On BP-complete query languages on $\mathcal{K}$-relations

Floris Geerts[1] & Antonella Poggi[2]

[1] University of Edinburgh, UK
[2] SAPIENZA Università di Roma, Italy

Overview

- Motivation/BP-completeness
- $\mathcal{K}$-relations
- A query language for $\mathcal{K}$-relations
- How to get BP-completeness in the setting of $\mathcal{K}$-relations
- Main result
- Conclusion

Original motivation

- Given two (fixed) relations S and T, does there exist a generic relational algebra query Q such that $Q(S) = T$?
- Example:

$$S = \begin{array}{|ccc|}\hline A & B & C \\ \hline a & d & a \\ b & a & c \\ a & e & a \\ c & a & b \\ \hline\end{array} \qquad T_1 = \begin{array}{|c|}\hline B \\ \hline b \\ c \\ d \\ e \\ \hline\end{array} \qquad T_2 = \begin{array}{|c|}\hline B \\ \hline c \\ d \\ e \\ \hline\end{array}$$

- $T_1 = \pi_B(\sigma_{A=C}(S)) \cup \rho_{A/B}(\pi_A(\sigma_{A\neq C}(S)))$ but there is no relational algebra query Q such that $T_2 = Q(S)$!
- Why??

BP-completeness

- Bancilhon and Paredaens independently studied this problem around 1978.
- They provided the following characterization:

There exists a generic relational algebra query between two given instances S and T iff

- $\mathrm{adom}(T) \subset \mathrm{adom}(S)$; and moreover
- $\mathrm{Aut}(S) \subseteq \mathrm{Aut}(T)$.

Here, adom() denotes the active domain, and Aut() is the set of automorphisms of a relation, i.e., mappings of the underlying domain that map tuples to tuples.

A query language that satisfies the above characterization is called BP-complete.

Example revisited

$$S = \begin{array}{|ccc|}\hline A & B & C \\ \hline a & d & a \\ b & a & c \\ a & e & a \\ c & a & b \\ \hline \end{array} \qquad T_1 = \begin{array}{|c|}\hline B \\ \hline b \\ c \\ d \\ e \\ \hline \end{array} \qquad T_2 = \begin{array}{|c|}\hline B \\ \hline c \\ d \\ e \\ \hline \end{array}$$

- $\mathrm{adom}(T_i) \subset \mathrm{adom}(S)$, for $i = 1, 2$.
- It is easily verified that $\mathrm{Aut}(S) \subseteq \mathrm{Aut}(T_1)$.
  - We already knew there was a query from S to T.
- However, consider h defined by: $a \mapsto a$, $b \mapsto c$, $c \mapsto b$, $d \mapsto e$ and $e \mapsto d$. Then $h \in Aut(S)$ but $h \notin Aut(T_2)$.
  - There is indeed no query from S to T!

A natural question ...

Can the BP characterization be extended to other data models?

- Given two instances S and T, where S, T are
  - relations (sets) $\rightarrow$ this is the case considered by BP
  - relations (bags - when multiplicities are taken into account)
  - probabilistic relations
  - Boolean c-tables
  - ...
- Informally: Does there exists a generic query Q in the corresponding query language such that $Q(S) = T$?

First, we need for a proper formalization of both the data models and query languages ...

$\mathcal{K}$-relations

- $\mathcal{K}$-relations are an extension of the standard relational data model.
- Given a semiring $\mathcal{K}$ (see later):
  - Each tuple $\bar{t}$ in a $\mathcal{K}$-relation R is assigned a value in $\mathcal{K}$, denoted by $R(\bar{t})$.
  - The underlying "classical" relation of $\mathcal{K}$-relation is called the support of R and is equal to $\text{supp}(R) = \{\bar{t} \mid R(\bar{t}) \neq 0\}$.
- $\mathcal{K}$-relations were recently introduced by Green et al. (PODS'07, "*Provenance semirings*") as an abstraction for various existing data models.
- It also allows to define new datamodels by varying the semiring $\mathcal{K}$...

