Exploring Relations between Answer Set Programs

Katsumi Inoue
National Institute of Informatics

Chiaki Sakama
Wakayama University

Symposium on Constructive Mathematics in Computer Science

October 25th, 2010

Relations between Answer Set Programs

- **LP**: the class of all logic programs
- Relation ☺ $\subseteq$ **LP** $\times$ **LP**
- P1, P2 $\in$ **LP** : logic programs
- P1☺P2 : comparison, correspondence, entailment, etc.
  - Equivalence/Non-equivalence
  - Inclusion/Non-inclusion/Disjoint
  - Generality/Specificity and Abstraction/Refinement
  - Preferred/Non-preferred and Higher/Lower Priority
  - Strength/Weakness and Better/Less Valued
  - More/Less Informativeness or Supportedness

Ordering Logic Programs

- ⌘ Equivalence of logic programs
  - ⌘ optimization, debugging, simplification, verification
  - ⌘ ordinary (weak)/strong/uniform/relativized/projected
- ⌘ Direction to refine/abstract programs (minimal revision)
  - ⌘ Inductive Logic Programming (ILP), Ontology
- ⌘ Synthesis of a common generalized/specialized theory from different sources of information [Sakama & Inoue, 2006,2007,2008]
  - ⌘ Multi-Agent Systems (MAS)

Comparing Nonmonotonic Programs

⌘ **Example**:

$$P1 = \{\ p \leftarrow \textit{not}\ \ q\ \}.$$

$$P2 = \{\ p \leftarrow \textit{not}\ \ q, \quad q \leftarrow \textit{not}\ \ p\ \}.$$

P1 has the single answer set: {p}

P2 has two answer sets: {p}, {q}

⌘ P1 is more informative than P2 in the sense that P1 has the **skeptical** consequences {p} which includes {}.

⌘ P2 is more informative than P1 in the sense that P2 has the **credulous** consequences {p,q} which includes {p}.

Generality Relations [Inoue & Sakama, 2006]

- ⌘ Smyth (#) and Hoare (♭) orderings between sets of answer sets.
- ⌘ Both #- and ♭-generalities are defined in a way that weakly equivalent programs belong to the same equivalence class induced by these orderings.
- ⌘ Both minimal upper and maximal lower bounds can be defined for any pair of programs in these generality orderings.
- ⌘ #-general programs entail more skeptical consequences, while ♭- general programs entail more credulous consequences.
- ⌘ Both strong #- and strong ♭-generalities are defined in a way that strongly equivalent programs belong to the same equivalence class induced by these orderings.
- ⌘ The proposed orderings can be applied to the class of extended disjunctive programs (EDPs).

Extensions (in the paper)

- General framework to compare semantic structures—comparing composite sets, which are sets of literal sets, under any pre-order between literal sets.
- Application of the comparison framework to any class of logic programs including nested programs, programs with aggregates, and any propositional formulas as well as under other semantics.
- Variations of generality relations — strong/uniform/relativized equivalence/inclusion/generality
  - comparison of generality with respect to contexts—robustness
  - satisfaction of classical inductive generality
  - application to formalizing abductive generality

Ordering on Powersets

⌘ *pre-order* $\leq$: binary relation which is reflexive and transitive

⌘ *partial order* $\leq$: pre-order which is also anti-symmetric

⌘ $\langle D, \leq \rangle$: *pre-ordered set* / *poset*

⌘ $\mathbf{P}(D)$: the power set of D

⌘ The **Smyth order**: for $X, Y \in \mathbf{P}(D)$,

$$X \models^{\#} Y \text{ iff } \forall x \in X \, \exists y \in Y. \; y \leq x$$

⌘ The **Hoare order**: for $X, Y \in \mathbf{P}(D)$,

$$X \models^{\flat} Y \text{ iff } \forall y \in Y \, \exists x \in X. \; y \leq x$$

⌘ Both $\langle \mathbf{P}(D), \models^{\#} \rangle$ and $\langle \mathbf{P}(D), \models^{\flat} \rangle$ are pre-ordered sets.

Ordering over the sets of literal sets

⌘ *Lit* : the set of all ground literals in the language

⌘ composite set Σ: $\Sigma \in \mathbf{P}(\mathbf{P}(Lit))$

⌘ A composite set represents a semantic structure of a logic program *P*, e.g., the answer sets (stable models), the supported models, the possible models, the minimal models, the classical models, the preferred answer sets of *P*.

⌘ The **Smyth order** $\Sigma 1 \models^{\#} \Sigma 2$ and the **Hoare order** $\Sigma 1 \models^{\flat} \Sigma 2$ can be defined for any two composite sets $\Sigma 1$, $\Sigma 2$.

⌘ If $\Sigma 1$ and $\Sigma 2$ represent the semantics of *P1* and *P2*, the Smyth/Hoare ordering gives generality relation between them.

