Alternation Turing Machines

Logan Woodbury

[2017-03-21 Tue]

Outline

Nondeterministic Turing Machines

Alternation

ATIME and ASPACE

Σ_i-alternating Turing Machines and Π_i-alternating Turing Machines

Nondeterministic Turing Machines

Basic Idea

Deterministic Finite Automata $\rightarrow$ Nondeterministic Finite Automata

Basic Idea

Deterministic Finite Automata $\rightarrow$ Nondeterministic Finite Automata

Turing Machine $\rightarrow$?

Basic Idea

Deterministic Finite Automata $\rightarrow$ Nondeterministic Finite Automata

Turing Machine $\rightarrow$ Nondeterministic Turing Machine

Basic Idea

Deterministic Finite Automata $\rightarrow$ Nondeterministic Finite Automata

Turing Machine $\rightarrow$ Nondeterministic Turing Machine

P $\rightarrow$ NP

Formal Description

$$M = (Q, \Sigma, \iota, \_, A, \delta)$$

1. Q is the set of states
2. Σ is the Tape Alphabet
3. ι is the initial state: $\iota \in Q$
4. $\_$ is the blank symbol: $\_ \in \Sigma$
5. A is the set of accept states: $A \in Q$
6. δ is the transition function: $\delta \subset (Q \backslash A x \Sigma) \rightarrow P(Q x \Sigma x \{L, R\})$

Example: SAT

Assign all possible assignments of variables concurrently

Check if any of them evaluate to true... concurrently!

If you find one that does, accept!

Alternation

Basic Idea

NTMs are kinda... easy to please?

Basic Idea

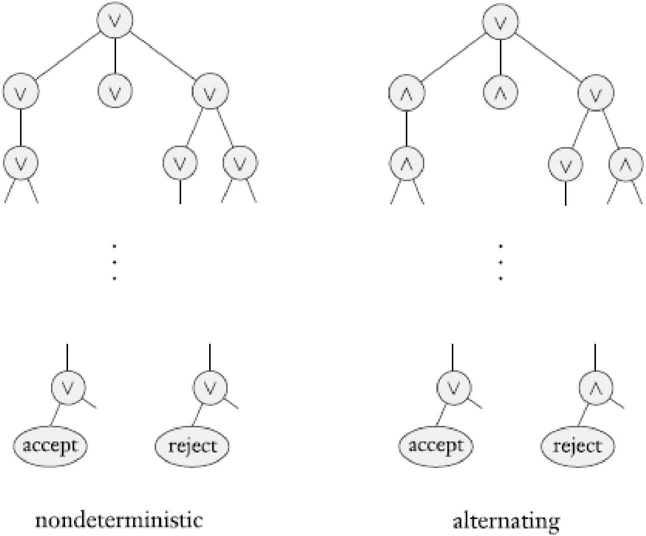
NTMs are kinda... easy to please?

All states are logical "or's"

Basic Idea

How about we have a system where we could have alternating or's and and's?

Basic Idea


accept
reject
nondeterministic
accept
reject
alternating

Basic Idea

Turing Machines $\rightarrow$ Nondeterministic Turing Machines $\rightarrow$ Alternating Turing Machines

P $\rightarrow$ NP $\rightarrow$ AP

Formal Description

$M = (Q, \Gamma, \delta, q_0, g)$

1. Q is the still set of states
2. Γ is now the Tape Alphabet
3. δ is still the transition function: $\delta : (Qx\Sigma) \rightarrow P(Qx\Gamma x\{L, R\})$
4. q_0 is now the initial state: $q_0 \in Q$
5. g is a function that specifies the *type* of each state
   $g : Q \rightarrow \{\wedge, \vee, accept, reject\}$

Examples: TAUT (Tautology)

$TAUT = \{\langle \Phi \rangle | \Phi \text{ is a tautology}\}$

1. Universally select all possible assignments to the variables of Φ ($\wedge$)
2. Evaluate these assignments to see if they are true
3. If all the assignments accept, accept! Otherwise... reject!

Examples: SEXY

L $=$ {S | S is a series of 1's in a positive multiple of 3, followed by an even amount of 0's, or the inverse (3x 0's followed by 2y 1's)}

Spent a lot of time on this one!

