Gegenbeispiel geleitete Abstraktionsverfeinerung für speicherprogrammierbare Steuerungen

Motivation

Speicherprogrammierbare Steuerungen (engl. Programmable Logic Controllers) sind Komponenten eines Steuerungssystem für großtechnische Anlagen und sind in IEC 61131-3 normiert. SPSen arbeiten zyklusorientiert, ein Programm belegt dabei innerhalb eines Zykluses die Ausgangsvariablen in Abhängigkeit der Eingangsvariablen und des internen Zustands. Dies ist so organisiert, dass sowohl das Einlesen aller Eingänge, als auch das Schreiben aller Ausgänge atomar erfolgt, die internen Zustände sind somit für die Außenwelt unsichtbar.

Aufgrund der Ergebnisse einer früheren Diplomarbeit besitzt der Lehrstuhl bereits die Möglichkeit, SPS-Programme im Format einer Anweisungsliste (AWL) mit Hilfe des Model-Checkers [mc]square zu untersuchen. Allerdings wird der Zustandsraum durch Aufzählung aller möglichen Belegungen der Eingangsvariablen aufgebaut, jeder neue Eingang wirkt sich also exponientiell auf die Größe des Zustandsraums aus. Auch werden sämtliche Ausgangsvariablen im Zustandsraum gespeichert, auch wenn sie für die Überprüfung einer Formel nicht relevant sind. Für Programme mit einer großen Menge an Ein- oder Ausgängen ist der Zustandsraum deswegen praktisch nicht mehr handhabbar.

Aufgabenstellung

Ziel dieser Diplomarbeit ist es, Model-Checking einer größeren Menge von Programmen zu ermöglichen, indem der Zustandsraum durch Zusammenfassung von Zuständen (Abstrahierung) verkleinert wird. Es wird dann nicht mehr auf konkreten Werten operiert, sondern auf Elementen einer Abstraktionsdomäne (z.B. Intervallen oder Bitmengen), die mehrere – möglichst viele – konkrete Werten zusammenfassen. Die Idee ist, eine gute Abstraktion des Verhaltens dynamisch zu finden, indem man den abstrakten Zustandsraum sukzessive verfeinert (engl. Refinement), bis der Wahrheitsgehalt der Formel entschieden werden kann. Da wir eine Überapproximation vorliegen haben, bedeutet die Abwesenheit eines Gegenbeispiels, dass die Formel wahr ist. Erhalten wir hingegen ein Gegenbeispiel in unserem abstrakten Modell, so kann man durch Vergleich mit der konkreten Simulation herausfinden, ob dies ein echtes Gegenbeispiel ist, oder – im Falle eines eigentlich nicht möglichen Zustandsüberganges – wo die Abstraktion verfeinert werden muss. Bei der Implementierung der Algorithmen in [mc]square werden verschiedene Abstraktionsdomänen getestet.

Student

- Sebastian Biallas

Betreuer

- Dr. rer. nat. Jörg Brauer
- Dr.rer.nat. Bastian Schlich

From:
https://embedded.rwth-aachen.de/ - **Informatik 11 - Embedded Software**

Permanent link:
https://embedded.rwth-aachen.de/doku.php?id=lehre:abschlussarbeiten:cegar_plc

Last update: **2010/01/29 14:48**