine Typformel einzusetzen, so daß die auftretenden Terme in dieser entstehenden Formel als klein zu erwarten sind. Der Algorithmus `SPLITTING` trägt diesem Aspekt Rechenschaft, denn es werden maximal Produkte mit 2^{s-1} Faktoren berechnet, während χ nach Satz 3.13 aus 2^s Faktoren besteht. In Kapitel 7.2 werden in Beispielen die Ergebnisformeln anhand verschiedener Komplexitätsmaße verglichen, abhängig davon, ob man das Splitting-Verfahren anwendet oder nicht.

In der Praxis zeigte sich, daß die Verwendung von aufgesplitteten Typ-0-Typformeln zu einem sehr deutlichen Effizienzgewinn bzgl. der Berechnungszeiten und zur Verringerung der Komplexität der Ergebnisformeln führt. Dies begründet sich dadurch, daß die Komplexität von Typformeln $T_{2d}^{(0)}$ mit steigendem d viel stärker wächst im Gegensatz zu aufgesplitteten Typ-0-Typformeln $T_{split}^{(d)}(\underline{a}, \underline{b})$, obwohl zur Berechnung von aufgesplitteten Typ-0-Typformeln Typformeln für Polynome von verschiedenen Typen berechnet werden müssen. Vor allem wirkt sich die Verwendung von `SPLITTING` positiv auf die Berechnungszeiten aus, da der zeitintensiven Berechnung von Typformeln höherer Grade ausgewichen wird. In der Praxis zeigte sich nämlich, daß eine Typformelbestimmung sprunghaft zeitintensiver wurde, wenn der Grad größer wurde. Dies erklärt den Effizienzgewinn bei den Berechnungszeiten.

Es stellt sich nun noch die Frage, ob man den Typ eines Produktes von Polynomen nicht vollständig über die Typen der einzelnen Faktoren charakterisieren soll. D.h.: Seien $f_1, \ldots, f_k \in \mathbb{Z}[\underline{U}][X]$ von gleichem Grad d mit ausschließlich reellen Nullstellen gegeben. Dann gilt aufgrund Satz 3.9 offensichtlich die Äquivalenz

$$T^{(0)}_{d \cdot k}\Big(\prod_{i=1}^{k} f_i\Big) \Longleftrightarrow \bigvee_{i_1,\dots,i_k \in \{-d,\dots,d\}, \sum_{j=1}^{k} i_j = 0} \Big(\bigwedge_{i=1}^{k} T^{(i_1)}_{d}(f_i)\Big).$$

Auf der rechten Seite dieser Äquivalenz wird also eine vollständige Fallunterscheidung nach dem Typ der einzelnen Faktoren gemacht. Um genauer zu untersuchen, wann eine solche vollständige Fallunterscheidung besser geeignet ist zur Charakterisierung des Typs eines Produkts, müsste man die Komplexitäten der Typformeln $T^{(i)}_{d}$ genauer untersuchen. In der Praxis zeigte sich aber wider Erwarten, daß die Berechnungszeiten und die Anzahl der atomaren Formeln in den Ergebnisformeln im Vergleich zur Anwendung des Algorithmus `SPLITTING` stiegen, wenn man den Typ 0 eines Polynomproduktes durch vollständige Fallunterscheidung nach den Typen der einzelnen Faktoren charakterisierte. Diese Beobachtung ergibt sich auch dadurch, daß nicht Typformeln von beliebig großem Grad in akzeptabler Zeit berechnet werden können, so daß keine intensiveren Vergleichsmöglichkeiten zur Verfügung stehen, ob bzw. wann eine vollständige Fallunterscheidung besser geeignet wäre. Es liegt zwar für genügend großes d und k die Vermutung nahe, daß eine vollständige Fallunterscheidung nach den Typen der einzelnen Faktoren günstiger ist als die bisherigen Charakterisierungen, jedoch diese großen d- und k-Werte für eine praktikable Anwendung viel zu groß sind, als daß mit Versuchen diese Vermutung überprüft werden kann.

Kapitel 6

QE in der Roboterkinematik

Dieses Kapitel zeigt Anwendungen der Hermiteschen Quantoreneliminati-on anhand von Beispielen in der Roboterkinematik. Es werden dafür typische Problemstellungen formalisiert und gelöst. Auf eine detaillierte Herleitung aller roboterkinematischen Grundlagen wird dazu verzichtet. Details, Roboter-Beispiele, sowie weiterführende Informationen finden sich in [3].

6.1 Kurzeinführung in die Roboterkinematik

Ein **Roboter** besteht aus einer Folge von Gliedern, die durch Gelenke miteinander verbunden sind, wobei der Fuß des Roboters fest im Raum verankert ist. Die Gelenke können von verschiedenen Arten sein, z.B.:

- Drehgelenke
- Torsionsgelenke
- Translationsgelenke

Abbildung 6.1 zeigt einen Beispielroboter mit diesen drei Gelenken. Alle Gelenke eines Roboters können verschiedene Einstellungen besitzen. Bei Dreh- und Torsionsgelenken wird diese Einstellung mit einem Winkel identifiziert, der je nach Robotertyp beschränkt sein kann. Bei Translationsgelenken wird die Einstellung mit einer variablen Armlänge/Strecke identifiziert. Der **Effektor** eines Roboters befindet sich am Ende des letzten Gliedes des Roboters, an dem sich z.B. ein Bohrer, Schweißer, etc. befinden kann. Der Roboter hat die Aufgabe, den Effektor in eine vorgegebene Position zu bringen, so daß der Effektor dort eine bestimmte Tätigkeit verrichten kann. Die Abbildung 6.3 zeigt einen weiteren Roboter aus der Industrie. Der **Arbeitsraum** eines Roboters besteht aus allen Punkten im dreidimensionalen Raum, die

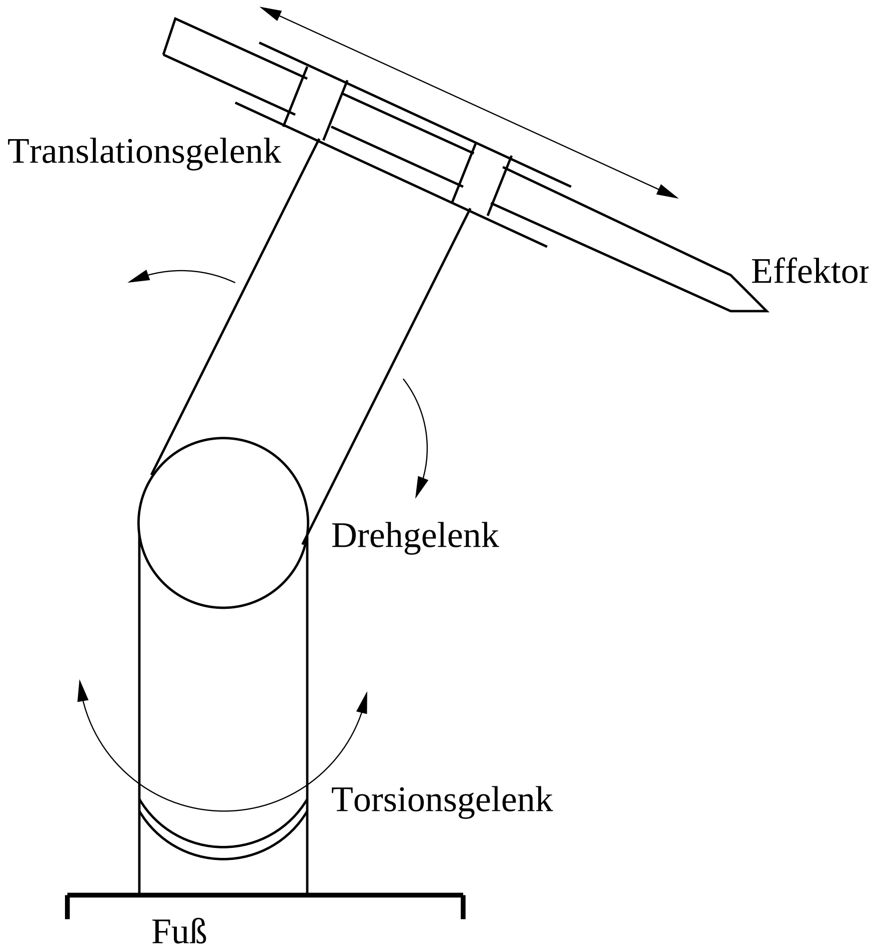

Abbildung 6.1: Beispiel-Roboter

der Effektor erreichen kann. Für weitergehende Informationen wird auf die Literatur verwiesen.

Wenn man nun ein einzelnes Gelenk eines Roboters betrachtet, so können für dieses Gelenk verschiedene Einstellungen gemacht werden. Durch Veränderungen dieser Einstellungen ändert sich ggf. die Position des Effektors im Raum, welche man mathematisch in Abhängigkeit von den Einstellungen der einzelnen Gelenke mit Hilfe von sog. *Transformationsmatrizen* beschreiben kann. Dazu werden Drehungen in $\mathbb{R}^3$ benötigt.

Alle Drehungen in $\mathbb{R}^3$ werden mittels speziellen orthogonalen Matrizen aus $SO(3) \subseteq \mathbb{R}^{3\times 3}$ beschrieben. Jede Drehung kann man aus 3 einzelnen Drehungen um eine Koordinatenachse zusammensetzen: Eine Drehung um einen Winkel α um die x-Achse, gefolgt von einer Drehung um einen Winkel β um die y-Achse, gefolgt von einer Drehung um einen Winkel γ um die z-Achse. Die Gesamtdrehung $R(\alpha,\beta,\gamma)$ kann als Produkt dieser drei einzelnen Drehungen dargestellt werden:

$$R(\alpha,\beta,\gamma) = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix}$$

Das Matrizenprodukt $R(\alpha,\beta,\gamma)$ ist wiederum eine spezielle orthogonale Matrix aus $SO(3)$ und beschreibt nun eine Drehung im dreidimensionalen Raum. $R(\alpha,\beta,\gamma)$ wird Rotationsmatrix genannt. Der Vollständigkeit halber sei erwähnt, daß dies nur eine Möglichkeit von vielen ist, Drehungen zu beschreiben. Dies erlaubt nun folgende Definition:

Definition 6.1 (Transformationsmatrix) *Eine* 4×4*-Matrix heißt* Transformationsmatrix, *falls sie folgende Struktur besitzt:*

$$\left(\begin{array}{ccc|c} & & & x \\ & R(\alpha,\beta,\gamma) & & y \\ & & & z \\ \hline 0 & 0 & 0 & 1 \end{array}\right)$$

wobei $R(\alpha,\beta,\gamma)$ *eine Rotationsmatrix in den Variablen* α,β,γ *ist.*

Jeder Übergang von einem Gelenk eines Roboters zum nächsten wird mit Hilfe von Transformationsmatrizen beschrieben. Ein Koordinatensystem besteht aus einer Menge von drei Vektoren, die zusammen den reellen Vektorraum $\mathbb{R}^3$ aufspannen und paarweise orthogonal zueinander sind. Man geht davon aus, daß in jedem Gelenk, am Fuß des Roboters und am Effektor jeweils ein Koordinatensystem sitzt, welche jeweils bis auf Drehung und Translation in $\mathbb{R}^3$ identisch sind. Jede Transformationsmatrix stellt eine Bewegung eines Koordinatensystems von einem Gelenk zum nächsten dar. Der

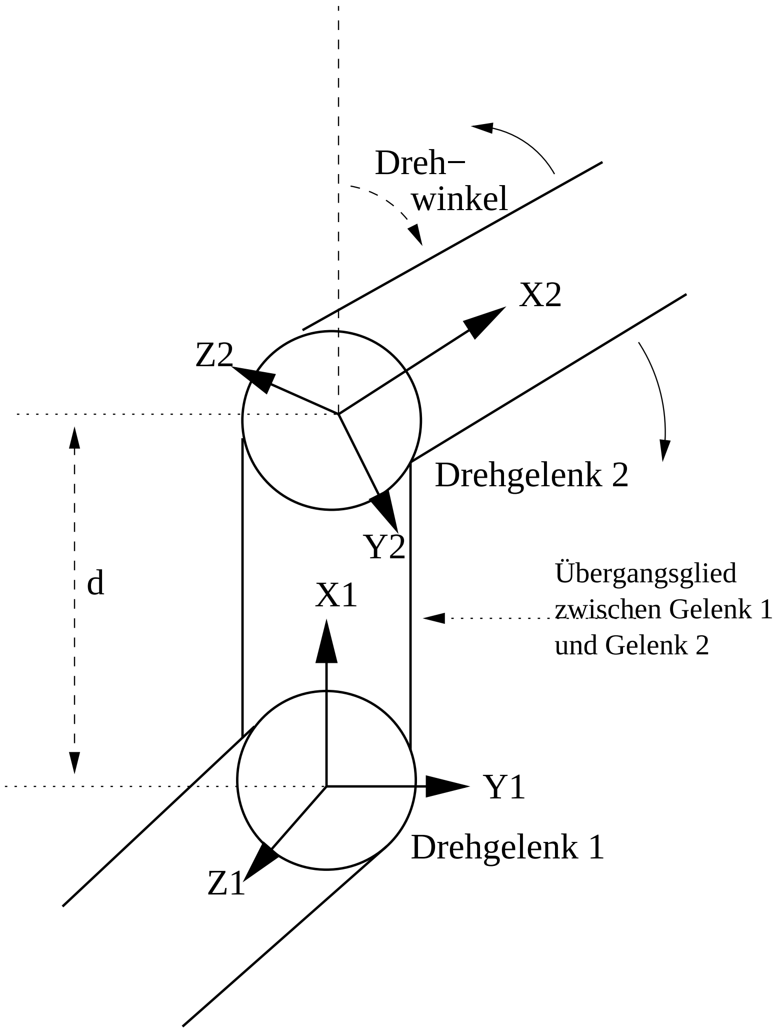

Abbildung 6.2: Skizze eines Robotergelenks

Aufbau einer Transformationsmatrix wird anhand eines Übergangs zwischen zwei Drehgelenken wie in Abbildung 6.2 erklärt.

