Formal Language Constrained Reachability and Model Checking Propositional Dynamic Logics

Martin Lange

School of Electr. Eng. and Comp. Sc., University of Kassel, Germany

29/09/2011

5th Workshop on Reachability Problems

joint work with Roland Axelsson

Reachability Problems

reachability problems have two parameters:

- structures: *where to find connection*
  finite graphs, pushdown graphs, Petri nets, . . .
- objectives: *what kind of connection*
  classes of formal languages

	finite	**infinite**
regular	easy	lots of work
non-regular	here	quickly undecidable

Motivation

Question

Given a finite, directed, Σ-edge-labeled graph $G = (V, \rightarrow)$ and $s, t \in V$.

Is it possible to decide effectively whether there is a path from s to t of the form

- $a^n b^n c^n$ for some $n \in \mathbb{N}$?
- ww for some $w \in \Sigma^*$?

Is it possible efficiently?

Outline

1. definition of three decision problems from
    - reachability theory
    - formal language theory
    - model checking
2. interreducibility
3. consequences

Formal Language Constrained Reachability

finite, directed graphs with edge labels from finite set Σ:
$G = (V, \rightarrow)$ with $\rightarrow \subseteq V \times \Sigma \times V$

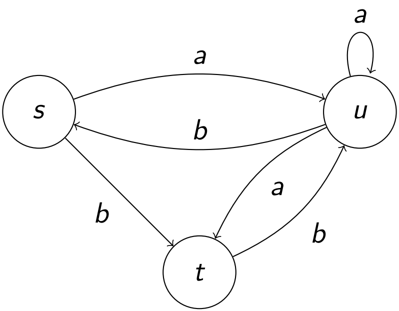
edge relation extends to words $w \in \Sigma^*$ inductively:

$$s \xrightarrow{\epsilon} t \quad \text{iff} \quad s = t$$
$$s \xrightarrow{aw} t \quad \text{iff} \quad \exists u. s \xrightarrow{a} u \wedge u \xrightarrow{w} t$$

Definition 1 (Formal Language Constrained Reachability)

Given $G = (V, \rightarrow)$, $s \in V$, $T \subseteq V$, and $L \subseteq \Sigma^*$, decide whether or not there is $t \in T$ and $w \in L$ s.t. $s \xrightarrow{w} t$.

Example



is $\{t\}$ reachable from s via $\{a^n b^n \mid n \in \mathbb{N}\}$?

yes, e.g. via $s \xrightarrow{a} u \xrightarrow{a} u \xrightarrow{a} t \xrightarrow{b} u \xrightarrow{b} s \xrightarrow{b} t$

Regular Intersection

Definition 2 (Regular Intersection)

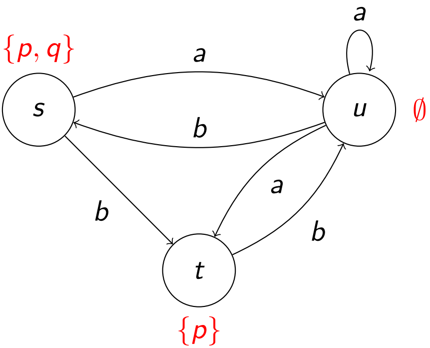
Given a formal language $L \subseteq \Sigma^*$ and an NFA for a regular language $R \subseteq \Sigma^*$ decide whether or not $L \cap R \neq \emptyset$

Remark

Let $\mathcal{C} \subseteq 2^{\Sigma^*}$. If $\mathcal{C}$ closed under intersections with regular languages and has decidable emptiness problem then regular intersection is decidable too.

