

CSE 105—Theory of Computability

Spring 2006

Lecture 2—April 6, 2006

Non-deterministic Finite Automata

Instructor: Neil Rhodes

Review

Reversing a regular set

$$\text{Reverse}(L) = \{w \mid \text{Reverse}(W) \in L\}$$

Idea:

- Old final states are all start states (more than one start state?)
- Old start state is new final state
- Arrows go backward (more than one arrow with same symbol leaving state?)

Example

- Reverse of: every 0 is immediately followed by at least 2 ones
 - Which means: every 0 is immediately preceded by at least 2 ones

Non-deterministic Finite Automata (NFA)

There can be many paths through the NFA

- If *any one* path ends in a final state (after consuming all the input), accept the string
- Differences from Deterministic Finite Automata (DFA):
 - Can have moves on ϵ
 - Can have multiple edges with the same symbol leaving a state
 - Need not have edges for all symbols from a state (if can't leave a state, that path fails)

Example

- Binary string contains 101101

Example

Binary strings divisible by 4

Union using NFA

Binary strings ending in 00 or with an even number of 1's

Union using DFA

Binary strings ending in 00 or with an even number of 1's

Regular sets

A set is regular if some Finite State Automata (FSA) exists that recognizes it.

- Regular sets are closed under:
 - Union
 - Intersection
 - Complement
 - Concatenation

Converting an NFA to an equivalent DFA

Idea: in the DFA, keep track of the set of states that the NFA could be in.

- Example: Every 0 is immediately followed by at least 2 ones

More formal description

Given an NFA, N , create a DFA whose:

- states are the powerset of the NFA
- start state is the set of all states reachable from N 's start state
- final states are those states containing N 's final states
- arrows are of form: from a state in DFA, on a symbol go to all states that can be reached from that symbol in set of states in NFA (make sure to follow ϵ -arrows)
- If desired, remove all unreachable states

Concatenating two regular sets

A string from the first set followed by a string from the second set

- Idea: guess where the first string ends
- Example: L = binary strings divisible by 4, M = every 0 immediately followed by 2 1s

Regular Expressions

Recognize the same sets as DFA and NFA.

- Equivalent

What is a regular expression?

- Used for pattern matching
 - grep
 - perl
 - etc.
- $(abc)^*(d|e)fgh$
 - Matches any number of repeated abc, followed by either d, or e, then followed by fgh
- Three operations:
 - Concatenation: regular expressions appear next to each other
 - Union: The | specifies a choice among alternative regular expressions
 - Kleene star: represents 0 or more repetitions of a regular expression
- Recursive definition:
 - RE =
 - symbol from alphabet
 - ϵ
 - $\emptyset$
 - RE_1RE_2
 - $(RE_1)^*$
 - $(RE_1 | RE_2)$

Examples

Binary strings divisible by 4

Binary strings with each 0 immediately followed by two 1s

Binary strings with each 0 immediately preceded by two 1s

Binary string divisible by 4 followed by string with an even number of 0s

Strings of length 0

Strings over $\{a, b, c\}$ of length 5

Empty set

Relationship between RE and FSA

Any RE can be converted to an equivalent NFA

- Recursive construction
 - For a symbol:

 - For concatenation:

 - For union:

 - For Kleene star:

Example

Convert $(ab|(b^*))^*c$ to a FSA

Converting NFA to RE

First, convert to Generalized NFA (GNFA).

Then convert GNFA with k states to one with $k-1$ states

- $k-1$ to $k-2$
- ...
- 3 to 2

Then, convert GNFA with 2 states to regular expression

What is GNFA

- NFA with:
 - transitions labeled with regular expressions
 - start state
 - no arrows in
 - arrows out to every other state
 - single accept state
 - no arrows out
 - arrow in from every other state
 - other states
 - Single arrow between every pair of states (except start, accept)

Converting NFA to GNFA

Add new start state

- ϵ transition to old start state
- appropriate transitions to other states

Add new final state

- ϵ transitions from old final states
- appropriate transitions to other states

Add new transitions

- $\emptyset$ transitions where no transitions exist (other than from new accept or to new start)

Example

Converting GNFA_k to GNFA_{k-1}

Pick a state d to rip out of the GNFA_k

- not start or final state

Patch up all other pairs of states

- If label from i to j was RE_{ij} new label is $(\text{RE}_{ij} \mid \text{RE}_{id}(\text{RE}_{dd}^*)\text{RE}_{dj})$

Example:

Example converting RE to FSA

Want RE for binary strings not divisible by 3