The Coalgebraic Intepretation of Brzozowski's Automata Minimisation Algorithm

Natalie Ravenhill

University of Birmingham

24th March 2015

Introduction

Automata minimisation = Giving the smallest possible automaton that accepts a given language

The two main different types of algorithms to do this:

- Partition Based *(Hopcroft, Moore, . . .)*
- Powerset Construction Based *(Brozozowski)*

We will study Brzozowski's Algorithm

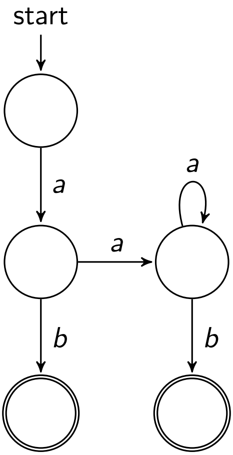
How does it work?

First we take a finite state automaton that accepts a language L.
Then we:

1. Reverse the automaton
2. Determinise the result by Powerset Construction
3. Take the reachable part
4. Repeat Steps 1 and 2

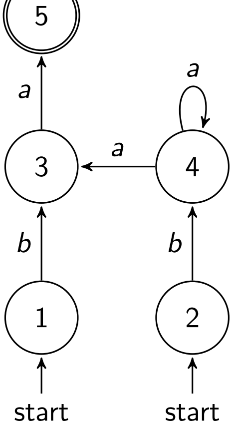
Now we have the minimal automaton that accepts L

Given an automaton M



which accepts the language $a(a)^*b$

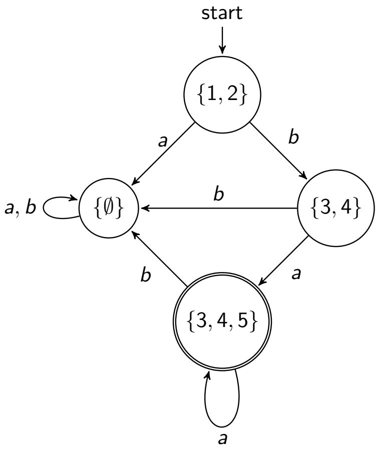
Reverse it:



Transitions

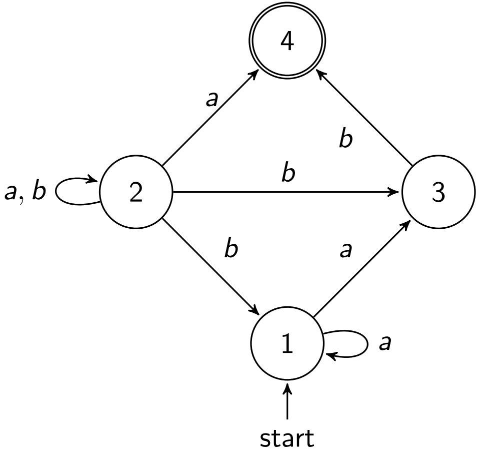
state	a	b
1	-	$\{3\}$
2	-	$\{4\}$
3	$\{5\}$	-
4	$\{3,4\}$	-
5	-	-

Take reachable part:



This will be the minimal automaton recognising the reverse language $ba(a)^*$

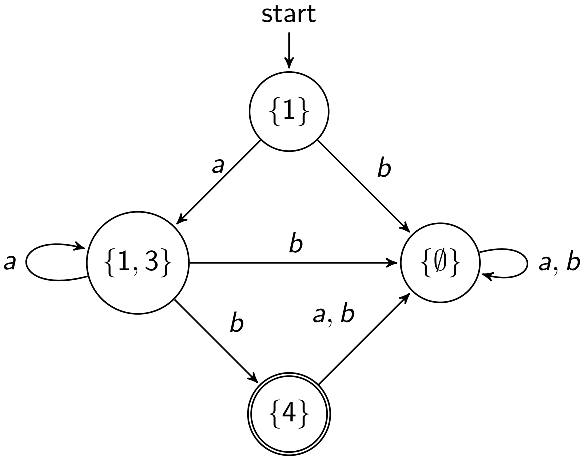
Reverse the new automaton:



Give transitions for powerset construction

state	a	b
1	{1,3}	-
2	{2,4}	{1,2,3}
3	-	{4}
4	-	-

Construct reachable part:



Now we have the minimal automaton that accepts $a(a)^*b$

How do we prove its correctness?