Model Checking PDL

PDL = modal logic interpreted over directed, edge- (Σ) and node-labeled ($2^{\mathcal{P}}$) graphs with accessibility relations closed under compositions and including tests



Syntax and Semantics

syntax defines formulas and programs inductively:

- $\mathcal{P} \subseteq \textsc{Form}$
- $\varphi, \psi \in \textsc{Form} \Longrightarrow \varphi \vee \psi, \neg\varphi \in \textsc{Form}$
- $\varphi \in \textsc{Form}$ and $L \subseteq \textsc{Prog}^* \Longrightarrow \langle L \rangle \varphi \in \textsc{Form}$
- $\Sigma \subseteq \textsc{Prog}$
- $\varphi \in \textsc{Form} \Longrightarrow \varphi? \in \textsc{Prog}$

semantics:

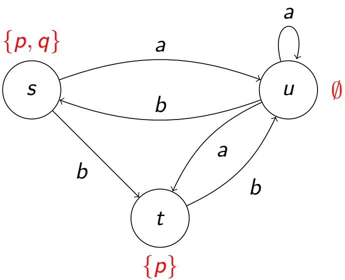
$$\vdots$$

$$s \xrightarrow{\varphi?} t \quad \text{iff} \quad s = t \text{ and } s \models \varphi$$

$$G, s \models \langle L \rangle \varphi \quad \text{iff} \quad \exists t. \exists w \in L. s \xrightarrow{w} t \text{ and } t \models \varphi$$

$$\vdots$$

Example



$G, s \models \langle \{a\,(\langle bb\rangle \neg q)?\,a\}\rangle p$

Definition 3 (Model Checking PDL)

Given $G = (V, \to, \lambda)$, $s \in V$ and $\varphi \in \textsc{Form}$, decide whether or not $s \models \varphi$ holds.

Parametrisation

goal: determine complexity and decidability of these three problems

answers clearly depend on classes of languages being used

formally consider problems parametrised by class $\mathcal{C}$ of formal languages

- $\textsc{Reach}[\mathcal{C}]$: reachability problem for finite digraphs and objectives $\mathcal{C}$
- $\textsc{RegISect}[\mathcal{C}]$: regular intersection problem for $\mathcal{C}$
- $\text{MC-PDL}[\mathcal{C}]$: model checking for PDL over languages from $\mathcal{C}$

Reductions

Theorem 4

a) $\textsc{Reach}[\mathcal{C}] \equiv_{\mathrm{lin}} \textsc{RegISect}[\mathcal{C}]$

b) $\textsc{Reach}[\mathcal{C}] \leq_{\mathrm{lin}} \mathrm{MC\text{-}PDL}[\mathcal{C}]$

c) $\mathrm{MC\text{-}PDL}[\mathcal{C}] \leq^{\mathrm{Turing}}_{\mathcal{O}(n^2)} \textsc{Reach}[\mathcal{C}]$

proof quite simple

benefit: transfers results from formal language theory to reachability and model checking

Reductions

(a) Reach[$\mathcal{C}$] $\leq$ RegISect[$\mathcal{C}$]

given $G = (V, \to)$, s, T and $L \in \mathcal{C}$, take NFA $\mathcal{A} = (V, s, \to, T)$

$s \xrightarrow{w} t$ for some $w \in L \iff L \cap L(\mathcal{A}) \neq \emptyset$

RegISect[$\mathcal{C}$] $\leq$ Reach[$\mathcal{C}$]

analogously

(b) Reach[$\mathcal{C}$] $\leq$ MC-PDL[$\mathcal{C}$]

given $G = (V, \to)$, s, T and $L \in \mathcal{C}$, take $G' = (V, \to, \lambda)$ with $q_T \in \lambda(t)$ iff $t \in T$

$s \xrightarrow{w} t$ for some $w \in L \iff G', s \models \langle L \rangle q_T$

Reductions

(c) MC-PDL[$\mathcal{C}$] $\leq^{\text{Turing}}$ Reach[$\mathcal{C}$]

