

Минимизация конечных детерминированных распознавателей

Конечный детерминированный распознаватель, содержащий наименьшее число состояний по сравнению с любым, эквивалентным ему распознавателем, называется *минимальным*. Множество всех эквивалентных между собой распознавателей называется *классом эквивалентных распознавателей*. Класс эквивалентных распознавателей может содержать несколько распознавателей с одинаковым, наименьшим числом состояний по сравнению с любым другим распознавателем из этого класса, но все они *изоморфны*, т.е. различаются только названиями состояний. Процесс получения минимального распознавателя, эквивалентного заданному, называется *минимизацией конечного распознавателя*.

Конечный распознаватель является минимальным, если он не содержит эквивалентных между собой состояний и состояний, недостижимых из начального. Поэтому процесс минимизации выполняется в два этапа:

- поиск и устранение состояний, недостижимых из начального;
- поиск и замена эквивалентных между собой состояний одним состоянием.

Первый этап минимизации.

Алгоритм поиска состояний, недостижимых из начального.

1. Определить множество состояний, достижимых из начального, используя алгоритм обхода графа (например в ширину или в глубину).
2. В результате исключения из множества состояний распознавателя состояний, достижимых из начального, получим множество состояний, недостижимых из начального.

Алгоритм устранения состояний, недостижимых из начального.

1. Исключить из множества состояний распознавателя состояния, недостижимые из начального.

При табличном способе задания распознавателя исключаются столбцы, соответствующие состояниям, недостижимым из начального, а при графовом способе — вершины, соответствующие исключаемым состояниям, и выходящие из них дуги.

Второй этап минимизации.

Второй этап минимизации можно выполнить разными способами.

Первый способ заключается в нахождении пары эквивалентных состояний и исключении одного из них до тех пор, пока есть эквивалентные состояния.

Алгоритм поиска пары эквивалентных состояний.

1. Перебирая пары состояний, проверять их на эквивалентность с помощью дерева проверки.
2. Поиск заканчивается, если проверяемые состояния эквивалентны или перебраны все пары состояний.

Алгоритм исключения одного из пары эквивалентных состояний (s_i, s_j) .

1. Исключить состояние s_i .
2. Перебрать все состояния. Если из некоторого состояния s_k существует переход под действием входного символа x в состояние s_i , то состоянием перехода состояния s_k под действием входного символа x сделать состояние s_j .

Второй способ заключается в **нахождении классов эквивалентных состояний**, которые в дальнейшем становятся состояниями минимального распознавателя. *Класс эквивалентных состояний* – это множество всех эквивалентных между собой состояний конечного распознавателя.

Состояния, для которых не существует различающих цепочек длины k , называются *k-эквивалентными*. Множество всех k -эквивалентных между собой состояний называется *классом k-эквивалентных состояний*.

Для нахождения классов эквивалентных состояний нужно последовательно находить классы 0-, 1-, 2-, ..., k -эквивалентных состояний, пока классы k -эквивалентных состояний не совпадают с классами $k-1$ -эквивалентных состояний. При совпадении классов k - и $k-1$ -эквивалентных состояний полученные классы представляют собой классы эквивалентных состояний.

Множество всех допускающих состояний образуют класс 0-эквивалентных состояний, т.к. все они допускают пустые цепочки. Множество всех отвергающих состояний образуют другой класс 0-эквивалентных состояний, т.к. все они отвергают пустые цепочки.

Для того, чтобы состояния были k -эквивалентными, они должны быть $k-1$ -эквивалентными и под действием одинаковых входных символов должны переходить в $k-1$ -эквивалентные состояния.

Процесс нахождения классов эквивалентных состояний можно представить построением таблиц переходов из состояний в классы k -эквивалентных состояний.

Пример.

Таблица переходов конечного распознавателя.

	↓				1			1	1	
	s0	s1	s2	s3	s4	s5	s6	s7	s8	
ц	s6	s5	s6	s4	s4	s6	s5	s8	s8	
.	s3	s3	s3			s7	s8			
+	s1									
-	s2									

Последний столбец в таблице переходов соответствует состоянию ошибки, а пустые клетки – переходам в состояния ошибки.

