

Различимость *UML*-описаний с использованием временных автоматов

Галимуллин Р.Ф.

Кафедра информационных технологий в исследовании дискретных структур
Национальный исследовательский Томский государственный университет
Томск, Российская Федерация
e-mail: nihilkhaos@gmail.com

Аннотация—В работе ставится задача различимости *UML*-описаний на основе преобразования конечно-автоматных диаграмм *UML* во временные автоматы с тайм-аутами, для которых известны отношения различимости и способы построения соответствующих различающих последовательностей.

Ключевые слова: *UML*; временной автомат с тайм-аутами

I. ВВЕДЕНИЕ

Как правило, документация при проектировании даже простых систем занимает не один десяток страниц, что существенно повышает сложность составления, проверки и использования такой документации. Дополнительные проблемы могут возникнуть в том случае, когда в разработке участвуют несколько научных групп, возможно, из разных регионов и/или стран, или при необходимости в кратчайшие сроки ввести нового сотрудника в курс дела. Для решения этих и многих других проблем существуют языки визуального моделирования, позволяющие унифицировать описание спецификации и, соответственно, значительно повысить производительность процесса проектирования. Один из таких языков, *the Unified Modelling Language (UML)*, стал *de facto* стандартом визуального моделирования [1]. *UML* широко используется при описании архитектуры программного обеспечения, электронных, механических и других типов устройств, бизнес-процессов и т.д. В общем случае, моделирование средствами *UML* осуществляется при помощи одной или более диаграмм. Диаграммы можно условно разделить на две категории: структурные диаграммы (*structural diagrams*) и поведенческие диаграммы (*behavioral diagrams*). В настоящем докладе мы рассматриваем одну из разновидностей поведенческих диаграмм - конечно автоматные диаграммы (*State Machine Diagram*). Программные средства, располагающие возможностью работы в среде *UML*, почти всегда позволяют автоматически создать шаблон программного кода по представленным диаграммам (т.н. кодогенерация) [2]. Однако нам не известны методы тестирования с гарантированной полнотой таких программных реализаций. Для синтеза тестов обычно используются автоматные модели, поэтому необходимо разработать процедуру перехода от *UML*-диаграммы к автомату

или композиции автоматов, по которым тесты могут быть построены известными способами [3].

II. ВРЕМЕННОЙ НЕДЕТЕРМИНИРОВАННЫЙ АВТОМАТ С ТАЙМ-АУТАМИ

Конечный автомат служит для описания поведения системы, переходящей из состояния в состояние под действием входных символов и производящей при этом выходные реакции, которые, в общем случае, зависят не только от вновь поступившего входного символа, но и от предыдущих входных символов. Если поведение автомата определено на всех входных последовательностях, то автомат называется *полностью определенным*. Если на некоторых входных последовательностях поведение автомата не определено, то автомат называется *частично определенным*, или *частичным*. Обычно, спецификация моделируется частичным автоматом, однако поведение тестируемой реализации системы описывается полностью определенным автоматом. Если входной символ, подаваемый на автомат, обрабатывается единственным образом, то автомат называется *детерминированным*, иначе - *недетерминированным* [4]. Введение недетерминизма позволяет расширить описательные способности модели, добавив возможность выбора следующего состояния и выходной реакции. Однако, как показывает практика, поведение большинства современных дискретных систем существенно зависит от времени, поэтому нашей рабочей моделью является модификация классического автомата, а именно временной автомат с тайм-аутами для входных и выходных символов (далее просто *временной автомат*). Если на временной автомат в текущем состоянии в течение определенного времени не было подано ни одного входного символа, то автомат может перейти в новое состояние-преемник; такие переходы называются переходами по временной задержке (*тайм-аут по входному символу* или просто тайм-аут в настоящем докладе). Кроме того, получив входной символ, автомат выдает выходной символ с некоторой задержкой, которая называется *тайм-аутом по выходному символу*.

III. РАЗЛИЧИМОСТЬ ВРЕМЕННЫХ НЕДЕТЕРМИНИРОВАННЫХ АВТОМАТОВ С ТАЙМ-АУТАМИ

Для детерминированных и недетерминированных автоматов известны различные отношения конформности (соответствия) между реализацией и спецификацией [4], а так же разработаны методы

синтеза проверяющих тестов с гарантированной полнотой относительно этих отношений. Поскольку рассматриваемый временной автомат является модификацией классического конечного автомата, то подобные отношения конформности (или, соответственно, различимости) можно ввести и для временных недетерминированных автоматов на множестве временных входных последовательностей.

