РЕДАКТОР СИНТАКСИЧЕСКИХ ДИАГРАММ

Ю.Д. Рязанов, А.В. Штырь

Белгородский государственный технологический университет им. В.Г. Шухова

Развитие информационных технологий приводит к необходимости создания новых трансляторов языков программирования. Для этого в настоящее время широко используются системы автоматизированного построения трансляторов (САПТ) [1 — 4]. Эффективность использования САПТ определяется языком описания спецификации транслятора. Как правило, спецификация транслятора в этих системах представляется в нотации РБНФ или ее модификациях. Одним из удобных способов задания спецификации транслятора является транслирующая синтаксическая диаграмма, которая практически не используется в САПТ. Здесь дается формальное определение транслирующей синтаксической диаграммы и описывается разработанное программное средство — редактор синтаксических диаграмм, которое может быть использовано в САПТ.

Транслирующую синтаксическую диаграмму будем задавать четверкой $D = (T, N, S, A, G)$, где T — конечное множество терминалов; N — конечное множество нетерминалов; $S \in N$ — начальный нетерминал; A — конечное множество действий, $G = (V, E)$ — ориентированный граф, где

$V = V_T \cup V_N \cup V_A \cup V_u \cup V_{вход} \cup V_{выход}$, где

$V_{вход}$ — множество точек входа, $|V_{вход}| = |N|$;

V_T — множество терминальных вершин;

V_N — множество нетерминальных вершин;

V_A — множество вершин действий;

V_u — конечное множество узлов;

$V_{выход}$ — множество точек выхода, $|V_{выход}| = |N|$;

$E = E_1 \cup E_2 \cup E_3 \cup E_4 \cup E_5 \cup E_6$, где

$E_1 \subseteq \{(a, b) \mid a \in V_{вход}, b \in V_u\}$ — множество входных дуг;

$E_2 \subseteq \{(a, b) \mid a \in V_u, b \in V_{выход}\}$ — множество выходных дуг;

$E_3 \subseteq \{(a, b) \mid a \in V_u, b \in V_T \cup V_N\}$ — множество дуг, выходящих из узлов;

$E_4 \subseteq \{(a, b) \mid a \in V_T \cup V_N \cup V_A, b \in V_u\}$ — множество дуг, входящих в узлы;

$E_5 \subseteq \{(a, b) \mid a \in V_N \cup V_T, b \in V_A\}$ — множество дуг входящих в вершину действие;

$E_6 \subseteq \{(a, b) \mid a \in V_u, b \in V_u\}$ — множество ε-дуг, соединяющих узлы.

Каждому нетерминалу соответствует связная компонента графа. Компонента именуется соответствующим нетерминалом, имеет только одну точку входа и одну точку выхода и

конечное множество вершин других типов. Точки входа и выхода на диаграмме компоненты не изображаются. Нетерминальная вершина изображается прямоугольником, в который вписан терминальный символ. Узел изображается на диаграмме жирной точкой. В точку входа не входит ни одна дуга и выходит конечное множество дуг (входные дуги компоненты). Узлы, в которые входят входные дуги, называются начальными. Из точки выхода не выходит ни одна дуга и входит конечное множество дуг (выходные дуги компоненты). Узлы, из которых выходят выходные дуги, называются заключительными. Каждая дуга, за исключением входных и выходных дуг, может выходить из узла и входить в терминальную или нетерминальную вершину или другой узел, выходить из терминальной, нетерминальной или вершины действия и входить в узел, либо выходить из терминальной или нетерминальной вершины и входить в вершину действие. В каждую терминальную, нетерминальную вершину и в вершину действия входит только одна дуга и выходит только одна дуга. На количество дуг, входящих в узлы и выходящих из них, ограничений нет. На *рисунке 1* приведен пример транслирующей синтаксической диаграммы.

