Deterministic expressions and unambiguous languages

Kamal Lodaya

The Institute of Mathematical Sciences, Chennai

Joint work with Paritosh Pandya and Simoni Shah (TIFR, Mumbai),

presented at IFIP TCS, Milano, September 2008

Regular languages

Language L over Σ^*.

- Recognized by automata
- Defined by regular expressions
- Specified by a formula of classical or temporal logic
  - Every a is eventually followed by b
  - $\Box(a \Rightarrow \Diamond b)$
  - $\forall x. \ (a(x) \ \Rightarrow \ \exists y. \ (x < y \land b(y)))$
- Specified algebraically by its syntactic monoid.

Expressiveness Language classes and their corresponding expressions/logics/algebra/automata.

Subclasses of regular languages

Some well known results.

Logic	Interval TL	TL	Reg. Exp.	Automata	Algebraic
$MSO[<]$	QDDC	QPTL	RE	DFA	Monoids
$FO[<]$, $FO^3[<]$	ITL	LTL	SF	CFA	Aperiodic Monoids
$FO^2[<]$	?	UTL	?	$po2dfa$	DA

- LTL $\stackrel{\text{def}}{=}$ TL[U,S,X,Y] (Manna, Pnueli)
- UTL $\stackrel{\text{def}}{=}$ TL[$\Diamond$,$\diamondminus$] (Etessami, Vardi, Wilke)

Answer A fragment of ITL/SF with left- and right-deterministic marked concatenation which we call Unambiguous interval temporal logic/ Unambiguous starfree expressions

Questions

- Membership for regular expressions is NLogSpace-complete
- Membership for starfree expressions is PSpace-complete (Petersen)
- Equivalence for regular expressions is PSpace-complete
- Equivalence for starfree expressions is nonelementary (Stockmeyer and Meyer)
- What about membership and equivalence for unambiguous starfree expressions?

Logic ITL

- Word $w \in \Sigma^+$
- $pos(w)$ set of positions in the word.
- $INTV(w) \stackrel{\text{def}}{=} \{[i,j] \in pos(w)^2 \mid i \leq j\}$
- Satisfaction $w, [i,j] \models D$
- $w \models D$ iff $w, [1, \#w] \models D$.
  $L(D) = \{w \in \Sigma^+ \mid w \models D\}$

Syntax and semantics

Let $a \in \Sigma$. Let D, D_1, D_2 range over formulas in ITL.

Abstract syntax of ITL:

$$pt \;\mid\; \lceil a \rceil \;\mid\; D_1 ; D_2 \;\mid\; D_1 \vee D_2 \;\mid\; \neg D$$

Semantics of ITL:

$w, [i,j] \models pt$ iff $i = j$

$w, [i,j] \models \lceil a \rceil$ iff for all $k : i \leq k \leq j : w[i] = a$

$w, [i,j] \models D_1 ; D_2$ iff for some $k : i \leq k \leq j$ and $w, [i,k] \models D_1$ and $w, [k,j] \models D_2$

Questions

- Modelchecking for ITL is PSpace-complete
- Satisfiability for QDDC and ITL is nonelementary (Stockmeyer and Meyer)
- What about modelchecking and satisfiability for $UITL$?

Results

Reg. Exp.	Membership	Emptiness	Interval TL	Modelchecking	Satisfiability
RE	NLogSpace	PSpace	QDDC	PSpace	Nonelementary
SF	PSpace	Nonelem.	ITL	PSpace	Nonelementary
USF	LogDCFL	PSpace	UITL	Ptime	PSpace

- UTL satisfiability is NP, $FO^2[<]$ satisfiability is NExpTime
- ITL/SF subclasses expressively equivalent to $po2dfa$
- Direct effective translations to $po2dfa$(unlike logics $FO^2[<]$, UTL)

Unambiguous languages

- Studied by Schützenberger (1976)
- A concatenation e_1e_2 is *unambiguous* if for any word w in $L(e_1e_2)$, there is a unique factorization $w = uv$ such that $u \in L(e_1)$ and $v \in L(e_2)$
- An unambiguous language is recognized by a finite monoid which satisfies for some n the equation $(xyz)^{2n} = (xyz)^n y (xyz)^n$
- Such monoids are said to belong to the pseudovariety DA
- Conversely, every monoid in DA recognizes an unambiguous language
- Unambiguous languages are precisely those definable in $FO^2[<]$ (Thérien and Wilke 1998)

Main results

- Unambiguous interval temporal logic (UITL)/ Unambiguous starfree expressions (USF)
- Unique parsability
- Membership in LogDCFL, emptiness in PSpace (NP over a fixed alphabet)
- Effective construction $A: \ UITL \rightarrow po2dfa$ such that $L(D) = L(A(D))$. The size $|A(D)| \ = \ \mathcal{O}(|D|^2)$.
- Effective construction $D: \ po2dfa \rightarrow UITL$ such that $L(A) = L(A(D))$. The size $|D(A)| \ = \ \mathcal{O}(2^{|A|})$.

