Guarded Negation Logics

Balder ten Cate
(UC Santa Cruz; LogicBlox⇝Google)

CSLI 2015

Based on joint work with Luc Segoufin, Vince Barany, Martin Otto, Michael Benedikt, Michael Vanden Boom, Thomas Colcombet, …

- **First-Order Logic (FO):**
  - $\phi ::= R(\mathbf{x}) \mid x = y \mid \phi \wedge \phi \mid \phi \vee \phi \mid \exists x \phi \mid \neg\phi(x)$

- Note:
  - Universal quantifier is treated as a defined connective; no function symbols (constants symbols are ok).
  - Negation-free formulas are equivalently known as Unions of Conjunctive Queries (UCQs)

- **Fixed-Point Logic (FO+LFP):**

$$\phi(\ldots, Z_1, \ldots, Z_n) \text{ where}_{\text{lfp}} \begin{cases} Z_1(\mathbf{x}) \leftrightarrow \phi_1(\mathbf{x}, \ldots, Z_1, \ldots, Z_n), \\ \cdots. \\ Z_n(\mathbf{x}) \leftrightarrow \phi_n(\mathbf{x}, \ldots, Z_1, \ldots, Z_n) \end{cases}$$

where $Z_1, \ldots, Z_n$ occurs only positively in $\phi_1, \ldots, \phi_n$.

- **Example**:

$\exists x\ Red(x) \wedge Z(x)$ where$_{\text{lfp}}$ $Z(u) \leftrightarrow Green(u) \vee \exists v(Edge(u,v) \wedge Z(v))$

- Negation-free fragment is also known as Datalog.

Undecidability

- The satisfiability problem for FO is undecidable ("Church's Curse")
- The satisfiability problem for FO+LFP is highly undecidable.
- The satisfiability problem for FO on finite structures is different but also undecidable ("Trakhtenbrot Trouble")

- General topic of this talk: taming FO and FO+LFP.

Motivation (Databases)

- Why we care
  - query optimization (testing if two queries are equivalent)
  - open-world query answering under constraints
  - view-based query rewriting (testing if a query can be equivalently written using only certain views)

- Approaches to taming FO logic
  - Syntactic fragments (e.g., modal logic)
  - Restricted classes of structures (e.g., trees)
  - Changing the definition of the semantics itself (dropping axioms)

Tree width

Trees

Grids

Trees

- Satisfiability of FO formulas on tree structures is decidable
- In fact, this holds even
  - for Guarded Second-Order logic (GSO) ...
  - ... on structures of "bounded tree width"

GSO

- Monadic Second-Order Logic (MSO):
  - Extension of FO with quantification over sets.
- Guarded Second-Order Logic (GSO):
  - Further extension of MSO with quantification over guarded relations (i.e,. relations containing only guarded tuples)

Guardedness

- A set of elements (in a structure) X is a guarded subset if it is either a singleton set or the elements co-occur in some fact (i.e., in some tuple in a relation).
  - Intuition: "the elements in X are related"
- A guarded relation (over the domain of a structure) is a relation that contains only guarded tuples. (Note: all unary relations are guarded)
- **Quiz**: Which of the following formulas is guaranteed to define a guarded relation (in all structures)?
  - $\phi_1(x,y) = R(x,y) \wedge \neg T(x,y)$
  - $\phi_2(x,y) = R(x,y) \vee \neg T(x,y)$
  - $\phi_3(x,y) = P(x) \wedge Q(y)$

Tree width

- **Tree width**: a measure of tree-likeness.
  - trees and forests have tree width 1,
  - cycles have tree width 2,
  - an n x m grid has tree width min{n,m}

- There are several equivalent ways to define tree width.
  - Via tree decompositions
  - Via Cops and Robber games,
  - Via the number of variables required to write the existential-positive diagram of the structure.

- The tree width of an infinite structure is the supremum of the tree width of its finite substructures

Grids versus bounded tree width

- Large grids have a high tree width.
- Conversely, graphs of high tree width contain large grids as a graph minor (Robertson & Seymour's Excluded Grid theorem).
- Thus, bounding the tree width ≈ disallowing large grids.

The Border of Decidability

- The following is decidable [Rabin; Seese; Courcelle]:
  - Given a $\phi \in$ FO and k>0, does ϕ have a model of tree width $\leq k$.
  - In fact the same holds for GSO.
- Conversely, if the GSO-theory of a class of structures C is decidable, then the structures in C have bounded tree width. [Seese 1991]
- In other words, "if it's decidable, it's because of bounded tree width"

- Approaches to taming FO logic
  - Syntactic fragments (e.g., modal logic)
  - Restricted classes of structures (e.g., trees)
  - Changing the definition of the semantics itself (dropping axioms)

Decidable Fragments

- Decidable fragments of first-order logic (FO):
  - Restricting quantifier alternation (e.g., Bernays, Schoenfinkel, 1928)
  - Restricting the number of variables (e.g., Scott 1962, Mortimer 1975)
  - Restricted quantification patterns (Andreka, vBenthem, Nemeti 1998)
  - NEW! Restricting the use of negation

Explaining the Good Behavior of Modal Logic

- “Why is modal logic so robustly decidable?” (Vardi 1996)
- “What makes modal logic tick?” (Andreka, van Benthem, Nemeti 1998)

Modal Logic

- The modal fragment:
  - $\phi(x) \;:=\; P_i(x) \;\mid\; \phi(x) \wedge \psi(x) \;\mid\; \neg\phi(x) \;\mid\; \exists y(Rxy \wedge \phi(y))$

- **Guarded Fragment (GF)**: restricted patterns of quantification.

