Modulare und Boolesche Abstraktion von SPS-Programmen

Thema

Arcade ist eine am Lehrstuhl Informatik 11 entwickelte Software zur Verifizierung von eingebetteten Systemen durch Model-Checking und statische Analysen. Teil davon ist Arcade.PLC, welches speziell auf die Verifizierung von speicherprogrammierbaren Steuerungen (SPS) ausgelegt ist. Eine (zyklusorientierte) SPS ist ein Gerät zur Steuerung oder Regelung einer Maschine oder Anlage, welches mit Ein- und Ausgängen ausgestattetet ist und periodisch ein Programm ausführt, welches die Ausgänge in Abhängigkeit von den Eingängen mit Werten belegt. Arcade.PLC unterstützt dazu bereits die IEC 61131-Programmiersprachen Strukturierter Text, Anweisungsliste und den proprietären STEP 7-Dialekt Statement List. Die Programme werden in eine vereinfachte Zwischensprache, die lediglich Zuweisungen, Sprünge, Aufrufe und bedingte Verzweigungen kennt, übersetzt, um eine einheitliche Basis für weitere Analysen zu haben. Das Model-Checking geschieht auf dem Zustandsraum, welcher durch die Simulation des SPS-Programmes in dieser Zwischensprache erzeugt wird. Da Model-Checking auf dem tatsächlichen Zustandsraum, aufgrund dessen Größe, praktisch unmöglich ist, implementiert Arcade.PLC Abstraktionsmethoden, welche die Zahl der benötigten Zustände reduzieren. Insbesondere wird nicht mit expliziten Werten, sondern mit Wertintervallen gearbeitet, um Verhalten, welches bei bestimmten Gruppierungen von Eingaben auftritt, in jeweils einem Zustand zusammenzufassen. Das Wertintervall eines Eingangs entspricht dabei zu Beginn dem Wertebereich des verwendeten Datentyps. Kommt es bei der Verwendung des Werteintervalls während der Simulation zu Nichtdeterminismus im Kontrollfluss, wird dieser dadurch aufgelöst, dass die relevanten Werteintervalle aufgespaltet und feiner gewählt werden. So führt z.B. die Anweisung „IF (A < 127)“ dazu, dass die möglichen Werteintervalle der Eingangsvariable A (vom Typ Byte) vom Intervall [0,255] auf die Intervalle [0, 126], [127,255] aufgespalten wird. Im Allgemeinen ist die Auflösung von Nichtdeterminismus im Kontrollfluss allerdings nicht einfach und stark von der verwendeten Abstraktionsdomäne abhängig, so dass bei der Auswertung von Ausdrücken mit mehr als einer Variable oder Funktionsaufrufen in vielen Fällen nur zu expliziten Werten aufgelöst werden können, wodurch die Zustandsmenge sich an der Stelle exponentiell vergrößert.

Zielsetzung

Ziel ist die Implementierung zweier Abstraktionsmethoden, welche die Zustandsexplosion bei Auswertungen von Funktionsaufrufen oder Berechnungen mit mehreren Variablen in bedingten Verzweigungen aufschieben soll, bis ein potenzielles Gegenbeispiel die Auflösung erzwingt. Die Abstraktionsmethoden sollen kompatibel zu bereits existierenden Funktionalitäten sein und im Vergleich zum Model-Checking ohne Abstraktionen ausgewertet werden.

Vorgehensweise

Ausgangspunkt ist die Zwischensprache. Es wird anhand einer statischen Analyse festgestellt, wo es

durch eine Auswertung einer Funktion oder einer Berechnung mit mehreren Variablen zu einer Zustandsexplosion kommen kann. Für solche Stellen wird eine neue Boolesche Eingabevariable erzeugt und der problematische Ausdruck durch diese ersetzt. Beim Model-Checking kann es nun sein, dass ein Gegenbeispiel zu einer Formel gefunden wird. Aufgrund der Überapproximation, die aus der Ersetzung des Ausdrucks resultiert, muss nun geprüft werden, ob das Gegenbeispiel realisierbar ist oder nur aus der Überapproximation resultiert. Dazu wird der ursprüngliche Ausdruck analysiert und falls nötig wieder eingefügt. Die Implementierung wird mit verschiedenen SPS-Programmen getestet.

Ansprechpartner

- Dr. rer. nat. Sebastian Biallas