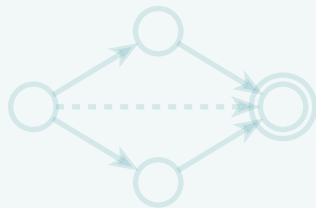


Лекция

16–17 марта

Недетерминированные автоматы



Недетерминизм

Две дороги разошлись в лесу, и я — я выбрал ту, что меньше была пройдена.

— Роберт Фрост

Сколько состояний нужно?

Задача: язык слов, у которых третий с конца символ равен 1.

Автомат должен помнить последние три символа.

Пример

Возможных хвостов длины 3 — восемь: 000, 001, ..., 111.

Разным хвостам нужны разные состояния: продолжение 1 разводит слова 00 и 10 — 001 отвергнуто, 101 принято.

Для хвостов длины k состояний нужно 2^k .

Детерминизм имеет цену: память последних k символов — 2^k состояний.

Хочется язык описания, где такая память не строится вручную.

Недетерминированный конечный автомат

Определение 1 (НКА)

Пятёрка $M = (Q, \Sigma, \delta, q_0, F)$, где Q, Σ, q_0, F — как у ДКА, а

$$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Q)$$

возвращает **множество** возможных следующих состояний.

Пример

$\delta(q_0, 0) = \{q_0, q_1\}$ — по нулю автомат попадает либо в q_0 , либо в q_1 .

Переход по ε меняет состояние бесплатно, ничего не читая.

НКА принимает слово, если существует хотя бы один путь в принимающее состояние, остальные пути игнорируются.

Что значит «угадывает»?

Недетерминизм часто описывают словом «угадывание».

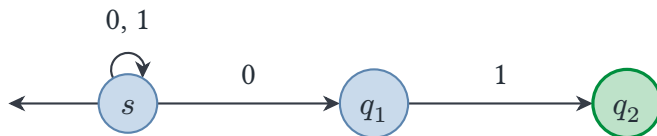
Это опасная метафора.

В определении языка действует только квантор существования: слово принято, если *существует* успешный путь.

Никакой случайности, никакого параллелизма, никакого алгоритма перебора — НКА это декларативное описание языка.

Исполнительная сторона появится позже: как прогнать все пути за один раз, расскажет конструкция подмножеств.

НКА для слов, заканчивающихся на «01»



Пример

Одна мысль: на каждом нуле автомат может «подозреть» начало финальной пары и проверить её следующим символом.

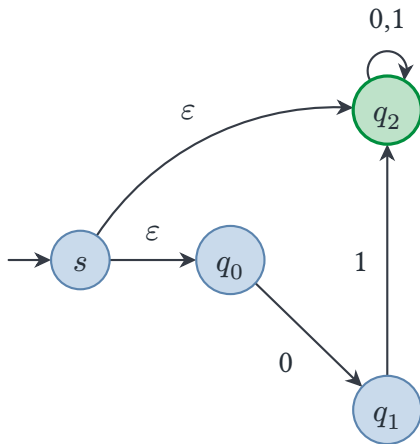
Прогон 10101: петля до четвёртого символа, затем $s \rightarrow q_1 \rightarrow q_2$ — принят.

Прогон 10110: последняя догадка умирает на 0, остальные кончают не там — отвергнут.

Ранняя догадка умирает в q_2 без переходов — для принятия достаточно одной удачной.

Эпсилон-переходы

ε -переход — смена состояния без чтения символа.



Прогон и эpsilon-замыкание

Пример

Автомат распознаёт язык «содержит подстроку 01, или пустое слово».

Слово ε принимается по переходу $s \rightarrow q_2$, слово 100 отвергнуто.

ε -замыкание состояния — все состояния, достижимые из него по цепочке ε -переходов.

Здесь ε -замыкание множества $\{s\}$ — это $\{s, q_2\}$: старт сразу «видит» принимающую ветвь.

Замыкание понадобится в конструкции подмножеств: начальное состояние ДКА — ε -замыкание старта НКА.

Сколько путей у НКА?

Каждая догадка — ветвь вычисления.