Simple proof of correctness given by *[Sakarovitch,2009]*

Coalgebraic proof given by *[Bonchi. et al, 2012]*

Benefits:

- helps with understanding of algorithm
- used to generalise Brzozowski to other types of Automaton (Weighted Automata , Kleene Algebra with Tests . . .)

Structure

To explain the coalgebraic proof of Brzozowski's algorithm, we must first understand some of the concepts behind it, including:

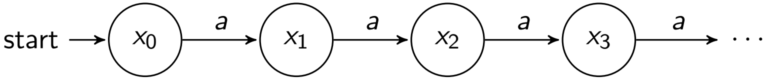
- Category Theory *(categories, functors, homomorphisms)*
- Universal Algebra and Coalgebra
- Reachability and Observability of Automata

Reachability

Definition

For any $x_i \in X$, there exists a word $w \in A^*$ such that applying that word to the initial state, we get x:

Example:



Observability

Definition

Each state in an a given automaton recognises a unique lanaguage (so has a unique behaviour).

Example:

$$\cdots \xrightarrow{a} x_{n-3} \xrightarrow{a} x_{n-2} \xrightarrow{a} x_{n-1} \xrightarrow{a} x_n$$

Automata as a Coalgebra

Rewrite transition function:

$$X \times A \longrightarrow X$$

As the coalgebra:

$$X \longrightarrow X^A$$

We represent an automaton as an algebra and a coalgebra

$$\begin{tikzcd} 1 \arrow[dr, "i"] & & 2 \\ & X \arrow[ur, "f"] \arrow[d, "t"] & \\ & X^A & \end{tikzcd}$$

Algebra = initial state function + transition structure

F-algebra

F-algebra where $F(X) = 1 + (A \times X)$

$$\begin{tikzcd}
1 \arrow[dr, "i"] & \\
& X \\
A \times X \arrow[ur, "t"] &
\end{tikzcd}$$

Reachability Algebraically

We can define the following automaton to represent reachability:

$$1 \xrightarrow{\epsilon} A^* \xrightarrow{\alpha} (A^*)^A$$

where:
$\epsilon : 1 \to A^*$
$\epsilon(^*) = x_0$

$\alpha : A^* \to (A^*)^A$
$\alpha(w)(a) = w \cdot a$

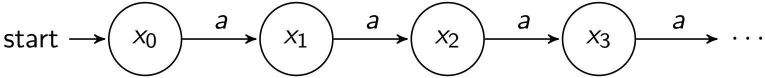
Reachability Algebra

which is the F-algebra $F(A^*) = 1 + (A \times A^*)$

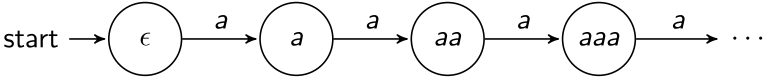
$$
\begin{tikzcd}
1 \arrow[dr, "\epsilon"] & \\
& A \\
A \times A^* \arrow[ur, "\alpha"] &
\end{tikzcd}
$$

Reachability Example

Original Example:



Reachability automaton:



Reachability Homomorphism

$$\begin{tikzcd} 1 \arrow[d, "\epsilon"] \arrow[dr, "i"] & \\ A^* \arrow[r, dotted, "r"] \arrow[d, "\alpha"] & X \arrow[d, "t"] \\ (A^*)^A \arrow[r, dotted, "r^A"] & X^A \end{tikzcd}$$

$r : A^* \to X$
$r(w) = i_w$

We represent an automaton as an algebra and a coalgebra

$$\begin{tikzcd} 1 \arrow[dr, "i"] & & 2 \\ & X \arrow[ur, "f"] \arrow[d, "t"] & \\ & X^A & \end{tikzcd}$$

Coalgebra = transition structure and final state function

Observability Coalgebraically

$$\begin{tikzcd} & 2 \\ X \arrow[ur, "f"] \arrow[r, dotted, "o"] \arrow[d, "t"] & 2^{A^*} \arrow[u, "\epsilon?"'] \arrow[d, "\beta"] \\ X^A \arrow[r, dotted, "o^A"] & (2^{A^*})^A \end{tikzcd}$$

