Temporal Logics Beyond Regularity

Martin Lange

Department of Computer Science
University of Århus

$\pi\lambda$ seminar, Sep. '06

Temporal Logics

for computer scientists most interesting because

- intuitive specification formalism
- possibility of automatic verification
- reasonable expressive power and complexity
- ...

research focused on very weak logics so far

trade-off between complexity and expressive power

but probably other reasons involved as well

In this talk

goal is a classification of highly expressive temporal logics w.r.t.

- relative expressive power
- complexity of model checking problem

less important (here):

- pragmatics, only addressed in few examples
- relation to automata, predicate logics, etc.

quite unimportant (here):

- SAT checking – because of undecidability
  see analogue in formal languages

Temporal Logics for Regular Properties

The Modal μ-Calculus

multi-modal logic + extremal fixpoint quantifiers

$$\varphi \quad ::= \quad q \mid X \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \langle a \rangle \varphi \mid [a]\varphi \mid \mu X.\varphi \mid \nu X.\varphi$$

interpreted over Kripke structures
$\mathcal{T} = (S, \{ \xrightarrow{a} \mid a \in A\}, L : S \to 2^P)$

semantics usually given as $[\![\varphi]\!]^{\mathcal{T}}_{\rho} \subseteq S$ with Knaster-Tarski

The Expressive Power of $\mathcal{L}_\mu$

Def.: $\mathcal{L}_\mu^k$ fragment with at most k fixpoint alternations

Bradfield'96, Lenzi'96

$$\mathcal{L}_\mu^0 \lneq \mathcal{L}_\mu^1 \lneq \mathcal{L}_\mu^2 \lneq \ldots \lneq \mathcal{L}_\mu$$

Kozen'83

$$PDL \lneq \mathcal{L}_\mu^0$$

Emerson/Halpern'86, Dam'92, ...

$$LTL \lneq CTL^* \lneq \mathcal{L}_\mu^1 \quad , \quad CTL \lneq \mathcal{L}_\mu^0$$

The Expressive Power of $\mathcal{L}_\mu$

Emerson,Jutla '91

Every $\mathcal{L}_\mu$-definable property is a regular tree language.

Janin,Walukiewicz '96

Every bisimulation-invariant regular tree language is $\mathcal{L}_\mu$-definable.

Examples of Non-Regular Properties

typical regular properties: safety, liveness, fairness, counting modulo fixed k, . . .

examples of non-regular, but nevertheless interesting properties:

- uniform inevitability,
  something holds on all paths at the same time
- unlimited counting like IO-buffer properties
- repetitions of unbounded sequences of actions
- . . .

Finite Structures and Non-Regular Properties

model checking non-regular properties on infinite structures mainly undecidable

regular properties suffice to describe (classes of) finite structures

but this might require

- exact knowledge of the size
- different specification formulas for different Kripke structures

Example

FIFO-buffer with k entries

$$L_k = \{w \in \{\mathtt{i}, \mathtt{o}\}^\omega \mid \forall u, v : w = uv \Rightarrow 0 \leq |u|_\mathtt{i} - |u|_\mathtt{o} \leq k\}$$

is regular for every k, but their limit

$$L = \{w \in \{\mathtt{i}, \mathtt{o}\}^\omega \mid \forall u, v : w = uv \Rightarrow |u|_\mathtt{o} \leq |u|_\mathtt{i}\}$$

not regular!

$\forall k \in \mathbb{N} : L_k \subseteq L$

using L only requires knowledge of boundedness, no knowledge about k

Temporal Logics for Non-Regular Properties

PDL of Context-Free Programs

by Harel, Pnueli, Stavi '83

PDL[CFG]

$$\varphi \ ::= \ q \mid \varphi \vee \varphi \mid \neg\varphi \mid \langle G \rangle \varphi$$

with G a context-free grammar

$$s \xrightarrow{a_1 \ldots a_n} t \quad \text{iff} \quad s \xrightarrow{a_1} \ldots \xrightarrow{a_n} t$$

$$s \xrightarrow{G} t \quad \text{iff} \quad \exists w \in L(G) \text{ with } s \xrightarrow{w} t$$

Examples of PDL[CFG] Formulas

- uniform inevitability impossible, not even $\mathtt{AF}q$
- $$[G]\mathtt{ff} \qquad \text{with} \quad G : \begin{array}{lcl} S & \rightarrow & b \mid aTb \\ T & \rightarrow & b \mid aTT \end{array}$$

