CIRCUMSCRIPTIVE REASONING

by

KENNETH JOHN HALLAND

submitted in part fulfilment of the requirements
for the degree of

MASTER OF SCIENCE

in the subject

COMPUTER SCIENCE

at the

UNIVERSITY OF SOUTH AFRICA

SUPERVISOR: PROF. W. LABUSCHAGNE

AUGUST 1994

PREFACE

The following conventions have been observed:

- Examples are marked by a solid line in the left-hand margin.

- Theorems are numbered in the form x.y, where x is the number of the relevant chapter and y indicates the sequence within the chapter. Corollaries to theorems are given the same numbers as the theorems from which they follow.

- 'Common sense' is spelt as 'common-sense' wherever the term is used as an adjective qualifying the noun 'reasoning'.

ACKNOWLEDGEMENTS

I must express my sincere gratitude to my supervisor, Willem Labuschagne, for his continual encouragement, enthusiasm and help, without which this work would never have been completed.

SUMMARY

We show how the non-monotonic nature of common-sense reasoning can be formalised by circumscription. Various forms of circumscription are discussed. A new form of circumscription, namely naïve circumscription, is introduced in order to facilitate the comparison of the various forms. Finally, some issues connected with the automation of circumscriptive reasoning are examined.

Key terms

Circumscription; Common-sense reasoning; Non-monotonic logic; Models; Satisfiability; Soundness; Completeness; Minimal models; Scope; Theorem-proving algorithms.

CONTENTS

CHAPTER ONE
INTRODUCTION

Non-monotonic logic attempts to formalise common-sense reasoning about some or other "system" or "world". The first task is to describe the system of interest (normally given in sentences of common speech) using a formal notation. (Although other languages for knowledge representation are known, we will consider only knowledge representation in first- and second-order logic.) The second task is to formalise the common-sense reasoning. Circumscription is an attempt to do this, and is most conveniently expressed in terms of second-order logic.

The present chapter surveys the technical machinery necessary for the first task. Subsequent chapters describe various forms of circumscription and assess the adequacy of these for the second task.

First-order logic

The description of a system of interest generally consists of statements concerning relations between, and operations on, specified "objects" in the system. To be able to represent our knowledge about the system, we firstly need a number of symbols to represent those objects we wish to single out, the relations between the objects and the operations on them.

The symbols form the *alphabet* of a first-order language, which may formally be viewed as the union of three disjoint sets:

- A set of logical symbols, usually containing
  - Parentheses: (and)
  - Sentential connective symbols: $\rightarrow$, $\neg$, $\wedge$, $\vee$ and $\leftrightarrow$ (representing the notions *if-then, not, and, or* and *if-and-only-if,* respectively)
  - Quantifier symbols: $\forall$ and $\exists$ (representing the notions *for all* and *for some,* respectively)
- A set of constant symbols, comprising
  - Zero or more individual constants (representing some or all of the objects in the system of interest)
  - Zero or more predicate constants (representing the relations between objects)

- Zero or more function constants (representing the operations on objects)

- A set of variable symbols, comprising
  - A countably infinite number of individual variables (intended to range over the objects in the system).

The choice of constant symbols will depend on the particular system being formalised.

In general, the choice of individual constants depends on the objects one wishes to name in the system. Consequently, the individual constant symbols may be single letters, e.g. *a*, *b* and *c*, numerals, e.g. *0*, *1* and *113*, or strings of characters, e.g. *Tweety*, *apple3* and *Nixon*.

The choice of predicate constants depends on the relevant relations holding between objects in the system. The predicate constant symbols may be single letters, e.g. *A*, *P* and *Q*, or strings of letters, e.g. *Bird*, *Pacifist* and *Ab*. Subscripts are sometimes appended to distinguish different predicates, e.g. P_1, P_{113}, Ab_1 and Ab_2.

Similarly, the choice of function constants depends on the operations on objects. The function constant symbols are strings of letters, e.g. *f*, *g* and *succ*. Subscripts are sometimes appended to distinguish different functions, e.g. f_1, f_2 and f_{113}.

Every predicate and function constant has a non-negative integer associated with it, denoting its *arity*. The arity of a predicate or function constant represents the number of arguments it takes.

A distinguished 2-ary predicate constant symbol, $=$, called *equality*, is often included in a first-order alphabet.