Число путей растёт с длиной слова стремительно: на каждом ветвлении пути удваиваются.

Пример

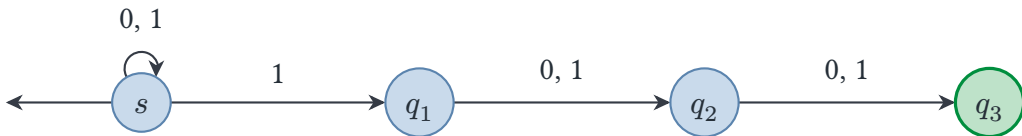
Слово длины n порождает до 2^n путей — перебор ветвей вручную разорителен.

Множества состояний НКА при этом всего 2^k , где $k = |Q|$.

Все пути сразу умещаются в конечный объект: множество состояний, где НКА может находиться после чтения префикса.

Эта идея сейчас превратится в конструкцию.

Третий с конца символ равен 1



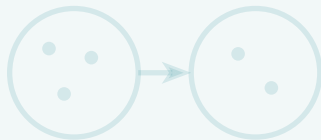
Пример

Четыре состояния: догадаться, что до конца осталось ровно три символа, и отсчитать их.

НКА — 4 состояния при любой длине слова.

Минимальный ДКА для этого языка требует $2^3 = 8$ состояний: он обязан различать все хвосты длины 3.

Почему именно 8 — ответит теорема Майхилла–Нерода в конце модуля.



Конструкция подмножеств

| *Бог не играет в кости.*

— Альберт Эйнштейн

Эквивалентность ДКА и НКА

Теорема 1 (Рабин—Скотт)

Для любого НКА существует ДКА, распознающий тот же язык.

Доказательство

Докажем построением.

Пусть дан НКА $N = (Q, \Sigma, \delta, q_0, F)$. Построим ДКА с множеством состояний $\mathcal{P}(Q)$:

- начальное состояние — ε -замыкание множества $\{q_0\}$;
- переход: $\delta'(S, a) = \varepsilon\text{-closure}\left(\bigcup_{q \in S} \delta(q, a)\right)$;
- принимающие: $F' = \{S \mid S \cap F \neq \emptyset\}$.

Состояние S — множество всех позиций, где НКА мог бы оказаться после чтения префикса.

ДКА симулирует все пути НКА одновременно, поэтому языки совпадают.

Рецепт детерминизации

1. Начать с ε -замыкания начального множества $\{q_0\}$.
2. Для каждого построенного множества S и символа a собрать все переходы по a из состояний S и замкнуть результат по ε .
3. Если такое множество уже встречалось — провести дугу в него, иначе завести новое состояние.
4. Закончить, когда у всех состояний заполнены переходы.
5. Принимающими объявить множества, содержащие хотя бы одно принимающее состояние НКА.

Рецепт всегда завершается: новых множеств не больше $2^{|Q|}$, и рано или поздно все дуги ведут в уже заведённые состояния.

Детерминизация на примере

Применим рецепт к НКА для языка «заканчивается на 01»:

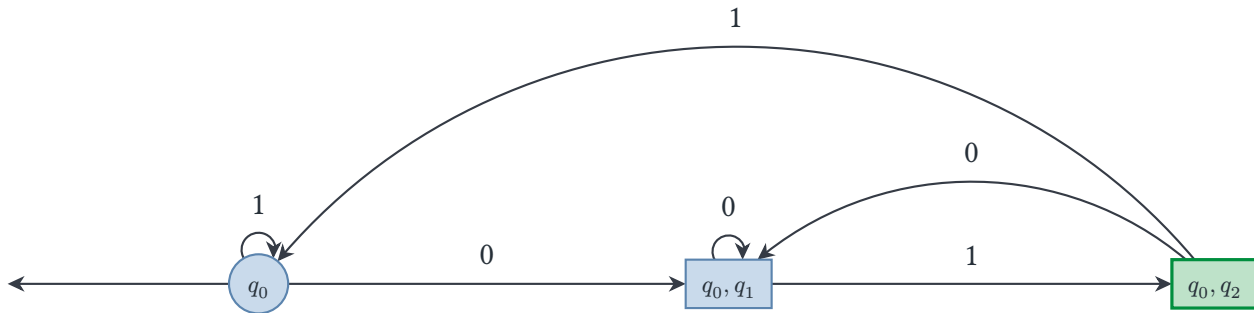
Множество S	$\delta'(S, 0)$	$\delta'(S, 1)$
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_0\}$

Пример

Достижимы три множества, принимающее среди них одно — $\{q_0, q_2\}$.

