**Автоматные грамматики и языки**

Рассмотрим автоматные грамматики и языки подробнее, имея целью построение алгоритмов распознавания этого класса языков.

**Граф автоматной грамматики**

Для каждой автоматной грамматики можно построить направленный граф по следующим правилам:

1. Каждому нетерминальному символу грамматики ставится в соответствие вершина графа, которая помечается этим символом.

2. При наличии правил вида

А →a

добавляется дополнительная вершина, которая помечается символом К.

3. Каждое правило вида

А→аВ

порождает дугу графа, ведущую из вершины А в вершину В.



Дуга помечается символом **а**.

4. Каждое правило вида

А →a

порождает дугу графа, ведущую из вершины А в вершину К



Дуга помечается символом **а**

5. Вершина, соответствующая начальному нетерминалу, помечается стрелкой.

 

6. Вершина К и вершины, соответствующие нетерминалам, для которых есть правило

А →ε

помечаются как конечные. Мы будем изображать их двойным кружком.

 

Построим граф автоматной грамматики G10 (рис. 9). Двум нетерминалам этой грамматики будут соответствовать вершины N и В (п. 1). Поскольку в грамматике есть несколько правил, в правой части которых записан единственный терминал, добавим вершину К (п. 2).



Рис. 9. Граф автоматной грамматики G10

Соединим вершины дугами, как это предписывается п. 3 и п. 4.

Вершину N пометим стрелкой как начальную (п. 5).

Граф автоматной грамматики может использоваться для порождения цепочек языка. Любой путь из начальной вершины графа в одну из конечных вершин порождает цепочку терминалов, соответствующую проходимым дугам. Эта цепочка принадлежит языку, порождаемому грамматикой. И, наоборот, для любой сентенции грамматики можно найти путь, ведущий из начальной вершины в одну из конечных и проходящий по дугам, помеченным символами этой сентенции.

Грамматика G10 порождает язык идентификаторов. Нетрудно убедиться, что для любого идентификатора найдется путь из вершины N в вершину К, а любой путь из N в К соответствует правильному идентификатору.

Граф автоматной грамматики идентичен диаграмме переходов конечного автомата — абстрактного устройства, являющегося моделью определенного класса реальных автоматических устройств и объектом изучения теории автоматов.

**Конечные автоматы**

Конечным автоматом (КА) называется пятерка:

A=(N,T,P,S,F),

где N — конечное множество состояний автомата;

Т— входной алфавит — конечное множество символов;

Р— функция переходов автомата (в общем случае неоднозначная), отображающая множество пар состояние - входной символ в множество состояний;

S — начальное состояние. S∈N;

F— множество конечных (финитных) состояний. F⊆N.

Конечный автомат действует следующим образом. Вначале он находится в состоянии S. На вход КА поступают символы, принадлежащие входному алфавиту.

Последовательность входных символов образует входную цепочку. Находясь в некотором состоянии и получив на вход очередной символ, автомат переходит в следующее состояние, определяемое значением функции переходов для данной пары символ-состояние, и считывает очередной символ.

В общем случае функция переходов может определять переход в несколько состояний для данной пары символ-состояние. В этом случае говорят о недетерминированном конечном автомате (НКА). Автомат останавливается, когда заканчиваются символы на его входе.

Если, прочитав входную цепочку а, автомат остановился в некотором состоянии В, говорят, что цепочка а перевела автомат из начального состояния в состояние В. Если В — одно из конечных состояний (В∈F), то говорят, что автомат принимает (допускает) цепочку а.

*Множество всех цепочек, переводящих конечный автомат А из начального в одно из конечных состояний (множество цепочек, принимаемых КА), образует язык L(A), принимаемый (допускаемый) КА.*

*Язык, порождаемый автоматной грамматикой G, совпадает с языком, принимаемым соответствующим конечным автоматом А.*

*L(G) = L(A).*

Как мы уже видели, КА может задаваться с помощью диаграммы переходов.

Например, граф автоматной грамматики G10, показанный на рис. 9, может считаться диаграммой переходов автомата A10. При этом

L(G10) = L(A10).

**Преобразование недетерминированного конечного автомата в детерминированный конечный автомат**

То обстоятельство, что при переходе от автоматной грамматики к КА мы получаем в общем случае недетерминированный конечный автомат (НКА), затрудняет его использование в роли распознавателя автоматного языка.

 Недетерминированность автомата выражается в том, что для некоторых вершин его диаграммы переходов имеется несколько дуг, выходящих из этих вершин и помеченных одним и тем же символом.