The individual variables are generally single letters (we restrict them to lower case letters near the end of the alphabet), e.g. *x*, *y* and *z*. Subscripts can also be appended to distinguish different variables, e.g. x_1, x_2 and x_{113}.

> As an example, consider a common system studied in Artificial Intelligence, namely a blocks world. In this system, a number of blocks of different size and colour are placed on a table. The blocks are named so that statements can be made about their properties and their relative positions (for instance, whether one block is stacked on another).

An example of an alphabet of symbols for the Blocks World is:

Individual constants: *a, b, c;* representing separate blocks

Predicate constants: *Grey, White, BiggerThan, SameSize, On;* representing properties of blocks and relationships between them.

Function constant: *top*; representing a function to determine which block is the topmost of a stack of blocks.

(All the logical symbols and all the individual variable symbols are included in the language.)

An alphabet S uniquely determines a first-order language, the expressions of which serve to represent statements about the system of interest, by the following two-stage process: firstly a subset of S* (the set of strings over the alphabet) whose members are called *terms* is defined and secondly a subset of S* whose members are called *well-formed formulas* (*wffs*) is defined. The set of wffs is the *first-order language*.

- Terms:

  The set of terms over an alphabet S is the smallest subset of S* such that

  · if c is an individual constant then c is a term

  · if x is an individual variable then x is a term

  · if $t_1,..,t_n$ are terms and f is an n-ary function constant then $f(t_1,..,t_n)$ is a term.

- Well-formed formulas:

  The set of wffs is the smallest subset of S* such that

  · if $t_1,..,t_n$ are terms and P is an n-ary predicate constant then $P(t_1,..,t_n)$ is a wff

  · if α and β are wffs and x is any individual variable, then the following are all wffs:

$(\neg\alpha)$	the negation of α
$(\alpha \rightarrow \beta)$	the conditional with antecedent α and consequent β
$(\alpha \wedge \beta)$	the conjunction of α and β
$(\alpha \vee \beta)$	the disjunction of α and β
$(\alpha \leftrightarrow \beta)$	the biconditional involving α and β
$\forall x(\alpha)$	the universal quantification of x in α
$\exists x(\alpha)$	the existential quantification of x in α.

In what follows, parentheses will often be omitted where ambiguity will not arise. When a language includes the equality predicate, we write it between the two terms forming its arguments. In other words, if t_1 and t_2 are two terms forming the arguments of the equality predicate, we write $t_1 = t_2$ rather than $=(t_1,t_2)$.

When we represent our knowledge about some system of interest in a first- order language, we in effect single out a set of wffs. Such a set of wffs is called a *set of axioms*. It is important to note that we make no distinction between a finite set of axioms and the conjunction of all its separate wffs. Generally speaking, when we represent our knowledge about some *real* (as opposed to *mathematical*) system, the result is a finite set of axioms.

Terms which contain no variables are called *ground terms*. Wffs which contain no sentential connective or quantifier symbols are called *atomic formulae* or simply *atoms*. Atoms which contain no variables are called *ground atoms*.

The axioms we consider will always be wffs of the kind called *sentences*. Sentences may be thought of as wffs in which either no variables occur or the variables which do occur are bound. More precisely:

- The *scope* of a quantifier is the shortest substring following it which is itself a wff.
- An occurrence of a variable x in a wff is *bound* iff it is in the scope of a $\forall x$ or $\exists x$, otherwise it is *free*.
- Wffs which contain no free occurrences of variables are called *sentences*.

A wff commencing with a number of universal (existential) quantifiers is often abbreviated by using only a single universal (existential) quantifier symbol:

$\forall x_1(\forall x_2(..\forall x_n(\alpha)..))$ is abbreviated to $\forall x_1..x_n(\alpha)$

and

$\exists x_1(\exists x_2(..\exists x_n(\alpha)..))$ is abbreviated to $\exists x_1..x_n(\alpha)$.

Consider, for example, the language given previously for the Blocks World. A set of axioms describing the Blocks World may be divided into two classes; those which are true for any number and arrangement of blocks and those which describe a particular state of affairs.

