Automata and Logics over Signals

Fabrice Chevalier[1] Deepak D'Souza[2] Raj Mohan[2]
Pavithra Prabhakar[3]

[1]LSV, Ecole Normale Superieure de Cachan, Cachan.

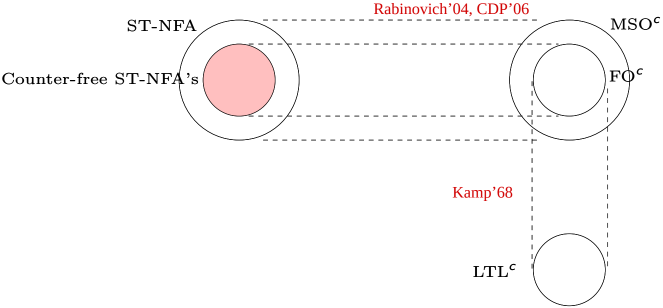
[2]Department of Computer Science & Automation
Indian Institute of Science, Bangalore.

[3]Department of Computer Science
University of Illinois at Urbana-Champaign.

29 January 2009

What this talk is about

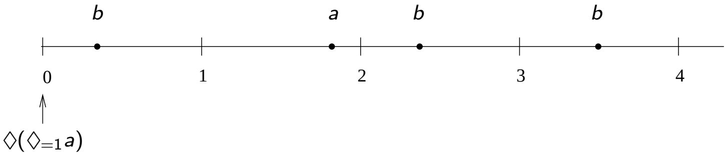
- Signals are piecewise-constant (or finitely-varying) functions.
- We consider natural formalisms over these models: automata, FO, MSO, LTL.
- Show connections between them, like for classical formalisms over words:



Some applications

- Signals are appropriate underlying models for continuous time logics (like words are for pointwise logics).
- Help to obtain following results for continuous time:
  - MSO logic characterisation for timed automata based on "input-determined" distance operators (eg. Event-recording automata).
  - Expressive completeness results for MTL/MITL (in general for logics with "input-determined" distance operators).
  - Counter-free automata characterisations for MTL/MITL.

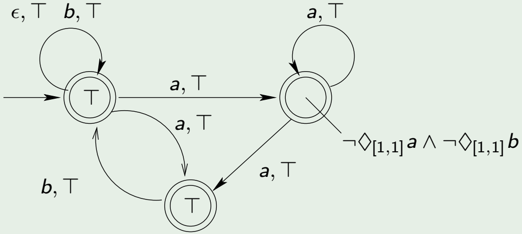
Pointwise vs continuous semantics



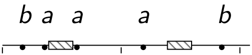
- In pointwise semantics:
  "There is a *action point* in the future from which an *a* occurs at a distance of 1 time unit". *False*
- In continous semantics:
  "There is a *time point* in the future from which an *a* occurs at a distance of 1 time unit" *True*
- Continuous semantics are typically more expressive [BCM05,DP06].

Example Eventual Timed Automaton (ETA) in continuous semantics

Continuous ETA

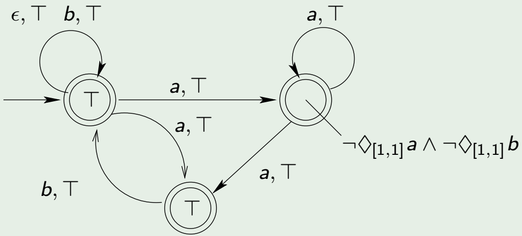


Accepts timed words in which there are “no insertions”

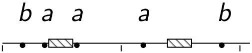


Example Eventual Timed Automaton (ETA) in continuous semantics

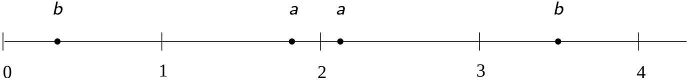
Continuous ETA



Accepts timed words in which there are "no insertions"

