Lecture Notes 7/8 – Nonmonotonic Reasoning and Circumscription

Patrick Doherty

February 7, 1998

1 Outline

1. Abnormality Theories
2. Model Preference Logics.
3. Circumscription.
   (a) Parallel Predicate Circumscription.
   (b) Prioritized Circumscription.
4. Model Preference Logics Con't.

2 Introduction

The purpose of this lecture is to introduce you to formal techniques which can be used to minimize the extensions of predicates. In the previous lecture, we considered the Monkey and Bananas problem and in an incremental manner modified the original axiomatization to make it more flexible and epistemologically adequate. Both the frame and qualification problems were introduced in this context and we showed how one could provide a potential solution to these problems *if* one could somehow augment FOPC with a tool enabling the restriction of extensions of various predicates to just those members of the extension that were provable from the axiomatization. In this lecture, we will present both a semantic and syntactic characterization of such a technique. We start by introducing abnormality theories [?], using an example similar to the Monkey and Bananas problem we considered previously. We then discuss model preference logics and their syntactic characterization in terms of circumscription. We introduce two types of circumscription and their corresponding preference semantics and conclude with a discussion. The tools and concepts discussed during this lecture will be used continually throughout the rest of the course in our investigation of the reasoning methods for action and change. There importance can not be underestimated.

3 Abnormality Theories

McCarthy [?] presents a "solution" to the frame problem along the lines sketched in the previous lecture. Rather than modify the Monkey and Bananas axiomatization, we will present a reified version of McCarthy's example in terms of an abnormality theory. In the following sections, we will consider formal techniques for reasoning with abnormality theories.

Laws of Motion

$$\forall x, l, s. \neg Ab(Aspect3(x), l, s) \rightarrow \\ Holds(At(x, l), Result(Move(x, l), s)) \tag{1}$$

$$\forall x, c, s. \neg Ab(Aspect4(x), c, s) \rightarrow \\ Holds(Color(x, c), Result(Paint(x, c), s)) \tag{2}$$

Frame Axioms

$$\forall x, e, s, l. Holds(At(x, l), s) \wedge \neg Ab(Aspect1(x), e, s) \rightarrow \\ Holds(At(x, l), Result(e, s)) \tag{3}$$

$$\forall x, e, s. Holds(Color(x, c), s) \wedge \neg Ab(Aspect2(x), e, s) \rightarrow \\ Holds(Color(x, c), Result(e, s)) \tag{4}$$

$$\forall x, l, s. Ab(Aspect1(x), Move(x, l), s) \tag{5}$$

$$\forall x, c, s. Ab(Aspect2(x), Paint(x, c), s) \tag{6}$$

Qualification Axioms

$$\forall x, l, s. \neg Holds(Clear(Top(x)), s) \vee \neg Holds(Clear(l), s) \\ \vee Holds(l = Top(x), s) \rightarrow Ab(Aspect3(x), Move(x, l), s) \tag{7}$$

State Constraints

$$\forall l, s. Holds(Clear(l), s) \equiv \neg \exists x. \neg Holds(Trivial(x), s) \wedge Holds(At(x, l), s) \tag{8}$$

$$\forall x, s. \neg Ab(Aspect7(x), s) \rightarrow \neg Holds(Trivial(x), s) \tag{9}$$

Every object has a number of aspects. Generally, an object is normal in most aspects, but quite often abnormal in some. The abnormality predicate Ab allows us to represent this notion. Ab takes three arguments, an aspect of the entity involved, the event or entity the aspect is abnormal to, and finally a situation. So for example, axiom 5 states that $aspect1$ of object x is abnormal relative to the action of moving x to location l in situation s. Axiom 6 states that $aspect2$ of an object is abnormal relative to the action of painting. Based on the present axiomatization, we would like to assume that any object is normal relative to all aspects with the exception of aspect 1 and 2, and sometimes 3.

By minimizing the extension of Ab, we in effect achieve a maximization of normal aspects. Why is this? Well, any aspect not shown to be abnormal can be assumed not to be, $\neg Ab(aspect_z(x), e, s)$. These conjectures can be derived in our modified axiomatization and used *as if* we had derived them in the standard way. Of course, conjecturing $\neg Ab(aspect_z(x), e, s)$ statements is not the same as deriving them once and for all. The distinction is that if such formulas were derived via a standard proof theory, adding additional axioms to the knowledge base would not affect their status as consequences of the axiomatization. On the other hand, conjectures, and any statement derived through the use of conjectures, may not be derivable if new statements are added to the axiomatization.

4 Model Preference Logics

Before considering circumscription and its many varieties, we will first take a look at model preference logics, of which circumscription is a syntactic characterization.[1]

Model preference logics are based on specifying criteria for choosing a preferred set of models from the models of a theory. In the case above, the preferred set of models for our axiomatization of the blocks world would be the models where the extension of Ab is as small as possible. If we have two models, M and M', and the extension of Ab in M is a subset of its extension in M', then M would be preferred to M'. Suppose the axiomatization only had two models and M was preferred. Generally, more consequences are true in M then in both M and M' together. The extra consequences are essentially conjectures. Why is this? If new information is added to the axiomatization, it may be the case that although M and M' are still models of the axiomatization, none is preferred to the other. In this case, the conjectures which were previously made, would have to be retracted. This is why our consequence relation is said to be nonmonotonic.