Так, автомат, изображенный на рис. 9, является недетерминированным.

Из вершины N исходят две дуги, помеченные символом а, из вершины В — по две дуги, помеченных символами а и b.

Было бы крайне желательно иметь возможность строить для автоматной грамматики детерминированный конечный автомат (ДКА).

**Теорема Клини**. *Для каждого НКА можно построить ДКА, допускающий тот же язык*.

Рассмотрим алгоритм построения ДКА, эквивалентного данному НКА.

Для иллюстрации алгоритма будем применять его к НКА A10 (см. рис. 9), принимающему язык идентификаторов.

1. Пусть исходный НКА имеет k состояний. Для построения ДКА возьмем 2k - 1 состояний, каждое из которых соответствует одному элементу множества всех подмножеств состояний исходного автомата, кроме пустого множества.

Автомат А10 имеет три (k = 3) состояния:

N, B и К.

У нового автомата будет 23 -1= 7 состояний. Они соответствуют таким множествам состояний исходного НКА:

 {N}, {В}, {К}, {N, В}, {В, К}, {N, К), {N, В, К}.

Будем обозначать состояния нового автомата просто последовательностями букв:

N, В, К, NB, ВК, NK, NBK.

2. Из каждого состояния S нового автомата направим не более чем один переход, помеченный данным символом, в такое состояние, которое соответствует множеству состояний НКА, в которые есть переходы по этому символу хотя бы из одного состояния НКА, образующего S.

У НКА A10 из состояния N есть переход по символу а в состояния В и К (см. рис. 9). Следовательно, из состояния N нового автомата дугу, помеченную символом а, направляем в состояние ВК.

Рассматривая состояние NB нового автомата, выясняем, что переходы из состояния N по символу а в исходном автомате есть в состояния В и К, из состояния В исходного автомата — также в состояния В и К.

Направляем дугу, помеченную а, из состояния NB в состояние ВК (рис. 10). Перехода по символу b из состояния N в исходном автомате нет. Из состояния В исходного автомата есть переходы, помеченные b, в состояния В и К. Направляем дугу b из состояния NB в состояние ВК.



Рис.10, Детерминированный конечный автомат А11, эквивалентный недетерминированному автомату А10

Аналогично формируем переходы из других состояний нового автомата А11,который будет эквивалентен A10. В нашем примере оказывается, что все дуги, помеченные как символом а, так и символом b, ведут в состояние ВК.

3. В качестве начального отметим состояние ДКА, имеющее то же обозначение, что и начальное состояние исходного НКА. Как конечные отметим все состояния ДКА, в которые входит хотя бы одно из конечных состояний исходного НКА.

В нашем примере начальным будет состояние N Конечными состояниями ДКА будут все состояния, включающие состояние К исходного автомата, то есть

К, ВК, NK,NBK (см. рис. 10).

Получившийся автомат является детерминированным (из любого состояния исходит не более одной дуги, помеченной данным символом) и принимает тот же язык, что и исходный недетерминированный автомат.

Детерминированный автомат А11 имеет больше состояний, чем исходный НКА А10. Нетрудно, однако, увидеть возможности упрощения получившегося ДКА.

Большинство его состояний (В, К, NB, NK, NBK) недостижимы из начального состояния, поэтому могут быть отброшены.

Получающийся после этого ДКА А12 показан на рис. 11.

Обозначение состояния ВК упрощено, оно снова названо просто В. Этот автомат не только детерминирован, но и проще исходного НКА A10.



Рис. 11. Минимальный детерминированный конечный автомат A12, распознающий идентификаторы

По диаграмме переходов получившегося автомата можно снова записать автоматную грамматику, порождающую язык идентификаторов, эквивалентную грамматике G10, но содержащую меньше правил:

G12: N→aB

 B→aB|bB|**ε**.

Кстати, смысл нетерминала В в новой грамматике сохранился — это «хвост», завершающий идентификатор.

Детерминированный конечный автомат можно рассматривать как распознаватель автоматного языка — устройство, с помощью которого просто и эффективно решается задача разбора для автоматной грамматики.

В связи с этим наряду с автоматами, принимающими (допускающими) некоторый язык, будем говорить об автоматах, распознающих язык.

Для любого автоматного языка можно построить детерминированный конечный автомат, распознающий этот язык.

Задача получения возможно более простого ДКА также имеет общее решение.

Для любого автоматного языка можно построить единственный ДКА, распознающий этот язык и имеющий минимально возможное число состояний.