COLLAPSING NON-DETERMINISTIC AUTOMATA

By
TARUN JANGPANGI
And
REMISH LEONARD MINZ
**NEED FOR MINIMIZATION**

- Finite Automata are useful for many important applications

**Applications**
- Implementing and Designing digital circuits.
- The “lexical analyzer” of a typical compiler.
- Software for verifying system of all types that has finite number of distinct states.

- In case of implementing digital circuits it is always required to minimize the automata so that
  - It occupies less fabrication area.
  - Make the digital circuit simpler for analysis.
  - Make the digital circuit cost effective.
ISOMORPHISM IN NON-DETERMINISTIC AUTOMATA

- With respect to minimization, nondeterministic automata are not necessarily unique up to isomorphism.

For example:
NFA for $0^+$

Both are minimal but not unique and not isomorphic.
Different algorithms for minimizing NFA

- Transformation reconstruction (by Kameda and Wiener)
- By autobisimulation
The method that we use for collapsing non-deterministic automata is “bisimulation”.

The collapsing relation between deterministic automata and bisimulation relation for nondeterministic automata are strongly related.

The latter generalizes the former in two significant ways.
- They work for non-deterministic automata
- They can relate two different automata.
BISIMULATION

Let
\[ M = (Q_M, \Sigma, \Delta_M, S_M, F_M), \]
\[ N = (Q_N, \Sigma, \Delta_N, S_N, F_N) \]
be two NFA's.

Let \( \approx \) be a binary relation relating states of \( M \) with states of \( N \);
i.e. \( \approx \) is subset of \( Q_M \times Q_N \).
Now for,
\[ B \subseteq Q_N \]
Define, \[ C_\approx (B) = \{ p \in Q_M \mid \exists q \in B, p \approx q \} \]
similarly for \( A \subseteq Q_M \)
Define, \[ C_\approx (A) = \{ q \in Q_N \mid \exists p \in A, p \approx q \} \]
The relation \( \approx \) can be extended in a natural way to subset of \( Q_M \) and \( Q_N \).
For \( A \subseteq Q_M \) and \( B \subseteq Q_N \)
\[ A \approx B \iff A \subseteq C_\approx (B) \text{ and } B \subseteq C_\approx (A) \]
\[ \iff \forall p \in A \exists q \in B, p \approx q \text{ and } \forall q \in B \mid \exists p \in A, p \approx q. \]
The relation is called Bisimulation if the following three conditions are met.

(i) \( S_M \approx S_N \);
(ii) If \( p \approx q \) then for all \( a \in \Sigma \), \( \Delta_M (p,a) \approx \Delta_N (q,a) \)
(iii) If \( p \approx q \), then \( p \in F_M \) iff \( q \in F_N \).

- We can say that \( M \) and \( N \) are Bisimilar if there exist a Bisimulation between them.
- The Bisimilarity class of \( M \) is a family of all the NFA\'s that are Bisimilar to \( M \).
Basic Properties of Bisimulation

- Bisimulation is Symmetric.
  - If $\approx$ is a bisimulation between $M$ and $N$ then its reverse
    \[
    \{(q, p) \mid p \approx q\}
    \]
    is a Bisimulation between $M$ and $N$.

- Bisimulation is Transitive.
  - If $\approx_1$ is a bisimulation between $M$ and $N$ and $\approx_2$ is a
    bisimulation between $N$ and $P$, then there composition $\approx_1 \circ \approx_2$
    \[
    = \{(p, r) \mid \exists q \ p \approx_1 q \text{ and } q \approx_2 r\}
    \]
    is a bisimulation between $M$ and $P$.

- The union of any nonempty family of bisimulation between $M$ and $N$ is a bisimulation
  between $M$ and $N$.
  - Let $\{\approx_i \mid i \in I\}$ be a non empty indexed set of bisimulations
    between $M$ and $N$.
    Define by,
    \[
    \approx = \bigcup_i \approx_i ;
    \]
    thus $p \approx q \iff \exists i \in I, p \approx_i q$. 

THEOREM: BISIMILAR AUTOMATA ACCEPT THE SAME SET.

Suppose \( \approx \) is a bisimulation between \( M \) and \( N \).

Then for any \( x \in \Sigma^* \)
\[ \hat{\Lambda}_M (S_M, x) \approx \hat{\Lambda}_N (S_N, x). \]

By the conditions for bisimulation
\[ \hat{\Lambda}_M (S_M, x) \cap F_m \neq \emptyset \text{ iff } \hat{\Lambda}_N (S_N, x) \cap F_N \neq \emptyset. \]

So by definition of acceptance for non deterministic automata, \( x \in L(M) \text{ iff } x \in L(N) \). Since \( x \) is arbitrary,
\( L(M) = L(N) \).
Let \( \approx \) be a bisimulation between \( M \) and \( N \). The **support** of \( \approx \) in \( M \) is a set \( C_{\approx} (Q_N) \) i.e. The set of states of \( M \) that are related by \( \approx \) to some states of \( N \).