Let's be a little more specific. Suppose we have an axiomatization of a particular blocks world scenario in a room. The room contains three blocks. Our language contains three constant symbols $b1$, $b2$, and $b3$, and several predicate symbols including $Block$. Suppose the axiomatization Γ contains, among others, the three axioms: $Block(b1)$, $Block(b2)$, $Block(b3)$ and the axiom

$$b1 \neq b2 \wedge b1 \neq b3 \wedge b2 \neq b3 \tag{10}$$

stating the constants denote different blocks. Axiom 10 is called the *unique names axiom*. We would like to represent the fact that the only objects that are blocks are just the objects named by $b1$, $b2$, and $b3$. One way to do this would be to add the following axiom to the axiomatization:

$$\forall x.Block(x) \rightarrow x = b1 \vee x = b2 \vee x = b3 \tag{11}$$

This would work fine if the number of blocks were finite and there was no other way of deriving $Block$ statements from the rest of Γ, but this is generally not the case. What we would like to do is to find a way to conjecture 11. One way to do this is to choose the models of Γ that contain the smallest set of blocks. So far, every model of Γ contains *at least* three blocks, one for each of the constants, but there are other models containing more blocks. If we let $M(\Gamma)_\prec$ contain the preferred (least blocks) models, then

$$M(\Gamma)_\prec \models \forall x.Block(x) \rightarrow x = b1 \vee x = b2 \vee x = b3 \tag{12}$$

which is precisely what we desire. Note that if new statements are added to Γ which allow us to derive the existence of one or more new blocks, then (12) is no longer a consequence of Γ, although the new set of preferred models would permit us to conjecture a new least set of blocks.

5 Circumscription

5.1 Parallel Predicate Circumscription

There are a number of versions of circumscription. We will start by considering parallel predicate circumscription. Let $\mathcal{L}$ be our logical language, P a set of predicate symbols we would like to minimize, and Z another set of symbols(predicate and/or function symbols), which will be allowed to vary across models. $\Gamma(P, Z)$ denotes a first-order sentence containing symbols P and Z.

[1] The following section loosely follows the tutorial in [**?**].

5.1.1 Semantic Characterization

Given $\Gamma(P, Z)$ we would like to define an ordering on models of $\Gamma(P, Z)$ where those that are minimal relative to the ordering are just the models with minimal extensions for those predicates in P. These models are assumed to have the same interpretation for all predicates and functions not in P or Z.

Let M_1 and M_2 be two models of $\Gamma(P, Z)$, where D_{M_1} and D_{M_2} are the domains of discourse for M_1 and M_2, respectively.

Definition 5.1 (Minimal Models)
We say that the model M_1 is P-minimal to M_2, written as $M_1 \preceq_{P;Z} M_2$, iff

1. $D_{M_1} = D_{M_2}$.
2. $Value(U, M_1) = Value(U, M_2)$ *for all* U *not in* P *or* Z.
3. $Value(U, M_1) \subseteq Value(U, M_2)$ *for all* $P_i \in P$.

The strict ordering, $\prec_{P;Z}$, on models is defined as

$$M_1 \preceq_{P;Z} M_2 \text{ and not } M_2 \preceq_{P;Z} M1.$$

The P-preferred models are those minimal relative to $\prec_{P;Z}$.

What does this definition tell us? Points one and two state that two models can only be compared if their domains are the same and the extensions of fixed predicates are the same. The third point states that the extension of members of P in M_1 are subsets of their counterparts in M_2.

Note that in order to use this definition that predicates and functions in our axiomatization must be classified into three sets:

1. Those that we want to minimize, P (only predicates),
2. those that we want to vary, Z (predicates and functions),
3. and those that we want to keep fixed.

Each combination represents a different minimization policy. Let's look at an example. The signature for L is $\Lambda = \langle\{Tweety, Fred\}, \{\}, \{Bird, Ab, Flies\}\rangle$. Suppose

$$\begin{aligned}
&\Gamma(Ab, Flies) = \\
&\quad \{\forall x. Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x), \\
&\quad\quad Bird(Tweety) \\
&\quad\quad \forall x. x = Tweety \vee x = Fred \\
&\quad\quad Tweety \neq Fred\}
\end{aligned} \tag{13}$$

Our minimization policy will be to minimize Ab and vary $Flies$. All other predicates and functions are fixed. $Tweety \neq Fred$ means that any model for $\Gamma(Ab, Flies)$ will have at least two entities in the domain of discourse. $\forall x. x = Tweety \vee x = Fred$ means that there are at most two entities in any model for $\Gamma(Ab, Flies)$. The following table describes the models for $\Gamma(Ab, Flies)$. For any two objects, a table isomorphic to the table shown is also a model.