Im Drehgelenk 1 ist das zugehörige Koordinatensystem $K_1 := \{X1, Y1, Z1\}$, bzgl. welchem nun Drehungen und Translationen durchgeführt werden. Das in Abbildung 6.2 skizzierte Drehgelenk 2 kann also Drehungen um einen Winkel θ um die Z1-Achse durchführen, allerdings keine Drehungen um die X1- und Y1-Achse. Je nach eingestelltem θ wird K_1 um θ gedreht und dann in Richtung des Übergangsgliedes zu Drehgelenk 2 verschoben, so daß man nun ein Koordinatensystem $K_2 := \{X2, Y2, Z2\}$ erhält, welches in Drehgelenk 2 sitzt. K_2 geht also aus K_1 durch Drehung und Translation hervor. Wenn T die zu diesem Gelenkübergang zugehörige Transformationsmatrix ist, so beschreibt die in T enthaltene Rotationsmatrix genau das eben genannte Drehverhalten in Abhängigkeit von θ. Der Vektor $(x\ y\ z)^T$ innerhalb von T beschreibt genau die Translation von K_1 entlang des Übergangsgliedes, in Abhängigkeit von der Übergangsgliedlänge d.

Die letzte Zeile einer Transformationsmatrix spielt nur aus Vereinfachungsgründen für Berechnungen eine Rolle, keine Technische. Zusammengefasst beschreibt also eine Transformationsmatrix die Lage eines Koordinatensystems, welches durch Drehung und Translation eines Ausgangskoordinatensystems entsteht.

Mit anderen Worten: Eine Transformationsmatrix kann man als affine Abbildung identifizieren:

$$\mathbb{R}^3 \to \mathbb{R}^3 : x \mapsto R(\alpha, \beta, \gamma) \cdot x + y \ .$$

Eine Transformationsmatrix zum Übergang zwischen Gelenk $i-1$ und Gelenk i wird mit T_i^{i-1} notiert. Der Index „0" bezeichnet dabei den Fuß des Roboters und der größte Index den Effektor. Man beachte dabei, daß auch im Fuß des Roboters und im Effektor jeweils ein Koordinatensystem sitzt.

Als **Weltkoordinatensystem** wird das kanonische Koordinatensystem bezeichnet, welches sich am Fuß des Roboters befindet. Wenn man nun zu jedem Gelenkübergang eines Roboters Transformationsmatrizen besitzt, so kann man die Lage des Koordinatensystems an der Effektorposition relativ zum Weltkoordinatensystem anhand der Einstellungen der einzelnen Gelenke wie folgt bestimmen:

Satz 6.2 *Seien $T_1^0, \ldots, T_n^{n-1}$ Transformationsmatrizen eines Roboters, wobei die Transformation T_1^0 als erste, T_2^1 als zweite, usw. ausgeführt wird. Dann ist das Produkt*

$$T_n^0 := T_1^0 \cdots T_n^{n-1}$$

wieder eine Transformationsmatrix. T_n^0 beschreibt die Gesamttransformation, d.h. sie beschreibt genau die Lage des Koordinatensystems im Effektor *relativ zum Weltkoordinatensystem in Abhängigkeit von den Gelenkeinstellungen.*

Beweis: Siehe [3], Seite 75. □

Es wird jetzt ein Roboter mit seinen Transformationsmatrizen vorgestellt: der Yasukawa Motoman L-3.

Der Yasukawa Motoman L-3(Abbildung 6.3) besteht aus 3 Dreh- und 2 Torsionsgelenken. Seine Transformationsmatrizen haben folgende Gestalt:

$$T_1^0 := \begin{pmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_2^1 := \begin{pmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin\theta_2 & -\cos\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_3^2 := \begin{pmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & l_2 \\ \sin\theta_3 & \cos\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_4^3 := \begin{pmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & l_3 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_5^4 := \begin{pmatrix} \cos\theta_5 & -\sin\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

mit $l_2, l_3 \in \mathbb{R}_>$. l_2 und l_3 entsprechen Armlängen des Roboters. Als abkürzende Notationen werden hier verwendet:

$$\begin{aligned} c_i &:= \cos\theta_i \quad \text{für } i \in \{1,\dots,5\}, \\ s_i &:= \sin\theta_i \quad \text{für } i \in \{1,\dots,5\}, \\ \underline{c} &:= (c_1, c_2, c_3, c_4, c_5), \\ \underline{s} &:= (s_1, s_2, s_3, s_4, s_5). \end{aligned}$$

Das Produkt dieser Matrizen ist

$$T_5^0 := T_1^0 \cdot T_2^1 \cdot T_3^2 \cdot T_4^3 \cdot T_5^4 = \begin{pmatrix} r_{11}(\underline{c},\underline{s}) & r_{12}(\underline{c},\underline{s}) & r_{13}(\underline{c},\underline{s}) & p_x(\underline{c},\underline{s}) \\ r_{21}(\underline{c},\underline{s}) & r_{22}(\underline{c},\underline{s}) & r_{23}(\underline{c},\underline{s}) & p_y(\underline{c},\underline{s}) \\ r_{31}(\underline{c},\underline{s}) & r_{32}(\underline{c},\underline{s}) & r_{33}(\underline{c},\underline{s}) & p_z(\underline{c},\underline{s}) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

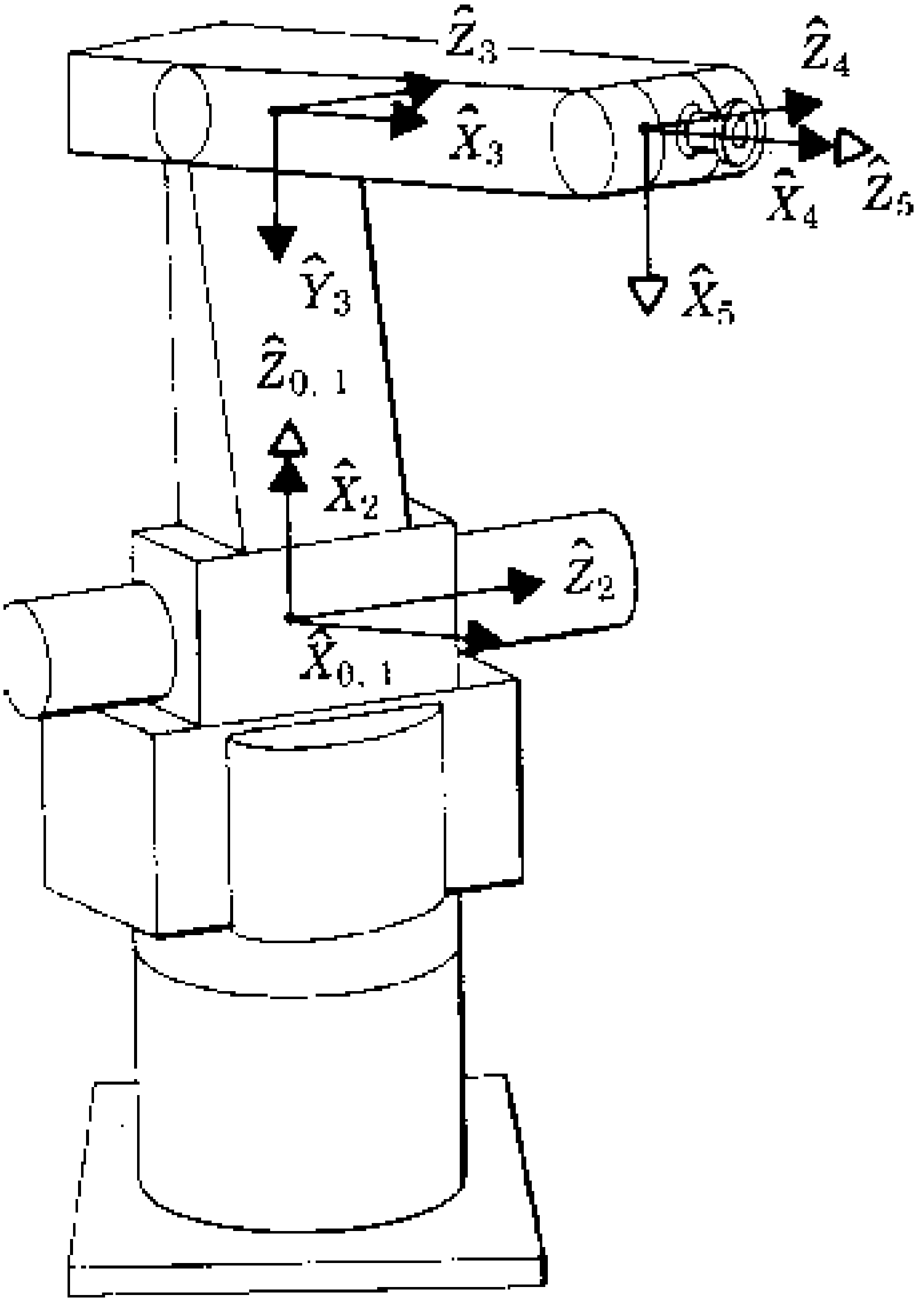

Abbildung 6.3: Yasukawa Motoman L-3

mit

$$
\begin{aligned}
r_{11}(\underline{c},\underline{s}) &= c_1((c_2c_3 - s_2s_3)c_4 - (c_2s_3 + c_3s_2)s_4)c_5 - s_1s_5,\\
r_{21}(\underline{c},\underline{s}) &= s_1((c_2c_3 - s_2s_3)c_4 - (c_2s_3 + c_3s_2)s_4)c_5 + c_1s_5,\\
r_{31}(\underline{c},\underline{s}) &= -((c_2c_3 - s_2s_3)s_4 + c_4(c_2s_3 + c_3s_2))c_5,\\
\\
r_{12}(\underline{c},\underline{s}) &= -c_1((c_2c_3 - s_2s_3)c_4 - (c_2s_3 + c_3s_2)s_4)s_5 - s_1c_5,\\
r_{22}(\underline{c},\underline{s}) &= -s_1((c_2c_3 - s_2s_3)c_4 - (c_2s_3 + c_3s_2)s_4)s_5 + c_1c_5,\\
r_{32}(\underline{c},\underline{s}) &= ((c_2c_3 - s_2s_3)s_4 + c_4(c_2s_3 + c_3s_2))s_5,\\
\\
r_{13}(\underline{c},\underline{s}) &= c_1((c_2c_3 - s_2s_3)s_4 + c_4(c_2s_3 + c_3s_2)),\\
r_{23}(\underline{c},\underline{s}) &= s_1((c_2c_3 - s_2s_3)s_4 + c_4(c_2s_3 + c_3s_2)),\\
r_{33}(\underline{c},\underline{s}) &= (c_2c_3 - s_2s_3)c_4 - (c_2s_3 + c_3s_2)s_4,\\
\\
p_x(\underline{c},\underline{s}) &= c_1(l_2c_2 + l_3(c_2c_3 - s_2s_3)),\\
p_y(\underline{c},\underline{s}) &= s_1(l_2c_2 + l_3(c_2c_3 - s_2s_3)),\\
p_z(\underline{c},\underline{s}) &= -l_2s_2 - l_3(c_2s_3 + c_3s_2).
\end{aligned}
$$

Die inverse Kinematik behandelt u. a. das Problem, ob der Effektor in gewisse vorgegebene Positionen im dreidimensionalen Raum gebracht werden kann. Es werden nun folgende Fragestellungen der inversen Roboterkinematik anhand des Yasukawa Motoman L-3 untersucht, welche mit Hilfe Hermitescher Quantorenelimination gelöst werden sollen:

1. Arbeitsraumprobleme:
   (a) Liegt ein Punkt $\underline{x} \in \mathbb{R}^3$ bzw. ein vorgegebener Weg in $\mathbb{R}^3$ im Arbeitsraum?
   (b) Kann der Effektor einem vorgegebenem Weg in $\mathbb{R}^3$ folgen, so daß der Effektor in jedem Punkt des Weges eine vorgegebene Orientierung einnimmt, d.h. daß die Drehung des Koordinatensystems im Effektor relativ zum Weltkoordinatensystem für jeden Punkt des Weges vorgegeben ist?
2. Sei ein Punkt $\underline{x} \in \mathbb{R}^3$ gegeben. Wie müssen gewisse Armlängen des Roboters beschaffen sein, damit der Punkt $\underline{x}$ im Arbeitsraum liegt?