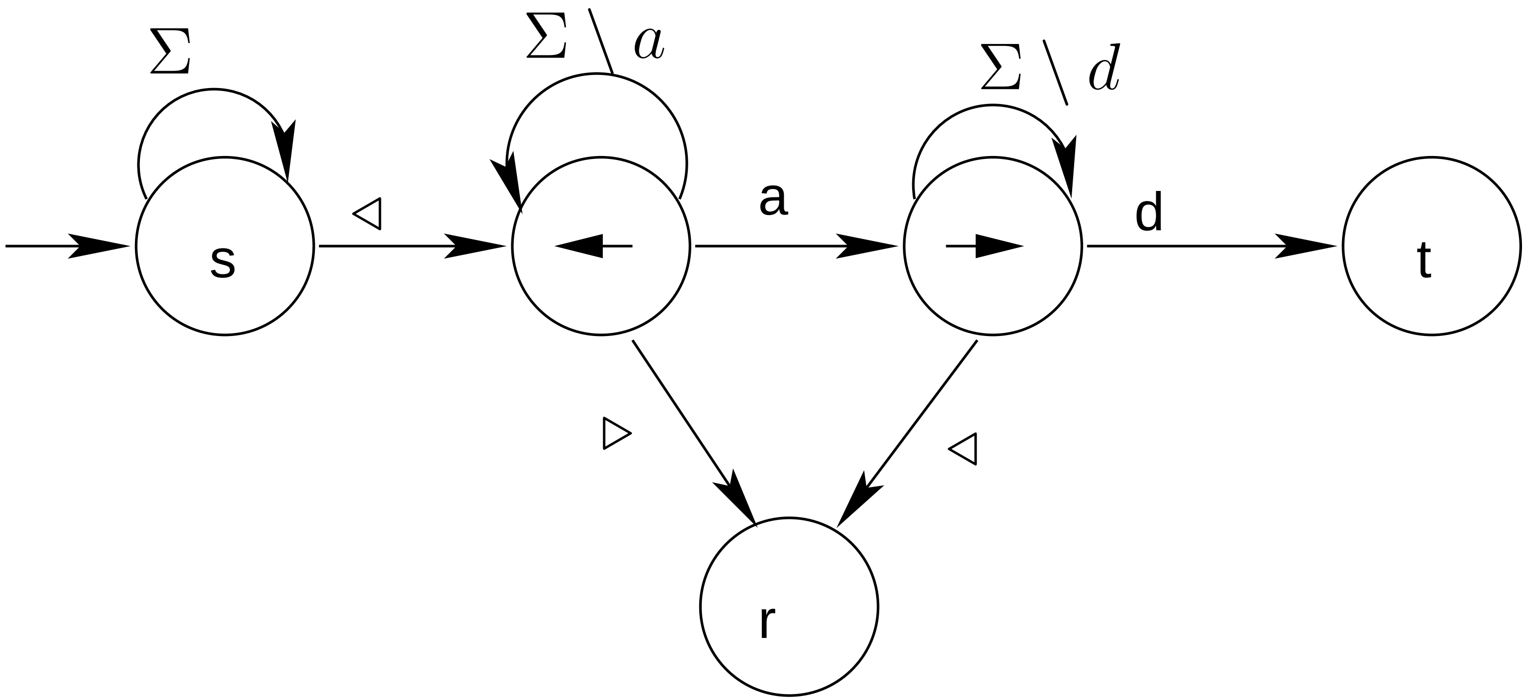
Partially ordered two-way DFA

Introduced by Schwentick, Thérien and Vollmer (2001).
A $po2dfa\ M = (Q, \leq, \Sigma, \delta, s, t, r)$ where