	$Tweety_I$	$Fred_I$	$Bird_I$	Ab_I	$Flies_I$
1	$\star$	$\bullet$	$\star$	$\star$	$\star$
2	$\star$	$\bullet$	$\star$	$\star$	$\star, \bullet$
3	$\star$	$\bullet$	$\star$	$\star$	$-$
4	$\star$	$\bullet$	$\star$	$-$	$\star$
5	$\star$	$\bullet$	$\star$	$-$	$\star, \bullet$
6	$\star$	$\bullet$	$\star$	$\bullet$	$\star$
7	$\star$	$\bullet$	$\star$	$\bullet$	$\star, \bullet$
8	$\star$	$\bullet$	$\star, \bullet$	$\star$	$\bullet$
9	$\star$	$\bullet$	$\star, \bullet$	$\star$	$\star, \bullet$
10	$\star$	$\bullet$	$\star, \bullet$	$\bullet$	$\star$
11	$\star$	$\bullet$	$\star, \bullet$	$\bullet$	$\star, \bullet$
12	$\star$	$\bullet$	$\star, \bullet$	$\star, \bullet$	$\star$
13	$\star$	$\bullet$	$\star, \bullet$	$\star, \bullet$	$\bullet$
14	$\star$	$\bullet$	$\star, \bullet$	$\star, \bullet$	$\star, \bullet$
15	$\star$	$\bullet$	$\star, \bullet$	$\star, \bullet$	$-$
16	$\star$	$\bullet$	$\star, \bullet$	$-$	$\star, \bullet$

It is easy to see that the preferred models of $\Gamma(Ab, Flies)$ are 4,5, and 16. Let $Mod_{\prec}(\Gamma(Ab, Flies))$ denote the Ab-preferred models of $\Gamma(Ab, Flies)$. It is easy to see that the following formulas are preferentially entailed:

$$Mod_{\prec}(\Gamma(Ab, Flies)) \models \neg Ab(Tweety), Flies(Tweety).$$

From the example, one can see why we must vary certain predicates. If our minimization policy minimized Ab without varying $Flies$ then the Ab-preferred models would be 4,5,8,10,15, and 16. In this case, we could not conjecture any new positive or negative atoms. The original form of circumscription did not allow varying predicates. The nonmonotonic behavior of the preferred models approach may be seen by adding $Ab(Tweety)$ to $\Gamma(Ab, Flies)$. The preferred models are then 1,2,3,8, and 9. It is clear that we can no longer entail $Flies(Tweety)$.

Note that to work with the preferred models approach, we first decide on an ordering among models to capture a particular feature of the phenomena we are investigating. We then choose a minimization policy relative to the signature of our language. Finally, we determine the preferred models relative to the policy and then use these as if they were the only models for our original premise set. In this particular example, determining the preferred models was still manageable due to the strong constraints built into the axiomatization. In the general case, there are far to many models to deal with directly. It would be nice if one could find a syntactic characterization of preferential entailment that could be used as the basis of a proof theory to determine preferential entailments. This of course, is an active area of research and a difficult one at that. In the next section, we will consider the original proposal of McCarthy which augments a premise set with additional premises in the form of first-order schemas or second-order formulas. The idea is that the augmented premise set will only have the preferred models as models.

5.1.2 Syntactic Characterization

Before considering syntactic characterizations of preferential entailment, we will introduce some convenient notation for describing circumscription axioms. Let U and V stand for predicate expressions of the same arity and Γ stand for a theory consisting of a finite set of first- or second-order formulas. We make no distinction between the theory Γ and the conjunction of all formulas in the theory.

Suppose we are given a theory Γ. We would like to find a syntactic characterization of the models of Γ that are minimal relative to the predicate U. In other words, given any two models of Γ, we will always prefer the one where U's extension is smaller. Let's first consider FOPC formulas in terms of comparing extensions of predicates setwise. Given a model M and two predicates U and V of the same arity, we denote their extensions in M by U_M and V_M, respectively.

- If $U_M \subseteq V_M$ then $Value(\forall x.U(x) \rightarrow V(x), M) = 1$.
- If $U_M \subset V_M$ then $Value(\forall x.U(x) \rightarrow V(x) \wedge \neg \forall x.V(x) \rightarrow U(x), M) = 1$.
- If $U_M \equiv V_M$ then $Value(\forall x.U(x) \rightarrow V(x) \wedge \forall x.V(x) \rightarrow U(x), M) = 1$.

Let U and V be predicates of the same arity. We will use the following abbreviations:

- $U \leq V$ stands for $\forall x.U(x) \rightarrow V(x)$.
- $U < V$ stands for $\forall x.U(x) \rightarrow V(x) \wedge \neg \forall x.V(x) \rightarrow U(x)$.
- $U = V$ stands for $\forall x.U(x) \rightarrow V(x) \wedge \forall x.V(x) \rightarrow U(x)$.

