CS 228 : Logic in Computer Science

S. Krishna

MSO on Words

- Signature $\tau = (Q_\Sigma, <, S)$, domain or universe = set of positions of a word
- MSO over words: Atomic formulae

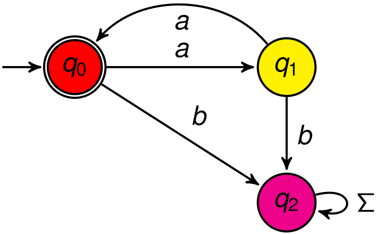
$$X(x) | Q_\Sigma(x) | x = y | x < y | S(x, y)$$

- Given a MSO sentence φ, $L(\varphi)$ defined as usual
- A language $L \subseteq \Sigma^*$ is MSO definable iff there is an MSO formula φ such that $L = L(\varphi)$
- Given an MSO sentence φ, is it satisfiable/valid?

MSO Expressiveness

- Clearly, $FO \subseteq MSO$
- $FO \subset$ Regular
- MSO=Regular

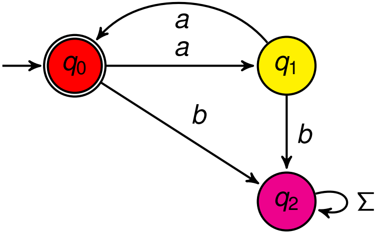
Regular Languages to MSO



Runs of words:

a *a* *a* *a*

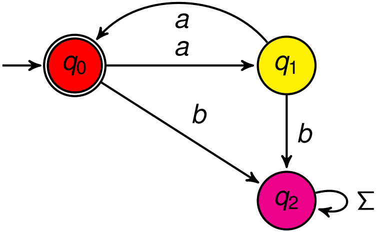
Regular Languages to MSO



Runs of words:

a	a	a	a	
q_0	q_1	q_0	q_1	q_0

Regular Languages to MSO

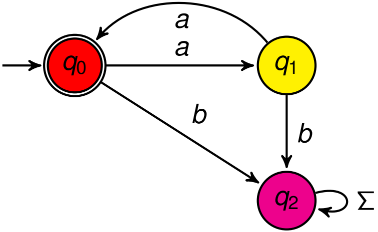


Runs of words:

a	a	a	a	
q_0	q_1	q_0	q_1	q_0

a	a	b	a

Regular Languages to MSO



Runs of words:

$$\begin{array}{ccccc} a & a & a & a & \\ q_0 & q_1 & q_0 & q_1 & q_0 \end{array}$$

$$\begin{array}{ccccc} a & a & b & a & \\ q_0 & q_1 & q_0 & q_2 & q_2 \end{array}$$

Regular Languages to MSO

Given a regular language L, and a DFA such that $L = L(A)$,

- Run of a word : at every position of the word, we are in some unique state

Regular Languages to MSO

Given a regular language L, and a DFA such that $L = L(A)$,

- Run of a word : at every position of the word, we are in some unique state

Position x	:	0	1	2	3	
		a	a	b	a	
		q_0	q_1	q_0	q_2	q_2

Regular Languages to MSO

Given a regular language L, and a DFA such that $L = L(A)$,

- Run of a word : at every position of the word, we are in some unique state

Position x	:	0	1	2	3	
		a	a	b	a	
		q_0	q_1	q_0	q_2	q_2

- For a state $q \in Q$, let X_q=the set of positions of the word where the state is q in the run

Regular Languages to MSO

Given a regular language L, and a DFA such that $L = L(A)$,

- Run of a word : at every position of the word, we are in some unique state

Position x	:	0	1	2	3	
		a	a	b	a	
		q_0	q_1	q_0	q_2	q_2

- For a state $q \in Q$, let X_q=the set of positions of the word where the state is q in the run
- $X_{q_0} = \{0, 2\}, X_{q_1} = \{1\}, X_{q_2} = \{3\}$

Regular Languages to MSO

Given a regular language L, and a DFA such that $L = L(A)$,

- Run of a word : at every position of the word, we are in some unique state

Position x	:	0	1	2	3	
		a	a	b	a	
		q_0	q_1	q_0	q_2	q_2

- For a state $q \in Q$, let X_q=the set of positions of the word where the state is q in the run
- $X_{q_0} = \{0, 2\}, X_{q_1} = \{1\}, X_{q_2} = \{3\}$
- The initial position of any word must belong to X_{q_0} : $0 \in X_{q_0}$