Прогон 10110: $\{q_0\} \rightarrow \{q_0\} \rightarrow \{q_0, q_1\} \rightarrow \{q_0, q_2\} \rightarrow \{q_0\} \rightarrow \{q_0, q_1\}$ — отвергнуто.

Множества играют роли состояний



Пример

Конструкция подмножеств для НКА с двумя стрелками из s — того, что ищет пару 01: $\{q_0\}$ — ожидание, $\{q_0, q_1\}$ — «кончается на 0», $\{q_0, q_2\}$ — «кончается на 01».

Получился в точности ДКА из детерминированной лекции: детерминизация — перевод описания, а не новая машина.

ДКА и НКА: сравнение

Свойство	ДКА	НКА
Переход	одно состояние	множество состояний
ε -переходы	запрещены	разрешены
Пути вычисления	ровно один	хотя бы один успешный
Дополнение языка	сменить F	сначала детерминизация
Компактность	может расти как 2^k	обычно линейная

Разница между моделями — в цене описания, а не в классе языков.

Каждый НКА превращается в ДКА, и обратное тривиально: ДКА — частный случай НКА.

Экспоненциальная цена детерминизма

Конструкция подмножеств строит до 2^k состояний — и в худшем случае меньше не бывает.

Пример

Язык «третий с конца символ равен 1»: НКА — 4 состояния, минимальный ДКА — $8 = 2^3$.

Для k -го с конца разрыв $k + 1$ против 2^k .

Булеан конечен, поэтому детерминизация всегда заканчивается. Цена — экспонента, арифметика булеана.

Недетерминизм в конечных автоматах устраняем, а для моделей с неограниченной памятью вопрос «можно ли убрать его без экспоненты» станет центральной загадкой теории сложности.

Ловушка: дополнение НКА

У ДКА дополнение языка — поменять принимающие состояния местами с отвергающими.

У НКА приём не работает: принятие — существование принимающего пути, а дополнение требует отсутствия *всех* принимающих путей.

Пример

Слово, у которого один путь успешен и десять неуспешных, НКА принимает.

Автомат с «инвертированной» F отвергнет его, хотя по языку дополнения оно должно приниматься.

Правильный путь: сначала конструкция подмножеств, затем смена принимающих состояний у ДКА.

Зачем же нужен НКА?

Вычислительной силы НКА не добавляет.

Добавляет он структуру описания:

- объединение языков — параллельный запуск двух автоматов;
- конкатенация — последовательное соединение;
- звезда — зацикливание.

Пример

В лексере каждое регулярное выражение компилируется в НКА, затем конструкцией подмножеств — в ДКА.

Описывать компактно удобно человеку, исполнять таблично — машине.

Лекция 6: что мы поняли?

- НКА: переход возвращает множество состояний, принять — существует успешный путь.
- ϵ -переходы меняют состояние бесплатно.
- Конструкция подмножеств симулирует все пути сразу: состояния ДКА — множества состояний НКА.
- Недетерминизм не расширяет класс языков — только экономит описание.
- Цена экономии — экспонента 2^k в худшем случае.

Вопросы для самопроверки

1. Почему слово «угадывает» — плохое описание НКА? Дайте точную формулировку через квантор.
2. Язык «третий с конца символ равен 1»: сколько состояний у НКА и сколько у минимального ДКА?
3. Что произойдёт, если в конструкции подмножеств забыть ε -замыкание?
4. Почему для дополнения языка НКА мало поменять принимающие состояния?

Чтение к следующей паре

- m23 — «Недетерминированные конечные автоматы», «Конструкция подмножеств».
- Флеш-опросник — в начале лекции 7.