Отвергающие состояния $\{s0, s1, s2, s3, s5, s6\}$ объединяем в класс $K1$ 0-эквивалентных состояний, а допускающие состояния $\{s4, s7, s8\}$ – в класс $K2$.

Таблица переходов в классы 0-эквивалентных состояний.

	K1							K2		
	s0	s1	s2	s3	s5	s6		s4	s7	s8
ц	K1	K1	K1	K2	K1	K1	K1	K2	K2	K2
.	K1	K1	K1	K1	K2	K2	K1	K1	K1	K1
+	K1	K1	K1	K1	K1	K1	K1	K1	K1	K1
-	K1	K1	K1	K1	K1	K1	K1	K1	K1	K1

Распознаватель из состояния $s0$ под действием входного символа ц переходит в состояние $s6$, которое принадлежит классу $K1$, поэтому в таблице на пересечении столбца $s0$ и строки ц записано $K1$.

Распознаватель из состояния $s1$ под действием входного символа $+$ переходит в состояние ошибки, которое принадлежит классу $K1$, поэтому в таблице на пересечении столбца $s1$ и строки $+$ записано $K1$.

Распознаватель из состояния $s3$ под действием входного символа ц переходит в состояние $s4$, которое принадлежит классу $K2$, поэтому в таблице на пересечении столбца $s3$ и строки ц записано $K2$.

Распознаватель из состояния ошибки под действием любого входного символа переходит в состояние ошибки, которое принадлежит классу $K1$, поэтому в таблице в столбце, соответствующем состоянию ошибки, записаны $K1$ для всех входных символов.

Таблица переходов в классы 0-эквивалентных состояний.

[illegible]

По таблице переходов в классы 0-эквивалентных состояний определяем классы 1-эквивалентных состояний. Состояния будут 1-эквивалентными, если они 0-эквивалентны и под действием одинаковых входных символов переходят в состояния, принадлежащие одинаковым классам 0-эквивалентных состояний, т.е. 1-эквивалентным состояниям в таблице переходов в классы 0-эквивалентных состояний соответствуют одинаково заполненные столбцы. Объединяем 1-эквивалентные состояния в классы и строим таблицу переходов в классы 1-эквивалентных состояний для определения 2-эквивалентных состояний.

Подготовка таблицы переходов в классы 1-эквивалентных состояний.

[illegible]

Таблица переходов конечного распознавателя.

	↓				1			1	1	
	s0	s1	s2	s3	s4	s5	s6	s7	s8	
□	s6	s5	s6	s4	s4	s6	s5	s8	s8	
.	s3	s3	s3			s7	s8			
+	s1									
-	s2									

Таблица переходов в классы 1-эквивалентных состояний.

[illegible]

Таблица переходов в классы 1-эквивалентных состояний.

[illegible]

В классе K1 2-эквивалентными состояниями будут s_0 , s_1 и s_2 . Состояние ошибки – отдельный класс 2-эквивалентных состояний. Остальные классы 1-эквивалентных состояний также являются и 2-эквивалентными. Классы 2-эквивалентных состояний не совпадают с классами 1-эквивалентных состояний, поэтому строим таблицу переходов в классы 2-эквивалентных состояний для определения классов 3-эквивалентных состояний.

Подготовка таблица переходов в классы 2-эквивалентных состояний.

[illegible]

Таблица переходов конечного распознавателя.

	↓				1			1	1	
	s0	s1	s2	s3	s4	s5	s6	s7	s8	
ц	s6	s5	s6	s4	s4	s6	s5	s8	s8	
.	s3	s3	s3			s7	s8			
+	s1									
-	s2									

Таблица переходов в классы 2-эквивалентных состояний.

[illegible]

Таблица переходов в классы 2-эквивалентных состояний.