Examples of PDL[CFG] Formulas

- uniform inevitability impossible, not even $\mathtt{AF}q$

- $$[G]\mathtt{ff} \qquad \text{with} \quad G: \begin{array}{lcl} S & \to & b \mid aTb \\ T & \to & b \mid aTT \end{array}$$

  no underflow of unlimited buffer, $a = \mathtt{in}$, $b = \mathtt{out}$

The Expressive Power of PDL[CFG]

PDL[CFG] and $\mathcal{L}_\mu$ are incomparable

- PDL[CFG] $\not\leq \mathcal{L}_\mu$: $\langle a^n b^n \rangle \mathtt{tt}$
- $\mathcal{L}_\mu \not\leq$ PDL[CFG]: $\mu X.q \vee [-]X$

but they trivially have a non-trivial common fragment: PDL

The Complexity of PDL[CFG]

Harel, Pnueli, Stavi '83

SAT checking PDL[CFG] is undecidable (Σ^1_1-complete)

result already holds for certain fixed CFGs

Koren, Pnueli '83

There are decidable non-regular fragments of PDL[CFG]

L. '05

Model Checking PDL[CFG] is P-complete

Next Logic . . .

Inflationary Fixpoints

idea taken from finite model theory: inflationary fixpoints instead of least

given a complete lattice M, a function $f : M \to M$

$$f^0 := \bot \qquad f^{\alpha+1} := f^\alpha \sqcup f(f^\alpha) \qquad f^\lambda := \bigsqcup_{\alpha<\lambda} f^\alpha$$

$\mathtt{ifp}\, f$ = value of this chain when stationary

slight problem: Békiè-Lemma only valid for monotonic f!

The Modal Iteration Calculus

by Dawar, Grädel, Kreutzer

MIC: extend $\mathcal{L}_\mu$ with inflationary fixpoints

$$\begin{aligned}\varphi &:= q \mid X \mid \varphi \vee \varphi \mid \neg\varphi \mid \langle a\rangle\varphi \mid \mathtt{ifp}\, X_i.\Phi \\ \Phi &:= (X_1 = \varphi_1, \ldots, X_n = \varphi_n)\end{aligned}$$

semantics as usual with inflationary fixpoint iteration and projection

Example

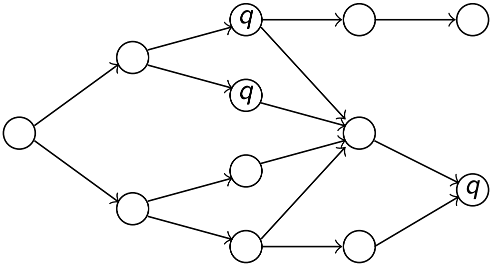
what does the following formula describe?

$$\mathtt{ifp}\, X. \left(\begin{array}{lcl} X & = & q \vee (\langle - \rangle tt \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{array} \right)$$

Example

what does the following formula describe?

$$\texttt{ifp}\,X.\begin{pmatrix} X & = & q \vee (\langle - \rangle tt \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



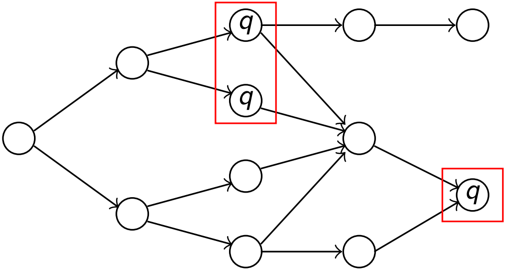
X

Y

Example

what does the following formula describe?

$$\texttt{ifp}\, X. \left(\begin{array}{lcl} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{array} \right)$$



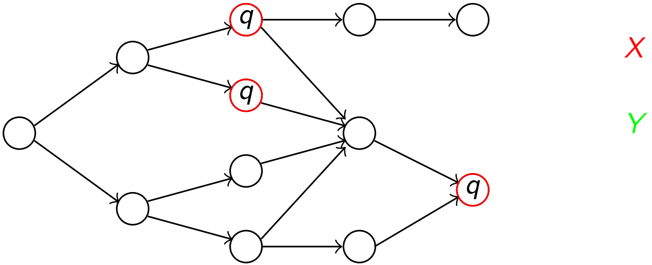
X

Y

Example

what does the following formula describe?