Regular Languages to MSO

- If a word *wa* is accepted, then
  - The last position x of the word satisfies $Q_a(x)$
  - For some state q, we have $X_q(x)$ and there is a transition $\delta(q, a) = q_f \in F$

Regular Languages to MSO

- If a word wa is accepted, then
  - The last position x of the word satisfies $Q_a(x)$
  - For some state q, we have $X_q(x)$ and there is a transition $\delta(q, a) = q_f \in F$

Position x	:	0	1	2	3	
		a	a	b	a	
		q_0	q_1	q_0	q_2	q_2

Regular Languages to MSO

- If a word *wa* is accepted, then
  - The last position x of the word satisfies $Q_a(x)$
  - For some state q, we have $X_q(x)$ and there is a transition $\delta(q, a) = q_f \in F$

Position x	:	0	1	2	3	
		a	a	b	a	
		q_0	q_1	q_0	q_2	q_2

- $Q_a(3)$ and $3 \in X_{q_2}$. $\delta(q_2, a) = q_2 \notin F$
- If x, y are consecutive positions in the word, and if $X_q(x) \wedge Q_a(x)$, then it must be that $X_t(y)$ such that $\delta(q, a) = t$

Regular Languages to MSO

- If a word wa is accepted, then
  - The last position x of the word satisfies $Q_a(x)$
  - For some state q, we have $X_q(x)$ and there is a transition $\delta(q, a) = q_f \in F$

Position x	:	0	1	2	3	
		a	a	b	a	
		q_0	q_1	q_0	q_2	q_2

- $Q_a(3)$ and $3 \in X_{q_2}$. $\delta(q_2, a) = q_2 \notin F$
- If x, y are consecutive positions in the word, and if $X_q(x) \wedge Q_a(x)$, then it must be that $X_t(y)$ such that $\delta(q, a) = t$
- $X_{q_0}(0)$, $X_{q_1}(1)$ and $Q_a(0)$. $\delta(q_0, a) = q_1$.
- $X_{q_1}(1)$, $X_{q_0}(2)$ and $Q_a(1)$. $\delta(q_1, a) = q_0$.

Regular Languages to MSO

Given a DFA $A = (Q, \Sigma, \delta, q_0, F)$, a word w is accepted iff it satisfies

$$\exists X_0 \exists X_1 \dots X_n \{[\forall x (X_0(x) \vee \dots \vee X_n(x)) \wedge \forall x \bigwedge_{i \neq j} \neg(X_i(x) \wedge X_j(x))] \wedge$$

Regular Languages to MSO

Given a DFA $A = (Q, \Sigma, \delta, q_0, F)$, a word w is accepted iff it satisfies

$$\exists X_0 \exists X_1 \dots X_n \{[\forall x (X_0(x) \vee \cdots \vee X_n(x)) \wedge \forall x \bigwedge_{i \neq j} \neg (X_i(x) \wedge X_j(x))] \wedge$$

$$[\exists x (\mathit{first}(x) \wedge X_0(x))] \wedge$$

Regular Languages to MSO

Given a DFA $A = (Q, \Sigma, \delta, q_0, F)$, a word w is accepted iff it satisfies

$$\exists X_0 \exists X_1 \ldots X_n \{[\forall x (X_0(x) \vee \cdots \vee X_n(x)) \wedge \forall x \bigwedge_{i \neq j} \neg (X_i(x) \wedge X_j(x))] \wedge$$

$$[\exists x (first(x) \wedge X_0(x))] \wedge$$

$$\forall x \forall y [S(x, y) \rightarrow \bigvee_{\delta(i,a)=j} [X_i(x) \wedge Q_a(x) \wedge X_j(y)]] \wedge$$

Regular Languages to MSO

Given a DFA $A = (Q, \Sigma, \delta, q_0, F)$, a word w is accepted iff it satisfies

$$\exists X_0 \exists X_1 \dots X_n \{[\forall x(X_0(x) \vee \dots \vee X_n(x)) \wedge \forall x \bigwedge_{i \neq j} \neg(X_i(x) \wedge X_j(x))] \wedge$$

