Intuiting Nondeterminism

- Nondeterministic machines are a serious departure from physical computers. How can we build up an intuition for them?
- There are two particularly useful frameworks for interpreting nondeterminism:
  - Perfect guessing
  - Massive parallelism
Perfect Guessing

\begin{itemize}
\item $q_0$
\item $a$
\item $q_1$
\item $b$
\item $q_2$
\item $a$
\item $q_3$
\end{itemize}
Perfect Guessing

\[\Sigma\]

\[
\begin{array}{c}
q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \\
\end{array}
\]

\[
\begin{array}{cccccc}
a & b & a & b & a & a
\end{array}
\]
Perfect Guessing

Transition diagram with states $q_0$, $q_1$, $q_2$, and $q_3$. The transitions are:
- From $q_0$: $a$ to $q_1$
- From $q_1$: $b$ to $q_2$
- From $q_2$: $a$ to $q_3$

Input alphabet $\Sigma = \{a, b\}$.
Perfect Guessing

\[
\begin{align*}
q_0 & \xrightarrow{\text{a}} q_1 & q_1 & \xrightarrow{\text{b}} q_2 & q_2 & \xrightarrow{\text{a}} q_3 \\
\text{start} & & & & \Sigma
\end{align*}
\]

\[
\begin{align*}
\text{a} & \quad \text{b} & \quad \text{a} & \quad \text{b} & \quad \text{a} & \quad \text{a}
\end{align*}
\]
Perfect Guessing

\[ \Sigma \]

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

\[ a b a b a b a a \]
Perfect Guessing

\[ \Sigma \]

\[
\begin{array}{cccc}
q_0 & \xrightarrow{a} & q_1 & \xrightarrow{b} & q_2 & \xrightarrow{a} & q_3 \\
\text{start} & & & & & & \\
\end{array}
\]

\[
\begin{array}{cccc}
a & b & a & b & a & a \\
\end{array}
\]
Perfect Guessing

δ

\begin{align*}
\text{start} & \rightarrow q_0 \\
q_0 & \xrightarrow{a} q_1 \\
q_1 & \xrightarrow{b} q_2 \\
q_2 & \xrightarrow{a} q_3
\end{align*}

a b a b a b a
Perfect Guessing

\[ a \rightarrow b \rightarrow a \rightarrow b \rightarrow a \rightarrow b \rightarrow a \]

Diagram:
- Start at state \( q_0 \)
- Move to state \( q_1 \) on input \( a \)
- Move to state \( q_2 \) on input \( b \)
- Move back to state \( q_0 \) on input \( a \)
- Move to state \( q_3 \) on input \( a \)
Perfect Guessing

\[
\begin{align*}
\Sigma & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad q_0 \\
\quad & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad q_1 \quad \quad \quad \quad \quad \quad \quad a \quad \quad \quad \quad \quad \quad \quad b \quad \quad \quad \quad \quad \quad \quad q_2 \quad \quad \quad \quad \quad \quad \quad a \quad \quad \quad \quad \quad \quad \quad q_3
\end{align*}
\]

\[
\begin{array}{ccccccc}
\text{a} & \text{b} & \text{a} & \text{b} & \text{a} & \text{a}
\end{array}
\]
Perfect Guessing

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

\[ \Sigma \]

Start

\[ a \ b \ a \ b \ a \ b \ a \]
Perfect Guessing

\( a \ b \ b \ a \ b \ a \ a \)
Perfect Guessing

• We can view nondeterministic machines as having *Magic Superpowers* that enable them to guess choices that lead to an accepting state.
  
  • If there is at least one choice that leads to an accepting state, the machine will guess it.
  
  • If there are no choices, the machine guesses any one of the wrong guesses.