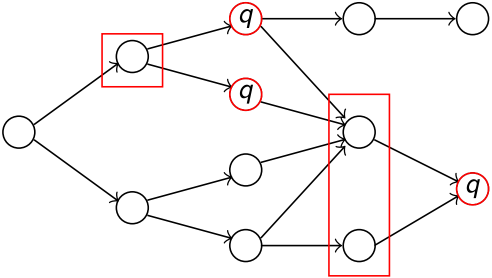
$$\texttt{ifp}\, X. \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



Example

what does the following formula describe?

$$\texttt{ifp}\, X . \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



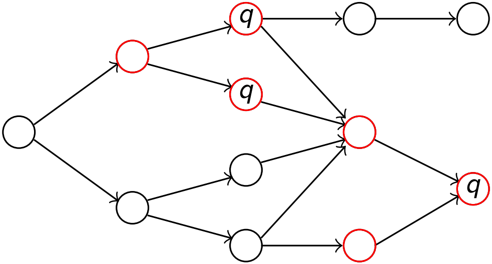
X

Y

Example

what does the following formula describe?

$$\texttt{ifp}\, X. \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



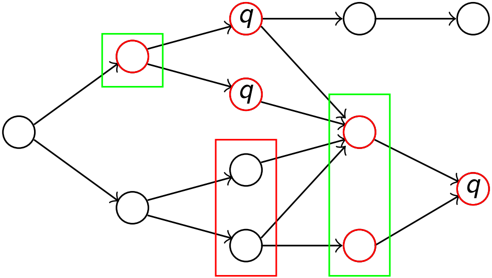
X

Y

Example

what does the following formula describe?

$$\mathtt{ifp}\,X.\begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



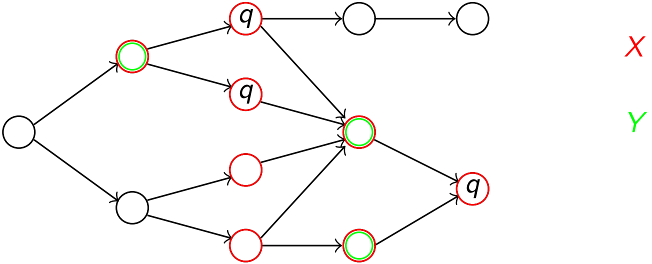
X

Y

Example

what does the following formula describe?

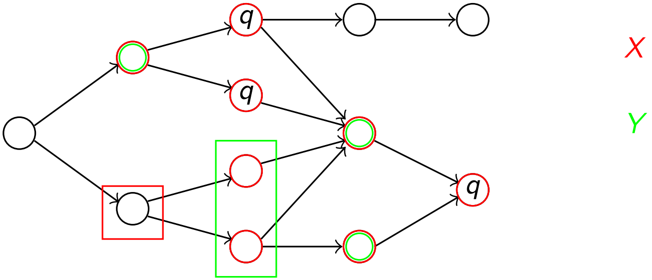
$$\texttt{ifp}\, X. \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



Example

what does the following formula describe?

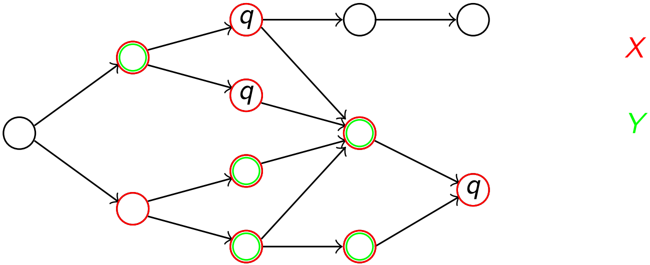
$$\texttt{ifp}\, X. \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



Example

what does the following formula describe?

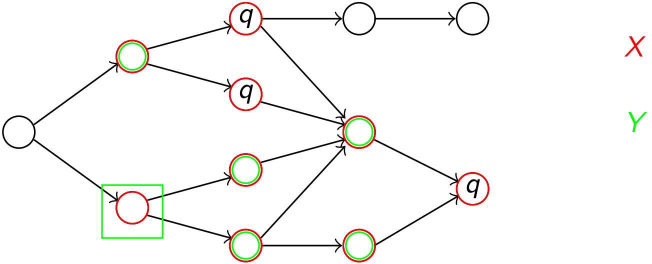
$$\mathtt{ifp}\, X. \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



Example

what does the following formula describe?