[illegible]

В классе $K1$ 3-эквивалентными состояниями будут $s1$ и $s2$. Состояние $s0$ – отдельный класс 3-эквивалентных состояний. Остальные классы 2-эквивалентных состояний также являются и 3-эквивалентными. Классы 3-эквивалентных состояний не совпадают с классами 2-эквивалентных состояний, поэтому строим таблицу переходов в классы 3-эквивалентных состояний для определения классов 4-эквивалентных состояний.

Подготовка таблицы переходов в классы 3-эквивалентных состояний.

[illegible]

Таблица переходов конечного распознавателя.

	↓				1			1	1	
	s0	s1	s2	s3	s4	s5	s6	s7	s8	
ц	s6	s5	s6	s4	s4	s6	s5	s8	s8	
.	s3	s3	s3			s7	s8			
+	s1									
-	s2									

Таблица переходов в классы 3-эквивалентных состояний.

[illegible]

Таблица переходов в классы 3-эквивалентных состояний.

	K1	K2		K3	K4	K5		K6		
	s0	s1	s2		s3	s5	s6	s4	s7	s8
ц	K5	K5	K5	K3	K6	K5	K5	K6	K6	K6
.	K4	K4	K4	K3	K3	K6	K6	K3	K3	K3
+	K2	K3	K3	K3	K3	K3	K3	K3	K3	K3
-	K2	K3	K3	K3	K3	K3	K3	K3	K3	K3

По таблице видно, что классы 4-эквивалентных состояний совпадают с классами 3-эквивалентных состояний, следовательно, классы 3-эквивалентных состояний представляют собой классы эквивалентных состояний.

Алгоритм формирования минимального распознавателя.

1. Множеством состояний минимального распознавателя сделать множество классов эквивалентных состояний исходного распознавателя. Класс эквивалентных состояний исходного распознавателя соответствует состоянию минимального распознавателя.

2. Начальному состоянию минимального распознавателя соответствует класс эквивалентных состояний, содержащий в себе начальное состояние исходного распознавателя.

3. Допускающим состояниям минимального распознавателя соответствуют классы эквивалентных состояний, содержащие в себе допускающие состояние исходного распознавателя.

4. Функция переходов минимального распознавателя определяется следующим образом. Если из состояний класса K_i под действием входного символа x исходный распознаватель переходит в состояния класса K_j , то в минимальном распознавателе состоянием перехода состояния K_i под действием входного символа x сделать состояние K_j .

Если классы эквивалентных состояний определялись с помощью таблиц переходов в классы k -эквивалентных состояний (см. пример выше), то для того, чтобы получить таблицу минимального распознавателя достаточно в последней таблице в каждом классе эквивалентных состояний оставить по одному столбцу, исключить строку, в которой записаны названия состояний исходного распознавателя, и отметить начальное и допускающие состояния.

Пример (продолжение).

Таблица переходов конечного распознавателя.

	↓				1			1	1	
	s0	s1	s2	s3	s4	s5	s6	s7	s8	
ц	s6	s5	s6	s4	s4	s6	s5	s8	s8	
.	s3	s3	s3			s7	s8			
+	s1									
-	s2									

Таблица переходов в классы 3-эквивалентных состояний.

	K1	K2		K3	K4	K5		K6		
	s0	s1	s2		s3	s5	s6	s4	s7	s8
ц	K5	K5	K5	K3	K6	K5	K5	K6	K6	K6
.	K4	K4	K4	K3	K3	K6	K6	K3	K3	K3
+	K2	K3	K3	K3	K3	K3	K3	K3	K3	K3
-	K2	K3	K3	K3	K3	K3	K3	K3	K3	K3

Таблица переходов минимального распознавателя.

	↓					1
	K1	K2	K3	K4	K5	K6
ц	K5	K5	K3	K6	K5	K6
.	K4	K4	K3	K3	K6	K3
+	K2	K3	K3	K3	K3	K3
-	K2	K3	K3	K3	K3	K3

Переобозначим состояния K1 через s0, K2 через s1, K4 через s2, K5 через s3, K6 через s4, состояние ошибки K3 и переходы в него исключим из таблицы.