$U \leq V$ expresses the fact that the extension of U is a subset of the extension of V. $U < V$ expresses the fact that the extension of U is a strict subset of the extension of V. $U = V$ expresses the fact that the extension of U is the same as the extension of V. If $\overline{U} = \{U_1, \ldots, U_i\}$ and $\overline{V} = \{V_1, \ldots, V_i\}$ are similar tuples of predicate expressions, then $\overline{U} \leq \overline{V}$ is used as an abbreviation for $\bigwedge_{i=1}^{n} [U_i \leq V_i]$. $\overline{U} < \overline{V}$ is an abbreviation for $\overline{U} \leq \overline{V}$ and $\neg \overline{V} \leq \overline{U}$, $\overline{U} \equiv \overline{V}$ is an abbreviation for $\overline{U} \leq \overline{V}$ and $\overline{V} \leq \overline{U}$. For example, if $\overline{U} = \{P, Q\}$ and $\overline{V} = \{P', Q'\}$ then $\overline{U} \leq \overline{V}$ is equivalent to $P \leq P' \wedge Q \leq Q'$.

There is one other technical point to be mentioned. Suppose Γ contains the formula $\forall x.\neg Ab(x) \rightarrow Flies(x)$. If we were to minimize Ab, then the extension of $Flies$ would have to vary across models. In other words, the smaller the extension of Ab, the larger the extension of $Flies$. In order to prefer the model with the smaller extension of Ab, we would have to deal with the extension of $Flies$ as a kind of *don't care* extension. When comparing two models, the extensions of $Flies$ are not relevant to the comparison. It is important that certain predicates can vary. Without this, we would not be able to derive any new conjectures. In this instance, we would like to say that most things fly. Minimizing Ab maximizes $\neg Ab$ which in turn maximizes $Flies$ as desired.

Our task is to syntactically characterize the formulas that are minimally entailed by Γ. Circumscription does this by augmenting the premises in Γ with either a first-order schema or a second-order formula. The additional premises constrain the models of Γ to be just those that are minimal relative to a particular minimization policy. Given Γ, suppose we would like to minimize P and let Z vary. We denote the axiom set as $\Gamma(P, Z)$ and the variable circumscription of P in Γ with predicate variables Z as $Circ(\Gamma(P, Z); P; Z)$. The idea is that the class of formulas

$$\{\alpha \mid Circ(\Gamma(P, Z); P; Z) \models \alpha\} \tag{14}$$

is the class of formulas minimally entailed by Γ. In other words, those formulas true in all P-preferred models of Γ.

The next step is to determine what $Circ(\Gamma(P, Z); P; Z)$ should be. We already know it includes $\Gamma(P, Z)$ because all preferred models must satisfy the original axiom set. In addition, we would like to add a set of premises which together state the following:

> "Given $\Gamma(P, Z)$, it is not the case that there are extensions, P' and Z' for P and Z where if substituted for P and Z in Γ will both
> - satisfy what holds for P and Z in Γ,
>   and
> - at the same time the extension of P' will be smaller than that of P."

In other words, any substitution we make will guarantee that P has the smallest extension possible that still staisfies the constraints placed on P by Γ. Here is the second order formula characterizing the statement:

$$\Gamma(P, Z) \wedge \neg \exists P', Z'.[\Gamma(P', Z') \wedge P' < P]. \tag{15}$$

Note that $\Gamma(P', Z')$ denotes the axiomatization $\Gamma(P, Z)$ where P and Z are replaced by P' and Z', respectively. Note that predicate expressions such as P' and Z' are represented as lambda expressions $(\lambda\bar{x}.F(\bar{x}))$, where F is an arbitrary formula. Here is an example: $(\lambda x.Bird(x)\wedge\neg Ostrich(x))$. Let's do some expansion and manipulation of 15 to put it in an equivalent form which sheds further light on the circumscriptive process.

$$\Gamma(P, Z)\wedge\neg\exists P', Z'.[\Gamma(P', Z')\wedge(\forall\bar{x}.P'(\bar{x})\rightarrow P(\bar{x}))\wedge(\neg\forall\bar{x}.P(\bar{x})\rightarrow P'(\bar{x}))]$$

$$\Gamma(P, Z)\wedge\forall P', Z'.\neg[\Gamma(P', Z')\wedge(\forall\bar{x}.P'(\bar{x})\rightarrow P(\bar{x}))\wedge(\neg\forall\bar{x}.P(\bar{x})\rightarrow P'(\bar{x}))]$$

$$\Gamma(P, Z)\wedge\forall P', Z'.\neg\Gamma(P', Z')\vee(\neg\forall\bar{x}.P'(\bar{x})\rightarrow P(\bar{x}))\vee(\forall\bar{x}.P(\bar{x})\rightarrow P'(\bar{x}))$$

$$\Gamma(P, Z)\wedge\forall P', Z'.\Gamma(P', Z')\rightarrow((\forall\bar{x}.P'(\bar{x})\rightarrow P(\bar{x}))\rightarrow(\forall\bar{x}.P(\bar{x})\rightarrow P'(\bar{x})))$$

$$\Gamma(P, Z)\wedge\forall P', Z'.[\Gamma(P', Z')\wedge(\forall\bar{x}.P'(\bar{x})\rightarrow P(\bar{x}))]\rightarrow(\forall\bar{x}.P(\bar{x})\rightarrow P'(\bar{x})). \tag{16}$$

It's also straightforward to show that 16 is equivalent to

$$\Gamma(P, Z)\wedge\forall P', Z'.[\Gamma(P', Z')\wedge(\forall\bar{x}.P'(\bar{x})\rightarrow P(\bar{x}))]\rightarrow(\forall\bar{x}.P(\bar{x})\equiv P'(\bar{x})). \tag{17}$$

