Efficient Fixpoint Methods for Approximate Query Answering in Locally Complete Databases

Alvaro Cortés[◇] Marc Denecker[◇]
Ofer Arieli[*] Maurice Bruynooghe[◇]

◇: Katholieke Universiteit Leuven, Belgium

∗: The Academic College of Tel-Aviv/Yaffo, Israel

Reiter's Closed World Assumption (1978)

In an arbitrary relational database, the Closed World Assumption ($\mathcal{CWA}$) rules as *false* all those tuples that are not in the database.

Train Time Table

Destination	*Time*
Brussels Centraal	8:03
Antwerpen Centraal	8:05
Ghent St. Pieters	8:13
Brugge	8:22

Reiter's $\mathcal{CWA}$ allows us to conclude that there is no train to Hasselt at 8:04, for instance.

Incomplete databases

The CWA applies to stand-alone databases storing *complete* (and correct) information about the world

What happens when the database is not complete?

Telephone

Name	*Telephone*
Leen Desmet	6531421
Leen Desmet	09-23314
Bart Delvaux	5985625

Department

Name	*Department*
Bart Delvaux	Computer Sci.
Leen Desmet	Philosophy
David Finner	Computer Sci.

This database does not store *complete knowledge* about collaborators from the Philosophy department.

$\Rightarrow$ The $\mathcal{CWA}$ is not the correct approach in this context

An alternative view: Open-World Assumption

The other extreme: The Open-World Assumption ($\mathcal{OWA}$)

- The world can be in any state in which all database atoms are true
- A common approach in data integration systems
- The $\mathcal{OWA}$ is often too incomplete and underestimates the knowledge in a database.

How to identify those parts of the database that *are* complete?

Different approaches were presented to specify that the database is partially complete.

- First approach : [Motro 88]
- We follow the approach of Local Closed-World Assumption of [Levy96] and [CDAB05]

A specification of the "areas" in the real world in which the tables of the database contain *all* (true) tuples.

Expressing Local Closed-World Assumptions

A *Local Closed-World Assumption* ($\mathcal{LCWA}$) is an expression [CDAB05] (extended from [Levy96]):

$$\mathcal{LCWA}(P(\overline{x}), \Psi[\overline{x}])$$

Where:

- $\Psi[\overline{x}]$: The window of expertise of the $\mathcal{LCWA}$
- $P(\overline{x})$: A database predicate, called the object of the $\mathcal{LCWA}$
- $\Psi[\overline{x}]$: The free variables in Ψ are a subset of $\overline{x}$

"For all $\overline{x}$ such that Ψ holds in the *real world*, if $P(\overline{x})$ is true in the real world, then $P(\overline{x})$ appears in the database"

Example of the $\mathcal{LCWA}$

Database knows about all the Telephone numbers of people in the CS (computer Science) department:

$$\mathcal{LCWA}(\mathsf{Telephone}(x, y), \mathsf{Dept}(x, \mathsf{CS}))$$

- Windows of expertise: $\mathsf{Dept}(x, \mathsf{CS})$
- Object of the $\mathcal{LCWA}$: $\mathsf{Telephone}(x, y)$

"For all persons x of the department of computer science, all true facts of the form Telephone (x, y) appear in the database."

Semantics of the $\mathcal{LCWA}$

Let D a set of ground atoms (a database) and the $\mathcal{LCWA}$ expression

$$\theta = \mathcal{LCWA}(P(\overline{x}), \Psi[\overline{x}]),$$

the meaning of θ given D is

$$\mathcal{M}_D(\theta) = \forall \overline{x} \Big(\Psi[\overline{x}] \supset \big(P(\overline{x}) \equiv \big(P(\overline{x}) \in P^D \big) \big) \Big)$$

Where "$P(\overline{x}) \in P^D$" is a shorthand for

$$\bigvee_{\overline{a} \in P^D} (\overline{x} = \overline{a})$$

Semantics of a Locally Complete Database

A locally closed database $\mathfrak{D}$ over Σ is pair $(D, \mathcal{L})$, where D is a database and $\mathcal{L} = \{\theta_1, \ldots \theta_m\}$ is a finite set of LCWAs.

