

Визуализатор работы конечных детерминированных автоматов

Пронин А.А.¹, Мещерякова О.В.²

¹ *gorka19800@gmail.com* ² *olyames@mail.ru*

Саратовский государственный университет имени Н.Г. Чернышевского

Статья посвящена разработке визуализатора работы одного и нескольких конечных детерминированных автоматов для ПК, с применением фреймворка window forms и языка программирования C#.

Ключевые слова: конечные автоматы, C#, window forms C#.

Введение

Многочисленные задачи моделирования в различных областях требуют наблюдения за поведением объекта, фиксации внутренних состояний и формирования определенных выходных реакций. Одним из удобных средств такого описания объекта стал конечный детерминированный автомат.

Ряд задач взаимодействия автомата с окружающей средой показали необходимость построения композиции автоматов с помощью последовательного и параллельного соединения. А это, в свою очередь, вызвало необходимость визуализации функционирования автомата.

Автоматы имеют широкое применение при разработке программного обеспечения, проверяющего соответствие заданному условию: «лексический анализатор» компилятора, программное обеспечение для разработки и проверки цифровых схем, и т.д. [1].

Основные понятия

Конечный детерминированный автомат представляется следующим множеством: $(S, X, Y, \delta, \lambda, S_0)$, где S – множество состояний автомата, X – множество входных сигналов, Y – множество выходных сигналов, $\delta: \delta(s \in S, x \in X) \rightarrow s' \in S$, $\lambda: \lambda(s \in S, x \in X) \rightarrow y \in Y$, S_0 – множество начальных состояний. Рассмотрим 2 типа соединений: последовательное и параллельное.

Параллельное соединение автоматов

Пусть даны автоматы $A_i = (S_i, A_i, B_i, \delta_i, \lambda_i)$, $S_{0i} = S_i, i \in \{1, 2\}$. Определим произведение автоматов A_1 и A_2 как $A_1 \times A_2 = (S_1 \times S_2, A_1 \times A_2, B_1 \times B_2, \delta, \lambda)$, где $\delta((s_1, s_2), (a_1, a_2)) = (\delta_1(s_1, a_1), \delta_2(s_2, a_2))$, $\lambda((s_1, s_2), (a_1, a_2)) = (\lambda_1(s_1, a_1), \lambda_2(s_2, a_2))$, для любых $(s_1, s_2), (a_1, a_2) \in A_1 \times A_2$.

Автомат $A_1 \times A_2$ часто называют параллельным соединением (или параллельной композицией A_1 и A_2 , так как S_1 и S_2 можно интерпретировать как множества состояний из двух параллельных блоков. Входы A_i воздействуют на S_i , порождая выходы $B_i, i = \{1, 2\}$; $A_1 \times A_2$ действует на $S_1 \times S_2$, порождая выходы $B_1 \times B_2$ (рис. 1).

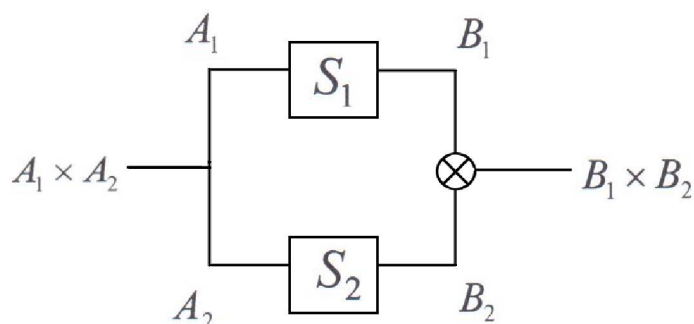


Рис. 1. Параллельное соединение автоматов

Последовательное соединение

Пусть A_1, A_2 , такие же, как и в параллельном соединении, но с дополнительным допущением: $A_2 = B_1$. Определим последовательное соединение (или последовательную комбинацию) автоматов A_1 и A_2 как автомат $A_1 \# A_2 = (S_1 \times S_2, A_1, B_2, \partial, \lambda)$, где

$\delta((s_1, s_2), a_1) = (\delta_1(s_1, a_1), \delta_2(s_2, \lambda_1(s_1, a_1)))$, $\lambda((s_1, s_2), a_1) = (s_2, \lambda_2(s_2, a_1))$, для любых $(s_1, s_2) \in S_1 \times S_2, a_1 \in A_1$. Этот автомат работает следующим образом: вход $a_1 \in A_1$, действуя на s_1 , определяет переход A_1 в состояние $s'_1 = \delta_1(s_1, a_1)$, и выход $b_1 = \lambda_1(s_1, a_1) \in B_1 = A_2$; выход b_1 действует на s_2 , преобразуя его в $s'_2 = \delta_2(s_2, b_1)$ и порождая выход $\lambda_2(s_2, b_1)$. После этого автомат $A_1 \# A_2$ оказывается в состоянии (s'_1, s'_2) (рис. 2).