Not exactly a particularly difficult problem to solve anyhow, but. . .

Examples: MIN-FORMULA

$MIN - FORMULA = \{\langle\Phi\rangle | \Phi$ is the smallest possible way to express that formula$\}$

1. *Universally* select all formulas ψ that are shorter than Φ ($\wedge$)
2. *Existentially* select an assignment to the variables of Φ ($\vee$)
3. Evaluate both Φ and ψ, accept if they have the same values, otherwise reject!

ATIME and ASPACE

TIME and (Relative Dimension In) SPACE

ATIME and ASPACE (ATARDIAS?)

$ATIME(t(n)) = \{L | L$ is decided by an $O(t(n))$ time alternating Turing Machine $\}$

$ASPACE(f(n)) = \{L | L$ is decided by an $O(f(n))$ space alternating Turing Machine $\}$

Relations!

For $f(n) \geq n$, we have
$ATIME(f(n)) \subseteq SPACE(f(n)) \subseteq ATIME(f^2(n))$

For $f(n) \geq \log n$, we have $ASPACE(f(n)) = TIME(2^{O(f(n))})$

Σ_i-alternating Turing Machines and Π_i-alternating Turing Machines

Definitions

Σ_i-alternating Turing machine is an alternating Turing machine that on the longest possible branch has *iruns* universal or existential steps

Σ_i-alternating Turing machines start with existential steps

Π_i-alternating Turing machines start with universal steps

Σ¡TIME, Π¡TIME, Σ¡SPACE, Π¡SPACE

…Not hard to figure out what all these are

$\Sigma_i P$ and $\Pi_i P$

$$\Sigma_i P = \cup_{k \in \Re} \Sigma_i TIME(n^k)$$

$$\Pi_i P = \cup_{k \in \Re} \Pi_i TIME(n^k)$$

$\Sigma_i P$ and $\Pi_i P$

$$\Sigma_i P = \cup_{k \in \Re} \Sigma_i TIME(n^k)$$

$$\Pi_i P = \cup_{k \in \Re} \Pi_i TIME(n^k)$$

$\Sigma_1 P$

$\Sigma_i P$ and $\Pi_i P$

$$\Sigma_i P = \cup_{k \in \Re} \Sigma_i TIME(n^k)$$

$$\Pi_i P = \cup_{k \in \Re} \Pi_i TIME(n^k)$$

$$NP = \Sigma_1 P$$

$\Sigma_i P$ and $\Pi_i P$

$$\Sigma_i P = \cup_{k \in \Re} \Sigma_i TIME(n^k)$$

$$\Pi_i P = \cup_{k \in \Re} \Pi_i TIME(n^k)$$

$$NP = \Sigma_1 P$$

$$coNP = \Pi_1 P$$

$\Sigma_i P$ and $\Pi_i P$

$$\Sigma_i P = \cup_{k \in \Re} \Sigma_i TIME(n^k)$$

$$\Pi_i P = \cup_{k \in \Re} \Pi_i TIME(n^k)$$

$$NP = \Sigma_1 P$$

$$coNP = \Pi_1 P$$

$$MIN - FORMULA \in \Pi_2 P$$

$\Sigma_i P$ and $\Pi_i P$

$$\Sigma_i P = \cup_{k \in \Re} \Sigma_i TIME(n^k)$$

$$\Pi_i P = \cup_{k \in \Re} \Pi_i TIME(n^k)$$

$$NP = \Sigma_1 P$$

$$coNP = \Pi_1 P$$

$$MIN - FORMULA \in \Pi_2 P$$

$$SEXY \in \Sigma_3 P$$

$\Sigma_i P$ and $\Pi_i P$

$$\Sigma_i P = \cup_{k \in \Re} \Sigma_i TIME(n^k)$$

$$\Pi_i P = \cup_{k \in \Re} \Pi_i TIME(n^k)$$

$$NP = \Sigma_1 P$$

$$coNP = \Pi_1 P$$

$$MIN - FORMULA \in \Pi_2 P$$

$$SEXY \in \Sigma_3 P$$

(It is also most definitely in P)

Conclusion