Intermezzo: semirings

- A semiring is an algebraic structure $\mathcal{K} = (\mathbb{K}, \oplus, \otimes, 0, 1)$ consisting of:
  - a set $\mathbb{K}$;
  - a binary operation $\oplus$, s.t. $(\mathbb{K}, \oplus, 0)$ is a commutative monoid with identity element 0;
  - a binary operation $\otimes$, s.t. $(\mathbb{K}, \otimes, 1)$ is a monoid with identity element 1;
  - and such that $\otimes$ distributes over $\oplus$; and finally
  - 0 is an annihilating element.
- $\mathcal{K}$ is commutative if $(\mathbb{K}, \otimes, 1)$ is a commutative monoid
  - We only consider commutative semirings.

Example

$R_1 =$

A	B	
a	b	true
b	c	true
c	a	true

$R_2 =$

A	B	
a	b	2
b	c	2
c	a	3

$R_3 =$

A	B	
a	b	$(b_1 \wedge b_2) \vee b_1$
b	c	b_1
c	a	$b_2 \vee b_3$

$R_4 =$

A	B	
a	b	$X \cap Y$
b	c	Y
c	a	$Y \cup Z$

$R_5 =$

A	B	
a	b	$xy + z$
b	c	$2x^2 + z$
c	a	$3y^3$

$\mathcal{K}_1 = (\mathbb{B}, \vee, \wedge, \text{false}, \text{true})$, $\mathcal{K}_2 = (\mathbb{N}, +, \times, 0, 1)$,
$\mathcal{K}_3 = (\text{PosBool}(X), \vee, \wedge, \text{false}, \text{true})$, $\mathcal{K}_4 = (\mathcal{P}(X), \cup, \cap, \emptyset, X)$, and
$\mathcal{K}_5 = (\mathbb{N}[X], +, \times, 0, 1)$ (used in the context of provenance).

The positive relational algebra $\mathcal{RA}^+_{\mathcal{K}}$ on $\mathcal{K}$-relations

union If $R_1, R_2 : U\text{-Tup} \rightarrow K$ then $R_1 \cup R_2 : U\text{-Tup} \rightarrow \mathbb{K}$ is defined by

$$(R_1 \cup R_2)(t) = R_1(t) \oplus R_2(t).$$

projection If $R : U\text{-Tup} \rightarrow \mathbb{K}$ and $V \subseteq U$ then $\pi_V(R) : V\text{-Tup} \rightarrow \mathbb{K}$ is defined by

$$(\pi_V R)(t) = \bigoplus_{t = t' \text{ on } V \text{ and } R(t') \neq 0} R(t').$$

selection If $R : U\text{-Tup} \rightarrow \mathbb{K}$ and the selection predicate $\mathbf{P}$ maps each U-tuple to either 0 or 1 depending on the equality or inequality of pairs of attributes, then $\sigma_{\mathbf{P}}(R) : U\text{-Tup} \rightarrow \mathbb{K}$ is defined by

$$(\sigma_{\mathbf{P}}(R))(t) = R(t) \otimes \mathbf{P}(t).$$

The positive relational algebra $\mathcal{RA}^+_\mathcal{K}$ on $\mathcal{K}$-relations

natural join If $R_i : U_i\text{-Tup} \longrightarrow \mathbb{K}$, for $i = 1, 2$, then $R_1 \bowtie R_2$ is the $\mathcal{K}$-relation over $U_1 \cup U_2$ defined by

$$(R_1 \bowtie R_2)(t) = R_1(t_1) \otimes R_2(t_2),$$

where $t_1 = t$ on U_1 and $t_2 = t$ on U_2.

renaming If $R : U\text{-Tup} \longrightarrow \mathbb{K}$ and $\beta : U \longrightarrow U'$ is a bijection then $\rho_\beta(R)$ is the $\mathcal{K}$-relation over U' defined by

$$(\rho_\beta R)(t) = R(t \circ \beta).$$

Green et al. showed that $\mathcal{RA}^+_\mathcal{K}$ coincides with the standard positive algebra in case of set, bag, conditional and probabilistic relations.

