

Graph query optimization using semi-join rewritings

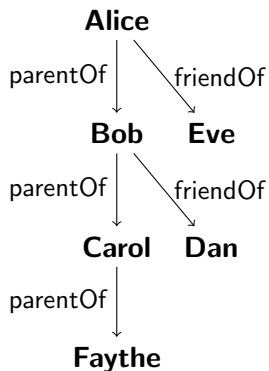
Jelle Hellings¹

`jelle.hellins@uhasselt.be`

Hasselt University, Martelarenlaan 42, 3500 Hasselt, Belgium

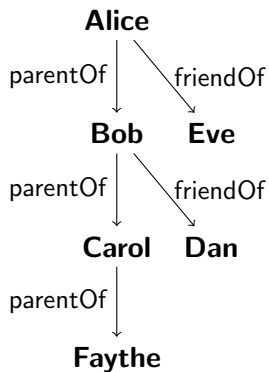
¹Joint work with Catherine L. Pilachowski, Dirk Van Gucht, Marc Gyssens, and Yuqing Wu.

Graphs and Graph Querying



Query: 'Great-grandparents and their friends'

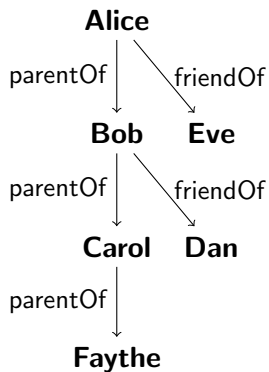
Graphs and Graph Querying



Query: 'Great-grandparents and their friends'

- ▶ (Great-grandparents, descendant):
parentOf ◦ parentOf ◦ parentOf

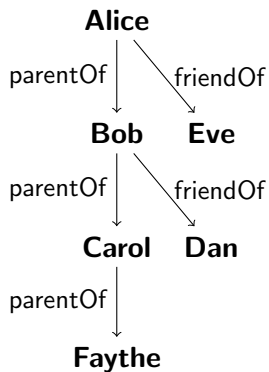
Graphs and Graph Querying



Query: 'Great-grandparents and their friends'

- ▶ (Great-grandparents, descendant):
 $\text{parentOf} \circ \text{parentOf} \circ \text{parentOf}$
- ▶ Great-grandparents:
 $\pi_1[\text{parentOf} \circ \text{parentOf} \circ \text{parentOf}]$

Graphs and Graph Querying



Query: 'Great-grandparents and their friends'

- ▶ (Great-grandparents, descendant):
 $\text{parentOf} \circ \text{parentOf} \circ \text{parentOf}$
- ▶ Great-grandparents:
 $\pi_1[\text{parentOf} \circ \text{parentOf} \circ \text{parentOf}]$
- ▶ Complete query:
 $\pi_1[\text{parentOf} \circ \dots \circ \text{parentOf}] \circ \text{friendOf}$

Graph Query Language

$\underline{id} \mid di \mid \underline{\ell} \mid \ell^\wedge \mid \pi_j[e] \mid \bar{\pi}_j[e] \mid \underline{e \circ e} \mid \underline{e \cup e} \mid e \cap e \mid e - e \mid \underline{[e]^*}$

- ▶ Regular Path Queries

Graph Query Language

$\underline{id} \mid di \mid \underline{\ell} \mid \underline{\ell}^{\wedge} \mid \underline{\pi_j[e]} \mid \bar{\pi}_j[e] \mid \underline{e \circ e} \mid \underline{e \cup e} \mid e \cap e \mid e - e \mid \underline{[e]^*}$

- ▶ Regular Path Queries
- ▶ Nested Regular Path Queries

Graph Query Language

$\underline{id} \mid di \mid \underline{\ell} \mid \underline{\ell}^{\wedge} \mid \underline{\pi_j[e]} \mid \bar{\pi}_j[e] \mid \underline{e \circ e} \mid \underline{e \cup e} \mid e \cap e \mid e - e \mid \underline{[e]^*}$

- ▶ Regular Path Queries
- ▶ Nested Regular Path Queries
- ▶ FO[3] augmented with transitive closure:

graph-navigational core of XPath, GXPath, SPARQL, ...

