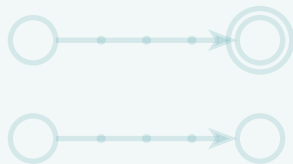


Лекция

30–31 марта

Минимизация автоматов



Различимость состояний

Что значит имя? Роза пахнет розой, хоть розой назови её, хоть нет.

— Уильям Шекспир

Зачем минимизировать?

Таблица переходов — память автомата в железе и в коде.

Лишние состояния — лишние строки таблицы, лишние проверки, лишние регистры.

Пример

Один и тот же язык «чётное число единиц» распознают автомат из двух состояний и автомат из четырёх.

Второй работает, но половина его состояний — балласт.

Вопрос имеет два этажа: как найти лишние состояния и почему после чистки не осталось ещё одно лишнее.

Ответы — алгоритм таблицы различимостей и теорема о единственности.

Эквивалентность состояний

Определение 1

Состояния p, q **эквивалентны**, если на любом слове w оба принимают или оба отвергают:

$$\hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F.$$

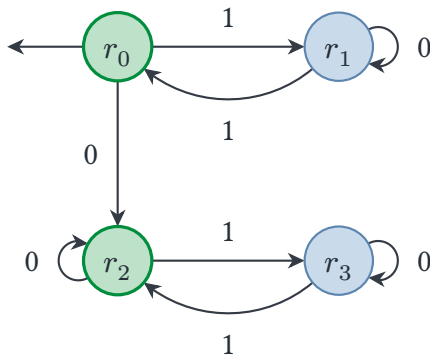
Иначе состояния **различимы**: есть слово-свидетель w , разводящее их ответы.

Пример

Свидетель может быть коротким: слово 1 различает q_0 и q_1 автомата чётности — из q_0 слово 1 отвергается, из q_1 принимается.

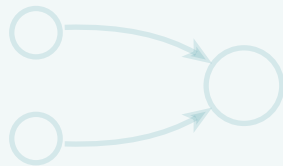
Эквивалентные состояния — одно состояние под двумя именами, их склейка и есть минимизация.

Четыре состояния вместо двух



Пример

Автомат распознаёт чётность единиц, но переходы по 0 ($r_0 \rightarrow r_2$, $r_1 \rightarrow r_1$) делают пары r_0, r_2 и r_1, r_3 неразличимыми: поведение на любое продолжение совпадает. Перебрать все слова нельзя — нужен алгоритм различимости.



Минимизация

Совершенство достигается не тогда, когда уже нечего прибавить, но когда уже ничего нельзя отнять.

— Антуан де Сент-Экзюпери

Таблица различимостей

Алгоритм Мура заполняет таблицу всех пар состояний:

1. Завести клетки для всех неупорядоченных пар $p \neq q$.
2. Пометить пары, где ровно одно состояние принимающее: их разводит пустое продолжение.
3. Итерация: если для пары (p, q) и символа a пара $(\delta(p, a), \delta(q, a))$ уже помечена — пометить (p, q) .
4. Повторять, пока таблица меняется.
5. Непомеченные пары — эквивалентные состояния: склеить их.

Различимость распространяется назад по дугам: если продолжение разводит образы, разводит и предшественников.

Пример: минимизация четырёх состояний

Для автомата $r_0, \dots, r_3$ сначала помечаются пары с ровно одним принимающим состоянием: $(r_0, r_1), (r_0, r_3), (r_1, r_2), (r_2, r_3)$.

Пара	по 0	по 1	итог
(r_0, r_2)	(r_2, r_2)	(r_1, r_3)	неразличимы
(r_1, r_3)	(r_1, r_3)	(r_0, r_2)	неразличимы

Остальные четыре пары помечены на первом шаге и дальше не участвуют.

Пример

Склейка $r_0 = r_2$ и $r_1 = r_3$ даёт двухсостоятельный автомат чётности.

Единственность минимального ДКА

Теорема 1 (Единственность)

Каждый регулярный язык имеет единственный минимальный ДКА с точностью до изоморфизма.

Доказательство

Докажем построением изоморфизма. Лемма: в минимальном ДКА любые два состояния различимы и каждое достижимо — иначе склейка или удаление дали бы автомат меньше. Пусть M_1, M_2 — два минимальных ДКА, сопоставим друг другу состояния, в которые ведёт слово x . Корректность: если x и y ведут в M_1 в одно состояние, они неразличимы, поэтому и в M_2 ведут в одно. Сюръективность — из достижимости, инъективность — из различимости. Начала, принимающие множества и переходы согласованы — построен изоморфизм.

Каноническая форма

Минимальный ДКА — портрет языка, а не удача конструктора.

Два автомата распознают один язык тогда и только тогда, когда их минимальные формы изоморфны.

Проверка эквивалентности — конечная процедура: минимизировать оба и сравнить.

Пример

Оптимизаторы регулярных выражений и инструменты анализа протоколов именно так проверяют эквивалентность: сравнением канонических форм.

Число состояний минимального ДКА — объективная мера сложности языка.

Откуда она берётся — сейчас.



Классы неразличимости

Напрасно пытаться посредством большего делать то, что может быть сделано посредством меньшего.