The *semantics* of $\mathfrak{D}$ is:

$$\mathcal{M}(\mathfrak{D}) = \mathcal{A} \wedge \bigwedge_{j=1}^{m} \mathcal{M}_D(\theta_j) \wedge \mathsf{UNA}(\Sigma) \wedge \mathsf{DCA}(\Sigma).$$

Where

- $\mathcal{A}$: The conjunction of atoms in D.
- $\mathsf{UNA}(\Sigma)$: Unique names axioms.
- $\mathsf{DCA}(\Sigma)$: Domain closure axioms.

Query Answering

We are interested in evaluating queries $\mathcal{Q}$ with respect to $\mathcal{M}(\mathfrak{D})$:

- $\bar{t}$ is a *certain answer* for $\mathcal{Q}[\bar{x}]$ in $\mathcal{M}(\mathfrak{D})$, if

$$\mathcal{M}(\mathfrak{D}) \models \mathcal{Q}[\bar{t}/\bar{x}].$$

  Set of certain answers: $Cert_{\mathfrak{D}}(\mathcal{Q}[\bar{x}])$.

- $\bar{t}$ is a *possible answer* for $\mathcal{Q}[\bar{x}]$ in $\mathcal{M}(\mathfrak{D})$, if

$$\mathcal{M}(\mathfrak{D}) \cup \mathcal{Q}[\bar{t}/\bar{x}] \text{ is satisfiable.}$$

  Set of possible answers: $Poss_{\mathfrak{D}}(\mathcal{Q}[\bar{x}])$.

Negative complexity result [CDAB05]:

- Checking whether $\bar{t}$ in $Cert_{\mathfrak{D}}(\mathcal{Q}[\bar{x}])$: $coNP - complete$.
- $Poss_{\mathfrak{D}}(\mathcal{Q}[\bar{x}])$ is $NP - complete$.

Query Answering II

We present a tractable method for *approximate query answering*

- Under approximation of certain answers.
- Over approximation of possible answers.

The approach:

- Posing fixpoint queries that symbolically describe the construction of a 3-valued structure that approximates $\mathfrak{D}$.
  - Efficient and sound.
  - In important cases, also complete.

Constructing a 3-valued approximation of $\mathfrak{D} = (D, \mathcal{L})$

The operator $\mathcal{A}pp_{\mathfrak{D}} : \mathfrak{L}^c \to \mathfrak{L}^c$ maps a three-valued structure $\mathcal{K}$ to a three-valued structure $\mathcal{K}' = \mathcal{A}pp_{\mathfrak{D}}(\mathcal{K})$ such that, for every predicate P of $\mathcal{R}(\Sigma)$ and every tuple $\bar{a}$,

$$P(\bar{a})^{\mathcal{K}'} = \begin{cases} \mathbf{t} & \text{if } P(\bar{a}) \in D, \\ \mathbf{f} & \text{if there exists } \mathcal{LCWA}(P(\bar{x}), \Psi_P[\bar{x}]) \in \mathcal{L} \text{ such that} \\ & \Psi_P[\bar{a}]^{\mathcal{K}} = \mathbf{t} \text{ and } P(\bar{a}) \notin D, \\ \mathbf{u} & \text{otherwise.} \end{cases}$$

Let $\mathcal{C}_{\mathfrak{D}}$ be the $\leqslant_p$-least fixpoint of $\mathcal{A}pp_{\mathfrak{D}}$.

- $\mathcal{C}_{\mathfrak{D}}$ 'approximates' $\mathcal{M}(\mathfrak{D})$.
- $Cert_{\mathcal{C}_{\mathfrak{D}}}(\mathcal{Q}[\overline{x}]) \subseteq Cert_{\mathfrak{D}}(\mathcal{Q}[\overline{x}]) \subseteq Poss_{\mathfrak{D}}(\mathcal{Q}[\overline{x}]) \subseteq Poss_{\mathcal{C}_{\mathfrak{D}}}(\mathcal{Q}[\overline{x}])$.

Motivating Fixpoint Queries for the LCWA

$\mathcal{C}_{\mathcal{D}}$ can be constructed in polynomial time in $|D|$.

But it needs to be recomputed each time the database changes...