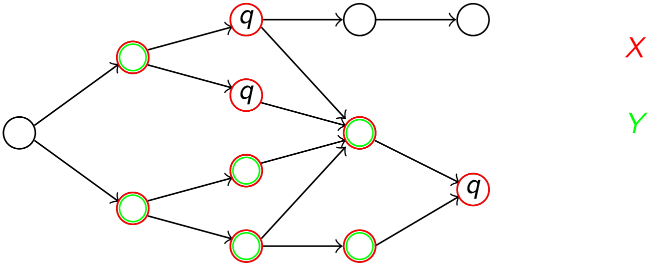
$$\texttt{ifp}\, X. \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$



Example

what does the following formula describe?

$$\texttt{ifp}\, X. \begin{pmatrix} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{pmatrix}$$

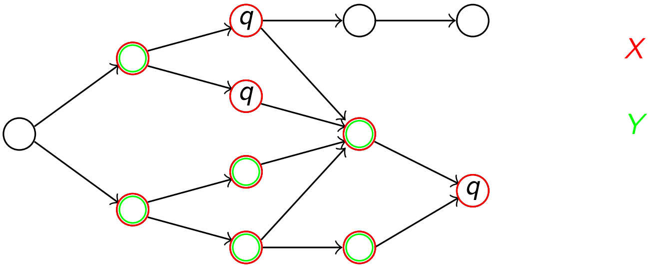


Example

what does the following formula describe?

$$\mathtt{ifp}\, X. \left(\begin{array}{lcl} X & = & q \vee (\langle - \rangle \mathtt{tt} \wedge [-](X \wedge \neg Y)) \\ Y & = & X \wedge \neg q \end{array} \right)$$

uniform inevitability



The Expressive Power of MIC

. . . w.r.t. context-free properties is not fully understood yet

Def.: 1MIC = MIC without simultaneous fixpoint inductions

Dawar, Grädel, Kreutzer '01

$\mathcal{L}_\mu \lneq$ *1MIC* $\lneq$ *MIC*

note: $\texttt{ifp}\, X.\varphi = \mu X.\varphi$ when $\varphi(X)$ is monotone

The Complexity of MIC

Dawar, Grädel, Kreutzer '01 / '04

- *Model checking MIC is PSPACE-complete*
- *for fixed formula in P*
- *SAT checking MIC is undecidable (not in arithmetic hierarchy)*
- *MIC has tree model property, no finite model property*

Next Logic . . .

The Intuition Behind Regularity for $\mathcal{L}_\mu$

recall syntax of $\mathcal{L}_\mu$

$$\varphi \ ::= \ q \mid X \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \langle a \rangle \varphi \mid [a]\varphi \mid \mu X.\varphi \mid \nu X.\varphi$$

formulas look like right-linear grammars!

variables $\approx$ non-terminals, $\langle a \rangle$,$[a]$ $\approx$ terminal symbols

to achieve non-regular effects: introduce sequential composition!

$\rightsquigarrow$ Fixpoint Logic with Chop (FLC) *Müller-Olm '99*

The Intuition Behind Regularity for $\mathcal{L}_\mu$

recall syntax of $\mathcal{L}_\mu$

$$\varphi \ ::= \ q \mid X \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \langle a \rangle \varphi \mid [a]\varphi \mid \mu X.\varphi \mid \nu X.\varphi$$

formulas look like right-linear grammars!

variables $\approx$ non-terminals, $\langle a \rangle$,$[a]$ $\approx$ terminal symbols

to achieve non-regular effects: introduce sequential composition!

$\rightsquigarrow$ Fixpoint Logic with Chop (FLC) *Müller-Olm '99*

The Intuition Behind Regularity for $\mathcal{L}_\mu$

recall syntax of $\mathcal{L}_\mu$

$$\varphi \quad ::= \quad q \mid X \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \langle a \rangle \mid [a] \mid \mu X.\varphi \mid \nu X.\varphi \mid \varphi;\varphi \mid \tau$$

formulas look like (alternating) context-free grammars!

variables $\approx$ non-terminals, $\langle a \rangle$,$[a]$ $\approx$ terminal symbols

to achieve non-regular effects: introduce sequential composition!