• No known physical analog for this style of computation – this is totally new!
Massive Parallelism

\[ \sum \]

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

[Sequence: a b a b a b a]
Massive Parallelism

\[ \Sigma \]

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

\[ a \ b \ a \ b \ a \ b \ a \]
Massive Parallelism

\[
\Sigma \\
\text{start} \\
q_0 \rightarrow a \rightarrow q_1 \rightarrow b \rightarrow q_2 \rightarrow a \rightarrow q_3
\]

\[
\text{a b a b a b a}
\]
Massive Parallelism

Σ

start

$q_0$ $q_1$ $q_2$ $q_3$

$a$ $b$ $a$ $a$ $a$ $b$ $a$ $a$
Massive Parallelism

The diagram shows a state transition diagram with states labeled \( q_0, q_1, q_2, q_3 \) and transitions labeled with symbols \( a \) and \( b \). The input alphabet is \( \Sigma \). The transitions are:

- From \( q_0 \) to \( q_1 \) on input \( a \)
- From \( q_1 \) to \( q_2 \) on input \( b \)
- From \( q_2 \) to \( q_3 \) on input \( a \)
- From \( q_3 \) to \( q_0 \) on input \( a \) (loop)

The sequence of inputs shown in the diagram is \( a \ b \ a \ b \ a \ a \).
Massive Parallelism

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

\[ \Sigma \]

a b a b a a
Massive Parallelism

\[ \Sigma \]

\[ \text{start} \rightarrow q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

\[ \text{a b a b a b a a} \]
Massive Parallelism

=start

$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3$

$a\ b\ a\ b\ a\ a$

$\Sigma$
Massive Parallelism

\[ \Sigma \]

\[
\begin{align*}
q_0 & \xrightarrow{a} q_1 \\
q_1 & \xrightarrow{b} q_2 \\
q_2 & \xrightarrow{a} q_3
\end{align*}
\]
Massive Parallelism

\[ a \ b \ a \ b \ a \]

Diagram:

- Start state: \( q_0 \)
- States: \( q_1, q_2, q_3 \)
- Transitions: 
  - \( a \rightarrow q_1 \)
  - \( b \rightarrow q_2 \)
  - \( a \rightarrow q_3 \)

Input alphabet: \( \Sigma \)
Massive Parallelism

\[ \sum \]

\[
\begin{array}{c}
q_0 \\
\rightarrow \\
q_1 \\
\rightarrow \\
q_2 \\
\rightarrow \\
q_3
\end{array}
\]

\[
\begin{array}{cccc}
a & b & a & b \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow \\
q_0 & q_1 & q_2 & q_3
\end{array}
\]

\[
\begin{array}{cccc}
a & b & a & b \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow \\
q_0 & q_1 & q_2 & q_3
\end{array}
\]
Massive Parallelism
Massive Parallelism

a b a b a a

\[ \sum \]

q₀ → q₁ → q₂ → q₃
Massive Parallelism

Start

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

\[ \Sigma \]

Input: \[ a, b, a, b, a, b, a \]
Massive Parallelism

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \]

\[ \sum \]

\[
\begin{array}{cccccc}
\text{start} & q_0 & q_1 & q_2 & q_3 \\
\end{array}
\]

\[
\begin{array}{cccccc}
a & b & a & b & a \\
\end{array}
\]
Massive Parallelism

\[ \Sigma \]

\( q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_3 \)

Input sequence: \( ababaaba \)
Massive Parallelism

\[ \sum \]

start \[ q_0 \] \[ \rightarrow \] \[ q_1 \] \[ a \rightarrow \] \[ q_2 \] \[ b \rightarrow \] \[ q_3 \] \[ a \rightarrow \]

\[ a \ b \ a \ b \ a \ b \ a \ a \]
We're in at least one accepting state, so there's some path that gets us to an accepting state.

Therefore, we accept!
Massive Parallelism

• An NFA can be thought of as a DFA that can be in many states at once.

• At each point in time, when the NFA needs to follow a transition, it tries all the options at the same time.

• (Here's a rigorous explanation about how this works; for reading after class).
  • Start off in the set of all states formed by taking the start state and including each state that can be reached by zero or more ε-transitions.
  • When you read a symbol a in a set of states S:
    - Form the set S’ of states that can be reached by following a single a transition from some state in S.
    - Your new set of states is the set of states in S’, plus the states reachable from S’ by following zero or more ε-transitions.
So What?

- Each intuition of nondeterminism is useful in a different setting:
  - Perfect guessing is a great way to think about how to design a machine.
  - Massive parallelism is a great way to test machines – and has nice theoretical implications.
- Nondeterministic machines may not be feasible, but they give a great basis for interesting questions:
  - Can any problem that can be solved by a nondeterministic machine be solved by a deterministic machine?
  - Can any problem that can be solved by a nondeterministic machine be solved *efficiently* by a deterministic machine?
- The answers vary from automaton to automaton.