Product Construction

Robert M. Keller
Harvey Mudd College
April 2014
The Product Construction

This construction can be used to show closure of regular languages under various operators, such as $\cap$, $-$, and $\oplus$. 
The Product Construction

Given two DFA’s over the same alphabet

\[(\Sigma, Q^1, q_0^1, \delta^1, F^1),\]
\[(\Sigma, Q^2, q_0^2, \delta^2, F^2)\]

the product is a DFA

\[(\Sigma, Q^1 \times Q^2, (q_0^1 q_0^2), \delta, F)\]

where \[\delta((q^1, q^2), \sigma) = (\delta^1(q^1, \sigma), \delta^2(q^2, \sigma)).\]
Product Machine Simulation

- The product machine simulates both original machines operating in parallel lock-step.

- We can prove by induction for the extended function $\delta$:

  $$\forall x \in \Sigma^* \delta((q^1, q^2), x) = (\delta^1(q^1, x), \delta^2(q^2, x))$$
Product Machine

logical combiner
(one of 16 possible)

input 0110
0110
0110
Product Construction Example
Construct the product of these two acceptors

State set will be \{a, b\} x \{c, d, e\}.
Product Construction

Accepting states depending on which operation is desired.
Choice of $F$ in a product machine

- The choice of $F$ depends on which language operator we want.

- If interested in $L \cap M$, then use
  
  $$F = \{(q^1, q^2) \mid q^1 \in F^1 \land q^2 \in F^2\}$$

- For $L \cdot M$, use
  
  $$F = \{(q^1, q^2) \mid q^1 \in F^1 \land q^2 \notin F^2\}$$

- For $L \cup M$, use
  
  $$F = \{(q^1, q^2) \mid q^1 \in F^1 \lor q^2 \in F^2\}$$
Two ways to get union

- We just saw how union can be achieved using the product machine.

- A second way is to use the NFA union, then convert to a DFA.
Testing Equivalence of Two DFA

- Two DFA are said to be equivalent if they accept exactly the same language.

- To test equivalence, use the product construction with

  \[ F = \{(q^1, q^2) \mid q^1 \in F^1 \iff q^2 \in F^2\} \]

- The machines are equivalent iff all \textit{reachable} state pairs of the product are in this \( F \).
Equivalence of Regular Expressions

• We can test whether two regular expressions are equivalent by constructing DFAs for each, then applying the previous result.

• We can test regular expression inclusion in a similar manner.

• We can check general regular expression identities by viewing each regular expression variable as if a letter in the alphabet.
Example

- Check the identity:
  \((R \ast S) \ast = (R \cup S) \ast\)

- NFA for \((R \ast S) \ast\):
NFA to DFA

- DFA for \((R*S)^*\):

Simplifies to
Identity Checking, continued

- NFA for \((R \cup S)^*\)
NFA to DFA

• DFA for \((R \cup S)^*\)

Simplifies to (as both states are accepting)
Product Construction

Simplified DFAs

Product:

Conclusion?
Counterexample to \((R*S)^* = (R \cup S)^*\)

- Let \(R = \{r\}, S = \{s\}\).

- In the diagram, pick the shortest sequence leading to a non-accepting state of the product machine.

- \(r \notin (R*S)^*, \text{ but } r \in (R \cup S)^*\)
Exercise

• Check the identity

\[(R^*S^*)^* = (R \cup S)^*\]
Beware of Special Cases

• In some special cases, identities may hold that do not hold in general.

• Examples:
  • Languages over 1-letter alphabets.
    $$RS = SR$$
  • Languages that contain $$\varepsilon$$.
    $$(R*S)^* = (R \cup S)^*$$ if $$\varepsilon \in L(S)$$
Axiom Systems for Regular Expressions

- Sound and complete axiomatizations exist, cf.

Salomaa’s System F1 (+ is union)

\[ A_1 \quad \alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma, \]
\[ A_2 \quad \alpha(\beta\gamma) = (\alpha\beta)\gamma, \]
\[ A_3 \quad \alpha + \beta = \beta + \alpha, \]
\[ A_4 \quad \alpha(\beta + \gamma) = \alpha\beta + \alpha\gamma, \]
\[ A_5 \quad (\alpha + \beta)\gamma = \alpha\gamma + \beta\gamma, \]
\[ A_6 \quad \alpha + \alpha = \alpha, \]

In \( A_1 \)–\( A_{11} \), \( \alpha, \beta \) and \( \gamma \) are arbitrary regular expressions. (In fact, the axioms are infinite axiom schemata.) There are two rules of inference.

R1 (Substitution). Assume that \( \gamma' \) is the result of replacing an occurrence of \( \alpha \) by \( \beta \) in \( \gamma \). Then from the equations \( \alpha = \beta \) and \( \gamma = \delta \) one may infer the equation \( \gamma' = \delta \) and the equation \( \gamma' = \gamma \).

R2 (Solution of equations). Assume that \( \beta \) does not possess e.w.p. Then from the equation \( \alpha = \alpha\beta + \gamma \) one may infer the equation \( \alpha = \gamma\beta^* \).