

Freie Theoreme für Curry

Stefan Mehner, Daniel Seidel, Lutz Straßburger und Janis Voigtländer

1. April 2014

Die funktional-logische Sprache Curry vereint Konzepte des funktionalen Paradigmas – Funktionen höherer Ordnung, Bedarfsauswertung und ein starkes Typsystem – mit solchen des logischen Paradigmas – Nichtdeterminismus und freie Variablen. Das Typsystem erlaubt parametrisch polymorphe Funktionen, die unabhängig vom Typ der Argumente auf syntaktisch einheitliche Weise definiert sind (wie etwa Konkatenation von Listen).

Diese Art der Polymorphie ist aus der funktionalen Sprache Haskell bekannt und erlaubt dort das Aufstellen so genannter *freier Theoreme*. Das sind semantische Gleichheiten, die allein aufgrund des Typs einer polymorphen Funktion gelten und verschiedene Instanziierungen der Funktion in Beziehung setzen. Solche Äquivalenzen können automatisch aus dem Typ der betreffenden Funktion generiert werden und dienen unter Anderem zur Laufzeitoptimierung.

Die zusätzlichen logischen Sprachprimitiven verhindern jedoch in Curry viele Transformationen, die in Haskell semantikerhaltend sind. In früheren Arbeiten wurden bereits Gegenbeispiele aufgezeigt aber auch Bedingungen formuliert, unter denen freie Theoreme in Curry trotz der logischen Primitiven gültig sein sollen.

Wir geben eine Übersetzung an, die ein Fragment von Curry in eine Zwischensprache namens SaLT (Set- and Lambda-Terms) abbildet und dabei den Nichtdeterminismus durch die Benutzung von Mengentypen explizit macht. Für SaLT definieren wir eine einfache denotationelle Semantik, welche auf partiell geordneten Mengen basiert und bezüglich welcher wir die Gültigkeit freier Theoreme für SaLT beweisen. Zusammen mit Beta- und Eta-Äquivalenz und einigen Umformungsgesetzen für Mengenwertige Terme erhalten wir so eine Vielzahl erlaubter Transformationen.

Das Hauptergebnis ist, dass sich freie Theoreme für Curry beweisen lassen, wenn man es mit der Semantik versieht, die sich aus der Übersetzung nach SaLT ergibt. Dabei dient die Zwischensprache nicht nur als Formalismus, in dem sich bequem rechnen lässt, sondern auch dazu, die zusätzlichen Voraussetzungen zu präzisieren: In SaLT kann man durch das Typsystem erzwingen, dass bestimmte Berechnungen deterministisch verlaufen. Als Nebenprodukt ergibt sich eine einfachere denotationelle Semantik für (einen Teil von) Curry, die als Grundlage weiterer Forschung dienen kann.