Datalog with Equality: Semantics and Evaluation

Martin E. Bidlingmaier

Overview: Extending Datalog by native equality

`mbid.me/eqlog-semantics`   `mbid.me/eqlog-algorithm`

`github.com/eqlog/eqlog`

- Allow equality in conclusions:

$$x \leq y \wedge y \leq x \implies x \equiv y$$

- Can encode partial functions $f : X \to Y$ as $f \subseteq X \times Y$ satisfying *functionality*:

$$f(x, y) \wedge f(x, y') \implies y \equiv y'$$

- Equal elements must behave the same wrt. other relations

- Can implement efficient type inference and Steensgaard's points-to analysis
- $\mathrm{Datalog} \subseteq$ RHL $\cong$ PHL $\supseteq$ essentially algebraic theories
- Semantics: (Weak) reflection of relational structures into subcategories of models
- Datalog/Eqlog evaluation $\triangleq$ orthogonal reflection/small object argument

Background: Datalog

Input:

- Datalog theory $\mathcal{T}$: Set of implications ("sequents", "rules")

$$\mathcal{F} \implies \mathcal{G}$$

  with $\mathcal{F}$ and $\mathcal{G}$ conjunctions of relation atoms

$$\bigwedge_{i=1}^{n} R_i(v_1, \ldots, v_{m_i}).$$

  Every variable in conclusion must also appear in premise.

  Example: $\mathrm{Edge}(u, v) \wedge \mathrm{Edge}(v, w) \implies \mathrm{Edge}(u, w)$

- A relational structure X: Elements ($=$ IDs) and relations over elements.

Output:

- Free relational structure $X' \supseteq X$ satisfying $\mathcal{T}$
- Obtained from X by repeatedly matching premises and adjoining conclusions

Datalog Evaluation

```
fn datalog(structure, sequents):
  loop:
    // 1. Match premises.
    let matches = [];
    for sequent in sequents:
      matches.push(find_matches(structure, sequent.premise));

    // 2. Apply conclusions.
    let has_changed = false;
    for (sequent, matches) in sequents.zip(matches):
      for match in matches:
        if apply_conclusion(structure, sequent.conclusion, match):
          has_changed = true;

    // Terminate if applying conclusions had no effect.
    if !changed:
      return structure;
```

Applications of Datalog

- Compute transitive closure of graph:

$$E(u,v) \wedge E(v,w) \implies E(u,w)$$

- Andersen's points-to analysis:
  1. If an expression e of the form `new Foo` is assigned to variable x, then x might point to e:

  $$\mathrm{Alloc}(x,e) \implies \mathrm{PointsTo}(x,e)$$

  2. If variable x is assigned to variable y somewhere, and x might point to e, then y might point to e:

  $$\mathrm{Assign}(x,y) \wedge \mathrm{PointsTo}(x,e) \implies \mathrm{PointsTo}(y,e)$$

- Composable framework for static code analyses: CodeQL

Datalog lacks equality. Non-applications:

- Congruence closure:

$$f(x,y) \wedge f(x,y') \implies y \equiv y'$$

- Steensgaard's points-to analysis: If x is assigned to y, then x and y have the same might-point-to set.

Relational Horn Logic (RHL)

Definition (RHL)

Like Datalog, but:

- Equality atoms: $u \equiv v$
- Sort quantification atoms: $u \downarrow$ or $u : s$ where s is sort symbol
- Variables can occur in conclusion only

Intuitive evaluation semantics:

- Equality is *congruence* wrt. all relations
  E.g.
  $$\mathrm{Edge}(u, v) \wedge \mathrm{Edge}(v, w)$$
  matches $(x, y), (y', z) \in \mathrm{Edge}$ if $y \equiv y'$ has been inferred
- Sort quantification in premise: Universal quantification over elements
- Sort quantification in conclusion: Try to find substitution, otherwise adjoin fresh elements

Semantics of RHL

Definition

A relational structure X consists of

- a carrier set X_s for each sort symbol s,
- a relation $r_X \subseteq X_{s_1} \times \cdots \times X_{s_n}$ for each relation symbol $r : s_1 \times \cdots \times s_n$.

Morphisms in Rel are maps on carriers preserving relations.

Definition

A relational structure X *satisfies* an RHL sequent $\mathcal{F} \implies \mathcal{G}$ if every interpretation I of $\mathcal{F}$ in X can be extended to an interpretation J of $\mathcal{F} \wedge \mathcal{G}$ in X.
Full subcategory of models $\mathrm{Mod}(\mathcal{T}) \subseteq \mathrm{Rel}$ for every RHL theory $\mathcal{T}$.