- $(Q, \leq)$ is a poset of states such that r (reject) and t (accept) are the only minimal elements.
- $\delta(q, a) \leq q$ (once a state is exited, it is not re-entered)
- $Q \ = \ Q_l \cup Q_r$ (left- and right-entered)
- $\delta : ((Q_l \cup Q_r) \times \Sigma) \to Q) \cup ((Q_l \times \{\triangleleft\}) \to Q \setminus Q_r) \cup ((Q_r \times \{\triangleright\}) \to Q \setminus Q_l)$
  (endmarkers to prevent falling off the end)

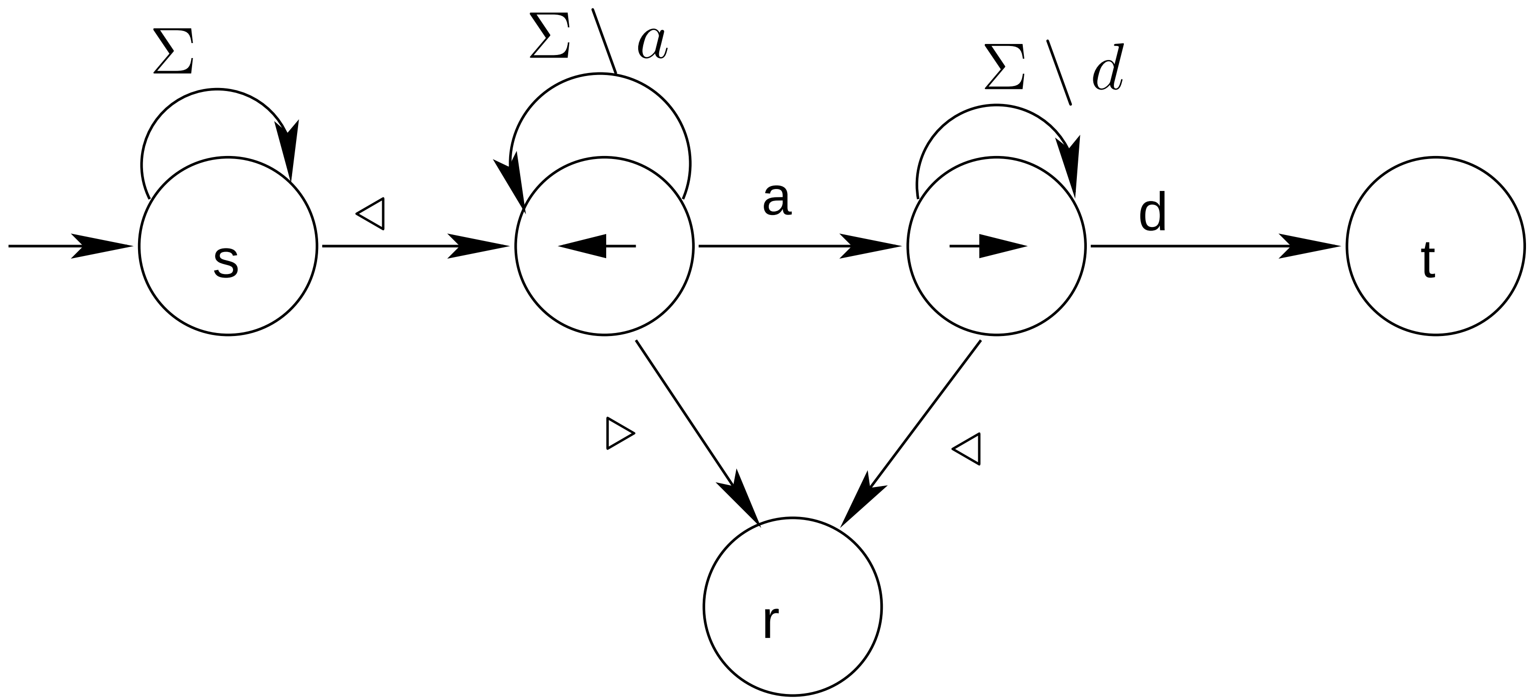
Example automaton

A $po2dfa$ accepting $\Sigma^* a(b+c)^* d(b+c+d)^*$.