Um diese Probleme in zulässige Formeln erster Stufe umzuwandeln, muß man die auftretenden Kosinus- und Sinusfunktionen kodieren: Zu jedem auftretenden Winkel θ gibt man sich zwei Variablen c_θ und s_θ vor, welche den Werten $\cos\theta$ bzw. $\sin\theta$ entsprechen und für welche die charakteristische Kreisgleichung $c_\theta^2 + s_\theta^2 = 1$ gelten soll. Damit ist durch c_θ und s_θ der Winkel

θ eindeutig festgelegt. Deshalb werden ab jetzt c_i und s_i als echte Variablen betrachtet und nicht mehr als Kurzform für Kosinus- und Sinuswerte von θ_i. Diese Variablen müssen somit folgende Bedingung erfüllen:

$$\hat{\varphi} := \bigwedge_{i=1}^{5} (c_i^2 + s_i^2 = 1) \ .$$

Die aufgeführten Fragestellungen werden als Formel erster Stufe kodiert. Die Hermitesche Quantorenelimination eignet sich gut, um diese Probleme zu behandeln, da es i. a. nur endlich viele Einstellungen der Winkel gibt, so daß am Effektor ein vorgegebener Punkt anliegt. Dadurch sind bei Anwendung der Hermiteschen Quantorenelimination auf diese Probleme die auftretenden Ideale i. a. als nulldimensional zu erwarten.

6.2 Arbeitsraumprobleme

In diesem Abschnitt wird die Hermitesche Quantorenelimination benutzt, um festzustellen, ob der Effektor in eine bestimmte vorgegebene Position und Lage gebracht werden kann. Seien ab jetzt dazu die Längen l_2 und l_3 fest vorgegeben mit $l_2 = l_3$=1.

Aufgabe 1: Sei ein Weg in $\mathbb{R}^3$ gegeben, etwa

$$w : [0,1] \to \mathbb{R}^3 : t \mapsto (w(t)_1, w(t)_2, w(t)_3) \ .$$

Für welche Werte von t liegt das Bild $w(t)$ im Arbeitsraum des Motoman?

Lösung: Der Effektor erreicht genau dann einen Punkt $\underline{x} \in \mathbb{R}^3$, wenn das Weltkoordinatensystem zum Punkt $\underline{x}$ bewegt werden kann. Man beachte, daß p_x, p_y, p_z nur von den Werten $c_1, s_1, c_2, s_2, c_3, s_3$ abhängen. Die zugehörige Formulierung des Problems ist damit:

$$\begin{aligned}\exists c_1 \exists s_1 \exists c_2 \exists s_2 \exists c_3 \exists s_3 \big(& c_1^2 + s_1^2 = 1 \wedge c_2^2 + s_2^2 = 1 \wedge c_3^2 + s_3^2 = 1 \wedge \\ & w(t)_1 = c_1(c_2 + (c_2c_3 - s_2s_3)) \wedge w(t)_2 = s_1(c_2 + (c_2c_3 - s_2s_3)) \wedge \\ & w(t)_3 = -s_2 - (c_2s_3 + c_3s_2) \wedge t \geq 0 \wedge t \leq 1\big) \ .\end{aligned}$$

Das Ergebnis der Hermiteschen Quantorenelimination angewandt auf diese Formel ist dann eine quantorenfreie Formel $\psi(t)$, so daß für alle $t_0 \in [0,1]$ gilt:

$$\psi(t_0) \quad \Leftrightarrow \quad \text{„}w(t_0) \text{ liegt im Arbeitsraum des Motoman.“}$$

Man beachte, daß man mit dieser Aufgabenstellung insbesondere feststellen kann, ob ein einzelner Punkt $\underline{x} \in \mathbb{R}^3$ sich im Arbeitsraum des Roboters befindet. Man wählt dazu den Weg, der konstant auf $\underline{x}$ abbildet, wobei dann die Formeln $t \geq 0$ und $t \leq 1$ gestrichen werden können.

Einige Beispiele demonstrieren nun konkrete Anwendungen dieser Aufgabenstellung:

Beispiel 6.3 *Liegt der Punkt* $(1,2,1) \in \mathbb{R}^3$ *im Arbeitsraum des Motoman? Die zugehörige Problemformulierung als Formel erste Stufe ist damit*

$$\begin{aligned}&\exists c_1 \exists s_1 \exists c_2 \exists s_2 \exists c_3 \exists s_3 \big(c_1^2 + s_1^2 = 1 \wedge c_2^2 + s_2^2 = 1 \wedge c_3^2 + s_3^2 = 1 \wedge \\ &\quad 1 = c_1(c_2 + (c_2c_3 - s_2s_3)) \wedge 2 = s_1(c_2 + (c_2c_3 - s_2s_3)) \wedge \\ &\quad\quad 1 = -s_2 - (c_2s_3 + c_3s_2)\big) \ .\end{aligned}$$

Das Ergebnis der Hermiteschen Quantorenelimination angewandt auf diese Formel ist false. *Somit kann der Motoman den Punkt* $(1,2,1)$ *nicht erreichen.* ◇

Beispiel 6.4 *Liegt der Punkt* $(1,1,0) \in \mathbb{R}^3$ *im Arbeitsraum des Motoman? Analog zum Beispiel 6.3 ist das Ergebnis der Hermiteschen Quantorenelimination* true, *d.h. der Motoman kann den Punkt* $(1,1,0)$ *erreichen.* ◇

Das nächste Beispiel zeigt eine kleine Abänderung der letzten beiden Beispiele. Alle auftretenden Winkel eines Roboters können i. a. nicht beliebig gewählt werden, sondern unterliegen mechanischen Restriktionen.

Beispiel 6.5 *Liegt der Punkt* $(0,1.5,1) \in \mathbb{R}^3$ *im Arbeitsraum des Roboters, wobei nachfolgende Winkelrestriktionen einzuhalten sind?*

Restriktionen: $\cos\theta_1 > 0.5$ *und* $\sin\theta_2 > 0.5$ *und* $\sin\theta_3 > 0.5$

Die Problemformulierung ist dann

$$\begin{aligned}&\exists c_1 \exists s_1 \exists c_2 \exists s_2 \exists c_3 \exists s_3 \big(c_1^2 + s_1^2 = 1 \wedge c_2^2 + s_2^2 = 1 \wedge c_3^2 + s_3^2 = 1 \wedge \\ &\quad 0 = c_1(c_2 + (c_2c_3 - s_2s_3)) \wedge 1.5 = s_1(c_2 + (c_2c_3 - s_2s_3)) \wedge \\ &\quad\quad 1 = -s_2 - (c_2s_3 + c_3s_2) \wedge c_1 > 0.5 \wedge s_2 > 0.5 \wedge s_3 > 0.5\big) \ .\end{aligned}$$

Das Ergebnis der Hermiteschen Quantorenelimination angewandt auf diese Formel ist false, *d.h. der Punkt ist mit den vorgegebenen Winkelrestriktionen nicht erreichbar. Vernachlässigt man diese Restriktionen, so liegt der Punkt* $(0,1.5,1)$ *jedoch im Arbeitsraum.* ◇

Die Berechnungszeiten zu diesen letzten drei Beispielen lagen bei nur 10 Millisekunden. Am nächsten Beispiel wird gezeigt, wie man charakterisieren kann, welche Punkte eines Weges im Arbeitsraum des Roboters liegen.

Beispiel 6.6 *Sei folgender Weg in* $\mathbb{R}^3$ *gegeben:*

$$w : [0,1] \to \mathbb{R}^3 : t \mapsto (t-1, 2t, t^2) \ .$$

Für welche Werte von t *liegt das Bild* $w(t)$ *im Arbeitsraum des Motoman?*

Das Ergebnis der Hermiteschen Quantoreneliminantion in diesem Fall wird in ca. 1,1 Sekunden berechnet, besteht aus 35 atomaren Formeln und ist:

$t-1 \leq 0 \wedge t \geq 0 \wedge (t^4+5t^2-2t+1 \neq 0 \wedge 5t^2-2t+1 \neq 0 \wedge t-1 < 0 \wedge t > 0 \wedge \neg(t^4+5t^2-2t-3 = 0 \wedge (25t^{20}-20t^{19}+14t^{18}-4t^{17}-1249t^{16}+2000t^{15}-600t^{14}-560t^{13}+22189t^{12}-48716t^{11}+51714t^{10}-39692t^9+7837t^8+17352t^7-30148t^6+25160t^5-15017t^4+6196t^3-1910t^2+372t-47 > 0 \vee 11t^2-6t+3 = 0) \vee t^4+5t^2-2t-3 \neq 0 \wedge (25t^{20}-20t^{19}+14t^{18}-4t^{17}-1249t^{16}+2000t^{15}-600t^{14}-560t^{13}+22189t^{12}-48716t^{11}+51714t^{10}-39692t^9+7837t^8+17352t^7-30148t^6+25160t^5-15017t^4+6196t^3-1910t^2+372t-47 > 0 \vee 5t^{20}-12t^{19}-90t^{18}+76t^{17}-805t^{16}+1416t^{15}+2236t^{14}-5072t^{13}+26585t^{12}-53444t^{11}+60614t^{10}-49212t^9+27589t^8-9016t^7-2048t^6+4840t^5-3697t^4+1716t^3-566t^2+116t-15 < 0 \wedge 55t^4-52t^3+38t^2-12t+3 < 0 \vee 5t^{20}-12t^{19}-90t^{18}+76t^{17}-805t^{16}+1416t^{15}+2236t^{14}-5072t^{13}+26585t^{12}-53444t^{11}+60614t^{10}-49212t^9+27589t^8-9016t^7-2048t^6+4840t^5-3697t^4+1716t^3-566t^2+116t-15 > 0 \wedge 55t^4-52t^3+38t^2-12t+3 > 0)) \vee t-1 = 0 \wedge \neg(t^{12}-32t^8+64t^6+464t^4-448t^2-768 > 0 \wedge t^4+4t^2-4 = 0 \vee t^4+4t^2-4 \neq 0 \wedge (t^{12}-32t^8+96t^6+624t^4-256t^2-256 > 0 \vee t^{12}-32t^8+64t^6+464t^4-448t^2-768 > 0)) \vee t = 0 \wedge \neg(t^8-4t^7+6t^6-36t^5+33t^4+160t^3-184t^2-80t+168 = 0 \wedge t^4+4t^2-8t-12 = 0 \wedge (4t^5+2t^4-44t^3+41t^2+22t-73 > 0 \vee t^4-4t^3+3t^2+2t-14 = 0) \vee t^4+4t^2-8t-12 \neq 0 \wedge (t^8-4t^7+6t^6-36t^5+33t^4+160t^3-184t^2-80t+168 > 0 \wedge t^6-2t^5+t^4-16t^3-8t^2+48t+36 > 0 \wedge t^4-4t^3+3t^2+2t-14 > 0 \vee t^6-2t^5+t^4-16t^3-8t^2+48t+36 > 0 \wedge 4t^5+2t^4-44t^3+41t^2+22t-73 > 0 \vee t^8-4t^7+6t^6-36t^5+33t^4+160t^3-184t^2-80t+168 < 0 \wedge t^6-2t^5+t^4-16t^3-8t^2+48t+36 > 0 \wedge t^4-4t^3+3t^2+2t-14 < 0)))$.
$\diamond$

Aufgabe 2: Sei wieder ein Weg

$$w : [0,1] \to \mathbb{R}^3 : t \mapsto (w(t)_1, w(t)_2, w(t)_3)$$

gegeben und außerdem zusätzlich eine parametrisierte Matrix

$$R(t) = \begin{pmatrix} r'_{11} & r'_{12} & r'_{13} \\ r'_{21} & r'_{22} & r'_{23} \\ r'_{31} & r'_{32} & r'_{33} \end{pmatrix} \in \mathbb{R}[t]^{3\times 3}$$

gegeben, so daß für jedes $t_0 \in [0,1]$ die Matrix $R(t_0)$ eine spezielle orthogonale Matrix ist. $R(t)$ soll die Orientierung des Effektors zum Zeitpunkt t vorgeben, d.h. die Drehung des Koordinatensystems an der Effektorposition relativ zum Weltkoordinatensystem zum Zeitpunkt t. Für welche $t_0 \in [0,1]$ existieren Einstellungen der Gelenke, so daß der Effektor derart zum Punkt $w(t_0)$ bewegt werden kann, daß dort das Koordinatensystem im Effektor die vorgegebene Orientierung $R(t_0)$ einhält?

