KNOWLEDGE REPRESENTATION, REASONING AND DECLARATIVE PROBLEM SOLVING

Jicheng Zhao

Department of Computer Science and Engg.
Arizona State University
Tempe, AZ 85287
chitta@asu.edu

http://www.public.asu.edu/∼cbaral

Class 5: Stratification and Splitting in Calculating Answer sets

Stratify and Local Stratify

- Dependency graph:
  - Predicate dependency graph (PDG): Vertexes of the graph are all predicates in the program. There is a positive (or negative) edge from vertex p to q if there is a rule r in the program such that p is in the head of r and q is a positive (or negative) literal in the body of r.
  - Atom dependency graph (ADG): Vertexes of the graph are all atoms in the program. There is a positive (or negative) edge from vertex p to q if there is a rule r in the program such that p is in the head of r and q is a positive (or negative) literal in the body of r.
- Stratify and locally stratify:
  - Stratify: If the PDG of the program does not have a cycle with a negative edge, the the program is stratified.

- Local stratify: If the ADG of the program does not have a cycle with a negative edge, the the program is locally stratified.
- If a program is stratified, then it is locally stratified.

Stratification in Computing answer sets

- If a program is stratified or locally stratified, then its answer set can be computed in polynomial time.
- To compute its answer set, we can state from $\emptyset$, calculate using T_π^1 operator recursively until it is stable.
- **Example 1** *Given program P:*

$$\begin{aligned} anc(X,Y) &\leftarrow par(X,Y) \\ anc(X,Y) &\leftarrow par(X,Z), anc(Z,Y) \\ par(a,b) & \\ par(b,c) & \end{aligned} \tag{1}$$

This program is stratified. The answer set calculated will be $\{par(a,b), par(b,c), anc(a,b), anc(b,c), anc(a,c)\}$.

- **Example 2** *Given program Q:*

$$\begin{aligned} p(a) &\leftarrow q(b) \\ q(b) &\leftarrow p(c) \end{aligned} \tag{2}$$

  This program is not stratified, but it is locally stratified. Calculating T_π^1 on $\emptyset$, we can find out its answer set $\emptyset$.

- Locally stratified $AnsProlog$ has unique answer set.

Splitting

- Even a program is not stratified, it can still be computed efficiently.
- In the examples above, we indeed made use of the notion of "bottom" in a program.
- Bottom: A subset of atoms of program p is a bottom of p iff
- We can compute all the answer sets of the bottom and then spread to the rest part of the program.

Splitting in computing answer sets

- **Example 3** *Given program P:*

$$
\begin{aligned}
c &\leftarrow not\ d \\
a &\leftarrow not\ b, c \\
b &\leftarrow not\ a
\end{aligned}
\tag{3}
$$

*Rule (**??**) is the bottom of the program. The answer set of the bottom is $\{c\}$. We spread it to the rest part ("top") of the program. Further, the program P has two answer sets: $\{c, a\}$, and $\{c, b\}$.*