Stratification for Variants of Default Logic

Jörg Ernst
Computer Science Institute, Kiel University, Preußerstr. 1-9, 24105 Kiel, Germany,
je@informatik.uni-kiel.de
Grigoris Antoniou
CIT, Griffith University, QLD 4111, Australia,
ga@cit.gu.edu.au

Default logic (DL) (Reiter 1980) is one of the most prominent approaches to nonmonotonic reasoning. One of the main problems with its applicability is that it is computationally harder than classical logic (Gottlob 1992). (Cholewinski 1994,1995) introduced and studied stratification of default theories to increase the efficiency of default reasoning. The idea is to to split the knowledge into smaller parts, and to apply reasoning in a local way. This paper shows how stratification can work for some important variants of Default Logic. These variants are Justified Default Logic (JDL), Rational Default Logic (RDL), and Constrained Default Logic (CDL).

The main idea of stratification is that if a default makes use of information from another default, then we should seek to apply it first. Let $T = (W, D)$ be a Default Theory, where W is an empty set of formulae and D the set of the two defaults $\delta = \frac{true:p}{p}$ and $\delta' = \frac{p:q}{q}$. In this case we should apply δ before δ' because δ' is not applicable to $Th(W)$[1].

A function ρ assigning a natural number to every default from D is called a *stratification function* iff for any $\delta, \delta' \in D$ the condition " If $Prop(cons(\delta)) \cap Prop(\delta') \neq \emptyset$ then $\rho(\delta) \leq \rho(\delta')$ (1)[2]" holds.

By assigning a number to each default, ρ decomposes D into layers (strata) $D_1, \ldots, D_k$ of ascending value under ρ. A default theory $T = (W, D)$ is called *stratified* iff W is consisent, $Prop(W) \cap Prop(cons(D) \cup just(D)) = \emptyset$, and there exists a stratification function for D. (Cholewinski 1994) shows that the decomposition of a set of defaults according to a stratification function preserves the extensions.

For stratification to work properly for default logic variants, we need to add new conditions for stratification functions. *Justified Default Logic* avoids running into failure by retracting a final, "fatal" step. The condition we need to add in order that stratification works properly is the following: If $Prop(just(\delta)) \cap Prop(cons(\delta')) \neq \emptyset$ then $\rho(\delta) \leq \rho(\delta')$ (2).

Rational Default Logic requires joint consistency of justifications of defaults contributing to an extension. Obviously there is a new interaction among defaults, and we need the following condition on ρ in order for stratification to work correctly: If $Prop(just(\delta)) \cap Prop(just(\delta')) \neq \emptyset$ then $\rho(\delta) = \rho(\delta')$ (3).

Finally *Constrained Default Logic* combines the properties of JDL and RDL. Thus we need all three conditions (1) – (3) in order that stratification works properly. For CDL, the correctness theorem of stratification looks as follows:

Theorem 1 *Let ρ be a stratification function for a finite set of defaults D which satisfies conditions* (1) – (3)*, and $D_1 \cup \ldots \cup D_l$ the decomposition of D according to ρ. Let W be a consistent set of formulae such that $Prop(W) \cap Prop(cons(D) \cup just(D)) = \emptyset$. Then (E, C) is an extension of $T = (W, D)$ in CDL, iff for all $0 \leq i \leq l$ there exists (E_i, C_i) such that $(E_l, C_l) = (E, C)$, $(E_0, C_0) = (Th(W), Th(W))$, and for all $1 \leq i \leq l$, (E_i, C_i) is an extension of the preconstrained default theory $T_i = (E_{i-1}, D_i, C_{i-1})$ in CDL.*

[1] where $Th(W)$ denotes the deductive closure of W.

[2] where $Prop(\delta)$ denotes the set of all propositional atoms occurring in δ.

References

Cholewinski, P. 1994. Stratified default logic. In *Computer Science Logic*, 456-470. Springer LNCS 933.

Cholewinski, P. 1995. Seminormal stratified default theories. *Annals of Math. and Art. Intel.*

Gottlob, G. 1992. Complexity results for nonmonotonic logics. *J. Logic and Computation* 2: 397-425.

Reiter, R. 1980. A logic for default reasoning. *Artificial Intelligence* 13: 81-132.