Опубликовано в материалах 2-й межвузовской научной конференции по проблемам информатики СПИСОК-2011, с. 368-369.

Я. М. Малаховски
Санкт-Петербургский государственный университет информационных технологий, механики и оптики

Применение систем типов для валидации и верификации автоматных программ

Валидация и верификация являются ключевыми требованиями при разработке ответственных систем. Автоматное программирование [1] позволяет значительно упростить процесс верификации, поскольку переход от автоматной программы к модели Крипке может быть произведен автоматически и изоморфно [2]. В работе [3] был предложен подход к реализации событийных конечных автоматов [1] на функциональных языках программирования и показаны его преимущества перед традиционными реализациями. При этом для валидации функций переходов использовались свойства алгебраических типов данных. В работе [4] было предложено обобщение данного подхода на структурные автоматы с переменными. Однако валидация производилась не в процессе компиляции, а в процессе конструирования автомата во время работы программы.

Валидация отдельного автомата обычно не требует много времени, поскольку, на практике, размер формул на дугах графа переходов весьма мал. При этом время валидации системы автоматов есть сумма времен валидации каждого автомата в отдельности. Таким образом, добавление нового автомата не сильно увеличивает задержку при запуске программы, а потому

результаты работы [4] являются пригодными для практического использования. Однако, для верификации данный подход не применим, поскольку время верификации системы автоматов есть функция от произведения мощностей множеств состояний каждого автомата. Таким образом, верификация программы должна быть частью процесса ее сборки и не может производится в процессе исполнения программы. Недостатком существующих средств верификации [2] является то, что спецификация программы, выраженная на языке, понятном верификатору, не является частью самой программы.

В настоящей работе предлагается метод верификации автоматных программ с использованием темпоральной логики CTL, производимый во время компиляции компилятором языка на котором выражены как автоматная программа, так и ее спецификация.

Разработанный подход основан на использовании зависимых систем типов в комбинации со встроенными предметно-ориентированными языками программирования (eDSL). При этом, при помощи встроенных предметно-ориентированных языков выражаются автоматные программы и спецификации на языке CTL, а проверка того, что автоматная программа удовлетворяет спецификации производится при помощи доказательства утверждений в зависимой системе типов функционального языка программирования.

Для апробации и практического применения предложенного подхода был выбран язык программирования Agda (версии 2). eDSL для описания автоматов был, насколько это возможно, перенесён из работы [4].

Литература

1. *Поликарпова Н.И., Шалыто А.А.* Автоматное программирование. СПб.: Питер. 2010. 176 с.
2. *Вельдер С.Э., Лукин М.А. Шалыто А.А., Яминов Б.Р.* Верификация автоматных программ. СПб.: ИТМО, 2011.
3. *Малаховски Я.М., Шалыто А.А.* Реализация конечных автоматов на функциональных языках программирования //Информационно-управляющие системы. 2009. №6, с. 30–33.
4. *Малаховски Я.М., Корнеев Г.А.* Валидация автоматов с переменными на функциональных языках программирования //Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий, механики и оптики. 2010. № 6, с. 73–77.