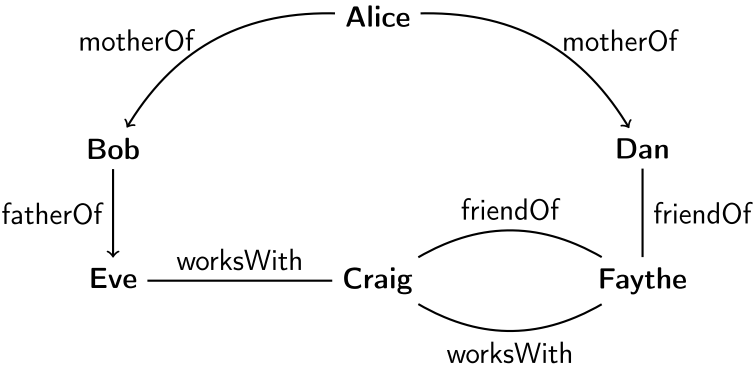
Relative Expressive Power of Downward Fragments of Navigational Query Languages on Trees and Chains

Jelle Hellings
Hasselt University

Joint work with Marc Gyssens, Yuqing Wu, Dirk Van Gucht, Jan Van den Bussche, Stijn Vansummeren, and George H. L. Fletcher

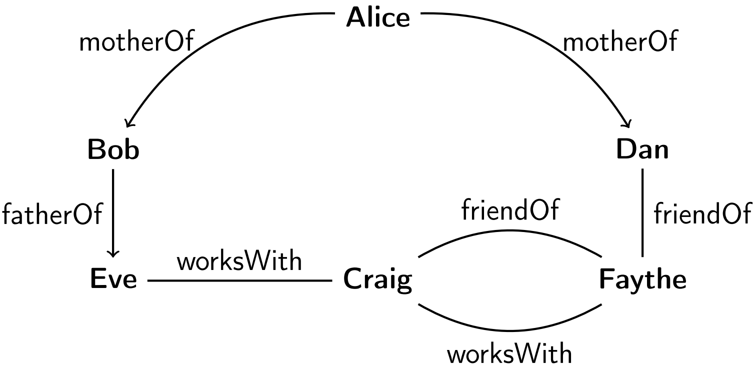
Overview

Querying graphs



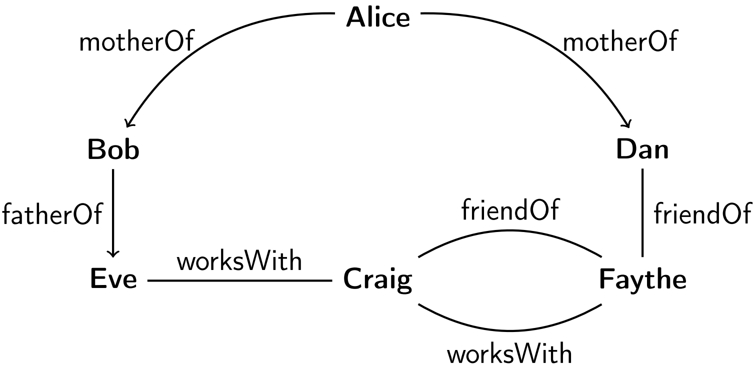
Querying graphs

$$\pi_1[\text{motherOf}] \circ [\text{motherOf} \cup \text{fatherOf}]^+$$



Querying graphs

$$\overline{\pi}_1[\mathsf{friendOf} \setminus \mathsf{worksWith}]$$



Navigational Expressions

$$e := \emptyset \mid \mathrm{id} \mid \ell \text{ (for } \ell \text{ an edge-label)} \mid e \circ e \mid e \cup e \mid$$
$$[e]^+ \mid \pi_1[e] \mid \pi_2[e] \mid \overline{\pi}_1[e] \mid \overline{\pi}_2[e] \mid e \cap e \mid e \setminus e$$

Question

Are all these operations necessary?
How does each operator influence expressive power?