$o : X \to 2^{A^*}$
$o = \{w \in A^* | f(x_w) = 1\}$

Applying both homomorphisms to the automaton

$$
\begin{tikzcd}
1 \arrow[d, "\epsilon"] \arrow[dr, "i"] & & 2 \\
A^* \arrow[r, dotted, "r"] \arrow[d, "\alpha"] & X \arrow[r, dotted, "o"] \arrow[d, "t"] \arrow[ur, "f"] & 2^{A^*} \arrow[u, "\epsilon?"'] \arrow[d, "\beta"] \\
(A^*)^A \arrow[r, dotted, "r^A"] & X^A \arrow[r, dotted, "o^A"] & (2^{A^*})^A
\end{tikzcd}
$$

r and o both apply to this automaton, so it is **minimal**.

Creating the Duality

Theorem

Let $\mathcal{X} = (X, t, i, f)$ be a determinsitic automaton and $\mathcal{R} = (2^X, 2^t, f, 2^i)$ be the reverse deterministic automaton of that automaton. If $\mathcal{X}$ is reachable, then $\mathcal{R}$ is observable and if $\mathcal{X}$ accepts a language L, then $\mathcal{R}$ accepts the reverse langauge $rev(L)$.

To prove this, we must construct the reverse language.

Contravariant Powerset Functor

Given sets A and B and a morphism $f : A \to B$

$$A \xrightarrow{f} B$$

applying the contravariant powerset functor $2^{(-)} : Set \to Set$ gives us

$$2^B \xrightarrow{2^f} 2^A$$

Reversed Automaton

$$\begin{tikzcd}
1 \arrow[dr, "f"] & & 2 \\
& 2^X \arrow[ur, "2^i"'] \arrow[d, "2^t"] & \\
& 2^{X^A} &
\end{tikzcd}$$

With reachability and observability

$$\begin{tikzcd}
1 \arrow[d, "\epsilon"] \arrow[dr, "f"] & & 2 \\
A^* \arrow[r, dotted, "R"] \arrow[d, "\alpha"] & 2^X \arrow[r, dotted, "O"] \arrow[d, "2^t"] \arrow[ur, "2^i"] & 2^{A^*} \arrow[u, "\epsilon?"'] \arrow[d, "\beta"] \\
(A^*)^A \arrow[r, dotted, "R^A"] & (2^X)^A \arrow[r, dotted, "O^A"] & (2^{A^*})^A
\end{tikzcd}$$

Getting Brzozowski from this

1. Do reverse construction of M
2. Take reachable part of $rev(M)$
3. Apply reverse construction to $rev(M)$
4. Take reachable part

Proving the theorem

We need to prove two lemmas:

Lemma

If a function $f : A \to B$ is surjective, then applying the contravariant poweset functor to it will give the injective function $2^f : 2^B \to 2^A$

Lemma

The function rev that maps a language to its reverse language is bijective.

If $\mathcal{X}$ is reachable, $\mathcal{R}$ is observable

- $\mathcal{X}$ is reachable, so r is surjective
- by Lemma 1, 2^r is injective
- by Lemma 2, *rev* is bijective
- $O = 2^r \circ rev$, so is injective overall
- O must be injective for $\mathcal{R}$ to be observable

If $\mathcal{X}$ accepts L, $\mathcal{R}$ accepts $rev(L)$

- In $\mathcal{X}$, observability is $o(i) = \{w \in A^* | f(i_w) = 1\}$
- Observability in $\mathcal{R}$ is $O(f) = \{w \in A^* | 2^i(f_w) = 1\}$
- Equivalently this is $\{w^R \in A^* | i_w \in f\}$
- We can get the reverse language of this using rev : $rev(\{w \in A^* | i_w \in f\}$
- Which we can rewrite as $rev(\{w \in A^* | f(i_w) = 1\}$
- This is the same as observability in the original automaton: $rev(o(i))$

Conclusion

Now we have discussed:

- the stages of Brzozowski's algorithm
- reachability and observabilty of automata and the duality between these properties
- Representing an automaton as both an algebra and coalgebra
- Reversing this representation of an automaton using the contravariant powerset functor
- How this can be used to prove the correctness of Brzozowski's Algorithm