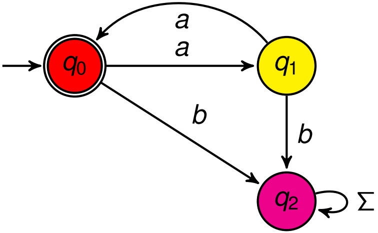
$$[\exists x(\mathit{first}(x) \wedge X_0(x))] \wedge$$

$$\forall x \forall y [S(x, y) \rightarrow \bigvee_{\delta(i,a)=j} [X_i(x) \wedge Q_a(x) \wedge X_j(y)]] \wedge$$

$$\exists x[\mathit{last}(x) \wedge \bigvee_{\delta(i,a)=j \in F} [X_i(x) \wedge Q_a(x)]]\}$$

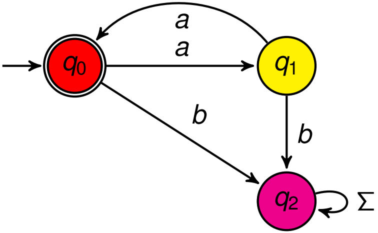
- $w \in L(A)$ iff $w \models \varphi$

Example : Regular to MSO



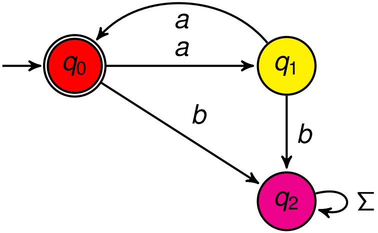
$\exists X_0 \exists X_1 \exists X_2 \{[\forall x(X_0(x) \vee X_1(x) \vee X_2(x)) \wedge \forall x[\neg(X_0(x) \wedge X_1(x)) \wedge$
$\neg(X_0(x) \wedge X_2(x)) \wedge \neg(X_1(x) \wedge X_2(x))] \wedge$

Example : Regular to MSO



$$\exists X_0 \exists X_1 \exists X_2 \{[\forall x(X_0(x) \vee X_1(x) \vee X_2(x)) \wedge \forall x[\neg(X_0(x) \wedge X_1(x)) \wedge$$
$$\neg(X_0(x) \wedge X_2(x)) \wedge \neg(X_1(x) \wedge X_2(x))] \wedge [\exists x(first(x) \wedge X_0(x))] \wedge$$

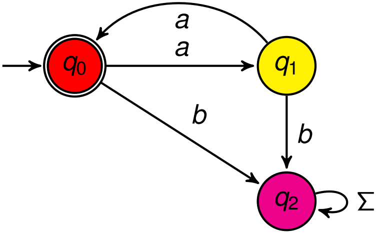
Example : Regular to MSO



$$\exists X_0 \exists X_1 \exists X_2 \{[\forall x (X_0(x) \vee X_1(x) \vee X_2(x)) \wedge \forall x [\neg(X_0(x) \wedge X_1(x)) \wedge$$
$$\neg(X_0(x) \wedge X_2(x)) \wedge \neg(X_1(x) \wedge X_2(x))] \wedge [\exists x (first(x) \wedge X_0(x))] \wedge$$

$$\forall x \forall y [S(x,y) \rightarrow [(X_0(x) \wedge Q_a(x) \wedge X_1(y)) \vee$$
$$(X_0(x) \wedge Q_b(x) \wedge X_2(y)) \vee (X_1(x) \wedge Q_a(x) \wedge X_0(y)) \vee$$
$$(X_1(x) \wedge Q_b(x) \wedge X_2(y)) \vee (X_2(x) \wedge Q_a(x) \wedge X_2(y)) \vee (X_2(x) \wedge Q_b(x) \wedge X_2(y))]]$$

Example : Regular to MSO



$$\exists X_0 \exists X_1 \exists X_2 \{[\forall x(X_0(x) \vee X_1(x) \vee X_2(x)) \wedge \forall x[\neg(X_0(x) \wedge X_1(x)) \wedge$$

$$\neg(X_0(x) \wedge X_2(x)) \wedge \neg(X_1(x) \wedge X_2(x))] \wedge [\exists x(first(x) \wedge X_0(x))] \wedge$$

$$\forall x \forall y[S(x,y) \rightarrow [(X_0(x) \wedge Q_a(x) \wedge X_1(y)) \vee$$