Example automaton

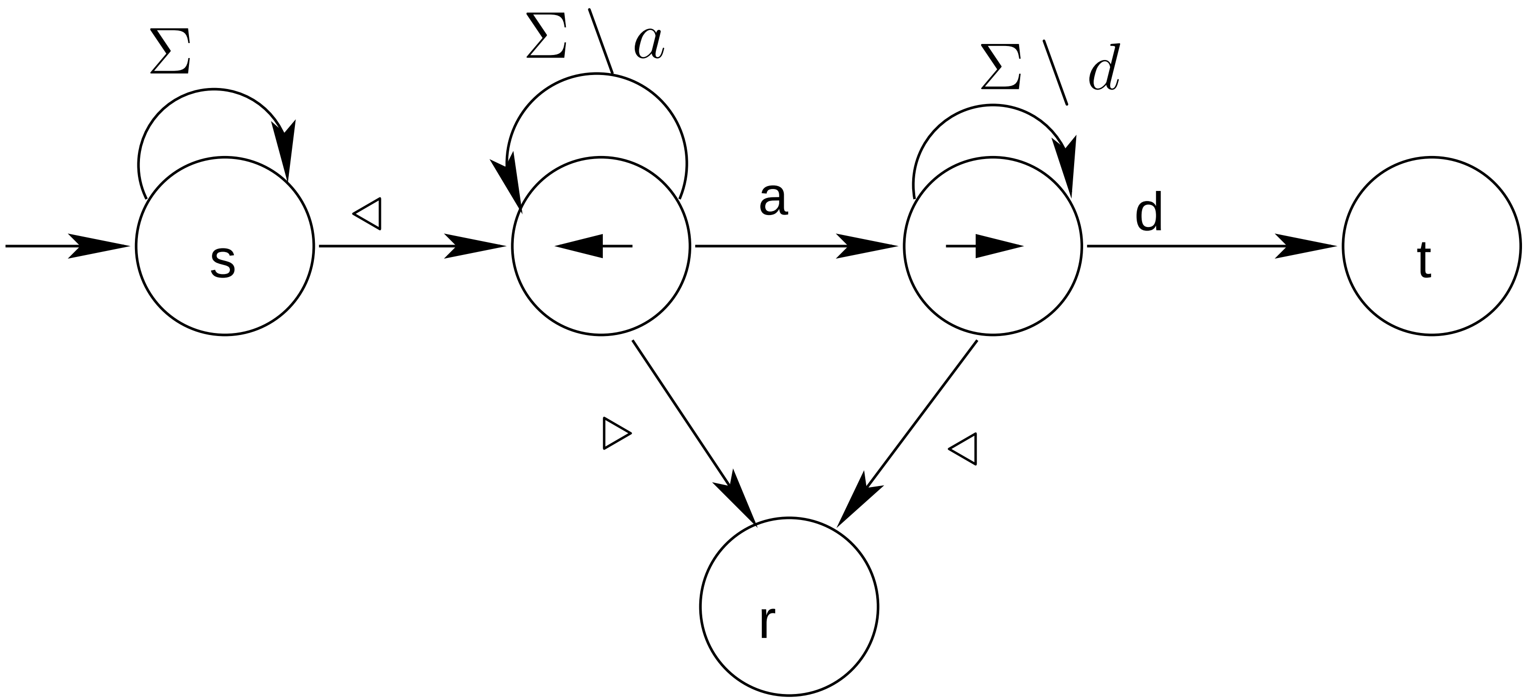
A $po2dfa$ accepting $\Sigma^* a(b+c)^* d(b+c+d)^*$.



Example Let $w' = \rhd abaab\textcolor{red}{d}dcb\lhd$. Then,
$M(w,*) = (t,6)$ (startable from anywhere).

Example automaton

A $po2dfa$ accepting $\Sigma^* a(b+c)^* d(b+c+d)^*$.



Turtle expression $\xrightarrow{\triangleleft}; \;\; \xleftarrow{a}; \;\; \xrightarrow{d}$

Run of $po2dfa$

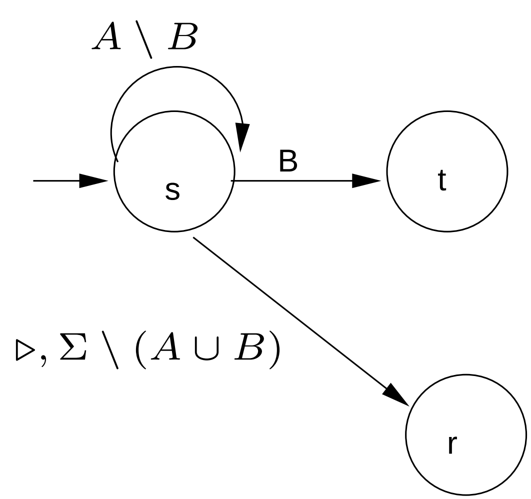
Given input w a $po2dfa$ $M = (Q, \leq, \delta, s, t, r)$ operates as follows:

- Scans the word $w' = \triangleright w \triangleleft$ in both directions.
- Configuration (q, i)
- $(q, i) \rightarrow (q', i')$ if $q, a, q' \in \delta$ and $w[i] = a$. Also, $i' = i + 1$ if $q' \in Q_l$ and $i' = i - 1$ if $q' \in Q_r$. We have $i' = i$ if $q' = t$ or $q' = r$,
- A run from position p_0 is a sequence of transitions $(q_0, p_0), (q_1, p_1), ...(q_f, p_f)$ with $q_0 = s$ and $q_f \in \{t, r\}$.
- $M(w, p_0) \ = \ (q_f, p_f)$.

Extended turtle expressions

Compositional way of constructing $po2dfa$, extending the turtle expressions originally defined by Schwentick, Thérien and Vollmer (2001).

- Acc accepts and Rej rejects without moving the head.
- Given $A, B \subseteq \Sigma'$, let $A \xrightarrow{B}$ be



- $A \xleftarrow{B}$

Extended turtle expressions

- $P?Q, R$ executes P first.
  If P accepts at position j then Q is executed from head position j.
  - Otherwise R is executed from head position j.
- $P;Q \stackrel{\text{def}}{=} P?Q, Rej$ and $\neg P \stackrel{\text{def}}{=} P?Rej, Acc$.

Unambiguous interval temporal logic

Property Between the last a and subsequent first d there must be no b.

Language Let $\Sigma = \{a, b, c, d\}$

$$\Sigma^* a c^* d \{b, c, d\}^*$$

Formula $D \stackrel{\text{def}}{=} (\top L_a ((\neg(\top F_b \top)) F_d \top))$

Logic UITL

Let $a \in \Sigma$. Let D, D_1, D_2 range over formulas in $UITL$.
(We use $\top$ for "true").

Abstract syntax of $UITL$:

$$\top \mid pt \mid D_1 F_a D_2 \mid D_1 L_a D_2 \mid D_1 \vee D_2 \mid \neg D$$
$$\mid \oplus D \mid \ominus D$$

Semantics

$w, [i, j] \models pt$ iff $i = j$

$w, [i, j] \models D_1 F_a D_2$ iff for some $k : i \leq k \leq j.\quad w[k] = a$ and
$\quad$ (for all $m : i \leq m < k.\ w[m] \neq a$) and
$\quad$ $w, [i, k] \models D_1$ and $w, [k, j] \models D_2$

$w, [i, j] \models D_1 L_a D_2$ iff for some $k : i \leq k \leq j.\quad w[k] = a$ and
$\quad$ (for all $m : k < m \leq j.\ w[m] \neq a$) and
$\quad$ $w, [i, k] \models D_1$ and $w, [k, j] \models D_2$

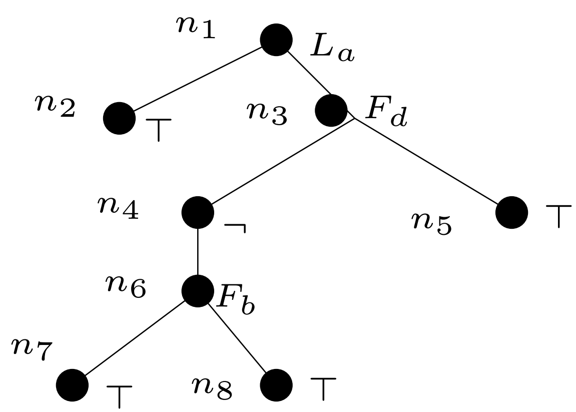
$w, [i, j] \models \oplus D$ iff $i < j$ and $w, [i + 1, j] \models D$