Query Evaluation

id | di | ℓ | ℓ^{\sim} | $\pi_j[e]$ | $\bar{\pi}_j[e]$ | $e \circ e$ | $e \cup e$ | $e \cap e$ | $e - e$ | $[e]^*$

Query Evaluation

id | di | l | l[^] | $\pi_j[e]$ | $\bar{\pi}_j[e]$ | $e \circ e$ | $e \cup e$ | $e \cap e$ | $e - e$ | $[e]^*$

- ▶ 'Easy to evaluate'

Query Evaluation

id | di | l | l[^] | $\pi_j[e]$ | $\bar{\pi}_j[e]$ | $e \circ e$ | $e \cup e$ | $e \cap e$ | $e - e$ | $[e]^*$

- ▶ 'Easy to evaluate'
- ▶ 'Expensive to evaluate'

Query Evaluation

id | di | l | l[^] | $\pi_j[e]$ | $\bar{\pi}_j[e]$ | $e \circ e$ | $e \cup e$ | $e \cap e$ | $e - e$ | $[e]^*$

- ▶ 'Easy to evaluate'
- ▶ 'Expensive to evaluate'

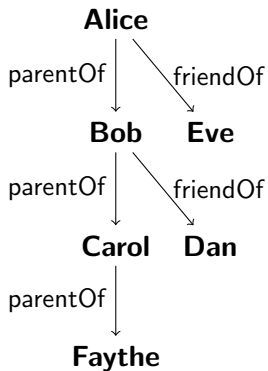
Idea: add partial alternatives for \circ and $[\cdot]^*$

$\pi_1[\text{parentOf} \circ \text{parentOf} \circ \text{parentOf}] \circ \text{friendOf}$

can be rewritten into

$\pi_1[\text{parentOf} \times (\text{parentOf} \times \text{parentOf})] \times \text{friendOf}$.

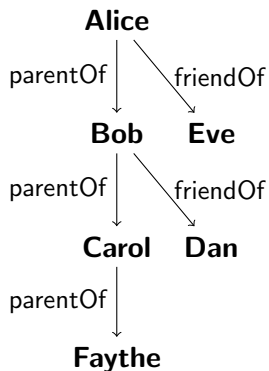
Query Optimization by rewriting?



Problem

$\text{parentOf} \circ \text{parentOf} \circ \text{parentOf}$
is not equivalent to
 $\text{parentOf} \times (\text{parentOf} \times \text{parentOf})$.

Query Optimization by rewriting?



Problem

$\text{parentOf} \circ \text{parentOf} \circ \text{parentOf}$

is not equivalent to

$\text{parentOf} \times (\text{parentOf} \times \text{parentOf})$.

Solution

j-test-equivalent rewriting: we have $e_1 \equiv_j e_2$, if, for every graph \mathcal{G} ,

$$\pi_j[e_1](\mathcal{G}) = \pi_j[e_2](\mathcal{G}).$$

Rewrite composition and transitive closure

- ▶ Rewrite \circ into \bowtie and \times
- ▶ Rewrite $[\cdot]^*$ into $\text{fp}_{j,\mathfrak{N}}[\cdot; \cdot]$ (fixpoint iteration)

$\text{id} \mid \text{di} \mid \ell \mid \ell^\frown \mid \pi_j[e] \mid \bar{\pi}_j[e] \mid e \circ e \mid e \cup e \mid e \cap e \mid e - e \mid [e]^* \mid$
 $e \times e \mid e \bowtie e \mid \mathfrak{N} \mid \text{fp}_{j,\mathfrak{N}}[e; e]$