Some axioms describing a certain state of affairs in the Blocks World are:

$$Grey(a) \wedge Grey(b) \wedge White(c)$$
$$BiggerThan(a,b) \wedge SameSize(b,c)$$
$$On(a,c)$$

Some further axioms making general statements about any number and arrangements of blocks are:

$$\forall x(Grey(x) \leftrightarrow \neg White(x))$$
$$\forall xy(\neg(BiggerThan(x,y) \vee SameSize(x,y)) \rightarrow BiggerThan(y,x))$$
$$\forall x(\neg\exists y(On(y,x)) \rightarrow top(x) = x)$$
$$\forall xyz(On(x,y) \wedge top(x) = z \rightarrow top(y) = z)$$

Semantics of a first-order language

When choosing individual, predicate and function constant symbols for a given system, we have an intended interpretation in mind for each of these symbols. Other interpretations may also be possible. In order to avoid ambiguity, it is necessary to be explicit about interpretations. We therefore make precise the concept of an interpretation.

An *interpretation* I of a first-order language consists of

- A *domain*: a non-empty set of "objects", denoted by $|\mathrm{I}|$
- A function from the constant symbols to their *denotations*, namely to members of, relations on, or functions over the domain. More precisely
  - Every individual constant, c_i, is mapped to a specific element of $|\mathrm{I}|$, namely c_i^{I}
  - Every n-ary predicate constant, P_i, is mapped to a specific n-ary relation on $|\mathrm{I}|$, i.e. a set of n-tuples of $|\mathrm{I}|^n$, namely P_i^{I}
  - Every n-ary function constant, f_i, is mapped to a specific n-ary function from $|\mathrm{I}|^n$ to $|\mathrm{I}|$, namely f_i^{I}.

For example, an interpretation I for the Blocks World language is:

- $|\ I\ |$ = {BLOCKA,BLOCKB,BLOCKC}
- a^I = BLOCKA, b^I = BLOCKB, c^I = BLOCKC
- $Grey^I$ = {BLOCKA,BLOCKB}, $White^I$ = {BLOCKC}

  $BiggerThan^I$ = {(BLOCKA,BLOCKB),(BLOCKA,BLOCKC)}

  $SameSize^I$ = {(BLOCKB,BLOCKC)}

  On^I = {(BLOCKA,BLOCKC)}
- top^I(BLOCKA) = BLOCKA

  top^I(BLOCKB) = BLOCKB

  top^I(BLOCKC) = BLOCKA.

Another interpretation I' is:

- $|\ I'\ |$ = {1,2,3,...}
- $a^{I'}$ = 2, $b^{I'}$ = 3, $c^{I'}$ = 4
- $Grey^{I'}$ = {1,3,5,...}, $White^{I'}$ = {2,4,6,...}

  $BiggerThan^{I'}$ = {(x,y) ∈ $|\ I'\ |^2$: x > y}

  $SameSize^{I'}$ = {(x,y) ∈ $|\ I'\ |^2$: x = y}

  $On^{I'}$ = {(x,y) ∈ $|\ I'\ |^2$: x is a factor of y less than y}
- $top^{I'}$(x) = 1 for all x ∈ $|\ I'\ |$.

A *valuation* in an interpretation I is a function v which assigns to every individual variable x_i an element of $|\ I\ |$, namely $v(x_i)$.

Note that if there are two or more elements in the domain, there are an infinite number of valuations.

Given a valuation v in an interpretation I of a first-order language, a corresponding term-value function ν is defined from the set of terms to the domain of I, as follows:

- $\nu(c_i) = c_i^I$, for every individual constant c_i
- $\nu(x_i) = v(x_i)$, for every individual variable x_i
- $\nu(f_i(t_1,..,t_n)) = f_i^I(\nu(t_1),..,\nu(t_n))$, for every function constant f_i, where $t_1,..,t_n$ are terms of the language.

A valuation v is said to *satisfy* a wff in an interpretation I under the following conditions:

- if α is an atomic wff, i.e. $\alpha = P_i(t_1,..,t_n)$, then v satisfies α iff the n-tuple $\langle v(t_1),..,v(t_n)\rangle$ is in the relation P_i^I
- v satisfies $(\neg\alpha)$ iff v does not satisfy α
- v satisfies $(\alpha \rightarrow \beta)$ iff v does not satisfy α or v satisfies β
- v satisfies $(\alpha \wedge \beta)$ iff v satisfies α and v satisfies β
- v satisfies $(\alpha \vee \beta)$ iff v satisfies α or v satisfies β or both
- v satisfies $(\alpha \leftrightarrow \beta)$ iff v satisfies both α and β or neither α nor β
- v satisfies $\forall x_i(\alpha)$ iff every valuation v' which differs from v at most in the value it assigns to x_i, satisfies α
- v satisfies $\exists x_i(\alpha)$ iff there is some valuation v' which differs from v at most in the value it assigns *to* x_i, which satisfies α.