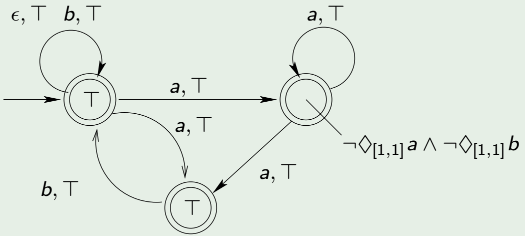


Accepts

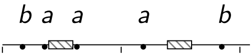


Example Eventual Timed Automaton (ETA) in continuous semantics

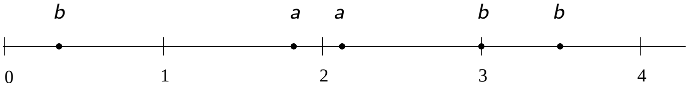
Continuous ETA



Accepts timed words in which there are “no insertions”



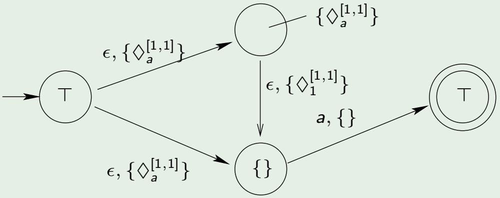
Rejects



Counter-free ETA's

- Characterise MTL^c-definable timed languages.
- Guards must be "proper" or "time-deterministic"
  - Specify exact set of guards to be satisfied.
- Automaton must be "fully canonical": no $g(\epsilon, g)g$ subwords possible.
- No counter in underlying graph.

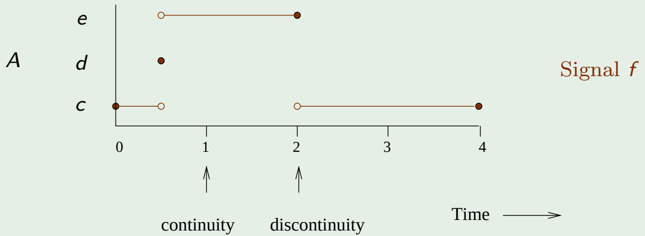
Counter-free ETA



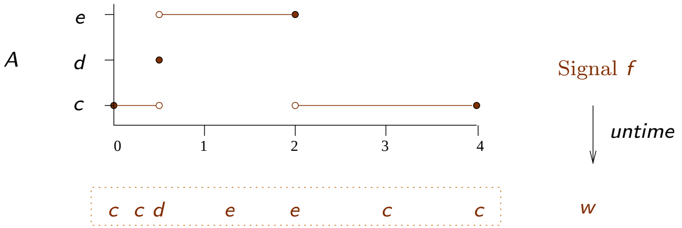
Finitely varying functions or Signals

- A *signal* over an alphabet A is a finitely varying function $f : [0, r] \rightarrow A$
- $t \in [0, r]$ is a point of continuity if there is $\epsilon > 0$ such that f is constant in $(t - \epsilon, t + \epsilon)$.
- finitely varying = finitely many discontinuities.

Example signal f

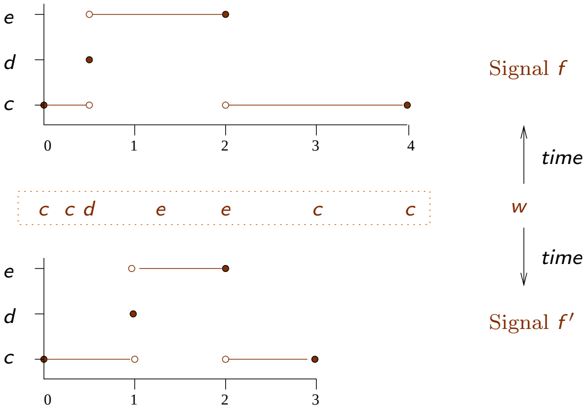


Untiming of a signal



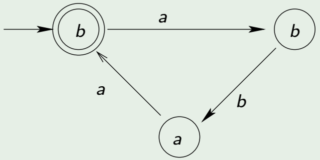
Such words are called canonical: elements of $A(AA)^*$ and no "aaa" at odd positions.