- **Here**: restricting the use of negation.
  - Unary negation (UNFO): allow only $\neg\phi(x)$ [tC & Segoufin 11]
  - Guarded negation (GNFO): allow also $G(\mathbf{x}) \wedge \neg\phi(\mathbf{x})$ [Barany, tC & Segoufin 11]

Guarded Fragment

- Guarded Fragment (GF) [Andreka, Van Benthem, Nemeti 1998]
  - $\phi ::= R(\mathbf{x}) \mid x = y \mid \phi \vee \phi \mid \phi \wedge \phi \mid \neg \phi \mid \exists \mathbf{x}\, G(\mathbf{xy}) \wedge \phi(\mathbf{xy}) \mid \exists x\, \phi(x)$
  - Unrestricted use of negation; restricted use of quantification.

- GF generalizes modal logic (and several other decidable formalisms) while preserving many desirable properties:
  - Decidable satisfiability problem
  - Finite model property
  - Even the fixpoint extension GF+LFP is decidable.
  - Key insight: GF+LFP has the bounded tree-width property (and is a fragment of GSO)
- Notable exception: Craig interpolation fails (although a weak form of interpolation holds for GF [Marx&Hoogland '00])

Failure of Craig Interpolation in GF

- Let $\phi(x_1) = \exists x_2,....,x_n$ ($G(x_1,...,x_n) \wedge$ DIRECTED-R-CYCLE$(x_1,...,x_n)$)
- Let $\psi(x_1) = P_0(x_1) \wedge \bigwedge_i \forall xy(P_i(x) \wedge R(x,y) \to P_{i+1}(y)) \to P_n(x_1)$
- $\phi(x_1)$ implies $\psi(x_1)$, and every interpolant is provably equivalent to $\exists x_2,....,x_n$ DIRECTED-R-CYLE$(x_1,\ldots,x_n)$. But this formula is not invariant for guarded bisimulations.
- **Conclusion**: GF lacks Craig interpolation. Any extension of GF with Craig interpolation must contain (a formula equivalent to) $\exists x_2,....,x_n$ DIRECTED-R-CYLE$(x_1,...,x_n)$

Guarded Negation

- Guarded Fragment (GF):
  - $\phi ::= R(\mathbf{x}) \mid x = y \mid \phi \vee \phi \mid \phi \wedge \phi \mid \neg\phi \mid \exists \mathbf{x}\, G(\mathbf{xy}) \wedge \phi(\mathbf{xy}) \mid \exists x\, \phi(x)$
  - Unrestricted use of negation; restricted use of quantification.
- Guarded Negation FO (GNFO):
  - $\phi ::= R(\mathbf{x}) \mid x{=}y \mid \exists x \phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid \neg\phi(x) \mid G(\mathbf{xy}) \wedge \neg\phi(\mathbf{x})$
  - Restricted use of negation; unrestricted use of existential quantification.
- Fixed-point Extension (GNFO+LFP):
  - Allow taking least fixpoints of formulas that define guarded relations.
- **Fact**: Every GF (+LFP) sentence is equivalent to a GNFO (+LFP) sentence.

Why Restricted Negation?

- A new way of looking at modal logic
- Gives rise to interesting new decidable fragments
- Restoring Craig interpolation (counterexample is now expressible!)
- The idea that “negation is dangerous” is prominent in DB theory, and most database queries in practice use a limited form of negation.

Examples

- Every node lies on a 3-cycle (uses only unary negation)

  $\forall x \exists yz.(Rxy \wedge Ryz \wedge Rzx) \equiv \neg\exists x \neg\exists yz.(Rxy \wedge Ryz \wedge Rzx)$

- No node lies on a 4-cycle (uses only Boolean negation)

  $\neg\exists xyzu.(Rxy \wedge Ryz \wedge Rzu \wedge Rux)$

- Ternary relation R is contained in S

  - $\forall xyz(Rxyz \rightarrow Sxyz) \equiv \neg\exists xyz(Rxyz \wedge \neg Sxyz)$

- R is transitive (**cannot** be expressed in GNFO)

  $\forall xyz.(Rxy \wedge Ryz \rightarrow Rxz) \equiv \neg\exists xyz(Rxy \wedge Ryz \wedge \neg Rxy)$

- GNFO and GNFO+LFP generalize several existing logics:
  - Modal logic, modal mu-calculus, various description logics,
  - Unions of conjunctive queries, monadic Datalog,
  - CTL*(X), Core XPath
- GNFO (+LFP) extends GF (+LFP) while preserving all its good properties. Plus, we have Craig interpolation.