$\leadsto$ Fixpoint Logic with Chop (FLC) *Müller-Olm '99*

The Semantics of FLC

lift semantics of modal μ-calculus to space of monotone functions of type $2^S \to 2^S$, e.g.

$$\begin{aligned}
[\![q]\!] &:= \lambda_.\{s \mid q \in L(s)\} \\
[\![\varphi \vee \psi]\!] &:= \lambda T.[\![\varphi]\!](T) \cup [\![\psi]\!](T) \\
[\![\langle a \rangle]\!] &:= \lambda T.\{s \mid \exists t \in T \text{ with } s \xrightarrow{a} t\} \\
[\![\mu X.\varphi]\!] &:= \bigsqcap\{f \text{ monotone} \mid [\![\varphi]\!]_{[X \mapsto f]} \sqsubseteq f\} \\
[\![\varphi;\psi]\!] &:= \lambda T.[\![\varphi]\!]([\![\psi]\!](T)) \\
[\![\tau]\!] &:= \lambda T.T
\end{aligned}$$

define $s \models \varphi$ iff $s \in [\![\varphi]\!](S)$

Example

$$\nu X.[b];\mathtt{ff} \wedge [a];(\nu Y.[b] \wedge [a];Y;Y);X$$

Example

$$\nu X.[b];\mathtt{ff} \wedge [a];(\nu Y.[b] \wedge [a];Y;Y);X$$

on all paths never more b's than a's

The Complexity of FLC

Müller-Olm '99

- *SAT for FLC is undecidable*
- *Model Checking is decidable*
- *FLC has tree model property*
- *no finite model property*

L./Stirling '02, L. '06

Model Checking FLC is EXPTIME-complete, even for fixed alternation-free formula

The Expressive Power of FLC

Cor. FLC $\not\leq$ MIC

...because data complexities are EXPTIME-hard, resp. in P

Def.: FLC^k fragment of at most k fixpoint alternations

L. '06

$$FLC^0 \lneq FLC^1 \lneq \ldots \lneq FLC$$

L., Somla '06

$$PDL[CFG] \lneq FLC^0$$

Next Logic . . .

Predicate Transformers

semantics of μ-calculus: predicate, i.e. $[\![\varphi]\!] : 2^S$

semantics of FLC: predicate transformer, i.e. $[\![\varphi]\!] : 2^S \to 2^S$

predicate transformer = first-order function

why not higher order?

$\rightsquigarrow$ Higher-Order Fixpoint Logic (HFL) Viswanathan^2 '04

Predicate Transformers

semantics of μ-calculus: predicate, i.e. $[\![\varphi]\!] : 2^S$

semantics of FLC: predicate transformer, i.e. $[\![\varphi]\!] : 2^S \to 2^S$

predicate transformer = first-order function

why not higher order?

$\leadsto$ Higher-Order Fixpoint Logic (HFL) Viswanathan^2 '04

Predicate Transformers

semantics of μ-calculus: predicate, i.e. $[\![\varphi]\!] : 2^S$

semantics of FLC: predicate transformer, i.e. $[\![\varphi]\!] : 2^S \to 2^S$

predicate transformer = first-order function

why not higher order?

$\rightsquigarrow$ Higher-Order Fixpoint Logic (HFL) Viswanathan[2] '04

Higher-Order Fixpoint Logic

syntax:

$$
\begin{array}{rcl}
\varphi & ::= & q \mid X \mid \varphi \vee \varphi \mid \neg\varphi \mid \langle a \rangle \varphi \mid \lambda(X^v : \tau)\,\varphi \mid \varphi\,\varphi \mid \mu(X : \tau)\,\varphi \\
v & ::= & + \mid - \mid ? \\
\tau & ::= & \mathcal{P} \mid \tau \to \tau
\end{array}
$$

typing rules guarantee well-formedness

semantics as elements of function spaces

Example

$$\neg\Big(\bigvee_{a\neq b}\big(\mu F.\lambda X.\lambda Y.(X\wedge Y)\vee(F\ \langle -\rangle X\ \langle -\rangle Y)\big)\ \langle a\rangle\mathtt{tt}\ \langle b\rangle\mathtt{tt}\Big)$$

