# *Closure Properties of Regular Languages*

*We show how to combine regular languages.*

## *Closure Properties*

A set is ***closed*** under an operation if applying that operation to any members of the set always yields a member of the set.

For example, the positive integers are closed under addition and multiplication, but not division.

# *Closure under Kleene*

**Fact.** *The set of regular languages is closed under each Kleene operation.*

That is, if $L_1$ and $L_2$ are regular languages, then each of $L_1 \cup L_2$, $L_1 L_2$ and ${L_1}^*$ is regular.

## *Proving Closure under Kleene*

The easiest approach is to show that the REs for $L_1$ and $L_2$ can be combined or adjusted to form the RE for the combination language.

Example: The RE for $L_1 L_2$ is obtained by writing down the RE for $L_1$ followed by the RE for $L_2$.

# *Closure under Complementation*

**Fact.** *The set of regular languages is closed under complementation.*

The complement of language $L$, written $\overline{L}$, is all strings not in $L$ but with the same alphabet.

The statement says that if $L$ is a regular language, then so is $\overline{L}$.

To see this fact, take ***deterministic*** FA for $L$ and interchange the accept and reject states.

# *Closure under Intersection*

**Fact.** *The set of regular languages is closed under intersection.*

One approach: Use de Morgan's law:

$$L_1 \cap L_2 = \overline{(\overline{L_1} \cup \overline{L_2})}$$

and that regular languages are closed under union and complementation.

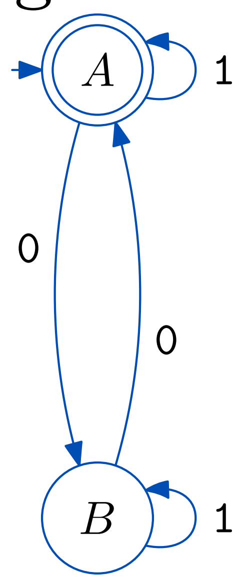
# *Product Construction for Intersection*

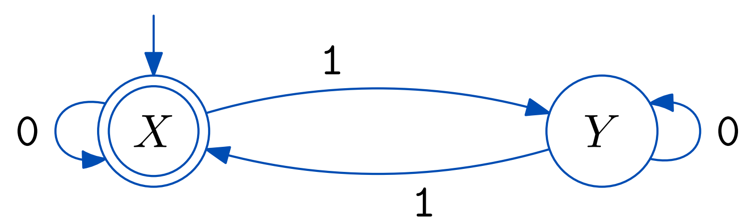
Each state in the ***product*** is pair of states from the original machines.

Formally, if $L_1$ is accepted by DFA $M_1$ with 5-tuple $(Q_1, \Sigma, q_1, T_1, \delta_1)$ and $L_2$ is accepted by DFA $M_2$ with 5-tuple $(Q_2, \Sigma, q_2, T_2, \delta_2)$. Then $L_1 \cap L_2$ is accepted by the DFA $(Q_1 \times Q_2, \Sigma, (q_1, q_2), T_1 \times T_2, \delta)$ where $\delta((r, s), \mathrm{x}) = (\delta_1(r, \mathrm{x}), \delta_2(s, \mathrm{x}))$.

# *Example: Even 0's and 1's*

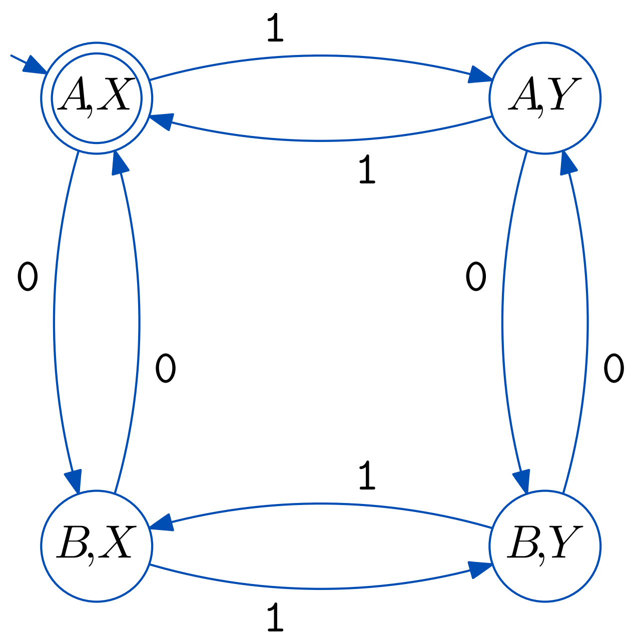
Suppose $L_1$ is the binary strings with an even number of 0's, and $L_2$ the binary strings with an even number of 1's. Then the FAs for these languages both have two states:




And so the FA for $L_1 \cap L_2$ has four states:

# *Product Construction for Even `0`'s and `1`'s*



# *Overview*

A regular language is one which has an FA or an RE. Regular languages are closed under union, concatenation, star, and complementation.