

ПРИМЕНЕНИЕ МЕТОДОВ РЕШЕНИЯ ЗАДАЧИ О ВЫПОЛНИМОСТИ КВАНТИФИЦИРОВАННОЙ БУЛЕВОЙ ФУНКЦИИ ДЛЯ ПОСТРОЕНИЯ УПРАВЛЯЮЩИХ КОНЕЧНЫХ АВТОМАТОВ ПО СЦЕНАРИЯМ РАБОТЫ И ТЕМПОРАЛЬНЫМ СВОЙСТВАМ

Е.В. Панченко, В.И. Ульянцев

Научный руководитель — кандидат наук, ассистент

Ф.Н. Царев

Аннотация: Данная статья рассматривает вопрос о применении управляющих автоматов при построении систем со сложным поведением. Предложен метод построения управляющего конечного автомата по заданному множеству сценариев работы и темпоральным свойствам, которые должны выполняться в результирующем автомате. Метод основан на сведении задачи к задаче выполнимости квантифицированной булевой функции. Описан алгоритм построения данной функции и основные составляющие полученной булевой формулы.

Ключевые слова: кванторы, булева функция, управляющие конечные автоматы, верификация.

Введение

Парадигма автоматного программирования используется для реализации систем со сложным поведением во многих промышленных отраслях [1]. В настоящее время существуют системы, насчитывающие более тысячи различных состояний, при записи их поведения в виде конечного автомата. Одним из используемых подходов, при попытке задать параметры системы на вход сторонним приложениям является передача сценариев работы системы, по которым эвристически строится необходимый конечный автомат.

Как правило, данный метод позволяет эффективно создавать автоматизированные системы по сценариям работы, описываемым, например, в функциональной спецификации продукта. Тем не менее, не все свойства системы можно описать с помощью сценариев работы.

В данной работе приведено решение, которое позволяет строить автоматы не только по сценариям работы, но также и по темпоральным свойствам, заданным с помощью подмножества языка LTL, с помощью сведения решения данной задачи к решению задачи выполнимости обобщенной булевой формулы.

Цель работы

В работе [2] было спроектировано решение, позволяющее решить задачу построения автомата по сценариям работы с помощью сведения ее к за-

даче SAT, задаче о выполнимости булевой формулы. После этого SAT-solver находил решение данной формулы, и на основании полученного примера алгоритм строил необходимый автомат.

Полученный автомат удовлетворял заданным сценариям работы, однако зачастую не соответствовал изначальным бизнес-требованиям, поскольку лист сценариев оказывался не полон. Целью настоящей работы является модификация метода построения управляющих автоматов, а именно добавление возможности использования LTL-формул в качестве дополнительного инструмента описания требуемой работы автомата.

Базовые положения исследования

На вход разрабатываемой программе подается список сценариев, а также набор темпоральных свойств работы системы, заданный с помощью синтаксиса языка линейной темпоральной логики LTL.

Сценарием работы является последовательность $T_1 \dots T_n$ троек $T_i = \langle e_i, f_i, A_i \rangle$, где e_i — входное событие, f_i — булева формула от входных переменных, задающая охранный условие, A_i — последовательность выходных воздействий. Автомат, находясь в состоянии S , удовлетворяет элементу сценария T_i , если из S исходит переход, помеченный событием e_i , последовательностью выходных воздействий A_i и охранным условием, тождественно равным f_i как булева формула. Автомат удовлетворяет сценарию работы $T_1 \dots T_n$, если он удовлетворяет каждому элементу данного сценария, находясь при этом в состояниях пути, образованного соответствующими переходами.

На формат входных LTL наложено ограничение в виде невозможности использовать характеристики состояний автомата, поскольку на этапе задания логики, состояний еще нет. Однако, это позволяет задавать свойства общего формата, что может быть полезно при создании систем с нуля. Было принято решение использовать в качестве переменных LTL формул входные и выходные воздействия. Таким образом, синтаксис входных данных включает в себя:

- Булевы связки ($!, \wedge, \vee$);
- Темпоральные операторы **X** (neXt) и **U** (Until);
- Выведенные темпоральные операторы **F**(Future), **G**(Globally in the future), **R**(Release);
- Предикаты:
 - $wasEvent(e)$ — переход совершен по событию e
 - $wasAction(z)$ — во время перехода было вызвано выходное воздействие z .