Preliminary answer

We can use basic rewriting:

$$e_1 \cap e_2 = e_1 \setminus (e_1 \setminus e_2)$$
$$\pi_i[e] = \overline{\pi}_i[\overline{\pi}_i[e]]$$
$$\overline{\pi}_i[e] = \mathrm{id} \setminus \pi_i[e]$$

Problem statement

We start with $\{\emptyset, \mathrm{id}, \cup, \circ\}$ and edge-labels

- Add any subset $\mathfrak{F}$ of $\{\pi, \overline{\pi}, +, \cap, \setminus\}$,
  We denote the resulting query language by $\mathcal{N}(\mathfrak{F})$
- Compare the expressive power of resulting languages
- Graphs: already fully studied by Fletcher et al.
- Trees: a few results are known (XML)

Definition

Let $\mathfrak{F} \subseteq \{+, \pi, \overline{\pi}, \cap, \setminus\}$.
$\underline{\mathfrak{F}}$ is the superset of $\mathfrak{F}$ obtained by "basic rewriting".

Example

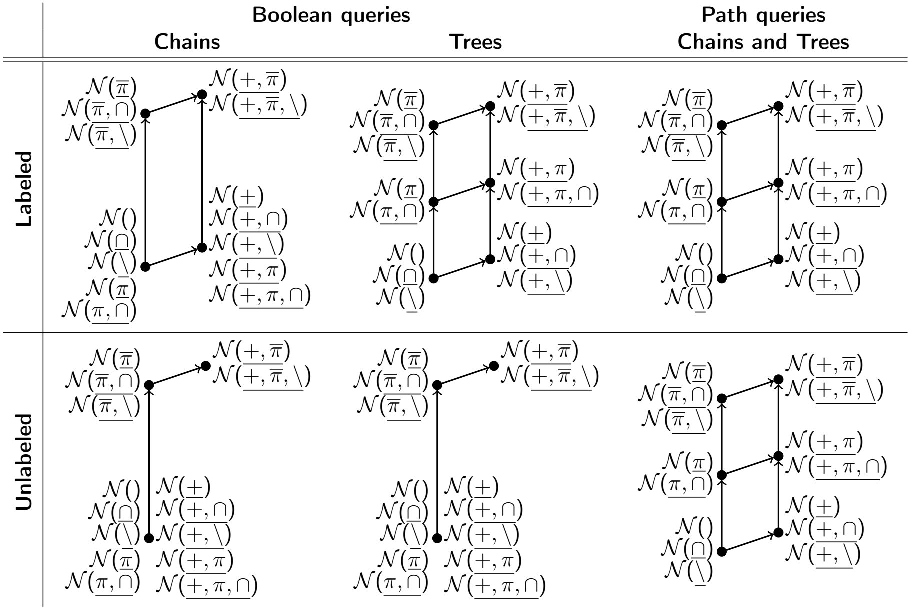
$\underline{\{\pi, \setminus\}} = \{\pi, \overline{\pi}, \cap, \setminus\}$

Overview

Results on trees and chains: the main results

- On labeled trees, we can do without $\cap$ and $\setminus$:
  we have $\mathcal{N}(\mathfrak{F}) \preceq_p \mathcal{N}(\underline{\mathfrak{F}} \setminus \{\cap, \setminus\})$
- For boolean queries:
  - On unlabeled trees, only $\overline{\pi}$ adds expressive power:
    we have $\mathcal{N}(+, \pi, \cap) \preceq_b \mathcal{N}()$
  - On labeled chains, we can do without π:
    we have $\mathcal{N}(+, \pi) \preceq_b \mathcal{N}(\mathfrak{F} \setminus \{\pi\})$

Results on trees and chains



Overview

The boolean collapse to $\mathcal{N}()$ on unlabeled trees

Theorem

Let $\mathfrak{F} \subseteq \{+, \pi, \cap\}$. On unlabeled trees we have $\mathcal{N}(\mathfrak{F}) \preceq_b \mathcal{N}()$.

