1 Weak Acyclicity

With each set of existential rules \( \mathcal{R} \) can be associated a graph, called the *predicate graph* of \( \mathcal{R} \), and here denoted by \( \text{PG}(\mathcal{R}) \). It is defined as follows:

- the vertices of \( \text{PG}(\mathcal{R}) \) 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 \( \rho \) in which there is a variable \( y \) appearing at position \( (p, i) \) in the body of \( \rho \) and at position \( (q, j) \) in the head of \( \rho \);
- there is a *special edge* from \( (p, i) \) to \( (q, j) \) if there is a rule \( \rho \) in which there is a variable \( y \) appearing at position \( (p, i) \) in the body of \( \rho \) and at some position in the head of \( \rho \), and a variable \( z \) appearing at position \( (q, j) \) in the head of \( \rho \).

1. Draw the predicate graph of the ruleset containing:
   - \( \forall x \forall y (r(x, y) \rightarrow \exists z r(y, z)) \);
   - \( \forall x \forall y (s(x, y) \rightarrow r(y, x)) \);
   - \( \forall x \forall y \forall z (r(x, y) \land r(y, z) \rightarrow r(x, z)) \).

2. A critical cycle is a cycle containing a special edge. Show that if \( \text{PG}(\mathcal{R}) \) does not contain any critical cycle (we say that \( \mathcal{R} \) is weakly-acyclic, w.a. for short), then the chase w.r.t. \( \mathcal{R} \) always terminates.

3. Determine the data and combined complexity of reasoning with respect to weakly-acyclic rulesets.


2 Graph of Rule Dependencies

We say that a rule \( \rho_2 \) depends on a rule \( \rho_1 \) if there exists a database \( D \) such that \( \rho_1 \) is applicable to \( D \) by \( \pi_1 \), and there exists a *new application* of \( \rho_2 \) on \( \alpha(D, \rho_1, \pi_1) \).

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 \( \rho_1 \) to \( \rho_2 \) if \( \rho_2 \) depends on \( \rho_1 \). A ruleset belongs to the class aGRD if its graph of rule dependencies is acyclic.

1. Compare the class of weakly acyclic ruleset and that of aGRD.

2. Split a ruleset \( \mathcal{R} \) in \( \mathcal{R}_1, \ldots, \mathcal{R}_n \) such that each \( \mathcal{R}_i \) corresponds to a strongly connected component of the graph of rule dependencies of \( \mathcal{R} \). Assume that for all \( i \), \( \mathcal{R}_i \) is weakly acyclic. What can you say of \( \mathcal{R} \)?

   A *piece unifier* of a query \( q \) with an existential rule \( \rho \) is a triple \( \mu = (Q', H', P_u) \) such that:
   - \( Q' \) is a non empty subset of \( Q \)
   - \( H' \) is a subset of the head \( H \) of \( \rho \)
• $P_u$ is partition on the terms of $H'$ and $Q'$ s.t.:
  
  – no two constants belong to the same class
  – if a class of $P_u$ contains an existential variable (thus, of $H'$), then the other terms are non-separating variable of $Q'$ (i.e., variables of $Q'$ that do not appear in $Q \setminus Q'$)
  – $u(H') = u(Q')$, where $u$ is a substitution being the identity on constants and assigning to two terms the same image if and only if their are in the same class in $P_u$.

1. Show that if $\rho_2$ depends on $\rho_1$, there is a piece unifier of the body of $\rho_2$ with $\rho_1$
2. Provide an example showing that the converse is not true.
3. Refine the notion of piece unifier to ensure that the converse holds.


3 Implementing the aGRD Check

DLGP is a textual format for the existential rules framework. It is described there: https://graphik-team.github.io/graal/doc/dlgp.html

Implement a piece of code with the following specifications:

• Input: a set of existential rules $R$, provided in the DLGP format
• Output: yes if and only if $R$ belongs to aGRD.

1. Design test cases.
2. Implement an aGRD checker – you can use the base classes from the Graal library: https://graphik-team.github.io/graal/doc/index


4 Rulesets as Query Languages

In this section, we consider rulesets as Boolean query languages, as follows: (i) the predicates can be taken either from $P_e$ or $P_i$; (ii) head atoms can only have predicates from $P_i$; (iii) there is a special nullary predicate $\text{goal} \in P_i$; (iv) the answer to a query $R$ on a database $D$ whose atoms are all of predicate $P_e$ is yes if and only if $D, R \models \text{goal}$.

1. Show that any query expressible by a w.a. ruleset is also expressible by full existential rules.
2. Show that any query expressible by an aGRD ruleset is also expressible by full existential rules.
3. Is any query expressible by aGRD also expressible by w.a.?
4. Is any query expressible by w.a. also expressible by aGRD?