Пример входного темпорального свойства, используемого при верификации часов с будильником в работе [3]:

$$\mathbf{F} (! (wasEvent(e) \wedge !wasAction(z))).$$

Разработанное программное средство в несколько шагов производит построение искомого конечного автомата, удовлетворяющего входным данным:

1. С помощью разработанного в работе [2] алгоритма по заданным сценариям работы создается булева формула, содержащая логические переменные $y_{a,b,e,f}$ (для каждой пары состояний результирующего автомата a и b , каждого события e , каждой формулы f , встречающейся в сценариях), соответствующие наличию перехода из состояния a в состояние b , помеченного событием e и формулой f в результирующем автомате.
2. С использованием подхода верификации моделей с ограничением на длину вычислений, Bounded Model Checking [3], входные темпоральные свойства «разворачиваются» в булеву функцию. Для этого используется понятие «обратного цикла» и ограничение его глубины поиска.

«Обратный цикл» — это цикл, образованный ребром, ведущим из какого-либо состояния пути в состояние, лежащее на данном пути ранее, как показано на Рис. 1.

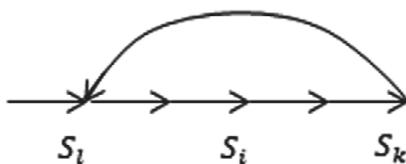


Рис. 1. Пример обратного $(k, 1)$ -цикла

Логика линейного предполагает, что некоторое утверждение будет выполняться для всех путей. Поэтому в работе [4] доказательство построено от противного, проверяя существование пути, на котором выполняется отрицание LTL формулы.

Однако, данный подход возможен только при верификации уже построенных автоматов. В нашем случае автомата еще только должен быть построен, поэтому используется квантор всеобщности, позволяя найти такое решение формулы, значения переменных $y_{a,b,e,f}$, чтобы темпоральные свойства выполнялись на всех путях построенного конечного автомата.

3. Для ограничения времени работы алгоритма устанавливается ограничение на длину цикла k . Таким образом, темпоральные свойства с помощью разложения на композицию темпоральных и булевых предикатов «разворачиваются» по циклу в формулу, размер которой линейно зависит от размера начальной формулы и константы k .
4. Полученные формулы объединяются в одну квантифицированную булеву функцию, проверяющую все бесконечные и конечные пути в радиусе k и содержащую кванторы существования и всеобщности.

5. Для решения полученной формулы используется программное средство *DepQBF*[5]. Полученные значения входных переменных с квантором существования используются для построения искомого управляющего конечного автомата.

Заключение

Разработан алгоритм автоматизированного построения управляющих автоматов по сценариям работы и темпоральным свойствам, основанный на сведении данных задач к проблеме о разрешимости квантифицированной булевой формулы. В настоящий момент производится сравнение качества и скорости работы с методами, реализованными в работах [3], [6].

Литература

1. *Поликарпова Н.И., Шалыто А.А.* Автоматное программирования. СПб: Питер, 2009
 2. *Ulyantsev V., Tsarev F.* Extended Finite-State Machine Induction using SAT-Solver / Proceedings of the Tenth International Conference on Machine Learning and Applications, ICMLA 2011, Honolulu, HI, USA // IEEE Computer Society, 2011. — V. 2. — P. 346–349.
 3. *Егоров К.В., Шалыто А.А.* Совместное применение генетического программирования и верификация моделей для построения автоматов управления системами со сложным поведением
 4. *Biere A., Cimatti A., Clarke E.M., Zhu Y., Strichman O.* Bounded Model Checking
 5. *DepQBF QBF-solver.* <http://lonsing.github.io/depqbf/>
 6. *Walkinshaw N., Bogdanov K.* Inferring Finite-State Models with Temporal Constraints
-