World of modal logic | World of database theory

GNFO

Guarded fragment — Unions of Conjunctive queries

GNFO+LFP

Guarded fixpoint logic — Monadic datalog

UN-bisimulations

GN-Bisimulations — Homomorphisms

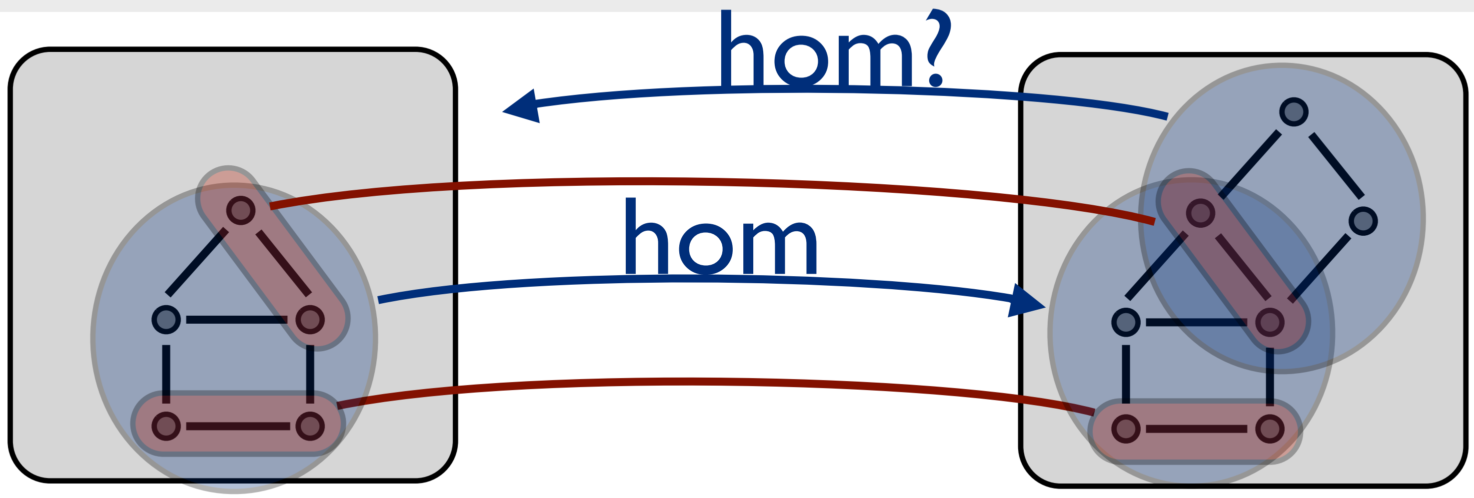
Querying the Guarded Fragment

- Barany-Gottlob-Otto 2010 ("Querying the guarded fragment"):
  - The following is 2ExpTime-complete and finitely controllable:

    Given a GF-sentence ϕ and a (Boolean) UCQ q, test if $\phi \models q$.

- GNFO is a common generalization of GF and UCQs.
  - The above question is equivalent to the (un)satisfiability of $\phi \wedge \neg q$.
  - Conversely, GNFO satisfiability reduces to querying the guarded fragment.
  - We show GNFP satisfiability is 2ExpTime-complete using the techniques from [Barany-Gottlob-Otto 2010] as well as [Barany-Bojanczyk 2011].

GN-Bisimulation Game



- The GN-bisimulation game:
  - Positions: pairs of guarded sets (**a**,**b**)
  - Moves:
    - Spoiler picks a finite set X in one of the structures.
    - Duplicator responds with a partial homomorphism h from X to the other structure, s.t. h(**a**)=**b**.
    - Spoiler picks guarded subsets (**c**,**d**) in h.

Recent Results (Highlights)

- Interpolation (BBtC MFCS 2013, BtCV LICS 2014; BtCV LICS 2015a)
  - UNFO and UNFO+LFP have Craig interpolation
  - Finite variable fragments of UNFO+LFP have uniform interpolation
  - GNFO has constructive interpolation (can compute the interpolant in 2ExpTime using mosaics)
- Boundedness and FO-definability: (BtCO VLDB 2012, BCtCV LICS 2015b)
  - We can decide if the least fixed point of a GNFO formula is reached after a fixed finite number of steps (2ExpTime-complete).
  - As a consequence, we can test whether the least fixed point of a GNFO formula is first-order definable.

Applications

- The decidability, interpolation, and boundedness theorems are used for several database applications including
  - query optimization (BtCO VLDB 2012, BtCT PODS 2014)
  - open-world query answering under constraints (deciding FO-definability —- BtCO VLDB 2012, BBtC MFCS 2013, BCtCV LICS 2015b)
  - querying under access restrictions: determinacy and rewritings (BBB ICDT 2013, BtCT PODS 2014)

- Thank you!