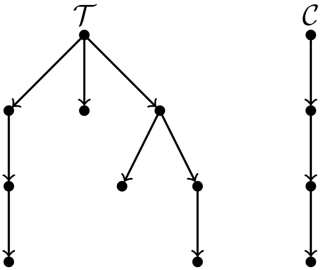
Definition (homomorphism)

A mapping $h : \mathbf{N}_1 \to \mathbf{N}_2$ is a *homomorphism* from $\mathcal{G}_1 = (\mathbf{N}_1, \mathcal{E}_1)$ to $\mathcal{G}_2 = (\mathbf{N}_2, \mathcal{E}_2)$ if $(m, n) \in \mathcal{E}_1$ implies $(h(m), h(n)) \in \mathcal{E}_2$.

Proposition

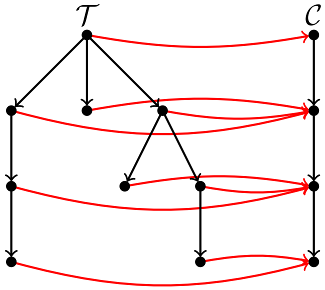
The language $\mathcal{N}(+, \pi, \cap)$ is closed under homomorphisms: if there is a homomorphism h from $\mathcal{G}_1$ to $\mathcal{G}_2$, then $h(e(\mathcal{G}_1)) \subseteq e(\mathcal{G}_2)$.

Proof: $\mathcal{N}(+, \pi, \cap)$ cannot distinguish trees from chains



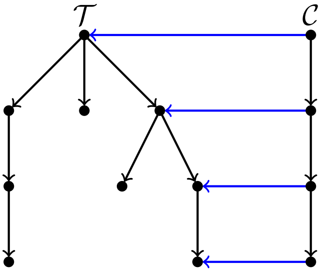
Proof: $\mathcal{N}(+, \pi, \cap)$ cannot distinguish trees from chains

- Provide homomorphism from tree $\mathcal{T}$ to chain $\mathcal{C}$



Proof: $\mathcal{N}(+, \pi, \cap)$ cannot distinguish trees from chains

- Provide homomorphism from tree $\mathcal{T}$ to chain $\mathcal{C}$
- Provide homomorphism from chain $\mathcal{C}$ to tree $\mathcal{T}$



Proof: $\mathcal{N}(+, \pi, \cap)$ can only query on depth

Conclusion

Even $\mathcal{N}(+, \pi, \cap)$ can only express queries of the form:

The height of the tree is at least k $\left(= \ell^k = \underbrace{\ell \circ \ldots \circ \ell}_{k \text{ terms}}\right)$

Theorem

Let $\mathfrak{F} \subseteq \{+, \pi, \cap\}$. On unlabeled trees we have $\mathcal{N}(\mathfrak{F}) \preceq_b \mathcal{N}()$.

Overview

Removing $\cap$ and $\setminus$ from simple queries

Example

$$\left[\ell^3\right]^+ \cap \left[\ell^7\right]^+ = \left[\ell^{21}\right]^+$$

$$\left[\ell^3\right]^+ \setminus \left[\ell^7\right]^+ = \left(\ell^3 \cup \ell^6 \cup \ell^9 \cup \ell^{12} \cup \ell^{15} \cup \ell^{18}\right) \circ \left(\left[\ell^{21}\right]^+ \cup \mathrm{id}\right)$$

Basic observations

- Expressions in $\mathcal{N}(+)$ are regular path queries
- Regular languages (expressions) are closed under $\cap$ and $\setminus$

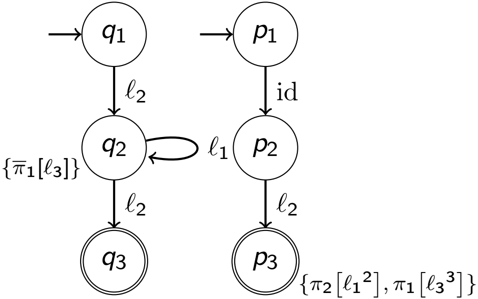
Question

How to generalize to π and $\overline{\pi}$?