Окончательная таблица переходов минимального распознавателя.

	↓				1
	s0	s1	s2	s3	s4
ц	s3	s3	s4	s3	s4
.	s2	s2		s4	
+	s1				
-	s1				

Упрощение алгоритма нахождения классов эквивалентных состояний

Упрощение заключается в том, что на первом шаге разбиваем множество состояний так, чтобы неэквивалентные состояния попали в разные классы. В этом случае на первом шаге мы можем получить больше классов, чем в исходном алгоритме и это сократит количество шагов алгоритма.

Пример.

Граф конечного детерминированного распознавателя A' .

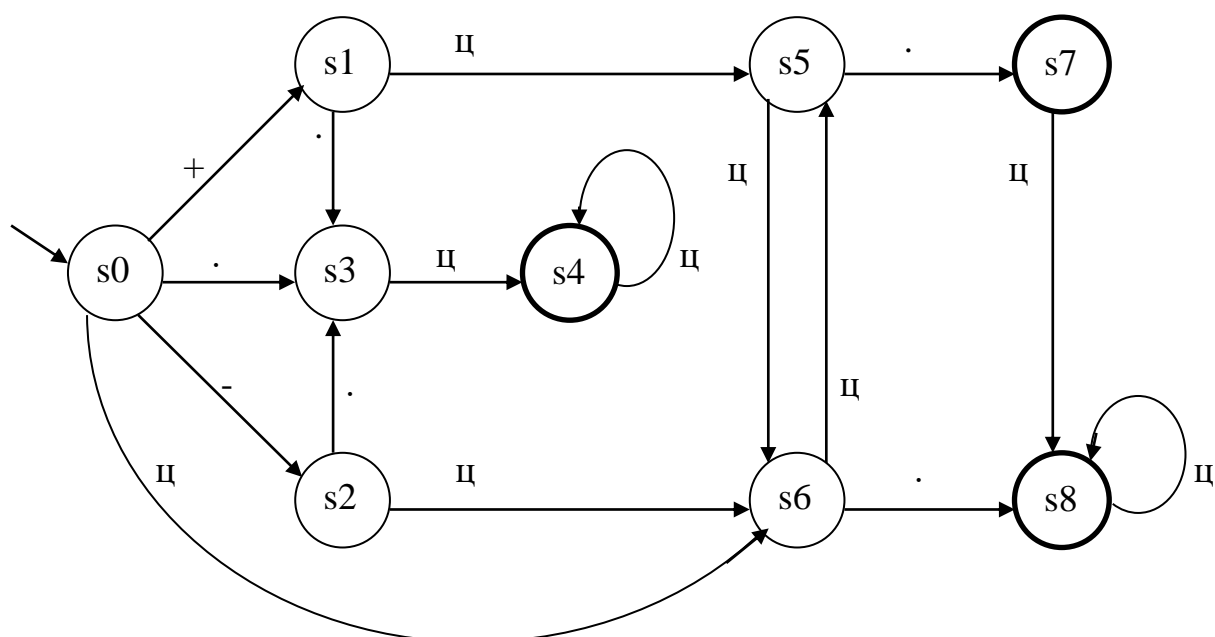


Таблица переходов конечного распознавателя.

	↓				1			1	1
	s0	s1	s2	s3	s4	s5	s6	s7	s8
ц	s6	s5	s6	s4	s4	s6	s5	s8	s8
.	s3	s3	s3			s7	s8		
+	s1								
-	s2								

Пустые клетки – это переходы в отвергающее состояние ошибки, из которого недостижимы допускающие состояния.

В примере из любого состояния (кроме ошибки), достижимо допускающее, поэтому ни одно состояние не эквивалентно состоянию ошибки. Состояние ошибки включаем в отдельный класс. Переход в состояние ошибки обозначаем пустой клеткой.

Одно из условий неэквивалентности двух состояний:

два состояния неэквивалентны, если по символу x распознаватель из одного состояния переходит в состояние ошибки, а из другого – в состояние, отличное от состояния ошибки.