— Уильям Оккам

Неразличимость по продолжению

Определение 2

Для языка L слова x, y **неразличимы по продолжению**, $x \simeq_L y$, если

$$\forall z \in \Sigma^* : \quad xz \in L \iff yz \in L.$$

Пример

Для языка слов, заканчивающихся на 01: продолжение 1 разводит ε и $0 - 1 \notin L$, а $01 \in L$.

Отношение $\simeq_L$ — эквивалентность на Σ^* : рефлексивность, симметричность и транзитивность наследуются от равенства ответов.

Это тот же язык классов эквивалентности и фактор-множеств, что строился для отношений в первом семестре.

Остаточные языки

Определение 3

Остаточный язык слова x относительно L :

$$C_x = \{z \in \Sigma^* \mid xz \in L\}.$$

Пример

Слова неразличимы тогда и только тогда, когда их остаточные языки совпадают.

Классы $\simeq_L$ — это слова с одним остаточным языком.

Остаточный язык — всё, что язык собирается спросить о слове в будущем.

Состояние автомата хранит в точности это — и ничего больше.

Классы для языка «01»

Для языка слов, заканчивающихся на 01, остаточных языков три.

Класс слов	Остаточный язык
слова, кончающиеся на 01	нужен пустой остаток
слова, кончающиеся на 0	нужна единица
остальные слова	нужна пара 01

Пример

Три класса — три состояния автомата q_0, q_1, q_2 из детерминированной лекции.

Пустое слово и 1 лежат в одном классе, 0 и 00 — во втором, 01 — в третьем.

Теорема Майхилла–Нерода

Теорема 2 (Майхилл–Нерод)

Для языка L следующие утверждения эквивалентны:

1. L регулярен.
2. Отношение $\simeq_L$ имеет конечный индекс.
3. Язык L — объединение классов правой конгруэнции конечного индекса (эквивалентности, согласованной с дописыванием символа справа).

Доказательство

Докажем по циклу импликаций.

Регулярность $\Rightarrow$ конечный индекс. Слова, ведущие ДКА в одно состояние, неразличимы, поэтому классов не больше, чем состояний.

Теорема Майхилла–Нерода [2]

Конечный индекс $\Rightarrow$ конгруэнция. Само отношение неразличимости — правая конгруэнция: $x \simeq_L y$ влечёт $xa \simeq_L ya$.

Конгруэнция $\Rightarrow$ регулярность. По классам строится фактор-автомат: переход $[x] \rightarrow [xa]$ корректен по свойству конгруэнции.

Принимающие классы — те, что содержат слова языка, и распознаётся ровно L .

Классы $\simeq_L$ — это и есть состояния самого экономного автомата.

Индекс отношения — точное число состояний минимального ДКА.

Нерегулярность через классы

Пример

Для $L = \{a^n b^n\}$ рассмотрим слова $a, a^2, a^3, \dots$

Для любых $i < j$ продолжение $z = b^i$ разводит их:

$$a^i b^i \in L, \quad a^j b^i \notin L.$$

Все степени a попарно различимы — классов бесконечно много.

Бесконечный индекс $\simeq_L$ — и язык $a^n b^n$ не регулярен.

В отличие от накачки, доказательство конструктивно: предъявлено бесконечное множество попарно различных слов.

Два инструмента рядом

Инструмент	Что даёт	Граница
Лемма о накачке	опровержение регулярности	необходимое условие, не достаточное
Теорема Майхилла–Нерода	критерий в обе стороны и минимальный автомат	нужно уметь считать классы

Число состояний минимального ДКА равно числу классов $\simeq_L$: мера сложности языка — в самом языке, а не в руках конструктора.

Лекция 8: что мы поняли?

- Состояния эквивалентны, если их не различает ни одно слово: склейка таких пар даёт минимизацию.
- Таблица различимостей находит все различимые пары — алгоритм Мура за $O(n^2 \cdot |\Sigma|)$.
- Теорема Майхилла–Нерода: классы неразличимости слов по продолжению — это состояния минимального ДКА, фактор-автомат по остаточным языкам.
- Минимальный ДКА единствен с точностью до изоморфизма — каноническая форма языка.
- Конечность модели делает все вопросы о ней решаемыми: пустота, эквивалентность, минимизация.

Вопросы для самопроверки

1. Почему в минимальном ДКА любые два состояния различимы? Что сломалось бы иначе?
2. Найдите слово-свидетель для пары (r_0, r_1) автомата с четырьмя состояниями и объясните, почему для пары (r_0, r_2) свидетеля нет.
3. Опишите классы $\simeq_L$ для языка слов с чётным числом единиц.
4. Докажите через Майхилла–Нерода нерегулярность языка палиндромов над $\{a, b\}$.

Чтение к следующей паре

- m23 — «Минимизация ДКА», «Теорема Майхилла–Нерода».
- Флеш-опросник — в начале лекции 9.
- Контрольная 2 (графы, автоматы и регулярные языки) — на этой неделе, на отдельной паре.
- Теорминимум 1 «Графы и автоматы» — на следующей неделе.