Lemma : A state of \( M \) is in the support of all bisimulation involving \( M \) iff it is **accessible**.

Proof : Let \( \approx \) be any arbitrary bisimulationaton between \( M \) and another automata.

then, Every start state of \( M \) is in support of \( \approx \) ; if \( p \) is in support of \( \approx \) then every element of the \( \Delta(p, a) \) is in the support of \( \approx \) for every \( a \in \Sigma \). From here we can say that that every accessible state of \( M \) is in support of \( \approx \).
**Autobisimulation**

An Autobisimulation is a bisimulation $\equiv_M$ between an automaton and itself.

Theorem: Any nondeterministic automaton $M$ has a coarsest autobisimulation $\equiv_M$.

The relation $\equiv_M$ is an equivalence relation.

Proof: let $B = \text{the set of all autobisimulation on } M$. Let $\equiv_M = \cup_{i \in B} \equiv_i$

Reflexive $= \text{due to presence of identity relation.}$

Symmetry $= (\text{by basic properties of bisimulation})$

Transitive $= (\text{by basic properties of bisimulation})$. 
HOW TO GET A MINIMAL NFA

Step 1. removing inaccessible states.
Step 2. collapse by maximal autobisimulation to get minimal NFA bisimilar to original NFA.

Suppose $M = (Q, \Sigma, \Delta, S, F)$ is our original automata.
$\equiv_M$ = our maximal autobisimulation on $M$.
Let $p$ = any state of $Q$
$[p] = \equiv$-equivalence class of $p$.
$[p] = \{ q | p \equiv q \}$
$\succeq = \{ (p, [p] | p \in Q) \}$
For any $A \subseteq Q$
$A' = \{ [p] | p \in A \}$
Lemma: For all $A, B \subseteq Q$,

(i) $A \subseteq C_{\equiv} (B) \iff A' \subseteq B'$

(ii) $A \equiv B \iff A' = B'$ and

(iii) $A \succeq A'$

Our minimized NFA is the Quotient automaton

$\mathcal{M}' = (Q', \sum, \Delta', S', F')$,

Where $\Delta' ([p], a) = \Delta(p, a)'$

$\Delta'$ is well defined because

$[p] = [q] \Rightarrow p \equiv q$

$\Rightarrow \Delta(p, a) \equiv \Delta(q, a)$ by definition of autobisimulation.

$\Rightarrow \Delta(p, a)' = \Delta(q, a)'$ by our lemma.
The relation $\geq$ is a bisimulation between $M$ and $M'$.

- As we have $S \geq S'$ and if $p \geq [q]$ then $p \equiv q$
  so $\Delta(p, a) \geq \Delta(p, a)' = \Delta'(\{p\}, a) = \Delta'([q], a)$

Similarly for final states
If $p \in F$ then $[p] \in F'$
also there exist $q \in [p]$ such that $q \in F$ then $q \equiv p$.

- The only autobisimulation on the $M'$ is the identity relation $=$.
- The quotient automaton $M'$ is the minimal automaton bisimilar to $M$ and is unique up to isomorphism.
If we take any automaton $N$ bisimilar to $M$ and we remove the inaccessible states and then collapse the resulting NFA by its maximal autobisimulation. We get an automaton isomorphic to $M'$. Where $\equiv_N = \text{maximal autobisimulation on } N$. $N' = \text{the quotient automaton}$. $M'$ and $N'$ are bisimilar gives one to one correspondence between the states of $M'$ and $N'$. 
AN ALGORITHM

- It computes the maximal bisimulation between any given pair of NFA’s M and N.
- In case of no bisimulations between M and N, the algorithm halts and reports failure.
- For the case M = N the algorithm computes the maximal autobisimulation.

The algorithm will mark pairs of states (p, q) where p ∈ Q_M and q ∈ Q_N.
A pair (p, q) will be marked when a proof is discovered that p and q cannot be related by any bisimulation.
Step 1: Write down table of all pairs \((p, q)\) initially unmarked.

Step 2: Mark \((p, q)\) if \(p \in F_M\) and \(q \notin F_N\) or vice versa.

Step 3: Repeat the following until no more changes occur: if \((p, q)\) is unmarked, and if for some \(a \in \Sigma\), either
- there exist \(p' \in \Delta_M(p, a)\) such that for all \(q' \in \Delta_N(q, a)\), \((p', q')\) is marked, or
- There exists \(q' \in \Delta_N(q, a)\) such that for all \(p' \in \Delta_M(p, a)\), \((p', q')\) is marked.

Then mark \((p, q)\).

Step 4: Define \(p \equiv q\) iff \((p, q)\) is never marked. If \(S_M \equiv S_N\) then \(\equiv\) is the maximal bisimulation between \(M\) and \(N\). If not, then no bisimulation between \(M\) and \(N\) exist.
THANK YOU