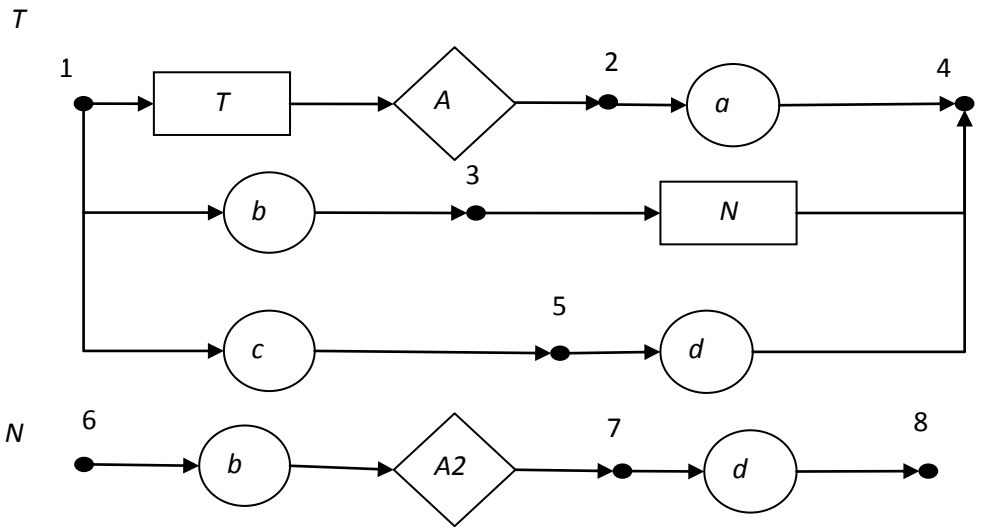

Рисунок 1 — Пример транслирующей синтаксической диаграммы.

Программа «Редактор синтаксических диаграмм» позволяет строить синтаксические диаграммы, которые соответствуют приведенному выше описанию. Пример диаграммы, построенной в данной программе, представлен на *рисунке 2*. Программа обладает множеством функций и возможностей. Главное меню содержит следующие пункты:

1) Изменить список нетерминалов;
2) Изменить список терминалов;
3) Создать диаграмму нетерминала;
4) Сформировать выходной файл;

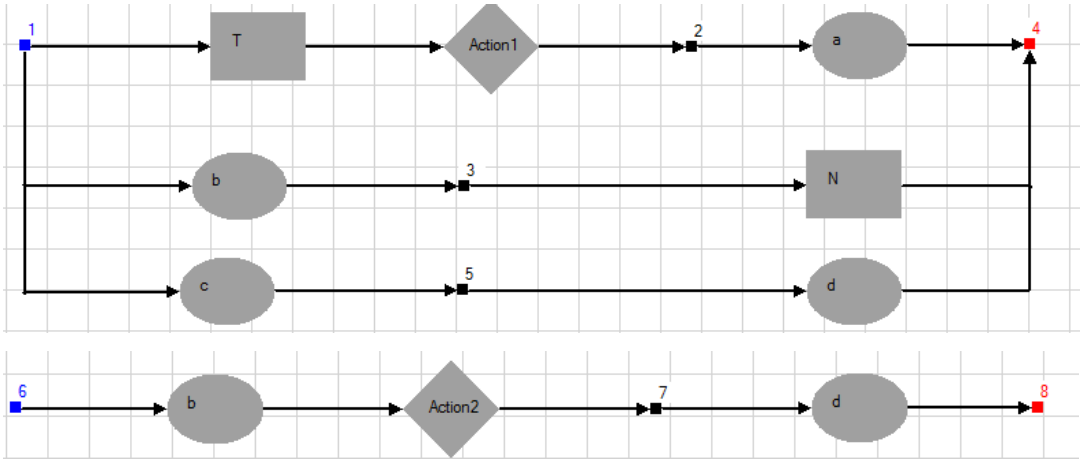

Рисунок 2 — Пример диаграммы, построенной в данной программе.

Интерфейс программы приведен на *рисунке 3*. Рабочая область программы разделена на три части — это панель инструментов, панель редактирования, панель параметров.