Example

$R_5 =$

A	B	
a	b	$xy + z$
b	c	$2x^2 + z$
c	a	$3y^3$

$R_5 \cup R_5 =$

A	B	
a	b	$2(xy + z)$
b	c	$2(2x^2 + z)$
c	a	$2(3y^3)$

$R_5 \bowtie R_5 =$

A	B	
a	c	$(xy + z)(2x^2 + z)$
b	a	$(2x^2 + z)(3y^3)$
c	b	$(3y^3)(xy + z)$

$\pi_A(R_5 \cup R_5 \bowtie R_5) =$

A	
a	$(xy + z)(1 + (2x^2 + z))$
b	$(2x^2 + z)(1 + (3y^3))$
c	$(3y^3)(1 + (xy + z))$

BP-completeness in the setting of $\mathcal{K}$-relations

- First, we reconsider the notion of automorphism, this time for $\mathcal{K}$-relations.
- Let S be a classical relation. Its set of *automorphisms* of S, denoted by $\text{Aut}(S)$, consists of all permutations h of $\text{adom}(S)$ such that $h(t) \in S$ iff $t \in S$.
- Let R be a $\mathcal{K}$-relation. The set of *automorphisms* of R, denoted by $\text{Aut}_{\mathcal{K}}(R)$, is defined as the automorphisms in $\text{Aut}(\text{supp}(R))$ such that

$$R(h(t)) = R(t), \forall t \in \text{supp}(R).$$

  I.e, only those automorphisms of $\text{supp}R$ are considered that also preserve the $\mathcal{K}$-values.

Automorphism of $\mathcal{K}$-relations (bag case)

$R_1 =$

A	B	
a	a	2
b	b	2

$R_2 =$

A	B	
a	a	1
b	b	2

$R_3 =$

A	B	
a	a	1
b	b	1

- When regarded as sets $\text{Aut}(R_1) = \text{Aut}(R_2) = \text{Aut}(R_3)$.
- But $\text{Aut}_{\mathcal{K}}(R_1) = \text{Aut}_{\mathcal{K}}(R_3)$, $\text{Aut}_{\mathcal{K}}(R_2) \subsetneq \text{Aut}_{\mathcal{K}}(R_1)$ for $\mathcal{K} = (\mathbb{N}, +, \times, 0, 1)$ (bag case).

BP-completeness in the setting of $\mathcal{K}$-relations

Definition

Let S be a $\mathcal{K}$-relation. The set of $\mathcal{K}$-relations that are definable from S, denoted by $\mathrm{DEF}_{\mathcal{K}}(S)$, is defined as:

$$\mathrm{DEF}_{\mathcal{K}}(S) = \{T \mid \mathrm{adom}(T) \subseteq \mathrm{adom}(S) \text{ and } \mathrm{Aut}_{\mathcal{K}}(S) \subseteq \mathrm{Aut}_{\mathcal{K}}(T)\}.$$

Definition

Let $\mathcal{L}$ be a query language, and S a $\mathcal{K}$-relation. The basic information of S with respect to $\mathcal{L}$ is the set of $\mathcal{K}$-relations:

$$\mathrm{BI}_{\mathcal{K},\mathcal{L}}(S) = \{T \mid Q(S) = T \text{ for some generic query } Q \in \mathcal{L}\}.$$

Definition

A query language $\mathcal{L}$ is BP-complete on $\mathcal{K}$-relations if $\mathrm{BI}_{\mathcal{K},\mathcal{L}}(S) = \mathrm{DEF}_{\mathcal{K}}(S)$ for any $\mathcal{K}$-relation S.

BP-completeness of $\mathcal{RA}^+_\mathcal{K}$??

- The result of Paredaens shows that $\mathcal{RA}^+_\mathcal{K}$ is *BP*-complete on $\mathcal{K}$-relations for $\mathcal{K} = (\mathbb{B}, \vee, \wedge, \text{false}, \text{true})$ (set case)
  - No difference is needed (although inequality is needed in selections).
- An easy induction on the structure of queries shows that:

Lemma

For any semiring $\mathcal{K}$, any $Q \in \mathcal{RA}^+_\mathcal{K}$ and any $\mathcal{K}$-relation S, we have that (i) $\text{adom}(Q(S)) \subseteq \text{adom}(S)$ and (ii) $\text{Aut}_\mathcal{K}(S) \subseteq \text{Aut}_\mathcal{K}(Q(S))$.