Timing a word to get signals

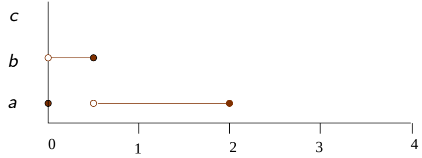


Automata accepting signals: ST-NFA's

ST-NFA $\mathcal{A}$

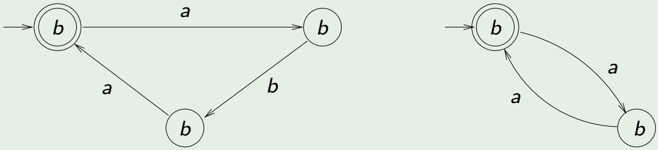


- State-and-Transition-labelled NFA's.
- Generates a classical "symbolic" language $L(\mathcal{A}) = \{abbaa, \ldots\}$.
- Generates a language of signals $S(\mathcal{A}) = timing(L(\mathcal{A}))$.



Canonical ST-NFA's

A non-canonical ST-NFA and its canonical version



- A canonical ST-NFA accepts only canonical words.
- Every ST-NFA can be converted to a signal language equivalent canonical one.

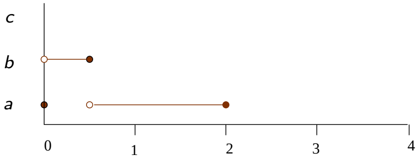
Logics over signals: FO^c

- Formulas of FO^c:

$$Q_a(x),\ x < y,\ \exists x(\varphi),\ \neg, \vee, \wedge.$$

- FO^c sentence describing point of continuity at x:

$$\exists y \exists z (y < x \wedge x < z \wedge \bigvee_{a \in \Sigma} \forall w (y < w < z \implies Q_a(w))).$$



More examples of FO^c sentences

- Subset W of domain of signal has a decreasing subsequence:

$$
\begin{aligned}
decseq(W) \;=\; & \exists l \exists a_0 (a_0 \in W \;\wedge l < a_0 \;\wedge \\
& \forall x((x \in W \wedge l < x) \implies \exists y(y \in W \wedge l < y < x)
\end{aligned}
$$

- Bounded subset W of domain of signal is infinite:

$$inf(W) = decseq(W) \vee incseq(W).$$

- Subset W of domain of signal is finitely-varying: Replace $x \in W$ by $\varphi_{disc}(x)$ in the formula

$$\neg inf(W).$$

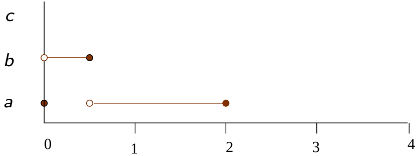
Logics over signals: MSO^c

- Formulas of MSO^c:

$$Q_a(x),\ x < y,\ \exists x \varphi,\ \exists X \varphi,\ \neg, \vee, \wedge.$$

- Second-order quantification ranges over *finitely-varying* subsets of domain $[0, r]$.
- MSO^c sentence describing existance of a dense subset with signal value a:

$$\exists X(\forall x(x \in X \implies Q_a(x)) \wedge \\ \forall x \forall y((x \in X \wedge y \in X \wedge x < y) \implies \exists z(z \in X \wedge x < z < y)))$$



Logics over signals: LTL^c

- Formulas of LTL^c:

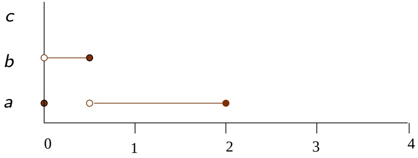
$$a,\ \theta U \theta,\ \theta S \theta,\ \neg, \wedge, \vee$$