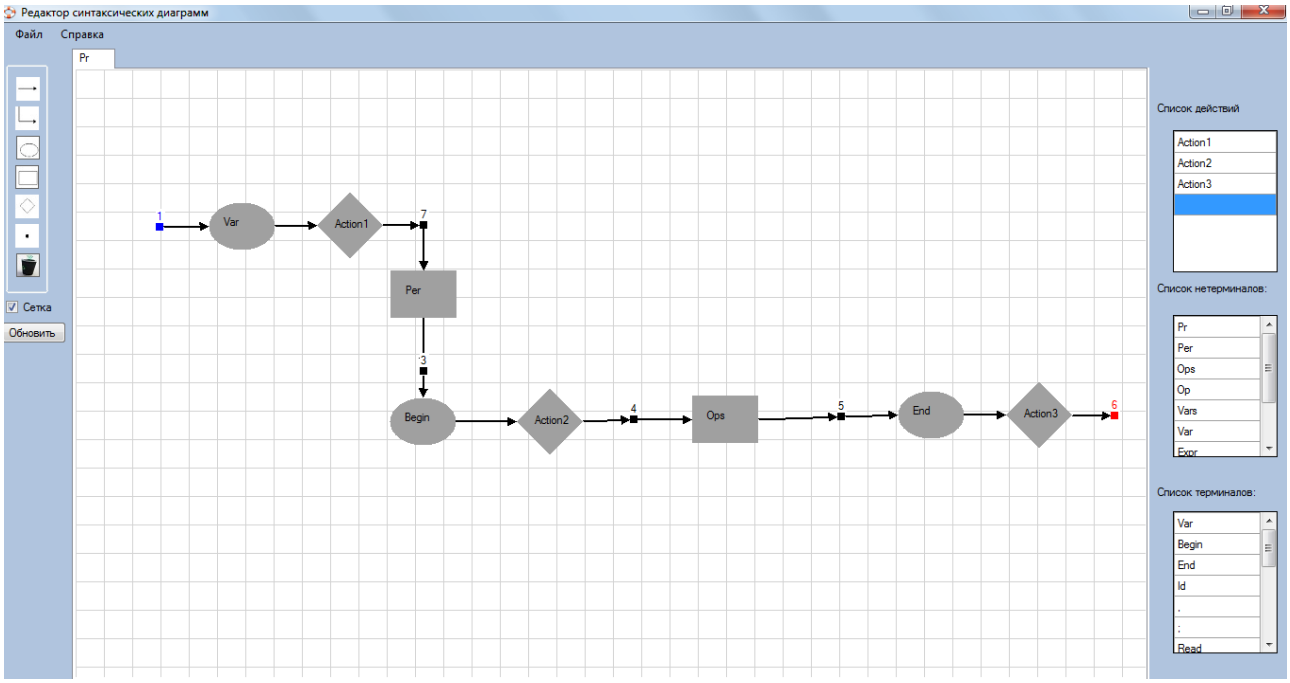

Рисунок 3 — Интерфейс программы.

После запуска программы для построения диаграммы необходимо задать список терминалов и нетерминалов. Затем выбрать пункт меню «Создать диаграмму нетерминала». Появится новая вкладка с окном редактирования.

В левой части рабочей области программы расположена панель инструментов, каждый из которых обозначен небольшой картинкой-пиктограммой. Соединительная прямая линия изображена прямой стрелкой, соединительная ломаная линия — ломаной стрелкой, тер-

минал — прямоугольником, нетерминал — овалом, действие — ромбом, узел — точкой, функция удаления — корзиной. Назначение инструментов следующее:

1) Соединительная прямая линия. Позволяет соединить терминал и узел, нетерминал и узел, нетерминал и действие, терминал и действие. Для соединения двух объектов необходимо последовательно нажать на эти объекты левой кнопкой мыши.

2) Соединительная ломаная линия. Позволяет соединить те же объекты что и прямая линия, но имеет точки перелома, количество которых задает пользователь. Для соединения объектов ломанной линией необходимо последовательно нажать объект из которого выходит линия, затем точки перелома линии и объект, в который входит линия.

3) Терминал. Позволяет добавить на панель редактирования объект терминал и задать его название из списка терминалов.

4) Нетерминал. Позволяет добавить на панель редактирования объект нетерминал и задать его название из списка нетерминалов.