- This property is also known as BP-preserveness.
- In order to show BP-completeness of $\mathcal{RA}^+_\mathcal{K}$, it remains to verify

$$\text{DEF}_\mathcal{K}(S) \subseteq \text{BI}_{\mathcal{K},\mathcal{RA}^+_\mathcal{K}}(S)$$

for any $Q \in \mathcal{RA}^+_\mathcal{K}$ and $\mathcal{K}$-relation S.

BP-completeness of $\mathcal{RA}^+_{\mathcal{K}}$?? In general, no!

Proposition

There exists a semiring $\mathcal{K}$ such that $\mathcal{RA}^+_{\mathcal{K}}$ is not BP-complete on $\mathcal{K}$-relations.

- Consider $\mathcal{K} = (\mathbb{N}, +, \times, 0, 1)$ (bag case)

$S =$

A	B	
a	a	2
b	b	2

$T =$

A	B	
a	a	1
b	b	1

- $\mathrm{Aut}_{\mathcal{K}}(S) = \mathrm{Aut}_{\mathcal{K}}(T)$ and hence $T \in \mathrm{DEF}_{\mathcal{K}}(S)$.
- However, for any $Q \in \mathcal{RA}^+_{\mathcal{K}}$, $Q(S)$ is either
  - empty, or
  - the empty tuple, or
  - such that it contains only tuples having *even multiplicity*.
- $T \notin \mathrm{BI}_{\mathcal{K},\mathcal{RA}^+_{\mathcal{K}}}(S)$.

Question: Can we extend $\mathcal{RA}^+_\mathcal{K}$ to get BP-completeness for the bag case?

- Consider $\mathcal{K} = (\mathbb{N}, +, \times, 0, 1)$ (bag case)

$S =$

A	B	
a	a	2
b	b	2

$T =$

A	B	
a	a	1
b	b	1

- It is standard practice to add duplicate elimination to the bag algebra:

$$\delta(R)(t) = 1 \qquad \text{for all } t \in \text{supp}(R).$$

- Hence, $T = \delta(S)$ (and our counterexample is resolved!)
- Still in the bag case (for now), denote by $\mathcal{RA}^{+,\delta}_\mathcal{K}$ the extension of $\mathcal{RA}^+_\mathcal{K}$ with δ.

Question: Is duplicate elimination enough?

- Recall, we are still in the bag case.

$$S = \begin{array}{|cc||c|}\hline A & B & \\ \hline a & a & 1 \\ b & b & 2 \\ \hline\end{array} \quad T = \begin{array}{|cc||c|}\hline A & B & \\ \hline a & a & 2 \\ b & b & 1 \\ \hline\end{array}$$

- $\mathrm{Aut}_{\mathcal{K}}(S) = \mathrm{Aut}_{\mathcal{K}}(T)$ and hence $T \in \mathrm{DEF}_{\mathcal{K}}(S)$.
- However, for any $Q \in \mathcal{RA}_{\mathcal{K}}^{+,\delta}$, $Q(S)$ it holds that
  - for any two tuples s and t in $Q(S)$, s occurs with *less or equal multiplicity* than t iff s contains a *less or equal number of b's* than t.
- Hence, $T \notin \mathrm{BI}_{\mathcal{K},\mathcal{RA}_{\mathcal{K}}^{+,\delta}}(S)$.

Question: Is duplicate elimination enough? No!

- Recall, we are still in the bag case.