16 and 17 provide the key to understanding how circumscription works. What we are interested in doing is to find two predicate expressions P' and Z' that make the antecedents of the universally quantified sentence true. The first antecedent $\Gamma(P', Z')$ guarantees that everything that holds for P and Z in Γ also holds for the substituted predicate expressions P' and Z'. The second antecedent $\forall\bar{x}.P'(\bar{x})\rightarrow P(\bar{x})$ guarantees that only those objects proven to be in the extension of P belong to the extension of P'. If the two antecedents are true then by 17, P is in fact equivalent to the substituted predicate expression P'. Here is the formal definition

Definition 5.2 (Parallel Circumscription)
Let $\overline{P}$ be a tuple of distinct predicate constants, $\overline{S}$ be a tuple of distinct predicate constants and/or function constants disjoint from $\overline{P}$, and $\Gamma(\overline{P}, \overline{S})$, a theory. The parallel circumscription *of $\overline{P}$ in $\Gamma(\overline{P}, \overline{S})$ with variable $\overline{S}$, written $Circ(\Gamma; \overline{P}; \overline{S})$, is the sentence*

$$\Gamma(\overline{P}, \overline{S})\wedge\forall\overline{P}', \overline{S}'.\neg[\Gamma(\overline{P}, \overline{S})\wedge\overline{P}' < \overline{P}]$$

where $\overline{P}'$ and $\overline{S}'$ are tuples of variables similar to $\overline{P}$ and $\overline{S}$, respectively.

Let's look at the example from section 5.1.1 and apply the circumscriptive technique. We would like to minimize Ab, vary $Flies$, with $Bird$, $Tweety$ and $Fred$ remaining fixed. Our premise set is

$$\begin{array}{c} \Gamma(Ab, Flies) = \\ \{\forall x.Bird(x)\wedge\neg Ab(x)\rightarrow Flies(x), \\ Bird(Tweety) \\ \forall x.x = Tweety\vee x = Fred \\ Tweety \neq Fred\} \end{array} \tag{18}$$

$\Gamma(Ab', Flies')$ is

$$\begin{array}{c} \Gamma(Ab', Flies') = \\ \{\forall x.Bird(x)\wedge\neg Ab'(x)\rightarrow Flies'(x), \\ Bird(Tweety) \\ \forall x.x = Tweety\vee x = Fred \\ Tweety \neq Fred\} \end{array} \tag{19}$$

$Circ(\Gamma(Ab, Flies); Ab; Flies)$ is

$$\Gamma(Ab, Flies) \wedge \forall Ab', Flies'.[\Gamma(Ab', Flies') \wedge \forall \bar{x}.Ab'(\bar{x}) \rightarrow Ab(\bar{x})] \rightarrow \forall \bar{x}.Ab(\bar{x}) \equiv Ab'(\bar{x}). \tag{20}$$

Our next task is to find two suitable predicate expressions to substitute for Ab' and $Flies'$. Using some wishful thinking, we can make the following educated guess:

$$Ab'(x) = (\lambda x.\bot),$$

where $\bot$ denotes falsity, and

$$Flies'(x) = (\lambda x.Bird(x)).$$

Next, we must show using $\Gamma(Ab, Flies)$ that the antecedents $\Gamma(Ab', Flies')$ and $\forall \bar{x}.Ab'(\bar{x}) \rightarrow Ab(\bar{x})$ in the universal formula are true. Let's start with $\Gamma(Ab', Flies')$ by substituting our choice for Ab' and $Flies'$.

$$\begin{aligned} &\Gamma(Ab', Flies') = \\ &\quad \{\forall x.Bird(x) \wedge \neg\bot \rightarrow Bird(x), \\ &\quad Bird(Tweety) \\ &\quad \forall x.x = Tweety \vee x = Fred \\ &\quad Tweety \neq Fred\} \end{aligned} \tag{21}$$

It is easy to see that all premises in $\Gamma(Ab', Flies')$ are true. Substituting our choice of Ab' into the second antecedent $\forall \bar{x}.Ab'(\bar{x}) \rightarrow Ab(\bar{x})$ results in $\forall \bar{x}.\bot \rightarrow Ab(\bar{x})$, which again is always true. Since we have shown that the antcedents in the universal formula are true, we may derive $\forall \bar{x}.Ab(\bar{x}) \equiv Ab'(\bar{x})$ using our choice of Ab'. Substituting in gives $\forall \bar{x}.Ab(\bar{x}) \equiv \bot$ which is equivalent to $\forall \bar{x}.\neg Ab(\bar{x})$. We can now use this conjecture to derive $\neg Ab(Tweety)$ which may then be used to derive that $Flies(Tweety)$, using the fact that $Bird(Tweety)$ is one of the premises.

Note that circumscription is a monotonic means of doing nonmonotonic reasoning. We derive statements from the augmented premise set using deduction. The nonmonotonic nature of circumscription is due to the fact that when a new premise is added to the premise set, the circumscription axiom used to augment the premise set is different from the previous axiom, but the actual reasoning is still done within the classical monotonic framework. This can be made clearer with an example. Suppose I add $Ab(Tweety)$ to my original premise set.