Lösung: Die zusätzliche Vorgabe einer zu erfüllenden Orientierung des Effektors wird folgendermaßen formuliert:

$$\varphi := \bigwedge_{i=1}^{3} \bigwedge_{j=1}^{3} r'_{ij} = r_{ij}(\underline{c}, \underline{s}) \ .$$

Die vollständige Problemformulierung ist nun

$$\exists c_1 \exists s_1 \exists c_2 \exists s_2 \exists c_3 \exists s_3 \exists c_4 \exists s_4 \exists c_5 \exists s_5 \big(\hat{\varphi} \wedge \varphi \wedge w(t)_1 = p_x(\underline{c}, \underline{s}) \wedge \\ w(t)_2 = p_y(\underline{c}, \underline{s}) \wedge w(t)_3 = p_z(\underline{c}, \underline{s}) \wedge t \geq 0 \wedge t \leq 1) \big).$$

Wiederum kann man mit Hilfe der Hermiteschen Quantorenelimination, angewandt auf diese Formel, eine quantorenfreie Formel $\psi(t)$ bestimmen, so daß für alle $t_0 \in [0,1]$ gilt:

$\psi(t_0) \quad \Leftrightarrow \quad$ „Der Effektor kann so zum Punkt $w(t_0)$ bewegt werden, daß der Effektor dort die vorgegebene Orientierung $R(t_0)$ einnimmt."

Folgendes Beispiel zeigt eine konkrete Anwendung dieser Aufgabenstellung:

Beispiel 6.7 *Sei der Weg*

$$w : [0,1] \to \mathbb{R}^3 : t \mapsto (t-1, t+1, 2t-1)$$

gegeben. Sei $R(t)$ die 3×3-Einheitsmatrix. Das Ergebnis der Hermiteschen Quantorenelimination angewandt auf diesen konkreten Fall wird in ca. 59 Sekunden berechnet, besteht aus 56 atomaren Formeln und ist:

$t-1 \leq 0 \wedge t \geq 0 \wedge (6t^2-4t+3 \neq 0 \wedge 2t-1 \neq 0 \wedge t-1 < 0 \wedge t > 0 \wedge \neg(6t^2-4t-1 = 0 \wedge (144t^{12} - 768t^{11} + 2320t^{10} - 4128t^9 + 5512t^8 - 5120t^7 + 2380t^6 + 40t^5 - 2699t^4 + 2544t^3 - 1994t^2 + 744t - 251 > 0 \vee 5t^2 - 2t + 5 = 0) \vee 6t^2 - 4t - 1 \neq 0 \wedge (144t^{12} - 768t^{11} + 2320t^{10} - 4128t^9 + 5512t^8 - 5120t^7 + 2380t^6 + 40t^5 - 2699t^4 + 2544t^3 - 1994t^2 + 744t - 251 > 0 \vee 864t^{11} - 4680t^{10} + 9840t^9 - 16340t^8 + 19184t^7 - 18006t^6 + 13172t^5 - 7083t^4 + 3002t^3 - 668t^2 + 38t + 69 > 0 \wedge 5t^4 - 2t^3 + 10t^2 - 2t + 5 < 0 \vee 864t^{11} - 4680t^{10} + 9840t^9 - 16340t^8 + 19184t^7 - 18006t^6 + 13172t^5 - 7083t^4 + 3002t^3 - 668t^2 + 38t + 69 < 0 \wedge 5t^4 - 2t^3 + 10t^2 - 2t + 5 > 0)) \vee 6t^2 - 4t + 3 \neq 0 \wedge t = 0 \wedge \neg(32832t^{18} - 199296t^{17} + 1134432t^{16} - 3685632t^{15} + 8482800t^{14} - 14828928t^{13} + 20679792t^{12} - 23623968t^{11} + 22409972t^{10} - 17740744t^9 + 11641246t^8 - 6203784t^7 + 2544403t^6 - 692060t^5 + 38494t^4 + 70736t^3 - 38841t^2 + 10188t - 1242 = 0 \wedge (2944t^{18} - 50560t^{17} + 226624t^{16} - 534528t^{15} + 724560t^{14} - 336800t^{13} - 1125696t^{12} + 3577440t^{11} - 6300088t^{10} + 8146344t^9 - 8368040t^8 + 7004936t^7 - 4829373t^6 + 2734054t^5 - 1256017t^4 + 455832t^3 - 124379t^2 + 23106t - 2259 > 0 \vee 28t^8 - 72t^7 + 230t^6 - 378t^5 + 475t^4 - 409t^3 + 240t^2 - 89t + 15 = 0) \wedge (308t^8 - 856t^7 + 1312t^6 - 1336t^5 + 889t^4 - 362t^3 + 57t^2 + 14t - 6 = 0 \vee 116t^8 - 248t^7 + 320t^6 - 256t^5 + 93t^4 + 38t^3 - 69t^2 + 32t - 6 = 0) \vee 308t^8 - 856t^7 + 1312t^6 - 1336t^5 + 889t^4 - 362t^3 + 57t^2 + 14t - 6 \neq 0 \wedge 116t^8 - 248t^7 + 320t^6 - 256t^5 + 93t^4 + 38t^3 - 69t^2 + 32t - 6 \neq 0 \wedge (32832t^{18} - 199296t^{17} + 1134432t^{16} - 3685632t^{15} + 8482800t^{14} - 14828928t^{13} + 20679792t^{12} - 23623968t^{11} + 22409972t^{10} - 17740744t^9 + 11641246t^8 - 6203784t^7 + 2544403t^6 - 692060t^5 + 38494t^4 + 70736t^3 - 38841t^2 + 10188t - 1242 < 0 \wedge 35728t^{16} - 175680t^{15} + 463040t^{14} - 833120t^{13} + 1102072t^{12} - 1093760t^{11} + 791120t^{10} - 361408t^9 + 18885t^8 + 130796t^7 - 125924t^6 + 66446t^5 - 20877t^4 + 2802t^3 + 520t^2 - 276t + 36 > 0 \wedge 168t^{12} - 544t^{11} + 1920t^{10} - 3948t^9 + 6804t^8 -$

$8892t^7 + 9553t^6 - 8209t^5 + 5667t^4 - 3048t^3 + 1211t^2 - 327t + 45 < 0 \lor 2944t^{18} - 50560t^{17} + 226624t^{16} - 534528t^{15} + 724560t^{14} - 336800t^{13} - 1125696t^{12} + 3577440t^{11} - 6300088t^{10} + 8146344t^9 - 8368040t^8 + 7004936t^7 - 4829373t^6 + 2734054t^5 - 1256017t^4 + 455832t^3 - 124379t^2 + 23106t - 2259 > 0 \land 35728t^{16} - 175680t^{15} + 463040t^{14} - 833120t^{13} + 1102072t^{12} - 1093760t^{11} + 791120t^{10} - 361408t^9 + 18885t^8 + 130796t^7 - 125924t^6 + 66446t^5 - 20877t^4 + 2802t^3 + 520t^2 - 276t + 36 > 0 \lor 32832t^{18} - 199296t^{17} + 1134432t^{16} - 3685632t^{15} + 8482800t^{14} - 14828928t^{13} + 20679792t^{12} - 23623968t^{11} + 22409972t^{10} - 17740744t^9 + 11641246t^8 - 6203784t^7 + 2544403t^6 - 692060t^5 + 38494t^4 + 70736t^3 - 38841t^2 + 10188t - 1242 > 0 \land 35728t^{16} - 175680t^{15} + 463040t^{14} - 833120t^{13} + 1102072t^{12} - 1093760t^{11} + 791120t^{10} - 361408t^9 + 18885t^8 + 130796t^7 - 125924t^6 + 66446t^5 - 20877t^4 + 2802t^3 + 520t^2 - 276t + 36 > 0 \land 168t^{12} - 544t^{11} + 1920t^{10} - 3948t^9 + 6804t^8 - 8892t^7 + 9553t^6 - 8209t^5 + 5667t^4 - 3048t^3 + 1211t^2 - 327t + 45 > 0)) \lor 5t^2 - 2t + 2 \neq 0 \land t - 1 = 0 \land \neg(225t^8 - 1080t^7 + 2156t^6 - 1464t^5 - 316t^4 + 528t^3 - 912t^2 + 320t - 176 > 0 \land 5t^2 - 2t - 2 = 0 \lor 5t^2 - 2t - 2 \neq 0 \land (225t^8 - 1080t^7 + 2156t^6 - 1464t^5 - 316t^4 + 528t^3 - 912t^2 + 320t - 176 > 0 \lor 75t^8 - 360t^7 + 772t^6 - 456t^5 + 100t^4 - 48t^3 - 24t^2 + 16t - 16 > 0)) \lor 4t^2 - 4t + 5 \neq 0 \land 2t - 1 = 0 \land \neg(3200t^{10} - 17024t^9 + 53088t^8 - 113536t^7 + 182824t^6 - 226952t^5 + 221554t^4 - 167076t^3 + 94947t^2 - 37226t + 8185 = 0 \land (7424t^{12} - 34816t^{11} + 112256t^{10} - 250624t^9 + 446944t^8 - 633376t^7 + 747976t^6 - 721136t^5 + 580497t^4 - 372438t^3 + 191882t^2 - 68870t + 17249 < 0 \lor 208t^6 - 480t^5 + 984t^4 - 1144t^3 + 1117t^2 - 604t + 309 = 0) \land (48t^4 - 96t^3 + 152t^2 - 136t + 63 = 0 \lor 4t^2 - 12t + 11 = 0) \lor 48t^4 - 96t^3 + 152t^2 - 136t + 63 \neq 0 \land 4t^2 - 12t + 11 \neq 0 \land (3200t^{12} - 17024t^{11} + 56288t^{10} - 130560t^9 + 235912t^8 - 340488t^7 + 404378t^6 - 394028t^5 + 316501t^4 - 204302t^3 + 103132t^2 - 37226t + 8185 > 0 \land 208t^8 - 480t^7 + 1192t^6 - 1624t^5 + 2101t^4 - 1748t^3 + 1426t^2 - 604t + 309 < 0 \land 192t^6 - 960t^5 + 2288t^4 - 3424t^3 + 3556t^2 - 2252t + 693 > 0 \lor 7424t^{12} - 34816t^{11} + 112256t^{10} - 250624t^9 + 446944t^8 - 633376t^7 + 747976t^6 - 721136t^5 + 580497t^4 - 372438t^3 + 191882t^2 - 68870t + 17249 < 0 \land 192t^6 - 960t^5 + 2288t^4 - 3424t^3 + 3556t^2 - 2252t + 693 > 0 \lor 3200t^{12} - 17024t^{11} + 56288t^{10} - 130560t^9 + 235912t^8 - 340488t^7 + 404378t^6 - 394028t^5 + 316501t^4 - 204302t^3 + 103132t^2 - 37226t + 8185 < 0 \land 208t^8 - 480t^7 + 1192t^6 - 1624t^5 + 2101t^4 - 1748t^3 + 1426t^2 - 604t + 309 > 0 \land 192t^6 - 960t^5 + 2288t^4 - 3424t^3 + 3556t^2 - 2252t + 693 > 0)))$. $\diamond$

6.3 Architekturprobleme

In diesem Abschnitt wird die Hermitesche Quantorenelimination benutzt, um festzustellen, wie die Architektur des Roboters beschaffen sein muß, so daß der Effektor gewisse Positionen im Raum einnehmen kann. Dazu sind hier die Armlängen l_2, l_3 des Motoman variabel.