$S =$

A	B	
a	a	1
b	b	2

$T =$

A	B	
a	a	2
b	b	1

- $\text{Aut}_{\mathcal{K}}(S) = \text{Aut}_{\mathcal{K}}(T)$ and hence $T \in \text{DEF}_{\mathcal{K}}(S)$.
- However, for any $Q \in \mathcal{RA}_{\mathcal{K}}^{+,\delta}$, $Q(S)$ it holds that
  - for any two tuples s and t in $Q(S)$, s occurs with *less or equal multiplicity* than t iff s contains a *less or equal number of b's* than t.
- Hence, $T \notin \text{BI}_{\mathcal{K},\mathcal{RA}_{\mathcal{K}}^{+,\delta}}(S)$.

Extending $\mathcal{RA}_{\mathcal{K}}^{+,\delta}$ even more ...

- Still in the bag case, suppose we add the difference operator $\setminus$.
- Resulting algebra is $\mathcal{RA}_{\mathcal{K}}^{*}$, i.e, $\mathcal{RA}_{\mathcal{K}}^{+}$ plus δ and $\setminus$.

$S =$

A	B	
a	a	1
b	b	2

$T =$

A	B	
a	a	2
b	b	1

- Easily verified that

$$T = (((\delta(S) \cup \delta(S)) \setminus S) \cup \delta(S).$$

- Hence, $T \in \mathsf{BI}_{\mathcal{K},\mathcal{RA}_{\mathcal{K}}^{*}}(S)$ (and also this counterexample is resolved!)

Question: Is $\mathcal{RA}_{\mathcal{K}}^{*}$ BP-complete on bags?

General strategy

- We will show that $\mathcal{RA}^*_{\mathcal{K}}$ is BP-complete on $\mathcal{K}$-relations for a large class of semirings
  - This class includes the bag semiring $(\mathbb{N}, +, \times, 0, 1)$.
- We need to introduce difference and duplicate elimination on $\mathcal{K}$-relations.
- Note: the introduction of difference was left open by Green et al. (PODS'07).

Adding difference to $\mathcal{RA}^+_\mathcal{K}$

We add difference to $\mathcal{RA}^+_\mathcal{K}$ in two steps:

- First, we introduce a so-called monus operator $\ominus$ on a class of semirings;
- Second, using $\ominus$ we can then define the difference $\setminus$ of two $\mathcal{K}$-relations.

We will see that we need some further assumptions on the semirings....

Adding monus to semirings

- Assume that $\mathcal{K}$ is naturally ordered. That is, the quasi-order $x \preceq y$ on $\mathbb{K}$ defined as $x \preceq y$ iff there exists a $z \in \mathbb{K}$ such that $x \oplus z = y$, must define a partial order on $\mathbb{K}$.
- We additionally require the following property (†): for each pair of elements $x, y \in \mathbb{K}$, the set $\{z \in \mathbb{K} \mid x \preceq y \oplus z\}$ has a (unique) smallest element.

Definition

Let $\mathcal{K}$ be a naturally ordered semiring that satisfies property (†). For any $x, y \in \mathbb{K}$, we define $x \ominus y$ to be the smallest element z such that $x \preceq y \oplus z$.

From here on, we call a commutative semiring $\mathcal{K}$ which can be equipped with a monus operator an *m*-semiring.

Examples

All semirings considered so far are naturally ordered.

- In case of $(\mathbb{B}, \vee, \wedge, \text{false}, \text{true})$ (set case), the monus corresponds to negation;
- In case of $(\mathbb{N}, +, \times, 0, 1)$ (bag case), the monus corresponds to the truncated minus $x \dot{-} y = \max\{0, x - y\}$ on $\mathbb{N}$;
- In case of $(\mathcal{P}(X), \cup, \cap, \emptyset, X)$ (prob. db), the monus correspons to complementation;
- In case of $(\text{PosBool}(X), \vee, \wedge, \text{false}, \text{true})$ (c-tables), the monus cannot be defined unless negated literals are added to the base set: $(\text{Bool}(X), \vee, \wedge, \text{false}, \text{true})$ in which case the monus corresponds to negation.

Difference operator

Let $\mathcal{K}$ be an m-semiring.

