Tutorial: Analyzing rule sets with Kiabora

1 Zoology of existential rule classes

Abstract rule classes

- A set of rules is a *finite expansion set* (fes) if its chase is finite.
- A set of rules is a *finite unification set* (fus) if every conjunctive query has a sound and complete UCQ-rewriting w.r.t. this set.

Deciding if a set of rules is a fus or a fes is undecidable in general but some recognizable rule classes are known to be fus or fes.

Concrete rule classes

This tutorial aims at discovering some classes of existential rules (among many other) that can be recognized and have interesting properties. The following notions are useful to study rule classes.

Definition 1 (Predicate graph)**.** The *predicate graph* of a set of existential rules $\mathcal{R}$, denoted by $\mathsf{PG}(\mathcal{R})$, is defined as follows:

- the vertices of $\mathsf{PG}(\mathcal{R})$ are the pairs (p, i), where p is a predicate appearing in $\mathcal{R}$ and i is an integer between 1 and the arity of p;
- there is a *normal edge* from (p, i) to (q, j) if there is a rule ρ in which there is a variable y appearing at position (p, i) in the body of ρ and at position (q, j) in the head of ρ;
- there is a *special edge* from (p, i) to (q, j) if there is a rule ρ in which there is a variable y appearing at position (p, i) in the body of ρ and at some position in the head of ρ, and an existential variable z appearing at position (q, j) in the head of ρ.

Definition 2 (Graph of rule dependencies)**.** We say that a rule ρ_2 depends on a rule ρ_1 if there exists a database D such that ρ_1 is applicable to D by π_1, and there exists π_2 such that ρ_2 is applicable to $\alpha(D, \rho_1, \pi_1)$ by π_2 and ρ_2 is not applicable to D by π_2.

The *graph of rule dependencies* of a ruleset $\mathcal{R}$ is a graph having as vertices the rules of $\mathcal{R}$ and an edge from ρ_1 to ρ_2 if ρ_2 depends on ρ_1.

Definition 3 (Frontier of a rule)**.** The *frontier* of a rule is the set of variables that occur both in the body and the head of the rule.

Definition 4 (Marked variables)**.** The marked variable set is built from a rule set by the following marking procedure:

1. for each rule ρ_i and for each variable x occurring in the body of ρ_i, if there is an atom of the head of ρ_i in which x does not occur, mark each occurrence of x in the body of ρ_i;
2. apply until a fixpoint is reached: for each rule ρ_i, if a marked variable x appears at position (p, k) in the body of ρ_i, then for each rule ρ_j (including $i = j$) and for each variable y appearing at position (p, k) in the head of ρ_j, mark each occurence of y in the body of ρ_j.

- **Full - Datalog - Range restricted (rr) :** All variables that appear in the head also occur in the body. Ensures fes.
- **Linear (lin) :** The body contains a single atom. Ensures fus.
- **Weakly acyclic (wa) :** The predicate graph does not contain any critical cycle, i.e. a cycle containing a special edge. Ensures fes.
- **Acyclic graph of rule dependencies (aGRD) :** The graph of rule dependencies is acyclic. Ensures fes and fus.
- **Sticky (s) :** Each marked variable occurs at most once in a rule body. Ensures fus.
- **Disconnected (disc) :** The frontier is empty. Ensures fes and fus.
- **Frontier-1 (fr1) :** The frontier contains only 1 variable.
- **Guarded (g) :** At least one atom in the body (called a guard) contains all the variables from the body.

2 Analyzing rule sets with Kiabora

Kiabora (`https://graphik-team.github.io/graal/downloads/kiabora`) is a tool dedicated to the analysis of a set of existential rules. Design rule sets with the following properties and check them on the online analyzer: `https://graphik-team.github.io/graal/downloads/kiabora-online`.

1. weakly acyclic and not full
2. weakly acyclic and not aGRD
3. aGRD and not weakly acyclic
4. linear and sticky, with a non-empty set of marked variables
5. linear and not sticky
6. sticky, with a non-empty set of marked variables, and not linear
7. disconnected
8. guarded, frontier 1 and not linear
9. guarded and not frontier 1
10. sticky, with a non-empty set of marked variables, and not weakly acyclic
11. with a cycle of size at least three in the graph of rule dependencies
12. with a critical cycle of size at least three in the predicate graph
13. a set of rules that is not known to be a fes or a fus but can be partitioned into a fes and a fus that are not independent but such that query answering can be done by chasing w.r.t. the fes then using UCQ-rewriting w.r.t. the fus
14. a set of rules that can be partitioned into a fes and a fus but such that query answering cannot be done by chasing w.r.t. the fes then using UCQ-rewriting w.r.t. the fus
15. a set that has none of the considered properties

Note: Kiabora uses DLGP syntax. The set of rules $\{r(x,y) \wedge p(y,z) \rightarrow r(x,z), t(x,y) \rightarrow p(x,y)\}$ is written as follows (variables must begin with an uppercase letter).

```
r(X,Z) :- r(X,Y), p(Y,Z).
p(X,Z) :- t(X,Y).
```