Aufgabe: Sei ein Punkt $\underline{x} := (x_1, x_2, x_3) \in \mathbb{R}^3$ gegeben. Wie müssen die Armlängen l_2 bzw. l_3 gewählt werden, so daß der Punkt $\underline{x}$ im Arbeitsraum des Motoman liegt?

Lösung: Die zugehörige Problemformulierung ist damit

$$\exists c_1 \exists s_1 \exists c_2 \exists s_2 \exists c_3 \exists s_3 \big(c_1^2 + s_1^2 = 1 \wedge c_2^2 + s_2^2 = 1 \wedge c_3^2 + s_3^2 = 1 \wedge \\ x_1 = c_1(l_2 c_2 + l_3(c_2 c_3 - s_2 s_3)) \wedge x_2 = s_1(l_2 c_2 + l_3(c_2 c_3 - s_2 s_3)) \wedge \\ x_3 = -l_2 s_2 - l_3(c_2 s_3 + c_3 s_2)\big) \ .$$

Die Hermitesche Quantorenelimination, angewandt auf diese Formel, liefert eine quantorenfreie Formel $\psi(l_2, l_3)$, so daß für alle $(l'_2, l'_3) \in \mathbb{R}^2$ gilt:

$\psi(l'_2, l'_3) \Leftrightarrow$ „Wenn man für die Armlänge l_2 den Wert l'_2 wählt und für l_3 den Wert l'_3, dann liegt $\underline{x}$ im Arbeitsraum des Motoman."

Anhand des folgenden Beispiels wird die Aufgabe konkret durchgerechnet:

Beispiel 6.8 *Sei $l_2 = 1$ und der Punkt $(1, 2, 1) \in \mathbb{R}^3$ gegeben.*
Wie muß l_3 gewählt werden, so daß dieser Punkt im Arbeitsraum des Motoman liegt?

Das Ergebnis der Hermiteschen Quantorenelimination angewandt auf diesen konkreten Fall wird in 20 Millisekunden berechnet, besteht aus 11 atomaren Formeln und ist:

$l_3 \neq 0 \wedge \neg(5l_3^8 - 140l_3^6 + 1440l_3^4 - 6440l_3^2 + 563 > 0 \wedge (2l_3^4 - 28l_3^2 + 131 = 0 \vee l_3^2 + 2l_3 - 5 = 0 \vee l_3^2 - 2l_3 - 5 = 0) \wedge (l_3^2 + 2l_3 - 5 = 0 \vee l_3^2 - 2l_3 - 5 = 0) \vee l_3^2 + 2l_3 - 5 \neq 0 \wedge l_3^2 - 2l_3 - 5 \neq 0 \wedge (5l_3^8 - 140l_3^6 + 1440l_3^4 - 6440l_3^2 + 563 > 0 \vee 2l_3^8 - 56l_3^6 + 573l_3^4 - 2534l_3^2 + 3275 > 0))$. $\diamond$

Diese Aufgabenstellung kann natürlich insofern erweitert werden, daß man nicht einen einzelnen Punkt sich vorgibt und fragt, wie die Armlängen beschaffen sein müssen, sondern daß man sich einen ganzen Weg vorgibt und fragt, wie die Armlängen beschaffen sein müssen, so daß der ganze Weg sich im Arbeitsraum des Roboters befindet. Diese und noch viele andere ähnliche Fragestellungen lassen sich analog als Formel erster Stufe formulieren und mit Hermitescher Quantorenelimination lösen. Jedoch gelangt man auch hier schnell an die Grenzen der Anwendbarkeit der Hermiteschen Quantorenelimination, je mehr Vorgaben und Bedingungen man macht.

Kapitel 7

Testbeispiele

Dieses Kapitel zeigt Anwendungsbeispiele der Hermiteschen Quantorenelimination. Dabei werden die Ergebnisse sowohl auf Berechnungszeit und Komplexität hin untersucht.

Sofern nicht explizit anders angegeben, wurden alle Berechnungen unter Einbeziehung aller bisher in dieser Arbeit aufgeführten Optimierungsvorschläge durchgeführt.

Dieses Kapitel teilt sich in 3 Teile auf: Der erste Teil beinhaltet einfachere Testbeispiele. Der zweite Teil zeigt anhand der Fragestellung der Existenz von rellen Nullstellen von Idealen unter Berücksichtigung von Nebenbedingungen die Unterschiede, wenn man die in dieser Arbeit vorgestellten Optimierungsvorschläge zu den Berechnungen zuläßt bzw. nicht zuläßt. Der dritte und letzte Teil beschäftigt sich mit Beispielen aus der Literatur.

7.1 Einfache Testbeispiele

Beispiel 7.1.1 Injektivität der Parabel auf der positiven x-Achse:

$$\forall x \forall y((x > 0 \land y > 0 \land x^2 = y^2) \to x = y)) \ .$$

Das Ergebnis *true* erhält man sofort, d.h. die Berechnungszeit ist so gering, daß der Rechner die dafür benötigte Zeit nicht mehr messen kann.
Eine kleine Abänderung ist folgende Formel:

$$\forall x \forall y(x^2 = y^2 \to (x = y \lor x = -y)) \ .$$

Die Quantoren der Formel werden ebenfalls sofort eliminiert. Das Ergebnis ist wiederum *true*. $\diamond$

Beispiel 7.1.2 Existenz von Schnittpunkten:
Seien für $a \in \mathbb{R}$ die Funktionen

$$f : \mathbb{R}^2 \to \mathbb{R} \;:\; (x, y) \mapsto x^2 + 1 \quad \text{und} \quad g : \mathbb{R}^2 \to \mathbb{R} \;:\; (x, y) \mapsto a \cdot y^2$$

gegeben. Die Graphen beider Funktionen schneiden sich, wenn folgende Formel erfüllt ist:

$$\exists x \exists y (x^2 + 1 = a \cdot y^2) \; .$$

Das Ergebnis der Quantorenelimination erhält man sofort und ist $a > 0$, d.h. für positives a besitzen die beiden Graphen mindestens einen Schnittpunkt. Ein ähnliches Problem stellt folgende Formel dar:

$$\exists x \exists y (x^2 + 1 = a \cdot y^2 + b) \; .$$

Das Ergebnis der Elimination ist

$$(a \neq 0 \wedge (a \cdot b - a \leq 0 \vee a > 0 \vee b - 1 \geq 0)) \vee (a = 0 \wedge b - 1 \geq 0) \; .$$

Das Ergebnis wird auch wiederum sofort berechnet. $\diamond$

Beispiel 7.1.3
Für ein univariates, normiertes Polynom vom Grad 2 charakterisiert folgende Formel, wann dieses eine relle Nullstelle mit Vielfachheit 2 besitzt:

$$\exists x \forall y (x^2 + ax + b = 0 \wedge (y \neq x \to y^2 + ay + b \neq 0)) \; .$$

Das Ergebnis wird sofort berechnet und besteht aus 15 atomaren Formeln:
$a^8 - 10a^6b + 2a^6 + 32a^4b^2 - 14a^4b + a^4 - 32a^2b^3 + 20a^2b^2 - 4a^2b - 4b^4 + 8b^3 - 4b^2 = 0 \wedge a^2 - 2b + 2 = 0 \wedge a^2 - 3b + 1 \neq 0 \wedge a \neq 0 \vee a^2 + b^2 - 2b + 1 = 0 \wedge a^2 - 2b + 2 \neq 0 \wedge a^2 - 4b \neq 0 \wedge a \neq 0 \vee a^2 - 2b + 2 \neq 0 \wedge (a^2 - b + 1 = 0 \vee a^2 - 4b = 0) \wedge (a^2 - 4b = 0 \vee a = 0) \wedge \neg(a^6 - 6a^4b + 2a^4 + 8a^2b^2 - 4a^2b + 4a^2 + 4b^3 - 4b^2 - 4b + 4 = 0 \vee a^4 + a^2b^2 - 6a^2b + a^2 - 4b^3 + 8b^2 - 4b > 0)$. $\diamond$

Beispiel 7.1.4
Sei $f : \mathbb{R}^2 \to \mathbb{R}^3 : (u, v) \mapsto (uv, u^2v, uv^2)$ gegeben. Es soll eine Formel $\psi(x, y, z)$ angegeben werden, so daß für ein Tripel $(a, b, c) \in \mathbb{R}^3$ gilt: $\psi(a, b, c) \Leftrightarrow (a, b, c) \in Bild(f)$.
Die Problemformulierung ist also

$$\exists u \exists v (x = uv \wedge y = u^2v \wedge z = uv^2) \; .$$

Das Eliminationsergebnis $\psi(x, y, z)$ erhält man sofort und besteht aus 5 atomaren Formeln:

$$(x^3 - yz = 0 \wedge x \neq 0) \vee (x = 0 \wedge y = 0 \wedge z = 0) \; . \diamond$$

7.2 Existenz reeller Nullstellen von Idealen

Dieser Abschnitt zeigt Beispiele für die Anwendung des in Kapitel 3 vorgestellten Quantoreneliminationverfahrens angewandt auf die Problemstellung, wann Ideale in $\mathbb{Q}[\underline{U}, \underline{X}]$ reelle Nullstellen besitzen, welche gewissen Nebenbedingungen genügen.

Es wird hierbei sehr viel Wert darauf gelegt, alle Beispiele mit und ohne Anwendung von vorgestellten Optimierungsmöglichkeiten berechnen zu lassen und die einzelnen Ergebnisse bzgl. ihres Umfangs und der benötigten Zeit zu deren Berechnung zu vergleichen. Die folgenden Beispiele wurden auf drei Arten jeweils berechnet:

1. Ohne Verwendung von Strukturkonstanten und ohne Anwendung des in Abschnitt 5.3 vorgestellten Splittingverfahrens `SPLITTING`.

2. Ohne Verwendung von Strukturkonstanten, aber mit Anwendung des Algorithmus `SPLITTING`.

3. Verwendung von Strukturkonstanten und Anwendung des Algorithmus `SPLITTING`.

Für all diese aufgeführten Fälle und alle Beispiele in diesem Abschnitt werden die Zeiten für die Ergebnisberechnungen und die Anzahl der atomaren Formeln im Ergebnis zum Vergleich aufgeführt. Die Anzahl der atomaren Formeln im Ergebnis kann sich dabei nur aufgrund (Nicht-)Anwendungen von `SPLITTING` ändern. Dabei bezeichnet

- $t_{Split,SK}$ die Zeit zur Berechnung des Ergebnisses *mit* Verwendung von Strukturkonstanten und *mit* Anwendung des Splittingverfahrens,

- t_{Split} die Zeit zur Berechnung des Ergebnisses *ohne* Verwendung von Strukturkonstanten, aber *mit* Anwendung des Splittingverfahrens,

- t_0 die Zeit zur Berechnung des Ergebnisses *ohne* Verwendung von Strukturkonstanten und *ohne* Anwendung des Splittingverfahrens,

- $\#AF_{Split}$ die Anzahl der atomaren Formeln im Ergebnis, wenn man zu dessen Berechnung das Splitting-Verfahren anwendet,

- $\#AF_0$ die Anzahl der atomaren Formeln im Ergebnis, wenn man zu dessen Berechnung das Splitting-Verfahren *nicht* anwendet.