5) Действие. Позволяет добавить на панель редактирования объект действие (процедуру, которую следует выполнить при трансляции).

6) Узел. Позволяет добавить на панель редактирования объект узел и задать тип узла (начальный, промежуточный, допускающий). Узлы разных типов отличаются цветом. Начальный узел — синего цвета, промежуточный — черного, допускающий — красного.

7) Удаление объекта. Позволяет удалять любой объект, расположенный на панели редактирования. Для этого необходимо выбрать данный пункт, а затем выбрать объект, который необходимо удалить.

8) Включение/отключение сетки. Позволяет включить или отключить разлиновку панели редактирования. Эта возможность помогает в выравнивании объектов.

По центру расположена панель редактирования. Это и есть главное окно, в котором происходит построение синтаксической диаграммы. На эту панель перетаскиваются объекты из панели инструментов.

Справа находится панель инструментов, которая содержит три основных списка: список действий, список нетерминалов и список терминалов.

Кроме основных функций существуют еще и дополнительные, которые используются для редактирования уже созданной синтаксической диаграммы:

1) Перемещение объекта. Любой объект, расположенный на панели редактирования, можно перетаскивать, при этом все входящие и выходящие линии тоже перемещаются вместе с объектом.

2) Переименовать объект. Для переименования необходимо дважды кликнуть по объекту, после чего появится дополнительное окно, в котором пользователь вводит новое имя объекта и нажимает кнопку «Сохранить».

3) Изменить тип узла. После двойного клика по объекту появляется дополнительное окно, в котором можно выбрать новый тип узла и нажать кнопку «Сохранить». В этом случае узел меняет свой цвет.

Синтаксическая диаграмма каждого нетерминала располагается в отдельной вкладке.

Программа формирует выходной файл, который описывает структуру синтаксической диаграммы. Этот файл может быть использован в САПТ для необходимых преобразований синтаксической диаграммы и генерации транслятора.

Программное обеспечение «Редактор синтаксических диаграмм» реализовано на языке C# и представляет собой совокупность следующих модулей:

1) Node.cs — класс для узлов.

Свойства: point — координаты узла,
label — номер узла,
type — тип узла;

Методы: Node_Load() — добавление узла на панель редактирования,
Node_MouseDoubleClick() — событие при двойном клике по узлу,
Node_MouseMove() — перемещение узла,

2) Components.cs — класс терминалов и нетерминалов.

Свойства: point — координаты компонента,
type — тип (термин или нетерминал),
name — название компоненты;

Методы: Components _Load() — добавление компонента на панель редактирования,
Components _MouseDoubleClick() — событие при двойном клике,
Components _MouseMove() — перемещение компонента;

3) Action.cs — класс действий.

Свойства: point — координаты действия,
Comp — компонент, после которого выполняется действие,
name — название действия;

Методы: Action _Load() — добавление действия на панель редактирования,
Action_MouseDoubleClick() — событие при двойном клике по действию,
Action _MouseMove() — перемещение действия,

4) Lines.cs — класс для линий.

Свойства: startPoint, endpoint — координаты начала и конца линии,

list — список переломных точек для ломаной линии,

startComp — компонент, из которого выходит линия,

endComp — компонент, в который входит линия.

MainModule.cs описывает логику работы программы, а также содержит реализацию ее интерфейса. В этом модуле описан класс формы mainForm, на которой размещены все основные компоненты для управления работой программы.

Таким образом, разработаны алгоритмы и программа для визуального представления синтаксических диаграмм, которая позволяет создавать, редактировать и представлять диаграммы в форме, удобной для дальнейшей обработки в САПТ.

Список литературы:

1. ANTLR.URL: http://www.antlr.org/ (дата обращения: 15.10.2013)

2. ASF+SDF.URL: http://www.cwi.nl/projects/MetaEnv/ (дата обращения: 15.10.2013)

3. Bison.URL: http://www.gnu.org/software/bison/ (дата обращения: 15.10.2013)

4. Coco/R.URL: http://www.ssw.uni-linz.ac.at/Research/Projects/Coco/ (дата обращения: 15.10.2013)