Théorie des Langages Rationnels

Minimizing Automata

Adrien Pommellet, LRE

April 6, 2023

Minimizing DFA

The whole pipeline

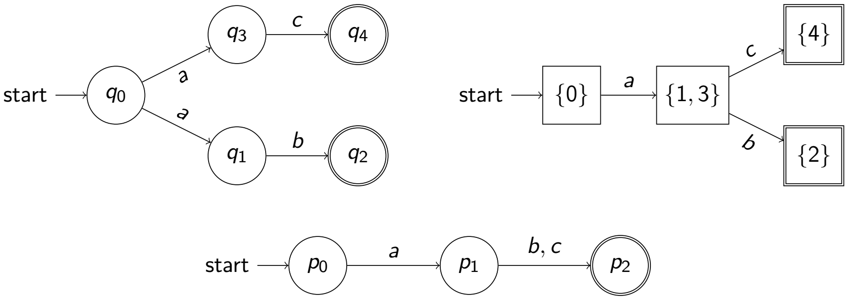
$$
\begin{array}{lcr}
\text{Regular expressions} & \cdots\cdots & \textit{n symbols} \\
\downarrow \text{Thompson's algorithm} & & \\
\varepsilon\text{-NFA} & \cdots\cdots & 2n \textit{ states} \\
\downarrow \text{Backward removal of } \varepsilon\text{-transitions} & & \\
\text{NFA} & \cdots\cdots & 2n \textit{ states} \\
\downarrow \text{Pruning procedure} & & \\
\text{pruned NFA} & \cdots\cdots & \leq 2n \textit{ states} \\
\downarrow \text{Determinization procedure} & & \\
\text{DFA} & \cdots\cdots & \leq 2^{2n} \textit{ states}
\end{array}
$$

Determinization creates an **exponential blow-up** of the number of states, trading a compact representation for computation speed.

Minimizing DFA

Optimizing what can be optimized

The determinization procedure may not yield the **smallest possible** DFA for a given language. Consider these equivalent pruned automata:



Minimizing DFA

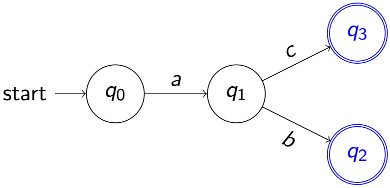
An equivalence relation

Indistinguishable states

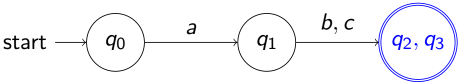
Two states q_1 and q_2 of a DFA $\mathcal{A}$ are said to be **indistinguishable** if for any word $w \in \Sigma^*$, from q_1 $\mathcal{A}$ accepts w if and only if from q_2 $\mathcal{A}$ accepts w. We also write $\mathcal{L}_{q_1}(\mathcal{A}) = \mathcal{L}_{q_2}(\mathcal{A})$.

Intuitively, starting a run from q_1 or q_2 yields the same result.

Our intuition is to regroup indistinguishable states in **equivalence classes** and **fuse** them together. Here, q_2 and q_3 accept the same language.



Yielding the following automata:



Our minimization algorithm is based on the following theorem:

Theorem (Myhill-Nerode)

Given a rational language L, there exists an unique DFA $\mathcal{A}$ with a minimal number of states (equal to L's number of indistinguishability classes) such that $\mathcal{L}(\mathcal{A}) = L$, known as L's ***canonical*** *automaton.*

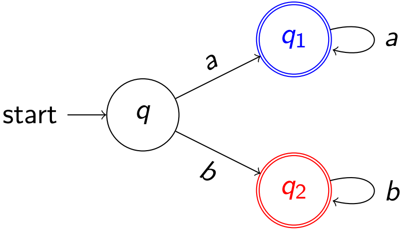
Thus, to find the canonical DFA, we need to be able to **decide** the indistinguishability relation.

Minimizing DFA

Proving indistinguishability

How can we prove indistinguishability when it's a property that applies to an infinite number of words?

It is much easier to prove **distinguishability**: if from q_1 we accept a word w (here, a) but from q_2 we do not, then q_1 and q_2 can't be fused.

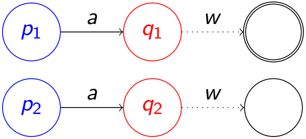


We will later use the following property:

Property

If two states p_1 and p_2 of a DFA $\mathcal{A}$ are indistinguishable and $p_1 \xrightarrow{a} q_1$, $p_2 \xrightarrow{a} q_2$, then q_1 and q_2 are **indistinguishable as well**.

Proof. By contradiction: were q_1 and q_2 distinguished by a word w, then p_1 and p_2 would be distinguished by $a \cdot w$. But they're indistinguishable.



Practical Application

Exercise 1. Guess the indistinguishability classes of $\mathcal{A}$.

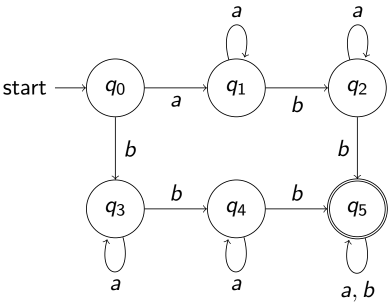


Figure 1: The DFA $\mathcal{A}$.

Answer

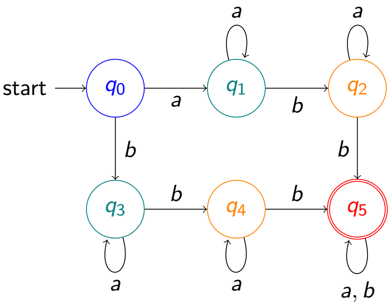


Figure 2: The indistinguishability classes of $\mathcal{A}$.