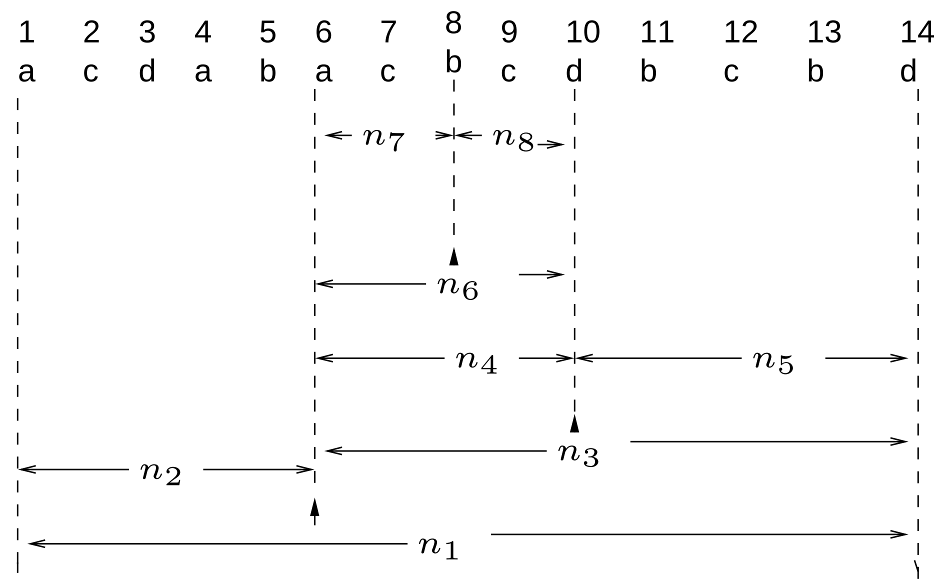
$w, [i, j] \models \ominus D$ iff $i < j$ and $w, [i, j - 1] \models D$

Note: $\lceil a \rceil$ is definable.

Example

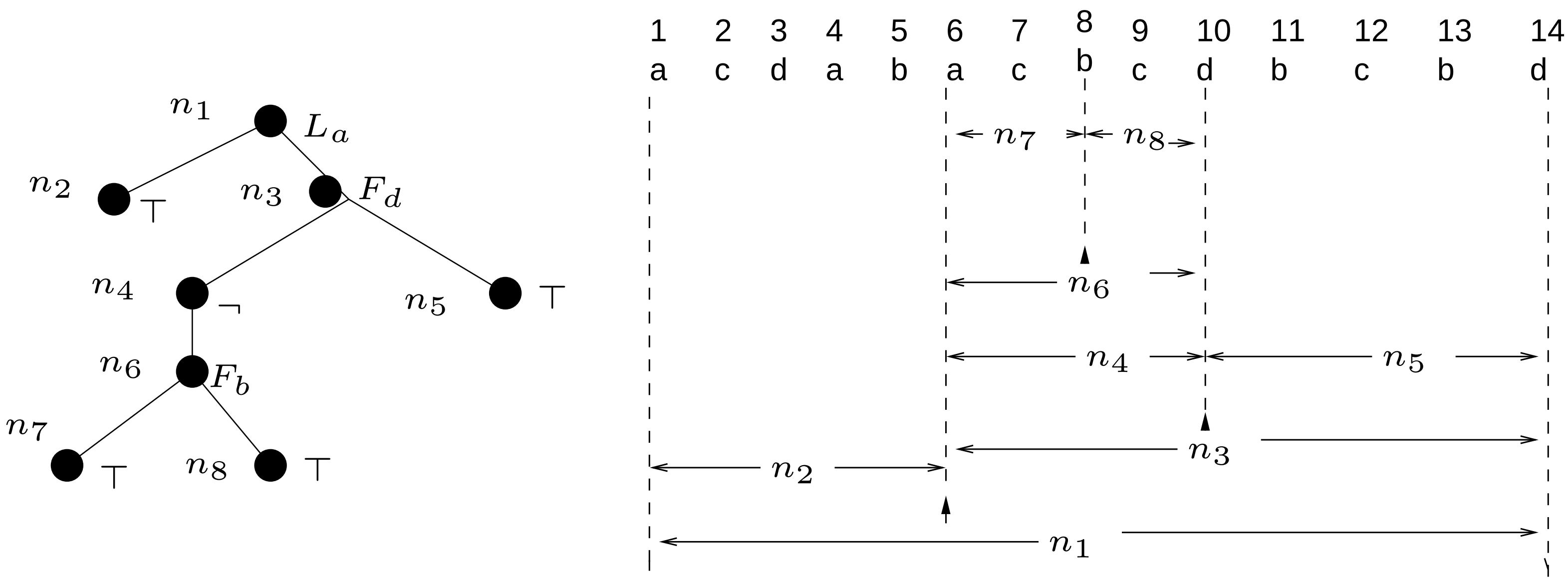
Formula $D \stackrel{\text{def}}{=} (\top L_a((\neg(\top F_b \top))F_d \top))$





Example

Formula $D \stackrel{\text{def}}{=} (\top L_a((\neg(\top F_b \top)) F_d \top))$



$w, [6, 14] \models n_4\ F_d\ n_5$ iff $w, [6, 10] \models n_4$ and $w, [10, 14] \models n_5$.