$$(X_0(x) \wedge Q_b(x) \wedge X_2(y)) \vee (X_1(x) \wedge Q_a(x) \wedge X_0(y)) \vee$$

$$(X_1(x) \wedge Q_b(x) \wedge X_2(y)) \vee (X_2(x) \wedge Q_a(x) \wedge X_2(y)) \vee (X_2(x) \wedge Q_b(x) \wedge X_2(y))]]$$

$$\wedge \exists x[last(x) \wedge (X_1(x) \wedge Q_a(x))]\}$$

MSO to Regular Languages

- Every MSO sentence φ over words can be converted into a DFA A_φ such that $L(\varphi) = L(A_\varphi)$.
- Start with atomic formulae, construct DFA for each of them.
- Conjunctions, Disjunctions, Negation easily handled via union, intersection and complementation of respective DFA
- Handling quantifiers?

Atomic Formulae to DFA

- $Q_a(x)$: All words which have an a. Need to fix a position for x, where a holds.

Atomic Formulae to DFA

- $Q_a(x)$: All words which have an a. Need to fix a position for x, where a holds.
- Think of a word $baab$ which satisfies $Q_a(x)$ as $\begin{matrix} baab \\ 0010 \end{matrix}$ or $\begin{matrix} baab \\ 0100 \end{matrix}$

Atomic Formulae to DFA

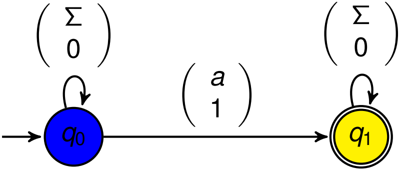
- $Q_a(x)$: All words which have an a. Need to fix a position for x, where a holds.
- Think of a word *baab* which satisfies $Q_a(x)$ as $\begin{matrix} baab \\ 0010 \end{matrix}$ or $\begin{matrix} baab \\ 0100 \end{matrix}$
- The first row is over Σ, and the second row captures a possible assignment to x

Atomic Formulae to DFA

- $Q_a(x)$: All words which have an a. Need to fix a position for x, where a holds.
- Think of a word *baab* which satisfies $Q_a(x)$ as $\begin{matrix} baab \\ 0010 \end{matrix}$ or $\begin{matrix} baab \\ 0100 \end{matrix}$
- The first row is over Σ, and the second row captures a possible assignment to x
- Think of an extended alphabet $\Sigma' = \Sigma \times \{0, 1\}$, and construct an automaton over Σ'.

Atomic Formulae to DFA

- $Q_a(x)$: All words which have an a. Need to fix a position for x, where a holds.
- Think of a word *baab* which satisfies $Q_a(x)$ as $\begin{matrix} baab \\ 0010 \end{matrix}$ or $\begin{matrix} baab \\ 0100 \end{matrix}$
- The first row is over Σ, and the second row captures a possible assignment to x
- Think of an extended alphabet $\Sigma' = \Sigma \times \{0, 1\}$, and construct an automaton over Σ'.
- Deterministic, not complete.



Simple Formulae to DFA

- $Q_a(x) \wedge X(x)$ means that the position x is in the set X, and letter a is true when $x = 1$.

Simple Formulae to DFA

- $Q_a(x) \wedge X(x)$ means that the position x is in the set X, and letter a is true when $x = 1$.
- Think of a word *baab* which satisfies $Q_a(x) \wedge X(x)$ as

baab		*baab*
0010	or	0100
$DD1D$		$D1DD$

  where D stands for *dont care*. X can have value 0 or 1 at D.

Simple Formulae to DFA

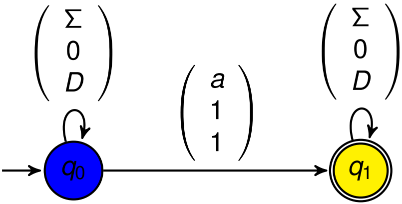
- $Q_a(x) \wedge X(x)$ means that the position x is in the set X, and letter a is true when $x = 1$.
- Think of a word *baab* which satisfies $Q_a(x) \wedge X(x)$ as

baab		*baab*
0010	or	0100
$DD1D$		$D1DD$

  where D stands for *dont care*. X can have value 0 or 1 at D.
- However, the position where $x = 1$ must belong to X.