Временным входным символом называется пара $(i, t) \in I \times (\mathbf{N} \cup \{0\})$, где t – время подачи входного символа i . Последовательность временных входных символов называется *временной входной последовательностью*. Временной выходной символ есть пара $(o, k) \in O \times (\mathbf{N} \cup \{0\})$, т.е. символ o выдается через время k . Соответственно, последовательность выходных временных символов есть *временная выходная последовательность*. Пусть α – временная входная последовательность, β – временная выходная последовательность, тогда пара α/β называется *временной трассой* автомата в данном состоянии. Множество всех временных трасс автомата S в данном состоянии s обозначим $traces_S(s)$. Два состояния s и p называются *эквивалентными*, если $traces_S(s) = traces_P(p)$. Полностью определенные временные автоматы S и P *эквивалентны*, если их начальные состояния эквивалентны, т.е. $traces_S = traces_P$; иначе, автоматы называются *различимыми*. Временные входные последовательности, для которых выходные реакции автоматов отличаются, называются *различающими последовательностями*. Временной автомат S есть *редукция* временного автомата P , если $traces_S \subseteq traces_P$. Различающие последовательности относительно эквивалентности и редукции для временных автоматов строятся так же как различающие последовательности для классических (невременных) автоматов. Два временных автомата называются *разделимыми*, если существует временная входная последовательность, такая, что при её подаче множества выходных реакций автоматов не пересекаются. Такая последовательность называется *разделяющей*. Существуют различные методы построения разделяющих последовательностей [5].

IV. ЗАКЛЮЧЕНИЕ

Для классических временных полуавтоматов существует инструментальное средство, позволяющее синтезировать временные автоматные модели по *UML* описанию. Этим средством является *UPPAAL*, представляющее собой результат сотрудничества университетов Уппсалы (*Uppsala, Sweden*) и Аалборга (*Aalborg, Denmark*) [6]. Однако известно, что множество временных автоматов, рассматриваемых в данной работе, не является подклассом множества классических временных полуавтоматов [7]. Таким образом, имеется возможность дополнительно описать некоторые возможные несоответствия проверяемой реализации. Изучение различных отношений конформности для используемой в данной работе модели временного автомата было начато лишь недавно [8,9], и методы синтеза проверяющих тестов с гарантированной полнотой для таких автоматов требуют дальнейших исследований.

- [1] Pitone D., Pitman N., “UML 2.0 in a Nutshell”, O’Reilly Media. Sebastopol, p. 240, 2005.
- [2] Рамбо Д., Блаха М. “UML 2.0. Объектно-ориентированное моделирование и разработка”, Питер. СПб, 2007, 544 с.
- [3] Petrenko A., Yevtushenko N., Bochman G.v., Dissouli R. “Testing in context: framework and test derivation”, Computer communications, vol.19, pp.1236 – 1249, 1996.
- [4] Евтушенко Н.В., Петренко А.Ф., Ветрова М.В. “Недетерминированные автоматы: анализ и синтез. Ч.1. Отношения и операции: учебное пособие”, Томск: Томский государственный университет, 2006, 142 с.
- [5] Gromov M., El-Fakih K., Shabaldina N., Evtushenko N. “Distinguishing non-deterministic timed finite state machines”, Formal Techniques for Distributed Systems. – Germany: Springer, 2009. – p.137 - 151
- [6] UPPAAL [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://www.uppaal.org/>.
- [7] Громов М.Л. “Разработка методов синтеза условных тестов для автоматных моделей с недетерминированным поведением”, дисс. канд. физ-мат. наук: 05.13.01. – Томск: ТГУ, 2009. – 154 с.
- [8] Громов М.Л., Евтушенко Н.В. “Синтез различающих экспериментов для временных автоматов”, Программирование. 2010. № 4. с. 1–11.
- [9] Shabaldina N., Galimullin R. “On Deriving Test Suites for Nondeterministic Finite State Machines with Time-Outs”, Programming and computer science, Vol.38, pp.127-133, 2012.