- We obtain $\mathcal{RA}_{\mathcal{K}}$ by extending $\mathcal{RA}^{+}_{\mathcal{K}}$ with the operator

difference If $R_1, R_2 : U\text{-Tup} \to \mathbb{K}$ then $R_1 \ominus R_2 : U\text{-Tup} \to \mathbb{K}$ is defined by

$$(R_1 - R_2)(t) = R_1(t) \ominus R_2(t).$$

The following is easily verified:

$\mathcal{RA}_{\mathcal{K}}$ coincides with the standard relational algebra on sets, bags, c-tables and probabilistic databases.

Adding duplicate elimination to $\mathcal{RA}_{\mathcal{K}}$

- We further restrict the class of *m*-semirings $\mathcal{K} = (\mathbb{K}, \oplus, \ominus, \otimes, 0, 1)$ to those that are finitely generated

  I.e., every element in $\mathbb{K}$ can be written as a finite sequence of sums ($\oplus$), monus ($\ominus$), products ($\otimes$) of a finite set of elements $\mathbf{k}_1, \ldots, \mathbf{k}_m$, called generators.
- We denote the set of generators of $\mathbb{K}$ by $\mathcal{G}en(\mathbb{K})$.
- Examples:
  - $\mathcal{G}en(\mathbb{B}) = \{\text{true}\}$;
  - $\mathcal{G}en(\mathbb{N}) = \{1\}$;
  - $\mathcal{G}en(\text{Bool}(X)) = X$;
  - $\mathcal{G}en(\mathcal{P}(X)) = X$;
  - $\mathcal{G}en(\mathbb{N}[X]) = \{1\} \cup X$.

Adding duplicate elimination to $\mathcal{RA}_{\mathcal{K}}$

Let $\mathcal{K} = (\mathbb{K}, \oplus, \otimes, \ominus, 0, 1)$ be a finitely generated *m*-semiring with generators $\mathcal{G}en(\mathbb{K}) = \{\mathbf{k}_1, \dots, \mathbf{k}_m\}$.

- We define the following set *duplicate elimination* operators:

duplicate elimination If $R : U\text{-Tup} \to \mathbb{K}$ and $\mathbf{k}_i$ is a generator of $\mathcal{K}$ then $\delta_{\mathbf{k}_i} : U\text{-Tup} \to K$ is defined by

$$(\delta_{\mathbf{k}_i}(R))(t) = \mathbf{k_i} \text{ for each } t \in R.$$

We denote by $\mathcal{RA}^*_{\mathcal{K}}$ the query language obtained by extending $\mathcal{RA}_{\mathcal{K}}$ with duplicate elimination.

Main result

Theorem

$\mathcal{RA}^*_{\mathcal{K}}$ is BP-complete on $\mathcal{K}$-relations for arbitrary finitely generated m-semirings $\mathcal{K}$.

- Proof is constructive: if the automorphism property holds, then a query Q in $\mathcal{RA}^*_{\mathcal{K}}$ is provided.
- Moreover, in the set case, i.e., $\mathcal{K} = (\mathbb{B}, \vee, \wedge, \text{false}, \text{true})$, the query Q collapses to the same query as given by Paredaens.
  - I.e., it is a query in $\mathcal{RA}^+_{\mathcal{K}}$.
- Implies a complexity bound on deciding the existence of a query between two $\mathcal{K}$-relations
  - Provided that one knows the complexity of checking the equality of two $\mathcal{K}$-values.

Concluding remarks

- We have seen that $\mathcal{RA}_{\mathcal{K}}^{+}$ is, in contrast to the set case, not BP-complete on general $\mathcal{K}$-relations.
- Extending with $\mathcal{RA}_{\mathcal{K}}^{+}$ with difference and duplicate elimination, results in $\mathcal{RA}_{\mathcal{K}}^{*}$ which is BP-complete on an large class of $\mathcal{K}$-relations.

Concluding remarks

- It is still interesting to look for a characterization for the existence of a $\mathcal{RA}^+_\mathcal{K}$ query between two $\mathcal{K}$-relations (is not trivial, even not in the bag case).
- The introduction of difference and duplicate elimination requires revisiting some of the results of Green et al.
- Is the monus the only way to incorporate difference in the semiring setting. There is huge amount of work regarding these algebraic structures.
- Expressive power of these query languages?