Our new premise set is

$$\begin{aligned} &\Gamma(Ab, Flies) = \\ &\quad \{\forall x.Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x), \\ &\quad Bird(Tweety) \\ &\quad Ab(Tweety) \\ &\quad \forall x.x = Tweety \vee x = Fred \\ &\quad Tweety \neq Fred\} \end{aligned} \tag{22}$$

$\Gamma(Ab', Flies')$ is

$$\begin{aligned} &\Gamma(Ab', Flies') = \\ &\quad \{\forall x.Bird(x) \wedge \neg Ab'(x) \rightarrow Flies'(x), \\ &\quad Bird(Tweety) \\ &\quad Ab'(Tweety) \\ &\quad \forall x.x = Tweety \vee x = Fred \\ &\quad Tweety \neq Fred\} \end{aligned} \tag{23}$$

$Circ(\Gamma(Ab, Flies); Ab; Flies)$ is

$$\begin{aligned}\Gamma(Ab, Flies)\wedge\forall Ab', Flies'.[\Gamma(Ab', Flies')\wedge\forall\bar{x}.Ab'(\bar{x})\rightarrow Ab(\bar{x})]\rightarrow \\ \forall\bar{x}.Ab(\bar{x}) \equiv Ab'(\bar{x}). \qquad (24)\end{aligned}$$

Again, we have to find two suitable predicate expressions to substitute for Ab' and $Flies'$. Again, using some wishful thinking, we can make the following educated guess:

$$Ab'(x) = (\lambda x.x = Tweety),$$

where $\perp$ denotes falsity, and

$$Flies'(x) = (\lambda x.Bird(x)\wedge x \neq Tweety).$$

Substituting in as before, we can show that the antecedents of the universal formula are true. In addition, we can no longer derive that $Flies(Tweety)$.

It can be shown that when using a second-order formula as the circumscription axiom that parallel predicate circumscription is both sound and complete relative to the preferred model semantics. Any formula derived is true in all preferred models and any formula true in all preferred models can be derived. There are a great many subtle technical issues involving the use of circumscription that deserve a whole course. We will avoid discussion of these issues because for most of the course it will suffice to think in terms of the preferred model semantics rather than any syntactic characterization.

5.2 Prioritized Predicate Circumscription

The use of abnormality theories introduces a number of problems, one of which involves a conflict between which predicates to minimize and in what order. Let's look at an example having to do with inheritance hierarchies. We would like to represent the following information. Normally, adults are employed. Normally, high school dropouts are not employed. All high school dropouts are adults. Mary is a high school dropout. Reasoning informally from the information, most would come to the conclusion that Mary is unemployed. Let's axiomatize this example and see what sort of conclusions we can derive.

$$\begin{gathered}\Gamma(\{Ab_1, Ab_2\}, \{Employed\}) = \\ \{\forall x.Adult(x)\wedge\neg Ab_1(x)\rightarrow Employed(x), \\ \{\forall x.Dropout(x)\wedge\neg Ab_2(x)\rightarrow\neg Employed(x), \\ \{\forall x.Dropout(x)\rightarrow Adult(x), \\ Dropout(Mary)\end{gathered} \qquad (25)$$

Using parallel predicate circumscription, and minimizing both Ab_1 and Ab_2, we find that there is a conflict in the minimization between Ab_1 and Ab_2. In some models, Ab_1 will be minimal and $Mary$ employed, while in others Ab_2 will be minimal with $Mary$ unemployed.In terms of an argument of specificity in inheritance hierarchies, we prefer the latter. Unfortunately, parallel predicate circumscription is too weak to allow us to derive this. It is easier to see the problem via a simple manipulation of formulas in 25. The problem becomes clearer if we look at the two formulas

$$\forall x.Adult(x)\wedge\neg Ab_1(x)\rightarrow Employed(x), \qquad (26)$$

$$\forall x.Dropout(x)\wedge\neg Ab_2(x)\rightarrow\neg Employed(x). \qquad (27)$$

The premise 27 can be rewritten as the equivalent

$$\forall x. Dropout(x) \wedge Employed(x) \rightarrow Ab_2(x). \tag{28}$$

Premises 26 and 28 together give

$$\forall x. Adult(x) \wedge Dropout(x) \wedge \neg Ab_1(x) \rightarrow Ab_2(x). \tag{29}$$

which is equivalent to both

$$\forall x. Adult(x) \wedge Dropout(x) \wedge \neg Ab_2(x) \rightarrow Ab_1(x) \tag{30}$$

and

$$\forall x. Adult(x) \wedge Dropout(x) \rightarrow (Ab_1(x) \vee Ab_2(x)). \tag{31}$$