- $\theta U \eta$ is *strict*: $\sigma, t \models \theta U \eta$ iff

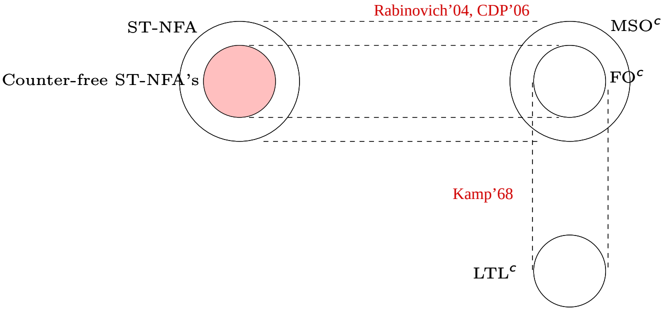
  $\exists t' :\ t < t' \leq dur(\sigma),\ \sigma, t' \models \eta,\ \text{and}\ \forall t'' :\ t < t'' < t', \sigma, t'' \models \theta.$

- Example: LTL^c formula describing points of continuity:

$$\bigvee_{a \in \Sigma} (a \wedge (aSa) \wedge (aUa)).$$



What we show


Rabinovich’04, CDP’06
ST-NFA
MSO^c
Counter-free ST-NFA’s
FO^c
Kamp’68
LTL^c

ST-NFA = MSO^c

From ST-NFA to MSO^c: The formula $\varphi_{\mathcal{A}}$ below describes when a signal is accepted by $\mathcal{A}$ ($e_1, \ldots, e_m$ are the transitions of $\mathcal{A}$):

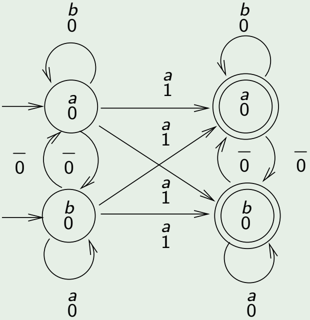
Sentence $\varphi_{\mathcal{A}}$

$$
\begin{aligned}
\exists X_1 \cdots \exists X_m \exists X (\forall x (\quad & (x \in X \iff \textstyle\bigvee_i x \in X_i) \wedge \\
& (\textstyle\bigwedge_{i \neq j} (x \in X_i \implies \neg x \in X_j)) \wedge \\
& (x \in X \iff \mathit{disc}(x)) \wedge \\
& (\mathit{first}(x) \implies \textstyle\bigvee_{i:\, p_i \in S} x \in X_i) \wedge \\
& (\mathit{last}(x) \implies \textstyle\bigvee_{i:\, q_i \in F} x \in X_i) \wedge \\
& (\textstyle\bigwedge_i (x \in X_i \implies (Q_{a_i}(x) \wedge ((\exists y(\mathit{consec}(x, y, X))) \implies \\
& \qquad \forall z((x < z \wedge z < y) \implies Q_{l(q_i)}(z))))))))).
\end{aligned}
$$

ST-NFA = MSO^c

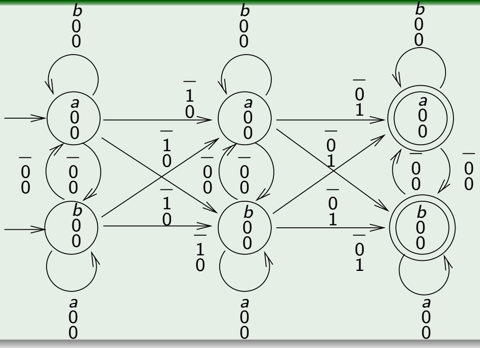
From MSO^c to ST-NFA: Inductively associated ST-NFA that accepts signals with interpretation built in.