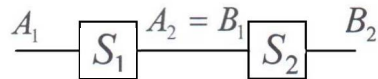


Рис. 2. Последовательное соединение автоматов

Описание работы программы

В случае функционирования одного автомата загружаются таблицы переходов – выходов (рис. 3). После ввода таблиц становится возможным наблюдать состояния и выходные сигналы для каждого начального состояния. В случае композиции автоматов необходимо построить соединение, расположить автоматы по уровням и для каждого элемента соединения первого уровня ввести множества входных сигналов, состояний. Для элементов иных уровней нужно ввести лишь таблицы переходов-выходов. Причем множества входных и выходных сигналов строятся в соответствии с определением соединения. Под параллельным соединением автоматов подразумевается набор параллельно соединенных конечных автоматов (рис. 4).

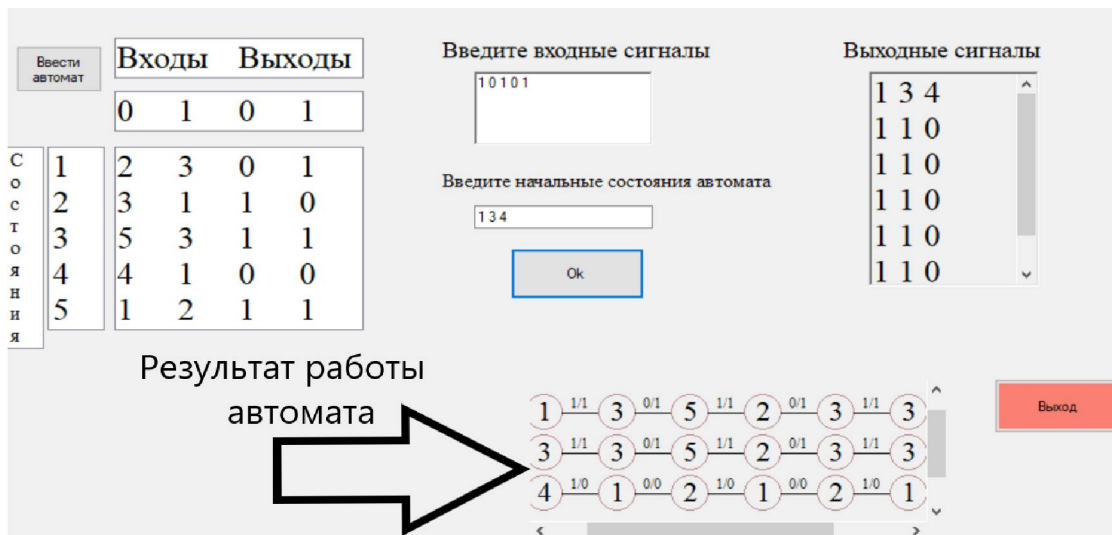


Рис. 3. Внешний вид окна программы в режиме работы одного автомата



Рис. 4 Внешний вид конструктора композиции автоматов.

Программная реализация

В качестве средств разработки использовался язык C# версии 5.0 [2], а также интерфейс программирования приложений, отвечающий за графический

интерфейс пользователя Windows Forms [3]. Язык программирования C# известен своей красотой и удобностью в использовании, особенно, при создании приложений с пользовательским интерфейсом. Конструктор интерфейса Windows Forms позволяет работать с элементами окна, даже при небольшом опыте разработки. Работа автомата реализуется двумя статическими методами, представляющими функции переходов и выходов. Реализация работы одного автомата довольно тривиально: для каждого из введенных начальных состояний необходимо получить реакции автомата на все введенные входные сигналы, после чего по результатам работы автомата происходит их графическая визуализация. Для реализации разбиения автоматов по группам, в зависимости от их расположения, использовались координаты ярлыков в конструкторе композиции автоматов.

Заключение

В результате работы удалось реализовать программу-визуализатор для проведения испытаний над одним или несколькими соединенными между собой автоматами. Предложенный визуализатор функционирования автомата будет полезен в качестве демонстратора при чтении лекций и проведении практических занятий по - различным дисциплинам, использующих автоматное моделирование и дисциплинам, связанным с архитектурой ЭВМ, а также при создании on-line курсов по моделированию. В дальнейшем планируется переход от формата приложения для ПК к формату веб-страниц, а также расширение функциональности программы – возможность проведения безусловных экспериментов над автоматами.

Список литературы

- [1] *Хоткрофт, Джон, Э., Мотвани, Раджив, Ульман, Джеффри, Д.* Введение в теорию автоматов, языков и вычислений, 2-е изд. : Пер. с англ. – М. : Издательский дом “Вильямс”, 2008. — 528 с (дата обращения: 05.09.2020).
- [2] *Албахари, Джозеф, Албахари, Бен* C# 5.0 правочник. Полное описание языка.. : Пер. с англ. – М. : Издательский дом “Вильямс”, 2014. – 1008 с (дата обращения: 17.06.2020).
- [3] Официальный ресурс компании Microsoft по интерфейсу Windows Forms. [Электронный ресурс] URL: <https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/?view=netframeworkdesktop-4.8>