Example

Graphs X satisfy $\mathrm{Edge}(u, v) \wedge \mathrm{Edge}(v, w) \implies \mathrm{Edge}(v, w)$ iff they are transitive.

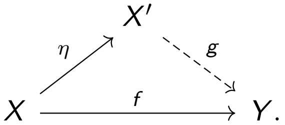
Example

Graphs X satisfy $v : \mathrm{Vertex} \implies \mathrm{Edge}(v, w)$ iff for every vertex $x \in X$ there exists a vertex $y \in Y$ with an edge from x.

Weak reflection

Definition

Let $\mathcal{T}$ be an RHL theory. Let $X \in \mathrm{Rel}$. A *weak reflection* is a map $\eta : X \to X'$ such that $X' \in \mathrm{Mod}(\mathcal{T})$ and for every $X \to Y$ with $Y \in \mathrm{Mod}(\mathcal{T})$ there exists g in



If g is always unique, then η is a *(strong)* reflection.

RHL evaluation = Computing weak reflections

Classifying structures

Definition

Let $\mathcal{F}$ be an RHL formula. The *classifying relational structure* $[\mathcal{F}]$ is such that

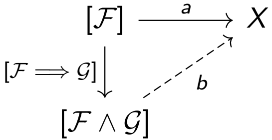
$$\mathrm{Interpretations}(\mathcal{F}, X) \cong \mathrm{Hom}([\mathcal{F}], X)$$

for all $X \in \mathrm{Rel}$.

Carrier of $[\mathcal{F}]$ are variables in $\mathcal{F}$ mod equality atoms in $\mathcal{F}$.

Proposition

$X \in \mathrm{Rel}$ satisfies $\mathcal{F} \implies \mathcal{G}$ iff it is injective to $[\mathcal{F} \implies \mathcal{G}]$:



□

The Small Object Argument

Proposition

Let

- $\mathcal{C}$ *be cocomplete,*
- $M \subseteq \operatorname{Mor}\mathcal{C}$ *be a set of morphisms of finitely presentable objects,*
- $X \in \mathcal{C}$.

Consider

$$X = X_0 \to X_1 \to X_2 \to \dots$$

such that

$$\begin{array}{ccc} \coprod_{(f,a)} A & \longrightarrow & \coprod_{(f,a)} B \\ \downarrow & & \downarrow \\ X_i & \longrightarrow & X_{i+1} \end{array} \quad (\text{pushout } \ulcorner)$$

where $f : A \to B$ *is in* M *and* $a : A \to X$*. Then* $X \to \operatorname{colim}_i X_i$ *is a weak reflection into the subcategory of* M*-injective objects.* □

Partial Horn Logic (PHL)

Definition

Like RHL, but:

- has function symbols in addition to relation symbols
- terms instead of variables only

Epic (epimorphic) PHL: No new variables in conclusions

Epic PHL $\hat{=}$ essentially algebraic theories

PHL is syntactic sugar over RHL:

Definition (Flattening)

Lowers PHL theory to RHL theory:

- Function symbols $f : s_1 \times \cdots \times s_n \to s$ to relations $f : s_1 \times \ldots s_n \times s$ and *functionality axiom*:

$$f(x_1, \ldots, x_n, y) \wedge f(x_1, \ldots, x_n, y') \implies y \equiv y'$$

- Replace terms $t = f(v_1, \ldots v_n)$ by fresh variable v_t and add atom $f(v_1, \ldots, v_n, v)$

RHL Evaluation

Incorporate elements of Datalog evalatution & congruence closure algorithms:

- Semi-naive evaluation: Don't consider matches that were found in previous iteration.
- Indices: Speed up matching; as in computation of SQL joins.
- Union-find: Represent semantic equality, canonical representative in each equivalence class.
- Normalization: Canonicalize elements in relations wrt. union-find.
- Occurence lists: Quickly find tuples a given element occurs in.

`Eqlog`: epic PHL $\to$ RHL $\to$ Rust library

`Egglog`: Forthcoming tool by egg e-graph library authors

Still open: How to properly detect that fixed point is reached?

Thanks!

- Datalog $\subseteq$ Relational Horn Logic $\cong$ PHL $\supseteq$ essentially algebraic theories
- Enable inference of equality, functions & terms, adding new elements during evaluation
- Can implement efficient type inference and Steensgaard's points-to analysis
- Semantics: Reflection of relational structures into reflective subcategories
- Datalog/Eqlog evaluation $\triangleq$ orthogonal reflection/small object argument
- `mbid.me/eqlog-semantics`
- `mbid.me/eqlog-algorithm`
- `github.com/eqlog/eqlog`
- Forthcoming egg-related paper at PLDI
- Palmgren and Vickers: *Partial Horn Logic and Cartesian Categories*.