- We partially avoid this by using fixpoint formulas that symbolically describe the construction of $\mathcal{C}_{\mathcal{D}}$.
- Certain or possible answers to queries can be computed by transforming the original query into a fixpoint/datalog query.
- It suffices to compute the relations that are relevant for the query rather than computing all the relations in $\mathcal{C}_{\mathcal{D}}$.

Fixpoint Queries for the LCWA: Preliminaries

Given a database vocabulary Σ, we introduce, for each element in $\mathcal{R}(\Sigma) = \{P_1, \ldots, P_n\}$, four new predicate symbols P_i^c, P_i^p, $P_i^{c\neg}$ and $P_i^{p\neg}$ of the same arity as P_i.

- Φ^c is the formula obtained by:
  - substituting $P_i^c(\bar{t})$ for each positive occurrence of $P_i(\bar{t})$ in Φ,
  - substituting $\neg P_i^{c\neg}(\bar{t})$ for each negative occurrence of $P_i(\bar{t})$ in Φ.

- Φ^p is, inversely, the formula obtained by:
  - substituting $P_i^p(\bar{t})$ for each positive occurrence of $P_i(\bar{t})$ in Φ
  - substituting $\neg P_i^{p\neg}(\bar{t})$ for each negative occurrence of $P_i(\bar{t})$ in Φ.

Fixpoint Queries

Let $\mathfrak{D} = (D, \mathcal{L})$ be a locally close database. For a query $\mathcal{Q}[\bar{x}]$ we introduce two new variables Q^c and $Q^{c\neg}$, the arity of which is the number of free variables of $\mathcal{Q}[\bar{x}]$, and define:

$$\Delta_{\mathcal{Q},\mathcal{L}} = \left\{ \begin{array}{l} Q^c(\bar{x}) \leftarrow \mathcal{Q}[\bar{x}]^c \\ Q^{c\neg}(\bar{x}) \leftarrow (\neg\mathcal{Q}[\bar{x}])^c \end{array} \right\} \cup \bigcup \left\{ \begin{array}{l} P_i^c(\bar{x}_i) \leftarrow P_i(\bar{x}_i) \\ P_i^{c\neg}(\bar{x}_i) \leftarrow \neg P_i(\bar{x}_i) \wedge (\Psi_{P_i}[\bar{x}_i])^c \end{array} \right\},$$

where the right union is over the database predicates P_i, and Ψ_{P_i} is the window of expertise of P_i.

Certain answers for $\mathcal{Q}[\bar{x}]$: $[\mathbf{lfp}_{Q^c,\Delta_{\mathcal{Q},\mathcal{L}}}](\bar{x})$

Possible answers for $\mathcal{Q}[\bar{x}]$: $\neg[\mathbf{lfp}_{Q^{c\neg},\Delta_{\mathcal{Q},\mathcal{L}}}](\bar{x})$

$\Rightarrow$ Both of these expressions are evaluated in D.

Fixpoint Queries: Example

Consider $\mathcal{LCWA}(\mathsf{Tel}(x,y), \mathsf{Dept}(x,\mathsf{CS}))$. Assume that no closure exists for the Dept relation, i.e. $\mathcal{LCWA}(\mathsf{Dept}(x,y), \mathbf{f})$.

Let $\mathcal{Q} = \mathsf{Tel}(\mathrm{BD}, 3962836)$

$$\Delta_{\mathcal{Q},\mathcal{L}} = \left\{ \begin{array}{l} Q^c \leftarrow \mathsf{Tel}^c(\mathrm{BD}, 3962836). \\ Q^{c\neg} \leftarrow \mathsf{Tel}^{c\neg}(\mathrm{BD}, 3962836). \\ \mathsf{Tel}^c(x,y) \leftarrow \mathsf{Tel}(x,y). \\ \mathsf{Tel}^{c\neg}(x,y) \leftarrow \neg\mathsf{Tel}(x,y) \wedge \mathsf{Dept}^c(x,\mathsf{CS}). \\ \mathsf{Dept}^c(x,y) \leftarrow \mathsf{Dept}(x,y). \\ \mathsf{Dept}^{c\neg}(x,y) \leftarrow \neg\mathsf{Dept}(x,y) \wedge \mathbf{f}. \end{array} \right\}$$

As $\mathbf{lfp}_{Q^{c\neg},\Delta_{\mathcal{Q},\mathcal{L}}}$ is true in D, $\mathsf{Tel}(\text{Bart Delvaux}, 3962836)$ is certainly false.