Unique parsability

Given a word w,

- Each node has unique associated interval (or none).

  $$Intv_w : Nodes \to INTV(w) \cup \{\mathbf{u}\}$$

  $Intv_w(n)$ depends only on the context of n.

- Each node of the form F_a or L_a has a unique “chopping” position (or none).

  $$cPos_w : MNodes \to INTV(w) \cup \{\mathbf{u}\}$$

- Truth of a node in its unique interval

  $$Eval_w : Nodes \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}.$$

Main results

- Unambiguous interval temporal logic (UITL)/ Unambiguous starfree expressions (USF)
- Unique parsability
- Membership in LogDCFL, emptiness in PSpace (NP over a fixed alphabet)
- Effective construction $A: \ UITL \to po2dfa$ such that $L(D) = L(A(D))$. The size $|A(D)| \ = \ \mathcal{O}(|D|^2)$.
- Effective construction $D: \ po2dfa \to UITL$ such that $L(A) = L(A(D))$. The size $|D(A)| \ = \ \mathcal{O}(2^{|A|})$.

Left and right endpoint scanners

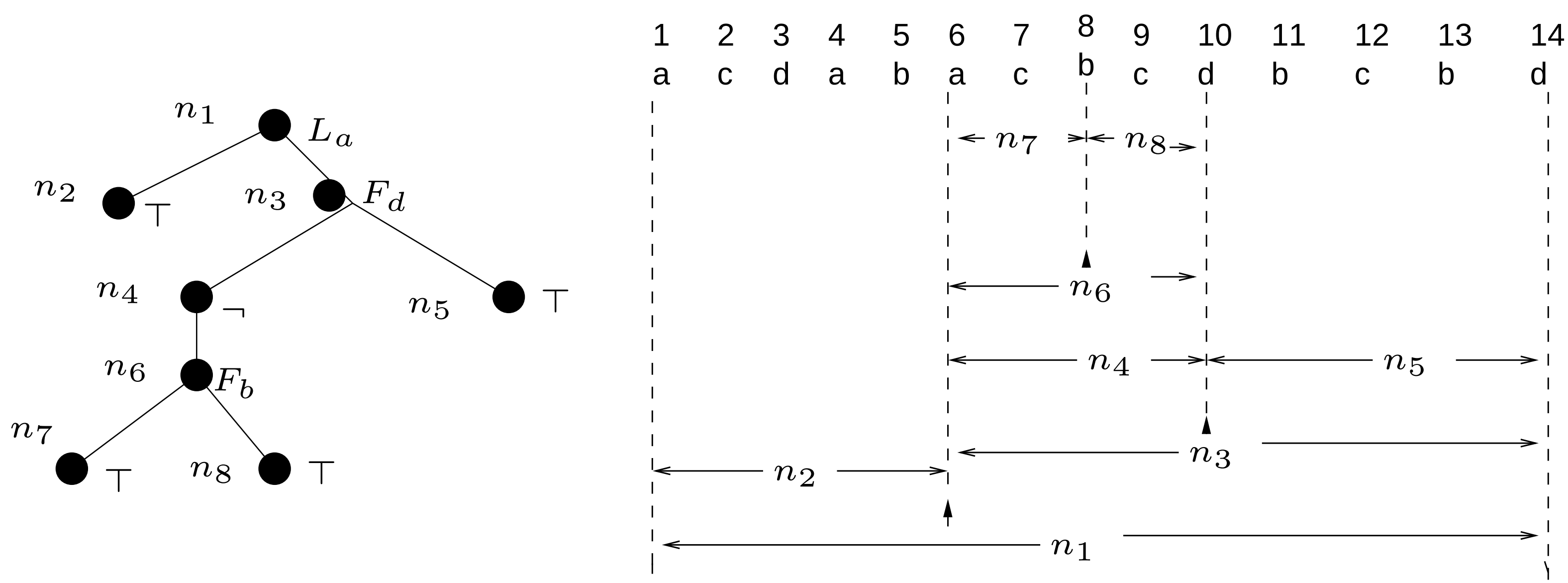
For each node n construct $po2dfa$

- $\mathcal{L}(n)$ which seeks and accepts at left endpoint of $Intv_w(n)$.
- $\mathcal{R}(n)$ which seeks and accepts at right endpoint of $Intv_w(n)$.
- Inductive construction based on depth of the node from $root$.