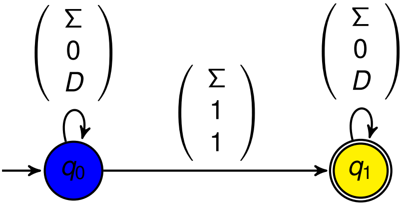
Simple Formulae to DFA

- The first row is over Σ, and the second row captures a possible assignment to x, and the third row captures a possible assignment to X.
- Think of an extended alphabet $\Sigma' = \Sigma \times \{0, 1\} \times \{0, 1\}$, and construct an automaton over Σ'.
- $Q_a(x) \wedge X(x)$: deterministic, not complete



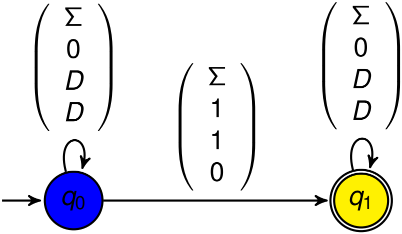
Simple Formulae to DFA

- $X(x)$



Simple Formulae to DFA

- $X(x) \wedge \neg Y(x)$
- $\Sigma' = \Sigma \times \{0,1\} \times \{0,1\} \times \{0,1\}$



Formulae to DFA

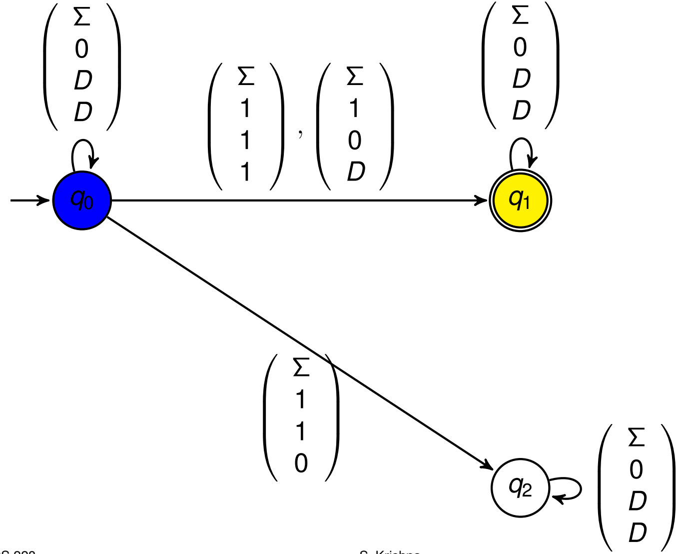
- Given $\varphi(x_1, \ldots, x_n, X_1, \ldots, X_m)$, an MSO formula over Σ, consider the extended alphabet

$$\Sigma' = \Sigma \times \{0, 1\}^{m+n}$$

- Assign values to x_i, X_j at every position as seen in the cases of atomic formulae
- Keep in mind that every x_i can be assigned 1 at a unique position

Handling Quantifiers

- $\exists X \exists Y \forall x [X(x) \rightarrow Y(x)]$



Points to Remember

- Given $\varphi(x_1, \ldots, x_n, X_1, \ldots, X_m)$, construct automaton for atomic MSO formulae over the extended alphabet $\Sigma \times \{0, 1\}^{m+n}$
- Intersect with the regular language where every x_i is assigned 1 exactly at one position
- Given a sentence $Q_{x_1} \ldots Q_{x_n} Q_{X_1} \ldots Q_{X_m} \varphi$, first construct the automaton for the formula $\varphi(x_1, \ldots, x_n, X_1, \ldots, X_m)$
- Replace $\forall$ in terms of $\exists$

Points to Remember

- Given the automaton for $\varphi(x_1, \ldots, x_n, X_1, \ldots, X_n)$, the automaton for $\exists X_i \varphi(x_1, \ldots, x_n, X_1, \ldots, X_n)$ is obtained by projecting out the row of X_i
- This may result in an NFA
- Determinize it and complement it to get a DFA for $\neg\exists x_i \varphi(x_1, \ldots, x_n, X_1, \ldots, X_n)$
- Intersect with the regular language where each of $x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n$ are assigned 1 exactly at one position

The Automaton-Logic Connection

Given any MSO sentence φ, one can construct a DFA A_φ such that $L(\varphi) = L(A_\varphi)$. If a language L is regular, one can construct an MSO sentence φ such that $L = L(\varphi)$.