General Extended Disjunctive Programs (or simply Programs)

⌘ Rules:

$L_1 ; \ldots ; L_k ; \textbf{not } L_{k+1} ; \ldots ; \textbf{not } L_l$

$\leftarrow L_{l+1} , \ldots, L_m , \textbf{not } L_{m+1} , \ldots, \textbf{not } L_n$

- ***not*** : negation as failure, L_i : literal, $0 \leqq k \leqq l \leqq m \leqq n$
- **Extended disjunctive program (EDP):** $k=l$.
- **Extended logic program (ELP):** $k=l \leqq 1$.
- **Positive disjunctive program (PDP):**

  EDP & $\forall L_i$: atom & $m=n$.
- **Normal logic program (NLP):** ELP & $\forall L_i$: atom.

Ordering Logic Programs

⌘ **LP**: the class of all programs

⌘ $A(P)$, $A(Q)$: the answer sets of $P, Q \in$ **LP**

- P is **more #-general than** Q :

$$P \models^{\#} Q \quad \text{iff} \quad A(P) \models^{\#} A(Q)$$

- P is **more ♭-general than** Q :

$$P \models^{\flat} Q \quad \text{iff} \quad A(P) \models^{\flat} A(Q)$$

⌘**<u>Theorem</u>**:

(1) $P \models^{\#} Q$ and $Q \models^{\#} P$ iff $min(A(P))=min(A(Q))$.

(2) $P \models^{\flat} Q$ and $Q \models^{\flat} P$ iff $max(A(P))=max(A(Q))$.

In case of EDP, $P \models^{\#/\flat} Q$ and $Q \models^{\#/\flat} P$ iff $A(P)=A(Q)$ (i.e. P≡Q).

Ordering Logic Programs

⌘ **Example**:

P1 = { p ← ***not*** q }.

P2 = { p ← ***not*** q, q ← ***not*** p }.

P3 = { p;q ← }.

P4 = { p ← ***not*** ¬p, q ← p }.

$\mathcal{A}$(P1) = {{p}}, $\mathcal{A}$(P2) = $\mathcal{A}$(P3) = {{p}, {q}}, $\mathcal{A}$(P4) = {{p,q}}

- P4 $\models^{\#}$ P1 $\models^{\#}$ P2
- P4 $\models^{\flat}$ P2 $\models^{\flat}$ P1
- P2 $\models^{\#}$ P3 $\models^{\#}$ P2, P2 $\models^{\flat}$ P3 $\models^{\flat}$ P2 (i.e., P2≡P3)

Skeptically / Credulously Entailed Literals in More / Less General Programs

⌘ *Skp*(*P*) : the set of skeptical consequences of *P*

⌘ *Crd*(*P*) : the set of credulous consequences of *P*

⌘ **Theorem**:

- If $P \models^{\#} Q$ then $Skp(P) \supseteq Skp(Q)$.
- If $P \models^{\flat} Q$ then $Crd(P) \supseteq Crd(Q)$.

Strong / Uniform / Relativized Generality

⌘ $P, Q \in \mathbf{LP}$, $C \subseteq \mathbf{LP}$

- P is **strongly / uniformly more #-general than** Q:
  $P \trianglerighteq^{\#} Q$ iff $P \cup R \models^{\#} Q \cup R$ for any $R \in \mathbf{LP}$ / $R \in \mathbf{P}(Lit)$.
- P is **strongly / uniformly more ♭-general than** Q:
  $P \trianglerighteq^{\flat} Q$ iff $P \cup R \models^{\flat} Q \cup R$ for any $R \in \mathbf{LP}$ / $R \in \mathbf{P}(Lit)$.

⌘ $P \trianglerighteq^{\#/\flat} Q$ implies $P \models^{\#/\flat} Q$.

⌘ **Relativized #/♭** -generality requests the above for any $R \in C$

⌘ Strong generality is equivalent to **strong inclusion** [Eiter *et al.*, 2005] (for finite programs).

⌘ Uniform generality for PDPs reduces to **classical inductive generality**, i.e., logical entailment.

Abductive Generality

- ⌘ **Explainable generality** [Inoue & Sakama, 2008] can be characterized by ♭-generality. That is, the ability to explain more observations corresponds to derive more credulous consequences when abducibles are represented as choice constructs:

  L ; ***not*** $L \leftarrow$ for any abducible literal L.

- ⌘ **Explanatory generality** [Inoue & Sakama, 2008] can be characterized by relativized ♭-generality, in which the set of abducibles is set to be the context for comparison.

- ⌘ Taking minimal upper/maximal lower bounds of two abductive programs can be applied to coordination of multiple abductive agents.

Conclusion: Generality as Comparison between Programs

- **Program Verification**: testing and debugging with respect to specification of a program (equivalence)
- **Program Development**: testing if a refined theory is more detailed from the previous theory (generality)
- **Problem Solving**: assessment of relative value of each theory/ontology (generality, preference)
- **Machine Learning**: criteria to compose better theories (generality, preference)
- **Multiagent Systems**: criteria to compare different information sources (informativeness, preference)