Такие состояния должны принадлежать разным подмножествам.

В одно подмножество можно включить два состояния, если они оба отвергающие или оба допускающие, и, если для любого x оба состояния переходят либо в состояние ошибки, либо в состояния, отличные от состояния ошибки.

В примере состояния s_0 и s_1 отвергающие, но по символу «+» из s_1 переход в состояние ошибки, а из s_0 – в состояние s_1 , поэтому s_0 и s_1 нужно включить в разные подмножества первого разбиения.

s_1 и s_2 включаем в одно подмножество первого разбиения, т. к. они оба отвергающие и для любого x оба состояния переходят либо в состояние ошибки, либо в состояния, отличные от состояния ошибки.

Таким образом, на первом шаге получаем разбиение:

$\{K_1=\{s_0\}, K_2=\{s_1, s_2, s_5, s_6\}, K_3=\{s_3\}, K_4=\{s_4, s_7, s_8\}, K_5=\{\text{ошибка}\}$

Таблица переходов конечного распознавателя.

	↓				1			1	1
	s0	s1	s2	s3	s4	s5	s6	s7	s8
ц	s6	s5	s6	s4	s4	s6	s5	s8	s8
.	s3	s3	s3			s7	s8		
+	s1								
-	s2								

Таблица переходов в подмножества первого разбиения

	K1	K2				K3	K4		
	s0	s1	s2	s5	s6	s3	s4	s7	s8
ц	K2	K2	K2	K2	K2	K4	K4	K4	K4
.	K3	K3	K3	K4	K4				
+	K2								
-	K2								

Отсюда видно, что K2 разбивается на два подмножества.

	K1	K2		K3		K4	K5		
	s0	s1	s2	s5	s6	s3	s4	s7	s8
ц	K3	K3	K3	K3	K3	K5	K5	K5	K5
.	K4	K4	K4	K5	K5				
+	K2								
-	K2								

Теперь подмножества не разбиваются, классы эквивалентности найдены. Для этого потребовалось сделать два шага алгоритма.

Другой способ формирования минимального распознавателя

Таблица переходов конечного распознавателя.

	↓				1			1	1
	s0	s1	s2	s3	s4	s5	s6	s7	s8
ц	s6	s5	s6	s4	s4	s6	s5	s8	s8
.	s3	s3	s3			s7	s8		
+	s1								
-	s2								

Разбиение на классы эквивалентных состояний

	K1	K2		K3		K4	K5		
	s0	s1	s2	s5	s6	s3	s4	s7	s8
ц	K3	K3	K3	K3	K3	K5	K5	K5	K5
.	K4	K4	K4	K5	K5				
+	K2								
-	K2								

$$K1 = \{s0\}$$

$$K2 = \{s1, s2\}$$

$$K3 = \{s5, s6\}$$

$$K4 = \{s3\}$$

$$K5 = \{s4, s7, s8\}$$

Формируем множество состояний минимального распознавателя.

Включаем в это множество по одному состоянию из каждого класса:

$$S = \{s0, s1, s5, s3, s4\}.$$

В таблице распознавателя оставляем столбцы, соответствующие состояниям минимального распознавателя:

	↓			1	
	s0	s1	s3	s4	s5
ц	s6	s5	s4	s4	s6
.	s3	s3			s7
+	s1				
-	s2				

Теперь в клетках таблицы есть состояния, не принадлежащие множеству состояний минимального распознавателя. Такие состояния нужно заменить на эквивалентные им состояния минимального распознавателя.

$$K1 = \{\mathbf{s0}\}$$

$$K2 = \{\mathbf{s1}, s2\}$$

$$K3 = \{\mathbf{s5}, s6\}$$

$$K4 = \{\mathbf{s3}\}$$

$$K5 = \{\mathbf{s4}, s7, s8\}$$

	↓			1	
	s0	s1	s3	s4	s5
ц	s5	s5	s4	s4	s5
.	s3	s3			s4
+	s1				
-	s1				

Минимальный распознаватель получен.