From 31, it is clear that if both Ab predicates are minimized there will be a conflict. In some preferred models Dropouts that are adults will be Ab_1, and in others, Ab_2. The best we can do is to derive

$$\neg Ab_1(Mary) \vee \neg Ab_2(Mary).$$

Suppose we were interested in representing the fact that being abnormal with respect to aspect1 is less preferred than being abnormal with respect to aspect2. In other words, we would prefer that Mary is a unemployed over her being employed due to the fact that dropouts are a subclass of adults and where there is a conflict between normative information coupled to classes, we would like to prefer information attached to the most specific class. In some sense, we would need to encode a priority for minimizing Ab_2 at the expense of Ab_1. One way to do this would be to first minimize Ab_2 while letting Ab_1 and *Employed* vary, taking the resulting preferred models and then minimizing Ab_1, fixing Ab_2 and letting *Employed* vary. This ordering on models is in fact used in Prioritized Circumscription, another version of circumscription used to prioritize the order in which we minimize our predicates. Rather than minimize all of our predicates in parallel, we minimize them in groups sequentially. In the following sections, we will consider both syntactic and semantic charcaterizations of prioritized circumscription.

5.3 Syntactic Characterization

Suppose, we would like to minimize a pair of predicate constants P_1 and P_2 in Γ. In parallel circumscription this is expressed as

$$\begin{aligned}\Gamma(\{P_1, P_2\}, \emptyset) \wedge \forall P_1', P_2' [\Gamma(\{P_1', P_2'\}, \emptyset) \wedge (P_1', P_2') \leq (P_1, P_2)] \\ \rightarrow (P_1, P_2) \leq (P_1', P_2')\end{aligned} \tag{32}$$

Recall also that for arbitrary pairs of similar predicate expressions (U_1, U_2) and (V_1, V_2),

$$(U_1, U_2) \leq (V_1, V_2) \text{ iff } (U_1 \leq V_1) \wedge (U_2 \leq V_2).$$

In order to minimize P_1 at a higher priority than P_2, we will simply use the circumscription axiom 32, but with a different ordering relation. Instead of $\leq$, we will replace it by the ordering $\lll$, where

$$(U_1, U_2) \lll (V_1, V_2) \text{ iff } (U_1 \leq V_1) \wedge ((U_1 \equiv V_1) \rightarrow (U_2 \leq V_2)).$$

The relation can be extended to similar tuples of predicate expressions

$$(U_1, \ldots, U_n) \lll (V_1, \ldots, V_n) \text{ iff } U_1 \lll V_1 \wedge \bigwedge_{i=2}^{n} ((\bigwedge_{j=1}^{i-1} U_j \equiv V_j) \rightarrow U_i \leq V_i)$$

and to similar lists of tuples of predicate expressions

$$(\overline{U_1},\ldots,\overline{U_n}) \lessapprox (\overline{V_1},\ldots,\overline{V_n}) \text{ iff } \overline{U_1} \lessapprox \overline{V_1} \wedge \bigwedge_{i=2}^{n}((\bigwedge_{j=1}^{i-1} \overline{U_j} \equiv \overline{V_j}) \rightarrow \overline{U_i} \leq \overline{V_i}).$$

We write $(\overline{U_1},\ldots,\overline{U_n}) \ll (\overline{V_1},\ldots,\overline{V_n})$ for

$$(\overline{U_1},\ldots,\overline{U_n}) \lessapprox (\overline{V_1},\ldots,\overline{V_n}) \wedge \neg (\overline{V_1},\ldots,\overline{V_n}) \lessapprox (\overline{U_1},\ldots,\overline{U_n}).$$

Given the preliminaries, suppose we have a tuple of predicate constants $\overline{P}$, that we would like to minimize. The first task is to divide $\overline{P}$ into disjoint sublists of predicate constants $\overline{P_1},\ldots,\overline{P_n}$. Predicates in $\overline{P_1}$ are minimized at highest priority Predicates in $\overline{P_2}$ are minimized at next highest priority, etc. This is expressed as $\overline{P^1} > \overline{P^2} >, \ldots > \overline{P^n}$.

Definition 5.3 (Prioritized Circumscription)
Let $\overline{P}$ be a tuple of distinct predicate constants, $\overline{S}$ be a tuple of distinct predicate constants and/or function constants disjoint from $\overline{P}$, and $\Gamma(\overline{P},\overline{S})$, a theory. The prioritized circumscription *of $\overline{P}$ in $\Gamma(\overline{P},\overline{S})$ with variable $\overline{S}$ with respect to priorities $\overline{P^1} > \overline{P^2} >, \ldots > \overline{P^n}$, written $Circ_{pc}(\Gamma; \overline{P^1} > \overline{P^2} >, \ldots > \overline{P^n}; \overline{S})$, is the sentence*

$$\Gamma(\overline{P},\overline{S}) \wedge \forall \overline{P}', \overline{S}'. \neg[\Gamma(\overline{P},\overline{S}) \wedge \overline{P}' \ll \overline{P}]$$

Here, $\overline{P}' \ll \overline{P}$ is $(\overline{P^1}' > \overline{P^2}' >, \ldots > \overline{P^n}') \ll (\overline{P^1} > \overline{P^2} >, \ldots > \overline{P^n})$.

5.4 Semantic Characterization

As for parallel circumscription, we can define an ordering om models which captures the prioritized minimization policy.