Rewrite composition and transitive closure

- ▶ Rewrite \circ into \ltimes and \rtimes
- ▶ Rewrite $[\cdot]^*$ into $\text{fp}_{j,\mathfrak{N}}[\cdot; \cdot]$ (fixpoint iteration)

$$\begin{array}{l} \underline{\text{id}} \mid \underline{\text{di}} \mid \underline{\ell} \mid \underline{\ell^{\smile}} \mid \underline{\pi_j[e]} \mid \underline{\bar{\pi}_j[e]} \mid e \circ e \mid \underline{e \cup e} \mid \underline{e \cap e} \mid \underline{e - e} \mid [e]^* \mid \\ \underline{e \ltimes e} \mid \underline{e \rtimes e} \mid \mathfrak{N} \mid \text{fp}_{j,\mathfrak{N}}[e; e] \end{array}$$

Analysis

- ▶ FO[2]

Rewrite composition and transitive closure

- ▶ Rewrite \circ into \times and \times
- ▶ Rewrite $[\cdot]^*$ into $\text{fp}_{j,\mathfrak{N}}[\cdot; \cdot]$ (fixpoint iteration)

id | di | ℓ | ℓ[∘] | π_j[e] | π̄_j[e] | e ∘ e | e ∪ e | e ∩ e | e − e | [e]^{*} |
e × e | e × e | ℑ | fp_{j,ℑ}[e; e]

Analysis

- ▶ FO[2] and FO[2]-like recursion
- ▶ For j -test-equivalent rewriting: only restrictions on \cap and $-$

Rewrite composition and transitive closure

- ▶ Rewrite \circ into \times and \times
- ▶ Rewrite $[\cdot]^*$ into $\text{fp}_{j,\mathfrak{N}}[\cdot; \cdot]$ (fixpoint iteration)

id | di | ℓ | ℓ[∘] | π_j[e] | π̄_j[e] | e ∘ e | e ∪ e | e ∩ e | e − e | [e]^{*} |
e × e | e × e | ℑ | fp_{j,ℑ}[e; e]

Analysis

- ▶ FO[2] and FO[2]-like recursion
- ▶ For j -test-equivalent rewriting: only restrictions on \cap and $-$
- ▶ Rewriting is sound and 'complete'

Rewrite composition and transitive closure

- ▶ Rewrite \circ into \times and \bowtie
- ▶ Rewrite $[\cdot]^*$ into $\text{fp}_{j,\mathfrak{N}}[\cdot; \cdot]$ (fixpoint iteration)

id | di | ℓ | ℓ[∘] | π_j[e] | π̄_j[e] | e ∘ e | e ∪ e | e ∩ e | e − e | [e]^{*} |
e × e | e ⋈ e | ℑ | fp_{j,ℑ}[e; e]

Analysis

- ▶ FO[2] and FO[2]-like recursion
- ▶ For j -test-equivalent rewriting: only restrictions on \cap and $-$
- ▶ Rewriting is sound and ‘complete’
- ▶ Rewriting results in a ‘small’ query: number of steps needed to evaluate the result is twice the length of the original query

Future Work

- ▶ Study (small extensions of) $FO[2]$ in more detail
- ▶ Further query optimization using information on the data
- ▶ Apply similar techniques to relational databases (SQL)

Fixpoints and transitive closure (example)

The transitive closure query

$$\pi_1[[\text{parentOf} \circ \bar{\pi}_1[\text{researcherAt}]]^* \circ \text{ownsPet}]$$

Fixpoints and transitive closure (example)

The transitive closure query

$$\pi_1[[\text{parentOf} \circ \bar{\pi}_1[\text{researcherAt}]]^* \circ \text{ownsPet}]$$

is equivalent to the FO[2]-like query

$$\text{fp}_{1,\mathfrak{N}}[\text{parentOf} \times \bar{\pi}_1[\text{researcherAt}] \times \mathfrak{N}; \text{ownsPet}].$$