Beispiel 7.2.1
Es sind die Quantoren der folgenden Formel zu eliminieren:

$$\begin{aligned}\varphi_1 \quad := \quad & \exists x \exists y (x - 3y - 3 = 0 \wedge y^2 + y - 9 = 0 \wedge x - a \neq 0 \wedge \\ & \quad x^{12}y^{15} - x^8y^{17} - 4x^2 + y^{17} \neq 0 \wedge x^{15} - 2y^{17} \neq 0) \ .\end{aligned}$$

Das Ergebnis der Elimination mit und ohne Anwendung des Splittingverfahrens ist

$$4246425929615149845821278117006905900002626071044461835938092694312426a^2$$
$$-90201476449233771556231587960193565302999494467426047969134628988024010a$$
$$+4792627149726777948458709094178497258547119534557404806646874517213425 21$$
$$\neq 0\ .$$

Es ergeben sich folgende Werte:

$t_{Split,SK}$	t_{Split}	t_0	$\#AF_{Split}$	$\#AF_0$
0,25 s	1,2 s	1,2 s	1	1

$\diamond$

Aufgrund der Größe der auftretenden Koeffizienten und Terme in den Ergebnisformeln wird ab jetzt auf die explizite Angabe von Ergebnisformeln verzichtet.

Beispiel 7.2.2
Es sind die Quantoren der folgenden Formel zu eliminieren:

$$\varphi_2 \ := \ \exists x(x^2 - 1 > 0 \wedge x^5 + ax^2 + bx + c = 0)\ .$$

Bei der Berechnung des Ergebnisses ergeben sich folgende Werte:

$t_{Split,SK}$	t_{Split}	t_0	$\#AF_{Split}$	$\#AF_0$
1,7 s	1,7 s	30,9 s	199	797

Man sieht an diesem Beispiel sehr deutlich, wie die Ergebnisformel durch Anwendung des Splitting-Verfahrens schneller berechnet und kleiner gemacht werden kann, gemessen an der Anzahl der atomaren Formeln. Diese Beobachtung wird man auch an den weiteren Beispielen in diesem Abschnitt machen. Die Verwendung von Strukturkonstanten wirkt sich in diesem Beispiel weder positiv noch negativ aus. $\diamond$

Beispiel 7.2.3
Es sind die Quantoren der folgenden Formel zu eliminieren:

$$\begin{aligned}\varphi_3 \ := \ & \exists x \exists y(x^5 + x^2y^5 - x^2y + 2x^2 + x + 1 = 0 \wedge y^5 - y + 1 = 0 \wedge \\ & x^3 + x^2 \neq 0 \wedge y^{10} + y^3 \neq 0 \wedge y^{18} + 10 \neq 0 \wedge x^{10} + x^6 + x \neq 0 \\ & \wedge x^{13} + x + y^{12} \neq 0 \wedge x^{12}y^2 + x^{10}y^5 \neq 0)\ .\end{aligned}$$

Bei der Berechnung des Ergebnisses *false* ergeben sich folgende Werte:

$t_{Split,SK}$	t_{Split}	t_0	$\#AF_{Split}$	$\#AF_0$
1,9 s	252 s	252 s	0	0

Hier zeigt sich also deutlich, welchen Effizienzgewinn die Verwendung von Strukturkonstanten mit sich bringen kann. $\diamond$

Beispiel 7.2.4
Es sind die Quantoren der folgenden Formel zu eliminieren:

$$\begin{aligned}\varphi_4 \quad := \quad & \exists x \exists y \exists z (z^2 - 2x + 1 = 0 \wedge x - 3y - 3 = 0 \wedge y^2 + y - 9 = 0 \wedge \\ & -a + x^2 z^5 - x^3 y^4 > 0 \wedge y^3 z^5 - x^4 y^5 - x^7 y^3 + 1 \neq 0 \wedge \\ & x^3 y^3 z^3 + x^2 y^2 - y^2 z^2 - 1 \neq 0) \ .\end{aligned}$$

Bei der Berechnung des Ergebnisses ergeben sich folgende Werte:

$t_{Split,SK}$	t_{Split}	t_0	$\#AF_{Split}$	$\#AF_0$
4,4 s	71 s	503 s	56	205

Hier zeigt sich also, daß sowohl die Anwendung des Splittingverfahrens als auch die Verwendung von Strukturkonstanten zu einem echten Effizienzgewinn führen. $\diamond$

Beispiel 7.2.5
Es sind die Quantoren der folgenden Formel zu eliminieren:

$$\begin{aligned}\varphi_5 \quad := \quad & \exists x \exists y (x^3 + x^2 + x - 1 = 0 \wedge 5y^3 - y^2 - 4 = 0 \wedge y^9 - xy^3 + 3 > 0 \wedge \\ & x^{12} y^8 - xy - 12x^2 - 9a > 0 \wedge y^{10} - x^2 - 9 \neq 0 \wedge x^2 y^2 - 3x^2 y \neq 0 \wedge \\ & x^5 y^6 - x^2 y^3 - 9 \neq 0 \wedge x^2 y^{10} - x^4 y^3 + x^2 y^5 - 7 \neq 0) \ .\end{aligned}$$

Bei der Berechnung des Ergebnisses ergeben sich folgende Werte:

$t_{Split,SK}$	t_{Split}	t_0	$\#AF_{Split}$	$\#AF_0$
5,4 s	15,2 s	897 s	68	288

Auch hier zeigt sich wie im vorherigen Beispiel ein deutlicher Effizienzgewinn durch Anwendung der vorgestellten Optimierungsmöglichkeiten. Die Anwendung des Splitting-Verfahrens macht sich hier enorm bemerkbar. $\diamond$

Beispiel 7.2.6
Es sind die Quantoren der folgenden Formel zu eliminieren:

$$\begin{aligned}\varphi_6 \quad := \quad & \exists x \exists y (x^3 - 2x^2 - x + 2 = 0 \wedge x^3 y - x^2 y + xy + 6y^2 - 2y^4 - 2y - 4 = 0 \wedge \\ & x^3 y^7 - x^2 y - 9a > 0 \wedge x^2 - b \neq 0 \wedge x^{15} y^{15} - x^{13} y^{10} + x^{12} y - 8 \neq 0 \wedge \\ & x^{10} y^{14} - x^3 y^9 - 9 \neq 0) \ .\end{aligned}$$

Bei der Berechnung des Ergebnisses ergeben sich folgende Werte:

$t_{Split,SK}$	t_{Split}	t_0	$\#AF_{Split}$	$\#AF_0$
8,9 s	25,1 s	ca. 23 min	249	651

Wiederum sieht man den deutlichen Effizienzgewinn durch Verwendung von Strukturkonstanten und Anwendung des Splitting-Verfahrens. Man beachte, daß in φ_6 zwei Parameter enthalten sind, nämlich a und b. Jeder zusätzliche Parameter in solch einer Eingabeformel macht sich i. a. sehr stark hinsichtlich der Berechnungszeiten negativ bemerkbar, falls keine Strukturkonstanten verwendet werden. $\diamond$

Beispiel 7.2.7
Die Beispielformel aus 7.2.5 wird vergrößert. Dann sind die Quantoren der folgenden Formel zu eliminieren:

$$\begin{aligned}\varphi_7 \quad := \quad & \exists x \exists y (x^3 + x^2 + x - 1 = 0 \wedge 5y^3 - y^2 - 4 = 0 \wedge y^9 - xy^3 + 3 > 0 \wedge \\ & x^{12}y^8 - xy - 12x^2 - 9a > 0 \wedge y^{10} - x^2 - 9 \neq 0 \wedge x^2y^2 - 3x^2y \neq 0 \wedge \\ & x^5y^6 - x^2y^3 - 9 \neq 0 \wedge x^2y^{10} - x^4y^3 + x^2y^5 - 7 \neq 0 \wedge \\ & x^4y^3 - axy + 9 \neq 0 \wedge x^3y^3 - 5x + 6y - 3 \neq 0) \ .\end{aligned}$$

Bei der Berechnung des Ergebnisses ergeben sich folgende Werte:

$t_{Split,SK}$	t_{Split}	t_0	$\#AF_{Split}$	$\#AF_0$
44,5 s	67,9 s	ca. 218 min	164	410

Hier sieht man sehr deutlich zum einen, welchen Zeitgewinn die Anwendung des Splitting-Verfahrens mit sich bringt, und zum anderen, daß die Anzahl der atomaren Formeln in der Ergebnisformel prägnant verringert wird. ◇

Sofern es in den Ergebnisformeln einen Unterschied betreffend der Anzahl der atomaren Formeln gibt in Abhängigkeit davon, ob man `SPLITTING` verwendet, kann man folgende Beobachtung machen: Durch die Anwendung des Splitting-Verfahrens werden die Anzahlen der atomaren Formeln in den Ergebnisformeln in etwa gedrittelt bis geviertelt, im Vergleich zur Berechnung der Ergebnisformeln ohne Verwendung von `SPLITTING`.

Bei ,,kleineren" Eingabeformeln kann der Effekt auftreten, daß durch die Berechnung der Strukturkonstanten mehr Zeit benötigt wird, als man durch deren Verwendung bei Multiplikationen in Restklassenalgebren später wieder gutmachen kann, im Vergleich zur Berechnung von Ergebnissen ohne Verwendung von Strukturkonstanten. Folgendes Beispiel zeigt dies:

Beispiel 7.2.8
Es sind die Quantoren der folgenden Formel zu eliminieren:

$$\varphi_8 := \exists x \exists y \exists z (z^4 + 3z^3 - 5z + 3 = 0 \wedge x^4 + x^3 - 2x - 4 = 0 \wedge y^4 - y^2 + 4y = 0) \ .$$

Das Ergebnis *false* erhält man bei der Berechnung mit Verwendung von Strukturkonstanten in 5,8 Sekunden. Werden keine Strukturkonstanten benutzt, um das Ergebnis zu berechnen, so erhält man das Ergebnis bereits in 4,3 Sekunden. Die Anwendung des Splitting-Verfahrens wirkt sich auf die Berechnungszeit weder positiv noch negativ aus. ◇

Das Beispiel 7.2.8 stellt aber die Verwendung von Strukturkonstanten nicht in Frage. Nur bei kleinen Beispielen, wie dem letzten, müssen relativ wenig Multiplikationen in Restklassenalgebren durchgeführt werden. Außerdem sind die auftretenden Reduktionsketten nicht sehr lang. Deswegen können die Strukturkonstanten nicht ihre echte Effizienz entfalten, welche umso mehr steigt, je mehr Multiplikationen durchgeführt werden und desto länger die Reduktionsketten bei Berechnung ohne Strukturkonstanten wären. Daher wirkt sich in obigem Beispiel die Verwendung von Strukturkonstanten aufgrund der Zeit zu deren Berechnung negativ auf die Berechnungszeit des Eliminationsergebnisses aus. In solchen Beispielen wird aber das Ergebnis in kurzer Zeit berechnet und somit ist es vernachlässigbar, wenn durch die Verwendung von Strukturkonstanten eine leicht erhöhte Berechnungszeit für das Eliminationsergebnis benötigt wird, im Vergleich zur Berechnung ohne Strukturkonstanten.

Weitere Vergleiche der resultierenden Ergebnisformeln:

Jetzt werden die in den bisher gezeigten Beispielen entstandenen Ergebnisformeln genauer untersucht. Bisher konnte man die Größe und Komplexität von Ergebnisformeln anhand der Anzahl der darin enthaltenen atomaren Formeln vergleichen. Es ist aber bisher nicht klar, ob die Reduktion der Anzahl der atomaren Formeln durch Anwendung des Splitting-Verfahrens nicht eine kompliziertere Gestalt(z.B. höherer Totalgrad, größere Koeffizienten) der einzelnen atomaren Formeln im Ergebnis mit sich bringt. Deshalb werden nun zu den Ergebnisformeln die darin enthaltenen atomaren Formeln auf ihre Komplexität hin überprüft, und dann verglichen, wie sich diese Komplexitäten verhalten, wenn man Splitting anwendet oder nicht.

Um diesen Komplexitätsvergleich der atomaren Formeln geeignet durchführen zu können, werden fünf Kriterien zu einer Ergebnisformel ψ betrachtet. Sei dazu $P \subseteq \mathbb{Z}[\underline{U}]$ die Menge aller Nicht-Nullpolynome, welche in den atomaren Formeln von ψ auftreten, wobei $\underline{U}$ die Menge aller Variablen von ψ repräsentiere. Zu einem Polynom $p = \sum_{t \in T(p)} a_t t \in \mathbb{Z}[\underline{U}]$ sei $|p| := \sum_{t \in T(p)} |a_t|$ die Summe der Absolutbeträge aller Koeffizienten in p und bezeichne $totdeg(p)$ den höchsten in p auftretenden Totalgrad. Mit diesen Notationen werden nun folgende fünf Kriterien in einer Ergebnisformel untersucht:

1. $D_{koeff} := (\sum_{p \in P} |p|)/(\sum_{p \in P} |T(p)|)$: Durchschnitt der Absolutbeträge aller in den atomaren Formeln auftretenden Koeffizienten. Z.B. ist für die Formel $2a - 1 > 0 \wedge b + 4d = 0$ der zugehörige Koeffizientenbetragsdurchschnitt 2.