Definition 5.4 (Minimal Models)
Let $\overline{P^1}, \overline{P^n}$ be disjoint sublists of all predicates constants from $\overline{P}$, $\overline{S}$ be a tuple of distinct predicate constants and/or function constants disjoint from $\overline{P}$, and $\Gamma(\overline{P},\overline{S})$, a theory. Suppose that M and N are models of Γ. M is said to be a $(\overline{P^1} >, \ldots, > \overline{P^n}); \overline{S})$-preferred model of N, written $M \lessapprox^{(\overline{P^1} >,\ldots,> \overline{P^n}; \overline{S})} N$, iff

1. *$D_M = D_N$.*
2. *$Value(U,M) = Value(U,N)$ for all constants U not in $\overline{P} \cup \overline{S}$.*
3. *For every $j = 1,\ldots,n$, if*

   $Value(P_l,M) = Value(P_l,N)$, for evry $P_l \in (\overline{P}^1,\ldots,\overline{P}^{j-1})$,

   then

   $Value(P_l,M) \subseteq Value(P_l,N)$, for evry $P_l \in \overline{P}^j$.

We write $M \ll^{(\overline{P^1} >,\ldots,> \overline{P^n}; \overline{S})} N$ if $M \lessapprox^{(\overline{P^1} >,\ldots,> \overline{P^n}; \overline{S})} N$, but not $N \lessapprox^{(\overline{P^1} >,\ldots,> \overline{P^n}; \overline{S})} M$.

A model of Γ is $(\overline{P^1} >, \ldots, > \overline{P^n}; \overline{S})$-minimal iff Γ has no model N such that $N \ll^{(\overline{P^1} >,\ldots,> \overline{P^n}; \overline{S})} M$. The set of all $(\overline{P^1} >, \ldots, > \overline{P^n}; \overline{S})$-minimal models of Γ will be denoted by $MOD^{(\overline{P^1} >,\ldots,> \overline{P^n}; \overline{S})}(\Gamma)$.

The definition above is the most general. To make things more concrete, let's consider minimizing just two predicate constants, Ab_1 and Ab_2 where $Ab_2 > Ab_1$. In other words, $\overline{P}^1 = \{Ab_2\}$ and $\overline{P}^2 = \{Ab_1\}$. We could then replace (iii) above with the following:

iii. $Value(Ab_2, M) \subseteq Value(Ab_2, N)$, or

If $Value(Ab_2, M) = Value(Ab_2, N)$

then

$Value(Ab_1, M) \subseteq Value(Ab_1, N)$.

As regards the syntactic characterization,

$$\overline{P}' \ll \overline{P} \equiv \{\{Ab_2'\}, \{Ab_1'\}\} \ll \{\{Ab_2\}, \{Ab_1\}\} \equiv$$

$$\begin{aligned}&[Ab_2' \leq Ab_2 \wedge (Ab_2' \equiv Ab_2 \rightarrow Ab_1' \leq Ab_1)] \rightarrow \\ &[Ab_2 \leq Ab_2' \wedge (Ab_2 \equiv Ab_2' \rightarrow Ab_1 \leq Ab_1')] \equiv\end{aligned}$$

$$\begin{aligned}&[(\forall x.Ab_2'(x) \rightarrow Ab_2(x)) \wedge ((\forall x.Ab_2'(x) \leftrightarrow Ab_2(x)) \rightarrow (\forall x.Ab_1'(x) \rightarrow Ab_1(x)))] \rightarrow \\ &\quad [(\forall x.Ab_2(x) \rightarrow Ab_2'(x)) \wedge ((\forall x.Ab_2(x) \leftrightarrow Ab_2'(x)) \rightarrow (\forall x.Ab_1(x) \rightarrow Ab_1'(x)))]\end{aligned}$$

5.5 Pros and Cons of Circumscription

1. Pros
   (a) Entirely embedded in the classical framework. first- or second- order logic.
   (b) After constructing the circumscription axiom, all deduction proceeds in monotonic logic. first-order logic plus nonmonotonicity reduces to classical second-order logic.
   (c) Possibilities of computing with fragments of theories to use logic programming etc.
2. Cons
   (a) How do we select the circumscriptive policy?
   (b) How do we choose the proper substitutions, or compute them automatically?

6 Model Preference Logics revisited

Yoav Shoham provides an elegant generalization of the idea of prefered models implicit in McCarthy's work and made more explicit in Lifschitz's work, that applies to logics other than FOPC. The idea is as follows:

1. Start with any standard logic, such as FOPC, Modal S4, or Intuitionistic logic, for example.
2. Define a partial order on the interpretations of the logic. The partial order might reflect a policy of preference based on the particular phenomona being modeled, such as temporal phenomona.
3. Shift the emphasis away from the particular way of defining the ordering, such as circumscription, to the semantics. Work with the models directly.

In a sense, this is the approach that will be used in the rest of the course.

7 Discussion

We have now presented both syntactic and semantic characterizations of model preference based on minimizing extensions of predicates. It all comes down to choosing the proper predicates to minimize, vary and fix, and to provide a policy of prioritization if needed. During the rest of the course, we will use this technique for reasoning about action and change. We will concentrate mostly on semantic characterizations of the phenomona we wish to study.