Candidatura PEx - Zugzwang

2024-01-05 - Next Research Lines

> After the base-setting work of "*An Algebraic Approach to Stochastic ASP*" these are the next tasks to consider. Is summary:
>
> 1. **Logic Programming** - Stratified & Non-stratified programs
> 2. **Computer Science** - Inductive Logic Programming
> 3. **Software** - Integration with Potassco and other frameworks
> 4. **Applications**

Line 1: Logic Programming - Stratified & Non-stratified programs

Line 1a - Logic Programs Structure and Properties

> *Stratified & non-stratified* programs are quoted in the "CREDAL" papers as important classes of logic programs.

Minimal example of a **non-stratified program**.

The following annotated LP, with clauses c_1, c_2, c_3 respectively, is non-stratified (because has a cycle with negated arcs) but no head is disjunctive:

```
0.3::a.                 % c1
b :- not c, not a.      % c2
c :- not b.             % c3
```

This program has three stable models:

$$
\begin{aligned}
m_1 &= \{ a, c \} \\
m_2 &= \{ \neg a, b \} \\
m_3 &= \{ \neg a, c \}
\end{aligned}
$$

> We should **investigate** *What are stratified programs and why are they important?* and how does our approach deals with such programs?

Line 1b - Investigate the expressiveness of PASP

Consider:

- Recursion
- Variables,
- Functional symbols,

Line 1c - The equivalence relation

Consider the cases where only $s \subseteq e$ and $e \subseteq s$. Or other refinements. Also consider the inconsistent and independent events.

Line 1d - Stability of the error function

Consider alternative error functions. See statistics, Kullback-Leibler divergence

Line 2: Computer Science - Inductive Logic Programming

> Proceed from scoring programs to support genetic algorithms or other program space exploration methods.

Scoring programs, as described in our paper, is just a step into **Inductive Logic Programming**. To go further, we need to explore algorithms that:

1. Use **background knowledge**, expressed as a PLP.
2. Consult **positive examples** that should be *soft* induced.
3. Consult **negative examples** that should be *soft* excluded.
4. Generate **PLPs** that are scored.
5. Recombine the **best scored** into a new *population*, using recombination rules.

> In order to do that, **PLPs must be expressed as data structures** to be manipulated. Also **recombination rules** must investigated before become formally expressed and supported with adequate methods.

Line 3: Software - Integration with Potassco and other frameworks

> Support annotated programs with zugzwang semantics.

- Bayesian Networks (BII Alice)
  - Generate an annotated asp program from a bayesian network and run it trough `clingo`.
  - Recover the stable models from the previous ste and compute the respective probabilities.
- Program Manipulation
  - Annotated ASP program *representation* and a *parser*.

Line 4: Applications

> Apply zugzwang to a few showcases, besides the theoretic corner stones (non-stratified, disjunctive, bayes networks), preferably based in real world scenarios, with complex structure and large datasets.

- (Stochastic) Plan Generation
- Yale-Shooting Problem
- (Stochastic) Situation Calculus
- Frame Problem
- Latent Facts - and core assumptions.
- Given a **Bayesian Network** (or a **Markov Networks**):
  - Represent it. (**done** for BNs; MNs?)
  - Solve the common probability tasks: join (**done**), marginals, conditionals, parameter learning, inferring unobserved variables, sample generation, *etc.*
- Given a *solved* ASP specification:
  - What is the marginal probability of the atom `a` ? (**done**)
  - What other probability queries are important to consider?
- Given an *unsolved* ASP specification:
  - What is the probability (distribution?) of the probabilistic fact `a` ?
  - What other questions are relevant? *E.g.* the distribution family of a fact?
- Given a *solved* ASP specification and a set of *samples*:
  - How do the probabilities inferred from the specification match the ones from the empiric distribution? (**done** might see alternative approaches)
- Given two *solved* ASP specification and a set of *samples*:
  - Which specification best describes the empiric distribution? (**done**)