2. *Median* der Beträge aller in den atomaren Formeln von ψ auftretenden Koeffizienten, d.h. es werden die Absolutbeträge der Koeffizienten a_t aus allen $p \in P$ gesammelt, aufsteigend geordnet und zu dieser entstehenden Folge der Median berechnet. Z.B. für $a - b + c > 0 \wedge 2a - 3b = 0$ ist die zugehörige Koeffizientenbetragsfolge (1 1 1 2 3) und somit der zugehörige Median der Koeffizientenbeträge 1.

3. $\#T := \sum_{p \in P} |T(p)|$: Gesamtanzahl der Terme aus allen atomaren Formeln. Z.B. sind in der Formel $a + bc > 0 \wedge 2a + 3b = 0$ 4 Terme enthalten.

4. $m := max\{totdeg(p) \mid p \in P\}$: Höchster auftretender Totalgrad in den atomaren Formel der Ergebnisformel ψ. Z.B. ist also in der Formel $x^2 - a > 0 \wedge abc - x = 0$ der höchste auftretende Totalgrad 3.

5. $D_{grad} := (\sum_{p \in P} totdeg(p))/|P|$: Durchschnitt der höchsten Totalgrade der atomaren Formeln von ψ. Z.B. ist für $x^2 - a > 0 \wedge abc - x = 0$ dieser Durchschnitt 2,5.

Die Tabellen in Abbildung 7.1 und 7.2 stellen zu einigen Beispielen von oben diese Kennzahlen zu den Ergebnisformeln gegenüber in Abhängigkeit davon, ob man das Splitting-Verfahren anwendet oder nicht. Zu den Durchschnitten und Medianen der Koeffizientenbeträge sind der Übersichtlichkeit halber lediglich Näherungswerte in Form von 10er-Potenzen angegeben Man sieht

Bsp.	$D_{koeff}^{(Split)}$	$D_{koeff}^{(ohneSplit)}$	Median bei Splitting	Median ohne Splitting
φ_2	1270	10^6	303	10^5
φ_4	10^{110}	10^{217}	10^{74}	10^{132}
φ_5	10^{139}	10^{268}	10^{72}	10^{206}
φ_6	10^{83}	10^{163}	10^{35}	10^{83}
φ_7	10^{161}	10^{312}	10^{81}	10^{238}

Abbildung 7.1: Komplexität der auftretenden Koeffizienten in den Ergebnisformeln, Teil I

Bsp.	$\#T^{(Split)}$	$\#T^{(ohneSplit)}$	$D_{Grad}^{(Split)}$	$D_{Grad}^{(ohneSplit)}$	$m^{(Split)}$	$m^{(ohneSplit)}$
φ_2	6397	152988	6,3	12,1	10	18
φ_4	240	1552	3,3	6,6	6	11
φ_5	208	1790	2,1	5,2	4	9
φ_6	942	4755	2,8	6,3	6	11
φ_7	668	4106	3,1	9,0	8	16

Abbildung 7.2: Komplexität der auftretenden Koeffizienten in den Ergebnisformeln, Teil II

in den Tabellen der Abbildungen 7.1 und 7.2 also, daß durch Verwendung des Splitting-Verfahrens zur Bestimmung der Ergebnisformeln nicht nur die Anzahlen der atomaren Formeln reduziert werden, sondern daß auch die Komplexitäten der atomaren Formeln sich verringern. Dies spricht klar für die Anwendung des Splitting-Verfahrens mittels Algorithmus `SPLITTING` aus Abbildung 5.3. Auffällig ist vor allem, daß der Median der Koeffizientenbeträge aus den atomaren Formeln stets deutlich unter dem Durchschnitt der Koeffizientenbeträge ist. Dies ist ein Hinweis darauf, daß viele auftretende Koeffizienten in den atomaren Formeln der Ergebnisformeln ,,kleinere" Beträge besitzen, und relativ wenige Koeffizientenbeträge ,,sehr groß" sind im Vergleich zu den kleineren Beträgen.

Fazit: Die Anwendung des Splitting-Verfahrens und die Verwendung von Strukturkonstanten führen in der Praxis i. a. zu einem deutlichen Effizienzgewinn.

Als letztes Beispiel in diesem Abschnitt wird ein Beispiel aus Kapitel 3.4.2.1 nachgereicht. Es wurde darauf hingewiesen, daß man bei der Hermiteschen Quantorenelimination nur Äste von Gröbnersystemen betrachten muß, deren Bedingung von mindestens einer reellen Spezialisierung der Parameter erfüllt wird, und restliche Äste streichen kann. Folgendes Beispiel zeigt, daß man durch Streichen solcher überflüssigen Äste zu einem Effizienzgewinn bei der Berechnungszeit des Ergebnisses kommen kann:

Beispiel 7.2.9:
Es soll der Quantor der folgenden Formel eliminiert werden:

$$\varphi_9 := \exists x\big((r^2+1)x^3 + ax^2 + 1 = 0 \wedge x^{10} - x^7 + 5x^4 - 3 > 0\big) \ .$$

Betrachtet man bei der Elimination nur Äste des auftretenden Gröbnersystems, deren Bedingung von mindestens einer reellen Spezialisierung erfüllt wird, so muß nur ein Ast eliminiert werden, da $r^2+1 \neq 0$ für alle $r \in \mathbb{R}$. Das Ergebnis erhält man dann in 65,4 Sekunden und besteht aus 57 atomaren Formeln.
Wird nicht untersucht, ob zu Bedingungen der Äste reelle Spezialisierungen existieren, welche diese erfüllen, so erhält man das Ergebnis in 71,1 Sekunden, bestehend wiederum aus 57 atomaren Formeln. Die Eliminationsergebnisse der „überflüssigen" Äste können also zu *false* simplifiziert werden. $\diamond$

7.3 Bekannte Testbeispiele

Beispiel 7.3.1: Davenport-Heintz-Problem
Das Davenport-Heintz-Problem ist ein bekanntes Testbeispiel für die reelle Quantorenelimination:

$$\exists c \forall b \forall a\big(((a = d \wedge b = c) \vee (a = c \wedge b = 1)) \rightarrow a^2 = b\big) \ .$$

Das Eliminationsergebnis erhält man sofort und ist die Formel:

$$d + 1 = 0 \vee d - 1 = 0 \ . \ \diamond$$

Beispiel 7.3.2: Descartessches Blatt
Sei

$$f : \mathbb{R}\backslash\{-1\} \rightarrow \mathbb{R}^2 : t \mapsto \left(\frac{3t}{1+t^3}, \frac{3t^2}{1+t^3}\right).$$

Zu bestimmen ist eine Formel $\psi(x,y)$, so daß für $(a,b) \in \mathbb{R}^2$ gilt:

$$\psi(a,b) \iff (a,b) \in Bild(f).$$

Es ist also der Quantor der folgenden Formel zu eliminieren:

$$\exists t\big(xt^3 + x = 3t \wedge yt^3 + y = 3t^2 \wedge t \neq 1\big).$$

Das Ergebnis $\psi(x,y)$ wird sofort berechnet und besteht aus 12 atomaren Formeln:

$(x^3 - 3xy + y^3 = 0 \wedge x^2 - 3x + y^2 \neq 0 \wedge 3x - y^2 \neq 0 \wedge x \neq 0 \wedge y \neq 0) \vee (x^2y^3 - 27x^2 - 18xy^2 + 81y = 0 \wedge xy + 3x - 9 \neq 0 \wedge xy - 9 \neq 0 \wedge x \neq 0 \wedge y = 0) \vee (x = 0 \wedge y = 0)$. $\diamond$

Beispiel 7.3.3: Quartic

Ziel ist es, zu einem univariaten Polynom f vom Grad 4 eine quantorenfreie Formel φ zu bestimmen, so daß $\varphi \Leftrightarrow f \geq 0$. Nach einer Anwendung einer Tschirnhausen-Transformation genügt es, univariate Polynome der Form $f(x) = x^4 + px^2 + qx + r$ zu betrachten. Zur Elimination wird die folgende Formulierung des Problems nach Gonzales-Vega [7] benutzt:

$$\forall x\big(x^4 + px^2 + qx + r \neq 0 \vee 4x^3 + 2px + q = 0\big).$$

Das Ergebnis der Elimination (mit und ohne Anwendung des Splitting-Verfahrens) ist folgende Formel, bestehend aus 12 atomaren Formeln:

$16p^4r - 4p^3q^2 - 128p^2r^2 + 144pq^2r - 27q^4 + 256r^3 = 0 \wedge (64p^{11} - 960p^9r + 128p^9 + 912p^8q^2 + 64p^7q^2 + 5504p^7r^2 - 1600p^7r + 64p^7 - 7136p^6q^2r + 1424p^6q^2 - 64p^6r^2 - 64p^6r + 3852p^5q^4 - 528p^5q^2r + 80p^5q^2 - 14720p^5r^3 + 7168p^5r^2 - 640p^5r + 648p^4q^4 + 17088p^4q^2r^2 - 8608p^4q^2r + 640p^4q^2 + 768p^4r^3 + 768p^4r^2 - 6588p^3q^4r + 4076p^3q^4 + 1664p^3q^2r^2 - 128p^3q^2r + 17408p^3r^4 - 13312p^3r^3 + 2048p^3r^2 + 4131p^2q^6 - 1620p^2q^4r + 396p^2q^4 - 14976p^2q^2r^3 + 11520p^2q^2r^2 - 2688p^2q^2r - 3072p^2r^4 - 3072p^2r^3 + 1458pq^6 + 12312pq^4r^2 + 288pq^4r + 1512pq^4 - 2304pq^2r^3 - 768pq^2r^2 - 6144pr^5 + 8192pr^4 - 2048pr^3 + 2916q^6r + 702q^6 + 9936q^4r^2 + 3024q^4r + 13824q^2r^4 + 8704q^2r^3 + 4608q^2r^2 + 4096r^5 + 4096r^4 > 0 \vee 8p^6 - 8p^5 - 64p^4r + 8p^4 - 134p^3q^2 + 56p^3r - 8p^3 + 90p^2q^2 + 144p^2r^2 - 48p^2r + 306pq^2r - 54pq^2 - 96pr^2 + 32pr + 27q^4 - 108q^2r + 28q^2 - 64r^3 + 64r^2 = 0) \vee 16p^4r - 4p^3q^2 - 128p^2r^2 + 144pq^2r - 27q^4 + 256r^3 \neq 0 \wedge (512p^{12}r - 128p^{11}q^2 - 512p^{11}r^2 + 384p^{10}q^2r - 10240p^{10}r^2 + 512p^{10}r - 64p^9q^4 + 11520p^9q^2r - 128p^9q^2 + 9728p^9r^3 - 512p^9r^2 - 2208p^8q^4 - 19584p^8q^2r^2 + 384p^8q^2r + 82944p^8r^3 - 9216p^8r^2 + 9504p^7q^4r - 64p^7q^4 - 136704p^7q^2r^2 + 10752p^7q^2r - 73728p^7r^4 + 8192p^7r^3 - 1296p^6q^6 + 80864p^6q^4r - 2080p^6q^4 + 211968p^6q^2r^3 - 9216p^6q^2r^2 - 344064p^6r^4 + 65536p^6r^3 - 12600p^5q^6 - 210240p^5q^4r^2 + 4416p^5q^4r + 616448p^5q^2r^3 - 116736p^5q^2r^2 + 278528p^5r^5 - 49152p^5r^4 + 75816p^4q^6r - 648p^4q^6 - 450816p^4q^4r^2 + 71424p^4q^4r - 909312p^4q^2r^4 + 73728p^4q^2r^3 + 753664p^4r^5 - 229376p^4r^4 - 8748p^3q^8 + 180576p^3q^6r - 11232p^3q^6 + 1078272p^3q^4r^3 - 32256p^3q^4r^2 - 1155072p^3q^2r^4 + 450560p^3q^2r^3 - 524288p^3r^6 + 131072p^3r^5 - 23814p^2q^8 - 684288p^2q^6r^2 + 10368p^2q^6r + 585216p^2q^4r^3 - 373248p^2q^4r^2 +$

