



Abschlussarbeit

Synthese elementarer Petrinetze mittels SAT-Solver

Hintergrund

Bei der Verwendung und Analyse von Petrinetzen werden oft Erreichbarkeitsgraphen erzeugt. Bei diesen handelt es sich formal um (beschriftete) Transitionssysteme. Der Algorithmus zur Erzeugung des Erreichbarkeitsgraphen basiert im wesentlichen auf wiederholtes Feuern von Transitionen, ist also leicht nachvollziehbar und seit langem bekannt.

Die Umkehrung wird als *Petrinetz-Synthese* bezeichnet. Bei dieser wird aus einem Transitionssystem ein Petrinetz erzeugt, dessen Erreichbarkeitsgraph isomorph zum ursprüngliche Erreichbarkeitsgraphen sein soll. Synthese-Algorithmen sind schwieriger als der Erreichbarkeitsgraphen-Algorithmus, auch der Berechnungsaufwand dieser Algorithmen ist groß; sie benötigen bereits bei kleinen Eingaben relativ viel Rechenzeit und Speicher.

Daher ist die Petrinetz-Synthese weiterhin Gegenstand aktueller Forschung. Es werden Algorithmen entwickelt, die schneller sind, jedoch nur funktionieren, wenn das Transitionssystem bestimmte Eigenschaften erfüllt oder wenn das erzeugte Petrinetz bestimmte Eigenschaften aufweisen soll.

Aufgabenstellung

Implementierung der Petrinetz Synthese für *elementare Petrinetze* unter Verwendung eines SAT-Solvers:

Elementare Petrinetze sind Petrinetze mit bestimmten Eigenschaften. Es gibt für sie einen speziellen Synthese-Algorithmus, für den wir aus einer früheren Abschlussarbeit bereits eine Implementierung haben. Diese Implementierung arbeitet mittels Backtracking, probiert also durch Annahmen, deren Auswirkungen analysiert werden und die ggf. wieder zurückgenommen werden, solange, bis eine Lösung gefunden wurde. Stattdessen ist auch eine Beschreibung mittels aussagenlogischen Formeln möglich. Aus deren erfüllenden Belegungen, die mittels eines SAT-Solvers bestimmt werden können, lassen sich elementare Petrinetz ablesen.

Da SAT-Solver sehr auf Geschwindigkeit optimiert sind, ist es gut vorstellbar, dass dieser Ansatz schneller ist, als die existierende Implementierung.

Die Implementierung soll in Java als im Rahmen unserer Software APT (Analyse von Petrinetzen und Transitionssystem) erfolgen.

Organisatorisches

- Möglicher Anfangszeitpunkt: jederzeit
- Art der Arbeit: **Bachelorarbeit**
- Nützliche Vorkenntnisse: Aussagenlogik, Petrinetze, Transitionssysteme

Ansprechpartner

Valentin Spreckels, M. Sc.

(0441) 7 98 – 24 11

spreckels@informatik.uni-oldenburg.de