For $Q_a(x)$

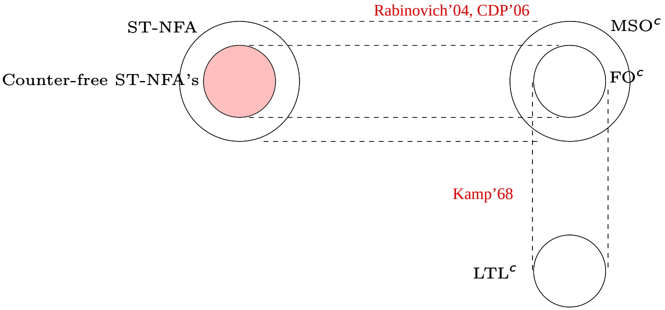


ST-NFA = MSO^c

For $x < y$

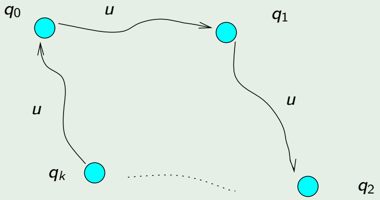


What we show

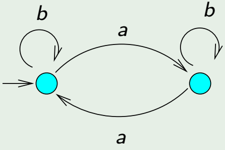

Rabinovich'04, CDP'06
ST-NFA
MSO^c
Counter-free ST-NFA's
FO^c
Kamp'68
LTL^c

Classical counter-free automata

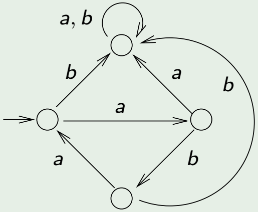
Counter in an automaton



Automaton with counter on "a"



Automaton without counter

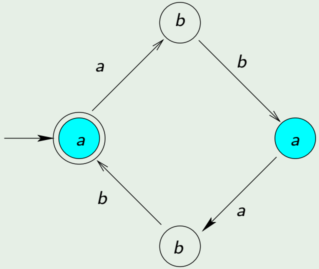


Counter-free ST-NFA's

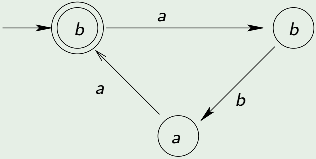
ST-NFA's that have

- No counter
- Are canonical

ST-NFA with counter



ST-NFA without counter

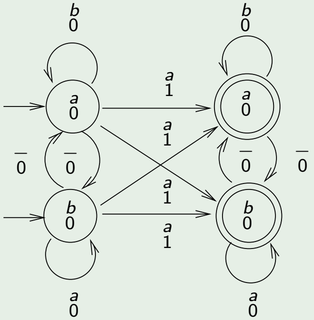


FO^c = counter-free ST-NFA

From FO^c to counter-free ST-NFA's:

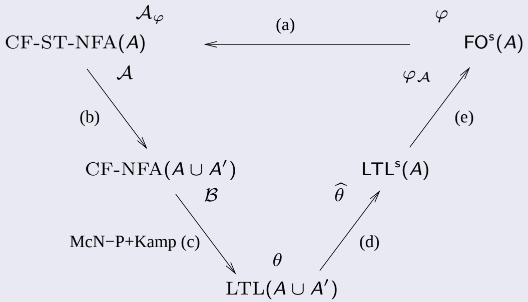
- Inductive construction associates a counter-free ST-NFA with open formulas.

For $Q_a(x)$



Counter-free ST-NFA's to FO

Route taken



Main step: LTL to LTL^c

For an LTL formula θ, construct an LTL^c formula $\widehat{\theta}$ which accepts timings of models of θ.
aUb is translated to:

$$\theta_{disc} \implies ((bUb) \;\vee\; (aU(\theta_{disc} \wedge b)) \;\vee\; (aU(\theta_{disc} \wedge a \wedge (bUb)))).$$

Summary of the talk

- Study natural formalisms over signals
- Automata-theoretic proof of decidability of MSO^c over reals.
- Proof of Kamp's theorem (LTL=FO) for signals using his result for words
- Counter-free automata characterisation of FO^c definable signal languages.
- Applications in expressive completeness + automata characterisation of real-time logics like MITL and MTL.