$1622016p^2q^2r^5 - 245760p^2q^2r^4 - 786432p^2r^6 + 393216p^2r^5 + 196830pq^8r - 1458pq^8 - 41472pq^6r^2 + 155520pq^6r - 1410048pq^4r^4 + 193536pq^4r^3 + 688128pq^2r^5 - 589824pq^2r^4 + 393216pr^7 - 131072pr^6 - 19683q^{10} - 2916q^8r - 20412q^8 + 279936q^6r^3 - 31104q^6r^2 + 221184q^4r^3 - 884736q^2r^6 + 294912q^2r^5 + 262144r^7 - 262144r^6 > 0 \wedge 8p^6 - 8p^5 - 64p^4r + 8p^4 - 134p^3q^2 + 56p^3r - 8p^3 + 90p^2q^2 + 144p^2r^2 - 48p^2r + 306pq^2r - 54pq^2 - 96pr^2 + 32pr + 27q^4 - 108q^2r + 28q^2 - 64r^3 + 64r^2 < 0 \wedge 16p^4r - 4p^3q^2 - 128p^2r^2 + 144pq^2r - 27q^4 + 256r^3 > 0 \vee 64p^{11} - 960p^9r + 128p^9 + 912p^8q^2 + 64p^7q^2 + 5504p^7r^2 - 1600p^7r + 64p^7 - 7136p^6q^2r + 1424p^6q^2 - 64p^6r^2 - 64p^6r + 3852p^5q^4 - 528p^5q^2r + 80p^5q^2 - 14720p^5r^3 + 7168p^5r^2 - 640p^5r + 648p^4q^4 + 17088p^4q^2r^2 - 8608p^4q^2r + 640p^4q^2 + 768p^4r^3 + 768p^4r^2 - 6588p^3q^4r + 4076p^3q^4 + 1664p^3q^2r^2 - 128p^3q^2r + 17408p^3r^4 - 13312p^3r^3 + 2048p^3r^2 + 4131p^2q^6 - 1620p^2q^4r + 396p^2q^4 - 14976p^2q^2r^3 + 11520p^2q^2r^2 - 2688p^2q^2r - 3072p^2r^4 - 3072p^2r^3 + 1458pq^6 + 12312pq^4r^2 + 288pq^4r + 1512pq^4 - 2304pq^2r^3 - 768pq^2r^2 - 6144pr^5 + 8192pr^4 - 2048pr^3 + 2916q^6r + 702q^6 + 9936q^4r^2 + 3024q^4r + 13824q^2r^4 + 8704q^2r^3 + 4608q^2r^2 + 4096r^5 + 4096r^4 > 0 \wedge 16p^4r - 4p^3q^2 - 128p^2r^2 + 144pq^2r - 27q^4 + 256r^3 > 0 \vee 512p^{12}r - 128p^{11}q^2 - 512p^{11}r^2 + 384p^{10}q^2r - 10240p^{10}r^2 + 512p^{10}r - 64p^9q^4 + 11520p^9q^2r - 128p^9q^2 + 9728p^9r^3 - 512p^9r^2 - 2208p^8q^4 - 19584p^8q^2r^2 + 384p^8q^2r + 82944p^8r^3 - 9216p^8r^2 + 9504p^7q^4r - 64p^7q^4 - 136704p^7q^2r^2 + 10752p^7q^2r - 73728p^7r^4 + 8192p^7r^3 - 1296p^6q^6 + 80864p^6q^4r - 2080p^6q^4 + 211968p^6q^2r^3 - 9216p^6q^2r^2 - 344064p^6r^4 + 65536p^6r^3 - 12600p^5q^6 - 210240p^5q^4r^2 + 4416p^5q^4r + 616448p^5q^2r^3 - 116736p^5q^2r^2 + 278528p^5r^5 - 49152p^5r^4 + 75816p^4q^6r - 648p^4q^6 - 450816p^4q^4r^2 + 71424p^4q^4r - 909312p^4q^2r^4 + 73728p^4q^2r^3 + 753664p^4r^5 - 229376p^4r^4 - 8748p^3q^8 + 180576p^3q^6r - 11232p^3q^6 + 1078272p^3q^4r^3 - 32256p^3q^4r^2 - 1155072p^3q^2r^4 + 450560p^3q^2r^3 - 524288p^3r^6 + 131072p^3r^5 - 23814p^2q^8 - 684288p^2q^6r^2 + 10368p^2q^6r + 585216p^2q^4r^3 - 373248p^2q^4r^2 + 1622016p^2q^2r^5 - 245760p^2q^2r^4 - 786432p^2r^6 + 393216p^2r^5 + 196830pq^8r - 1458pq^8 - 41472pq^6r^2 + 155520pq^6r - 1410048pq^4r^4 + 193536pq^4r^3 + 688128pq^2r^5 - 589824pq^2r^4 + 393216pr^7 - 131072pr^6 - 19683q^{10} - 2916q^8r - 20412q^8 + 279936q^6r^3 - 31104q^6r^2 + 221184q^4r^3 - 884736q^2r^6 + 294912q^2r^5 + 262144r^7 - 262144r^6 < 0 \wedge 8p^6 - 8p^5 - 64p^4r + 8p^4 - 134p^3q^2 + 56p^3r - 8p^3 + 90p^2q^2 + 144p^2r^2 - 48p^2r + 306pq^2r - 54pq^2 - 96pr^2 + 32pr + 27q^4 - 108q^2r + 28q^2 - 64r^3 + 64r^2 > 0 \wedge 16p^4r - 4p^3q^2 - 128p^2r^2 + 144pq^2r - 27q^4 + 256r^3 > 0)$.

Diese Ergebnisformel wird in ca. 0,25 Sekunden berechnet, unabhängig davon, ob Strukturkonstanten verwendet werden und ob man das Splitting-Verfahren anwendet. $\diamond$

Beispiel 7.3.4: Enneper
Die Enneper-Fläche ist gegeben durch die folgende Funktion:

$$f : \mathbb{R}^2 \to \mathbb{R}^3 : (u,v) \mapsto (3u + 3uv^2 - u^3, 3v + 3u^2v - v^3, 3u^2 - 3v^2).$$

Ziel ist es, eine quantorenfreie Formel zu bestimmen, die charakterisiert, wann ein Tupel $(x,y,z) \in \mathbb{R}^3$ im Bild von f liegt. Also müssen die Quantoren der folgenden Formel eliminiert werden:

$$\exists u \exists v \big(x = 3u + 3uv^2 - u^3 \wedge y = 3v + 3u^2v - v^3 \wedge z = 3u^2 - 3v^2\big).$$

Das Ergebnis der Elimination dieser Formel (mit und ohne Anwendung des Splitting-Verfahrens) besteht aus 164 atomaren Formeln. Diese Ergebnisformel erhält man in 2,6 Sekunden, unabhängig davon, ob man Strukturkonstanten zur Berechnung benutzt. Aus Komplexitätsgründen wird an dieser Stelle auf eine explizite Angabe des Ergebnisses verzichtet. $\diamond$

Beispiel 7.3.5: Das Ellipsenproblem
Für mehr Informationen zu diesem Problem betrachte man [9]. Das Original-Ellipsenproblem ist zur Anwendung der Hermiteschen Quantorenelimination nicht geeignet. Wenn man aber einige triviale Fälle ausschließt, kommt man zu folgender Problemformulierung:

$$\exists x \exists y \big(a > 0 \wedge b > 0 \wedge c > 0 \wedge a - b \neq 0 \wedge b + 1 > 0 \wedge \\ b^2(x-c)^2 + a^2y^2 - a^2b^2 = 0 \wedge x^2 + y^2 - 1 = 0 \wedge 4b^2(x-c)y \neq 4a^2xy \big).$$

Das Eliminationsergebnis wird in 21,4 Sekunden berechnet und enthält 38 atomare Formeln. Wiederum wird auf eine explizite Angabe des Ergebnisses verzichtet. $\diamond$

Zusammenfassung

In dieser Arbeit wurde ein Verfahren zur Quantorenelimination von Formeln erster Stufe über der Sprache der angeordneten Ringe vorgestellt, welches auf dem Zählen reeller Nullstellen von Idealen unter Nebenbedingungen basiert. Das verwendete Verfahren zum Zählen dieser Nullstellen geht auf den frz. Mathematiker Hermite zurück. Es wurde bei der Vorstellung des Verfahrens gezeigt, wie die einzelnen Rechenschritte möglichst effizient durchgeführt werden können. Neben mehreren kleineren Optimierungsmöglichkeiten wurden eine verallgemeinerte Variante von Strukturkonstanten und aufgesplittete Typ-0-Typformeln eingeführt. Deren Verwendung bei der vorgestellten Quantorenelimination zeigte sich als besonders positiv, so daß damit die Berechnungszeiten verkürzt werden konnten als auch die Komplexität der Ergebnisformeln verringert werden konnte. In Zusammenhang mit aufgesplitteten Typ-0-Typformeln und Auswahlmöglichkeiten von Variablen wurden auch vom Autor neu entworfene Algorithmen präsentiert. Anhand von verschiedenen Beispielen wurden diese Effizienzgewinne in der Praxis demonstriert. Außerdem wurden Abschätzungen zu Typformeln geliefert, da die Ergebnisformeln im Wesentlichen aus mehreren Typformeln aufgebaut sind.

Ferner wurden Anwendungsmöglichkeiten der Hermiteschen Quantorenelimination gezeigt. Zum einen wurde vorgestellt, wie man dieses Eliminationsverfahren generisch anwenden kann, d.h. wenn man Annahmen über Parameter machen darf. Dabei zeigte sich anhand von Beispielen aus der Geometrie, daß geometrische Eigenschaften automatisch bewiesen werden können und die entstehenden Annahmen durchaus anschaulich interpretierbar sind. Zum anderen wurden Fragestellungen aus der Roboterkinematik vorgestellt, zu deren Lösung sich Hermitesche Quantorenelimination gut eignet.

Literaturverzeichnis

[1] Eberhard Becker und Thorsten Wörmann: On the trace formula for quadratic forms and some applications. In *Proc. of RAGSQUAD*, Berkeley, 1991.

[2] Thomas Becker, Volker Weispfenning: Gröbner Bases, A Computational Approach to Commutative Algebra. Springer-Verlag, 1998. ¡

[3] John J. Craig: Introduction to Robotics, Mechanics & Control. Addison-Wesley, 1986.

[4] Andreas Dolzmann: Relle Quantorenelimination durch parametrisches Zählen von Nullstellen. Diplomarbeit, Universität Passau, 1994.

[5] Andreas Dolzmann: Examples for Real Quantifier Elimination with Parametric Real Root Count. Universität Passau.

[6] Andreas Dolzmann, Thomas Sturm: A New Approach for Automatic Theorem Proving in Real Geometry. Technical Report MIP-9611, Universität Passau, Mai 1996.

[7] L. Gonzales-Vega: A combinatorical algorithm solving some quantifier elimination problems. In *Proc. of the Collins Symposium*. Linz, 1993.

[8] Otto Kerner, Joseph Maurer, Jutta Steffens, Thomas Thode und Rudolf Voller: Vieweg Mathematik Lexikon. Vieweg-Verlag, 1995.

[9] D. Lazard: Quantifier Elimination: Optimal Solution for Two Classical Examples. *Journal of Symbolic Computation*, 5(1&2):261-266. 1988.

[10] P. Pedersen, M.-F. Roy und A. Szpirglas: Counting real zeroes in the multivariate case. In *Computational Algebraic Geometry*, pages 203-224, Boston, 1993.

[11] Gunter Ritter: Skript zur Vorlesung „Grundlagen der Stochastik". Universität Passau, 1998.

[12] Günter Scheja, Uwe Storch: Lehrbuch der Algebra, Volume 2. Teubner, Stuttgart, 1988.

[13] Uwe Storch, Hartmut Wiebe: Lehrbuch der Mathematik, Band 2 Lineare Algebra(2.Auflage). Spektrum-Verlag, 1999.

[14] Thomas Sturm: Real Quantifier Elimination in Geometry. Doctoral Dissertation. Universität Passau, Dezember 1999.

[15] Volker Weispfenning: A New Approach to Quantifier Elimination for Real Algebra. In *Quantifier Elimination and Cylindrical Algebraic Decomposition*. Springer-Verlag, Wien, New York, 1998.

[16] Volker Weispfenning: Skript zur Vorlesung „Computeralgebra". Universiät Passau, April 2001.

Eidesstattliche Erklärung

Ich versichere, daß ich diese Arbeit selbständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet. Diese Arbeit habe ich in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Passau, 30. September 2003

Lorenz A. Gilch