

Induktive Programsynthese mit Igor2*

Helmut Grohne

Institut für Informatik, Universität Bonn

grohne@cs.uni-bonn.de

Zusammenfassung

Induktive Programsynthese bezeichnet den Prozess eine über Eingabe- und Ausgabebeispielen spezifizierte Funktion zu berechnen. Üblicherweise ist die generierte Funktion allgemeiner und einfacher als die gegebenen Beispiele. Das an der Universität Bamberg von Martin Hoffman entwickelte Werkzeug Igor2 setzt diesen Prozess für eine Teilsprache von Haskell um. Eine Spezifikation der Funktion *snoc* könnte Beispielsweise wie folgt aussehen.

$$\begin{aligned} \textit{snoc} &:: a \rightarrow [a] \rightarrow [a] \\ \textit{snoc} \ a \ [] &= [a] \\ \textit{snoc} \ b \ [a] &= [a, b] \\ \textit{snoc} \ c \ [a, b] &= [a, b, c] \\ \textit{snoc} \ d \ [a, b, c] &= [a, b, c, d] \end{aligned}$$

Daraus synthetisiert Igor2 folgende Funktion.

$$\begin{aligned} \textit{snoc} \ a0 \ [] &= [a0] \\ \textit{snoc} \ a0 \ (a1 : a2) &= a1 : \textit{snoc} \ a0 \ a2 \end{aligned}$$

In diesem Vortrag werden die Fähigkeiten und die Arbeitsweise von Igor2 vorgestellt. Darauf aufbauend werden ein paar Erweiterungen des Algorithmus vorgestellt.

*<http://www.cogsys.wiai.uni-bamberg.de/publications/ecai08demo.pdf>