Moore's Minimization Algorithm

Sketching the method

We will use a **refinement** algorithm:

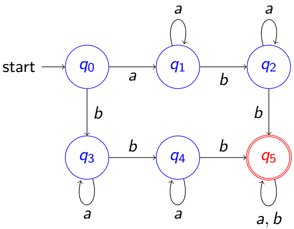
1. Start from a simplistic model.
2. Spot a mistake in the model.
3. Refine the model to remove that mistake.
4. Iterate until there is no mistake left.

The **practical** implementation is the following for a **complete** DFA:

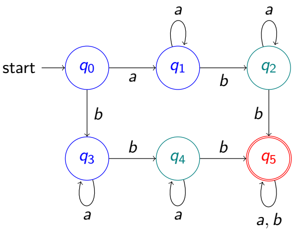
1. Claim that there are only two indistinguishability classes: the accepting states F and the non-accepting states $Q \setminus F$, distinguished by ε.
2. Check that if two states belong to the same class, their successors for a given letter do as well.
3. Split the classes if it is not the case.
4. Iterate until the classes are stable.

Moore's Minimization Algorithm
An example I

Classes	a	b
	01 lead to 2 3 4	1 0 3 24
$\{0, 1, 2, 3, 4\}$	1 2 3 4	2 3 4 5
$\{5\}$		



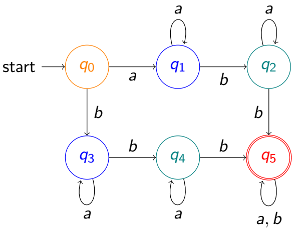
Classes	a	b
	01 3	1 0 3
$\{0,1,3\}$	1 3	2 3 4
$\{2,4\}$		
$\{5\}$		



Moore's Minimization Algorithm

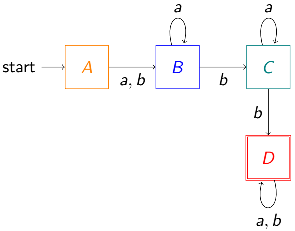
An example III

Classes	a	b
$\{0\}$	0 1	0 3
$\{1,3\}$	1 3 1 3	1 3 2 4
$\{2,4\}$	2 4 2 4	24 5
$\{5\}$	5 5	5 5



Once a table without contradictions has been found, a **minimal** DFA can be designed: to each class we match a state, the **initial** class is the one that contains the original initial state, and any class that contains an accepting state is **accepting**.

Classes	a	b
$A = \{0\}$	1	3
$B = \{1, 3\}$	1 3	2
$C = \{2, 4\}$	2 4	5
$D = \{5\}$	5	5



Moore's Minimization Algorithm

A summary

1. Use a **table** whose columns are labelled by the alphabet Σ and whoses lines by the indistinguishability classes.
2. Start with the two classes F and $Q \setminus F$.
3. For each letter, explore the successors of each class, keeping a **trace** of which state lead to which successor.
4. There's a **conflict** if two distinguishable states appear in the same cell.
5. To solve it, the problematic class is **split** into subclasses.
6. Iterate until no further conflict can be found.
7. Design a DFA based on the table.

Exercise 2. Minimize the automaton $\mathcal{A}$.

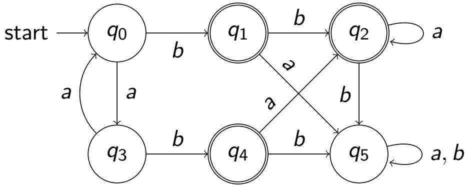
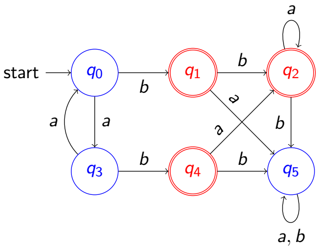


Figure 3: The DFA $\mathcal{A}$.

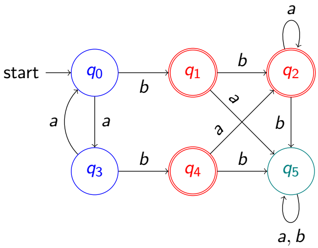
Answer I

Classes	a	b
	3 0 5	0 3 5
$\{0, 3, 5\}$	0 3 5	1 4 5
$\{1, 2, 4\}$		



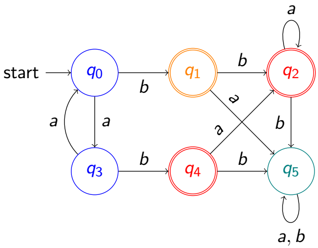
Answer II

Classes	a	b
$\{0, 3\}$	$\overset{3}{0}\ \overset{0}{3}$	$\overset{0}{1}\ \overset{3}{4}$
$\{1, 2, 4\}$	$\overset{2,4}{2}\ \overset{1}{5}$	
$\{5\}$		



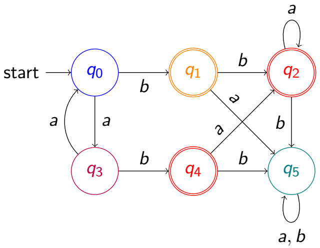
Answer III

Classes	a	b
$\{0,3\}$	$\overset{3\ 0}{0\ 3}$	$\overset{0\ 3}{1\ 4}$
$\{1\}$		
$\{2,4\}$		
$\{5\}$		



Answer IV

Classes	a	b
$\{0\}$	0 3	0 1
$\{3\}$	3 0	3 4
$\{1\}$	1 5	1 2
$\{2,4\}$	24 2	24 5
$\{5\}$	5 5	5 5



Answer V

Classes	a	b
$A = \{0\}$	3	1
$B = \{3\}$	0	4
$C = \{1\}$	5	2
$D = \{2, 4\}$	2	5
$E = \{5\}$	5	5