Condition automata

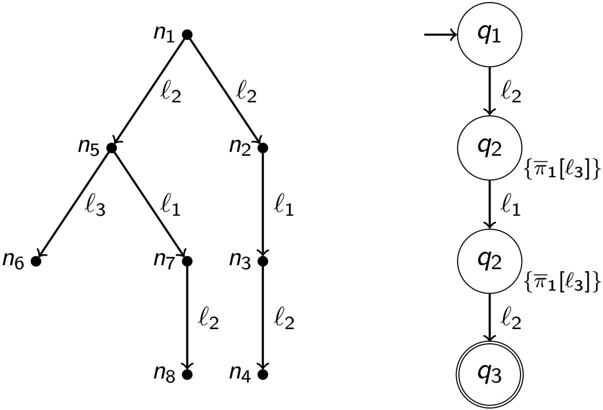
Definition (condition automaton)

A *condition automaton* is a 7-tuple $\mathcal{A} = (S, \Sigma, C, I, F, \delta, \gamma)$.



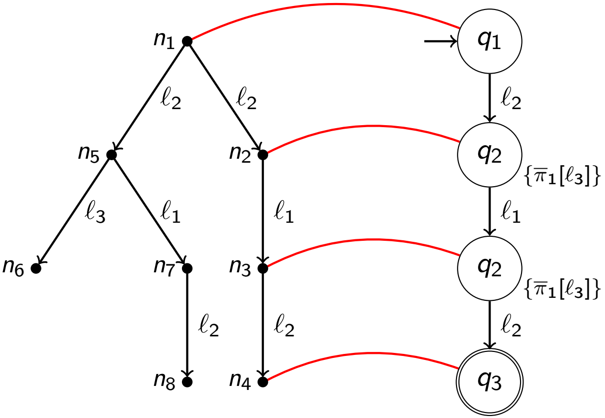
Semantics of condition automata

- Take a path in automaton from initial to final state
- Map to a path in a tree from m to n with equal labeling
- State s maps to node k: k satisfies the conditions of s



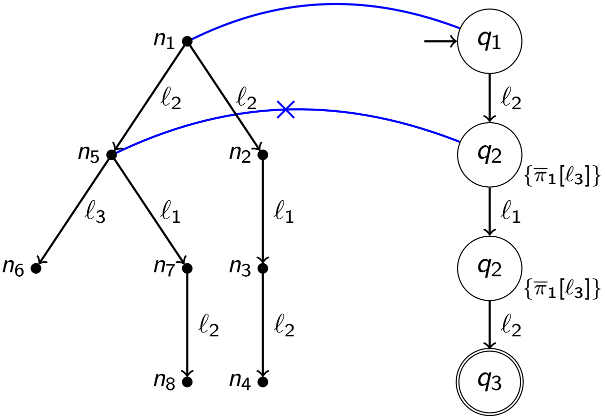
Semantics of condition automata

- Take a path in automaton from initial to final state
- Map to a path in a tree from m to n with equal labeling
- State s maps to node k: k satisfies the conditions of s



Semantics of condition automata

- Take a path in automaton from initial to final state
- Map to a path in a tree from m to n with equal labeling
- State s maps to node k: k satisfies the conditions of s



Condition automata and navigational expressions

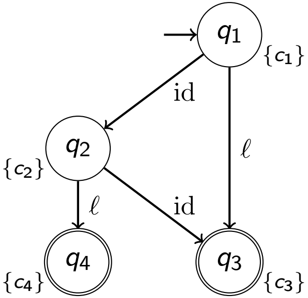
Proposition

Let $\mathfrak{F} \subseteq \{+, \pi, \overline{\pi}\}$. The class of condition automata specified for $\mathcal{N}(\mathfrak{F})$ in the following table is path-equivalent with $\mathcal{N}(\mathfrak{F})$.