Lemma If $Intv_w(n) = [i, j]$ then $\mathcal{L}(n)(w, *) = (t, i)$ and $\mathcal{R}(n)(w, *) = (t, j)$.

Example endpoint scanners

Formula $D \stackrel{\text{def}}{=} (\top L_a((\neg(\top F_b \top)) F_d \top))$



- $\mathcal{L}(n_4) \stackrel{\text{def}}{=} \stackrel{\triangleleft}{\longrightarrow};\ \stackrel{a}{\longleftarrow}$
- $\mathcal{R}(n_4) \stackrel{\text{def}}{=} \mathcal{L}(n_4);\ \stackrel{d}{\longrightarrow}$.

Subformula automaton

Theorem For each node n we construct $po2dfa$ $\mathcal{M}(n)$ such that

- If $Eval_w(n) = t$ then $\mathcal{M}(n)(w, *) = (t, j)$ for some j.
- If $Eval_w(n) = f$ then $\mathcal{M}(n)(w, *) = (r, j)$ for some j.

Construction of $\mathcal{M}(n)$ By structural Induction.

- $\mathcal{M}(\top) \quad = \quad Acc$.
- $\mathcal{M}(\neg n) \quad = \quad \mathcal{M}(n)?Rej, Acc$.
- $\mathcal{M}(n_1 \vee n_2) \quad = \quad \mathcal{M}(n_1)?Acc, \mathcal{M}(n_2)$.

Automaton for F_a construct

Let $n = n_1 \; F_a \; n_2$. Then $\mathcal{M}(n) \;\; =$

$$\mathcal{L}(n); \xrightarrow{a};$$

Check if head is in $Intv_w(n)$;

$$\mathcal{M}(n_1); \;\; \mathcal{M}(n_2)$$

Main difficulty: How to check if current head position is in $Intv_w(n)$?

Proof: Carry the context around and use it.

Converse

- Effective construction $D: \ po2dfa \rightarrow UITL$ such that $L(A) = L(A(D))$. The size $|D(A)| \ = \ \mathcal{O}(2^{|A|})$.
- Proof idea: Describe the run of the $po2dfa$ in $UITL$.
- Proof: Carry the context around and use it. This is considerably more tedious than in the forward direction and yields a disjunction over many cases which in general gives an exponential upper bound for the size of the formula.
- Open: A logic $UITL^+$ which can polynomially (even linearly) encode $po2dfa$?

Summary

- A subclass of starfree expressions with the same expressive power as the unambiguous languages of Schützenberger (1976).
- Closing the piecewise testable languages with left- and right-deterministic operators has the power of unambiguous closure. Independently proved by Kufleitner and Weil (2008) using algebraic techniques.
- A "deterministic" logic admitting unique parsing of its models.
- A tractable fragment of ITL giving size $O(n^2)$ $po2dfa$ construction and polynomial time modelchecking.
- Expressively weak. To describe an automaton of n states requires an $O(2^n)$ length formula.

More open questions

Reg. Exp.	Membership	Emptiness	Interval TL	Modelchecking	Satisfiability
SF	PSpace	Nonelem.	ITL	PSpace	Nonelementary
USF	LogDCFL	PSpace	UITL	Ptime	PSpace

- An interval logic expressively equivalent to $FO^2[<, S]$? Its automata and expressiveness?
- There is an infinite quantifier alternation hierarchy in $FO[<]$, for which the only (not decidable) characterization known is in terms of semidirect products of monoids. It is known that $FO^2[<]$ corresponds to the second level (more precisely $\Delta_2[<]$). Where do we lose polynomial time?
- Can we tighten the NC^1 lower bound for membership/modelchecking (the same as for propositional logic)?

Acknowledgements

The authors thank Manfred Kufleitner, Meena Mahajan and Pascal Weil for suggesting improvements to this work.
For making this visit possible, the speaker thanks

- DST and RFBR
- The organizers of this workshop
- The commuters and train staff of the city of Birmingham (UK)

And thank you all for your patience!