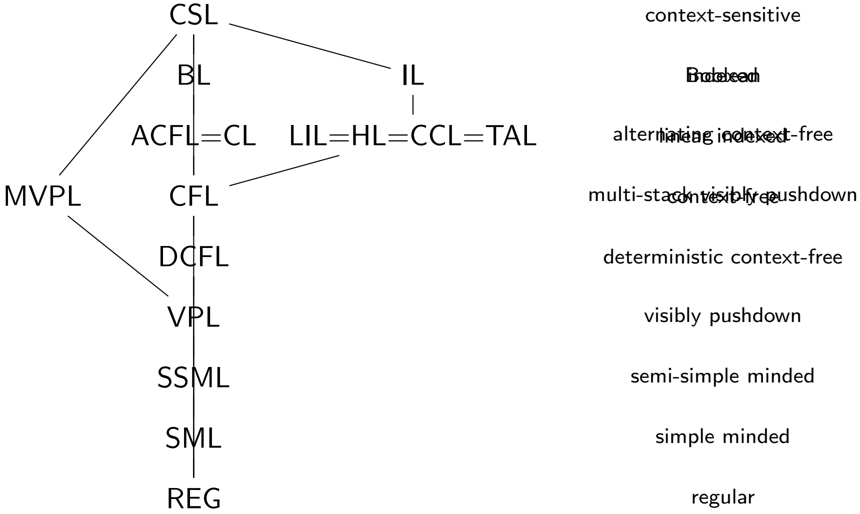
model checking algorithm for PDL[$\mathcal{C}$] with oracle for $\textsc{Reach}[\mathcal{C}]$

$\texttt{MC}(\varphi, G = (V, \rightarrow, \lambda)) =$

$\quad$ `case` φ `of`

q	:	`return` $\{v \mid q \in \lambda(v)\}$
$\psi_1 \vee \psi_2$	:	`return` $\texttt{MC}(\psi_1, G) \cup \texttt{MC}(\psi_2, G)$
$\neg\psi$	:	`return` $V \setminus \texttt{MC}(\psi, G)$
$\langle L \rangle \psi$	:	let Φ be top-level test formulas used in L
		$\rightarrow' := \{(v, \vartheta?, v) \mid \vartheta \in \Phi, v \in \texttt{MC}(\vartheta, G)\ \}$
		$G' := (V, \rightarrow \cup \rightarrow', \lambda)$
		$T := \texttt{MC}(\psi, G')$
		`return` $\{v \in V \mid (v, T) \in \texttt{Reach}(L)\ \}$

Classes of Formal Languages



The Picture Now

language class $\mathcal{C}$	$\textsc{RegISect}[\mathcal{C}]$	$\textsc{Reach}[\mathcal{C}]$	MC-PDL$[\mathcal{C}]$
ACFL,CL,BL,**CSL**	undec. [Landweber'63]	undec. [Barrett et al.'00]	undec.
MVPL	2EXPTIME [LaTorre et al.'07, Atig et al.'08]	2EXPTIME	
IL	EXPTIME [Aho'68, Tanaka/Kasai'07]	EXPTIME	
LIL,HL,CCL,TAL	PTIME [Gazdar'88],↓	PTIME	
DCFL, **CFL**	PTIME [Bar-Hillel et al.'61],↓	PTIME [Barrett et al.'00],↓	PTIME [Lange'05],↓
SML, SSML, VPL	PTIME ↑,[Lange'11]		
REG	NLOGSPACE [Hunt'73]		PTIME [Fischer/Ladner'79, folk.]

Re-Consider Introductory Questions

Question

Given a finite, directed, Σ-edge-labeled graph $G = (V, \to)$ and $s, t \in V$.

Is it possible to decide effectively whether there is a path from s to t of the form

- $a^n b^n c^n$ for some $n \in \mathbb{N}$? **yes**$\in$ PTIME
- ww for some $w \in \Sigma^*$? **yes**$\in$ EXPTIME

Is it possible efficiently?

$\{ww \mid w \in \Sigma^*\}$ is an indexed language (IL)

$\{a^n b^n c^n \mid n \in \mathbb{N}\}$ is a linear indexed language (LIL)