Navigational language	Class of condition automata
$\mathcal{N}()$	$\{+, \pi, \overline{\pi}\}$-free and acyclic.
$\mathcal{N}(\pi)$	$\{+, \overline{\pi}\}$-free and acyclic.
$\mathcal{N}(\pi, \overline{\pi})$	$\{+\}$-free and acyclic.
$\mathcal{N}(+)$	$\{\pi, \overline{\pi}\}$-free.
$\mathcal{N}(+, \pi)$	$\{\overline{\pi}\}$-free.
$\mathcal{N}(+, \pi, \overline{\pi})$	no restrictions.

Condition automata and intersect

- Basically: use cross-product construction
- Take care of id-transitions by first removing them

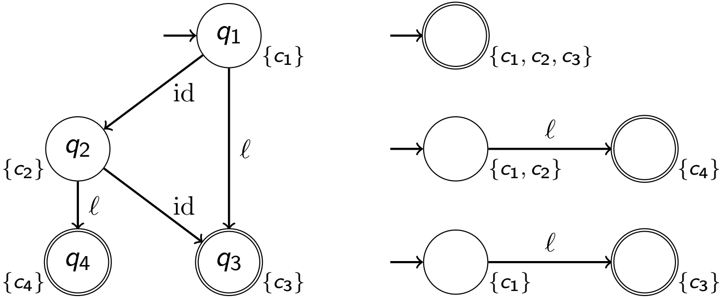


Proposition

Condition automata are closed under $\cap$.

Condition automata and intersect

- Basically: use cross-product construction
- Take care of id-transitions by first removing them



Proposition

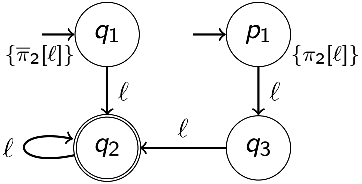
Condition automata are closed under $\cap$.

Condition automata and difference

- Difference: in terms of $\cap$ and complement: $S \setminus T = S \cap \overline{T}$
- In our setting: restrict $\overline{T}$ to the downward complement $\overline{T}_{\downarrow}$

Definition (deterministic condition automaton)

- For each node n: there exists exactly one initial state s such that n satisfies s.
- If node n satisifies state q, then, for each edge (n, ℓ, m) there exists exactly one transition (q, ℓ, p) such that p satisfies m.



Condition automata and downward complement

- Downward complement of deterministic condition automaton
  Swap the final states
- Conclusion: if we can construct a deterministic condition automaton, then condition automata are closed under $\setminus$

Proposition

For every condition automaton there is a path-equivalent deterministic condition automaton.

In this construction $\overline{\pi}$ is introduced if π was already used.

Theorem

On labeled trees we have $\mathcal{N}(\mathfrak{F}) \preceq_p \mathcal{N}(\underline{\mathfrak{F}} \setminus \{\cap, \setminus\})$.

Overview

Conclusions and Future Work

- Full characterization of the expressive power of downward navigational expressions
  - On trees and on chains
  - For boolean queries and path queries
- Typical non-downward operators are omitted
  - Next: we also include converse and diversity
  - We have some initial results

Results on graphs

Proposition (Fletcher et al. ICDT'11)

Let $\mathfrak{F}_1, \mathfrak{F}_2 \subseteq \{+, \pi, \overline{\pi}, \cap, \setminus\}$.

- *Labeled graphs:*
  - $\mathcal{N}(\mathfrak{F}_1) \preceq_b \mathcal{N}(\mathfrak{F}_2)$: *if* $\mathfrak{F}_1 \subseteq \underline{\mathfrak{F}_2}$.
- *Unlabeled Graphs:*
  - $\mathcal{N}(\mathfrak{F}_1) \preceq_p \mathcal{N}(\mathfrak{F}_2)$: *if* $\mathfrak{F}_1 \subseteq \underline{\mathfrak{F}_2}$.
  - $\mathcal{N}(\mathfrak{F}_1) \preceq_b \mathcal{N}(\mathfrak{F}_2)$: *if* $\mathfrak{F}_1 \subseteq \underline{\widetilde{\mathfrak{F}_2}}$ *or if*

$$\mathfrak{F}_1 \subseteq \{\pi\} \text{ and } \mathfrak{F}_2 = \mathfrak{F}_1 \cup \{+\}.$$