Fixpoint queries: Soundness & completeness

Answers obtained by fixpoint queries are sound w.r.t. to $\mathcal{C}_{\mathfrak{D}}$.

Given a locally closed database $(D, \mathcal{L})$ and a query $\mathcal{Q}[\bar{x}]$.

Let $(\mathcal{R}^c_{\mathcal{Q}}, \mathcal{R}^{c\neg}_{\mathcal{Q}}, \mathcal{R}^c_1, \mathcal{R}^{\neg c}_1, \ldots, \mathcal{R}^c_n, \mathcal{R}^{\neg c}_n)$ be the relations defined by $\Delta_{\mathcal{Q},\mathcal{L}}$ in D.

Then, for all $1 \leq i \leq n$,

$$\mathcal{R}^c_i = \{\bar{d} \mid P_i(\bar{d})^{\mathcal{C}_{\mathfrak{D}}} = \mathbf{t}\}, \quad \mathcal{R}^{c\neg}_i = \{\bar{d} \mid P_i(\bar{d})^{\mathcal{C}_{\mathfrak{D}}} = \mathbf{f}\},$$

$$\mathcal{R}^c_{\mathcal{Q}} = \{\bar{d} \mid \mathcal{Q}(\bar{d})^{\mathcal{C}_{\mathfrak{D}}} = \mathbf{t}\}, \quad \mathcal{R}^{c\neg}_{\mathcal{Q}} = \{\bar{d} \mid \mathcal{Q}(\bar{d})^{\mathcal{C}_{\mathfrak{D}}} = \mathbf{f}\}.$$

Answers obtained by fixpoint queries are 'sound' and 'complete' w.r.t. to $\mathcal{C}_{\mathfrak{D}}$.

Fixpoint queries: Soundness & completeness

But query answering in $\mathcal{C}_{\mathfrak{D}}$ is not always complete w.r.t $\mathcal{M}(\mathfrak{D})$.

Let $D = \emptyset$ and $\mathcal{L} = \{\mathcal{LCWA}(P, R),\ \mathcal{LCWA}(Q, R \supset \neg P)\}$.

In this case $\mathcal{M}(\mathfrak{D}) = (R \supset \neg P) \wedge ((R \supset \neg P) \supset \neg Q)$

This theory entails $\neg Q$.

The fact that in this case the window of expertise of the second LCWA is exactly the meaning of the first LCWA is not captured by $\mathcal{C}_{\mathfrak{D}}$, and so $Q^{\mathcal{C}_{\mathfrak{D}}} = \mathbf{u}$.

Restricting the LCWA databases

The *LCWA dependency graph* of $\mathcal{L}$ is the directed graph on $\mathcal{R}(\Sigma)$:

- A directed edge from predicate Q to P iff there exists $\mathcal{LCWA}(P(\overline{x}), \Psi[\overline{x}]) \in \mathcal{L}$ such that Q occurs negatively in Ψ.

A *hierarchically closed database* $\mathfrak{D}$ is a locally closed database in which the LCWA dependency graph is cycle-free.

Completeness

Let $\mathfrak{D} = (D, \mathcal{L})$ be a hierarchically closed database such that every window of expertise in $\mathcal{L}$ is a conjunction of literals.

If $\mathcal{Q}[\bar{x}]$ is a conjunction of literals, then

$$Cert_{\mathcal{C}_{\mathfrak{D}}}(\mathcal{Q}[\bar{x}]) = Cert_{\mathfrak{D}}(\mathcal{Q}[\bar{x}]).$$

If $\mathcal{Q}[\bar{x}]$ is a disjunction of literals, then

$$Poss_{\mathcal{C}_{\mathfrak{D}}}(\mathcal{Q}[\bar{x}]) = Poss_{\mathfrak{D}}(\mathcal{Q}[\bar{x}]).$$

Conclusions and Ongoing Work

Tractable methods for approximate querying in locally closed databases, based on standard fixpoint techniques.

- Current research refining the class of LCWA for which the method is complete.
- Preliminary results that incorporate integrity constraints and views. ⇒ XSB implementation.
- Also, safety issues.