Example

$$\neg\Big(\bigvee_{a \neq b} (\mu F.\lambda X.\lambda Y.(X \wedge Y) \vee (F\ \langle - \rangle X\ \langle - \rangle Y))\ \langle a \rangle \mathtt{tt}\ \langle b \rangle \mathtt{tt} \Big)$$

bisimilarity to a word model

conjecture: this is not FLC-definable
fact: it is also MIC-definable

The Expressive Power of HFL

Def.: HFL^k fragment restricted to functions of order k only

note: $\mathrm{HFL}^0 = \mathcal{L}_\mu$

Viswanathan, Viswanathan '04

- *FLC* $\leq$ *HFL*^{1}
- *satisfiability is undecidable*
- *HFL has the tree model property*

The Expressive Power of HFL

Thm.: MIC $\lneq$ HFL^1

Proof sketch:
strictness because FLC $\leq$ HFL^1 and FLC $\not\leq$ MIC

for inclusion:

$$\mathtt{ifp}\, X.\varphi(X) \;\equiv\; \Big(\mu F.\lambda X.X \vee \big(F\,(X \vee \varphi(X))\big)\Big)\; \mathtt{ff}$$

note that F is monotone, Békìc principle applies for simultaneous fixpoint inductions

The Expressive Power of HFL

Thm.: MIC $\lneq$ HFL^1

Proof sketch:
strictness because FLC $\leq$ HFL^1 and FLC $\not\leq$ MIC

for inclusion:

$$\mathtt{ifp}\, X.\varphi(X) \;\equiv\; \Big(\mu F.\lambda X.X \vee (F\ (X \vee \varphi(X)))\Big)\ \mathrm{ff}$$

note that F is monotone, Békìc principle applies for simultaneous fixpoint inductions

The Complexity of HFL

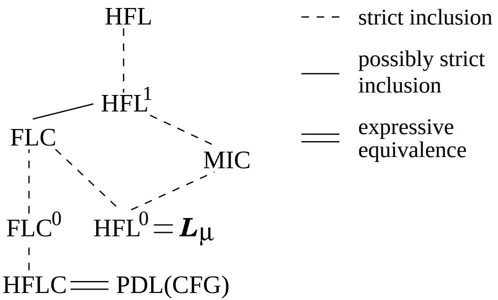
L.,Somla '05; Axelsson, L., Somla '0x

- *HFL^k model checking is k-ExpTime-complete for every $k \geq 1$*
- *already true for data complexity*
- *model checking 1-state structures is non-elementary*

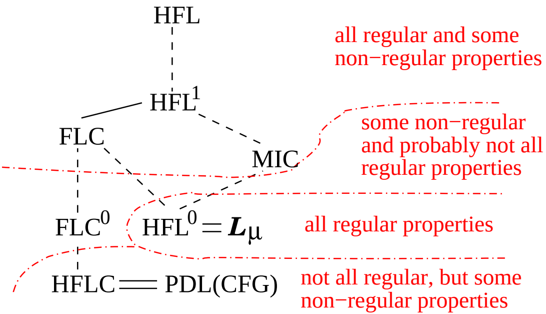
Cor.: $\mathrm{HFL}^1 \lneq \mathrm{HFL}^2 \lneq \ldots \lneq \mathrm{HFL}$

Overview

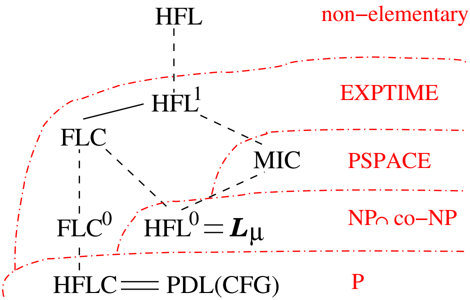
Temporal Logics Beyond Regularity



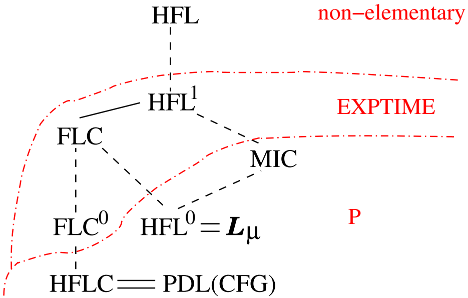
Non-Regular Properties



The Combined Model Checking Complexities



The Fixed Formula Model Checking Complexities



Further Work

- Is $\mathrm{FLC} = \mathrm{HFL}^1$, or is $\mathrm{MIC} \not\leq \mathrm{FLC}$?
- Is there an FLC^k that captures all regular properties?
- What about PDL[ACFG], PDL[CSG], ...?
- etc.