A wff is *true* in an interpretation I iff it is satisfied by all possible valuations in I. A wff is *false* in an interpretation I iff it is not satisfied by any valuations in I. Sentences have the property that they are satisfied either by all valuations in I or by none.

A wff is *logically valid* iff it is true in every interpretation. A wff is *contradictory* iff it is false in every interpretation.

Given a set of wffs A, an interpretation M is called a *model* of A iff all wffs of A are true in the interpretation M. A is said to be *satisfiable* iff it has a model, otherwise it is *unsatisfiable*.

Both interpretations given above for the Blocks World, namely I and I', are models of the axioms because all the axioms are true in the respective interpretations.

We note the importance of the general axioms about any number and arrangements of blocks. They restrict the class of models of the axioms to a much smaller class of models containing the intended interpretation. Although they do not prevent the "unintended" second interpretation from being a model, they do prevent a large class of "nonsense" interpretations, for example the following interpretation J:

- $|J| = \{BLOCKA,BLOCKB,BLOCKC\}$
- a^J = BLOCKA, b^J = BLOCKB, c^J = BLOCKC
- $Grey^J = \{BLOCKA,BLOCKB,BLOCKC\}$

$White^J$ = {BLOCKA,BLOCKC}

$BiggerThan^J$ = {(BLOCKA,BLOCKB),(BLOCKB,BLOCKA)}

$SameSize^J$ = {(BLOCKB,BLOCKC)}

On^J = {(BLOCKA,BLOCKC),(BLOCKB,BLOCKC)}

top^J(BLOCKA) = BLOCKB

top^J(BLOCKB) = BLOCKC

top^J(BLOCKC) = BLOCKA.

To see that interpretation J is not a model of the axioms, consider any valuation v in J with v(x) = BLOCKB. The valuation does not satisfy the first (general) axiom, namely $\forall x(Grey(x) \leftrightarrow \neg White(x))$, because the valuation v' identical to v except that v'(x) = BLOCKA does not satisfy $Grey(x) \leftrightarrow \neg White(x)$ since BLOCKA $\in Grey^J$ and BLOCKA $\in White^J$.

If A is a set of wffs and α is a single wff, then A *semantically entails* α iff every valuation in every model of A satisfies α. We write A $\models \alpha$ to indicate that A semantically entails α. We write $\models \alpha$ iff α is true in every interpretation, i.e. α is logically valid.

One of the properties of $\models$ is monotonicity: if A $\models \alpha$ then A $\cup$ $\{\beta\}$ $\models \alpha$. In other words, if α is entailed by a set of axioms A, then this remains the case regardless of how we may choose to augment A by additional information. One of the main characteristics of common-sense reasoning is that it moves from knowledge (represented by A) to a 'conclusion' (perhaps better regarded as a conjecture) which may be need to be retracted in the light of additional information. In other words, the formalisation of common-sense reasoning must involve some way of getting around the monotonicity of $\models$.

Second-order logic

Second-order languages are very similar to first-order languages, as will become apparent. The principle difference is that second-order languages have not only variables ranging over individual objects in the domain of interpretation but also variables taking relations and functions as values. The need for the second-order variables is illustrated in the following example:

> Consider the problem of expressing the principle of mathematical induction as a well-formed formula. The principle of mathematical induction is based on the fact that the set of natural numbers $\mathbb{N}$ is the smallest set containing 0 and having the property that if x belongs to it, so does the successor of x. Any property P defines a subset of $\mathbb{N}$. The principle of mathematical induction merely states that the subset defined by P, if it contains 0 and is closed under the formation of successors, must be equal to $\mathbb{N}$ - it cannot be a proper subset, since $\mathbb{N}$ is the smallest set with the relevant characteristics. For a property P, we can express this as follows:
>
> $$(P(0) \wedge \forall x(P(x) \rightarrow P(succ(x)))) \rightarrow \forall x(NatNum(x) \rightarrow P(x))$$
>
> The problem with this statement is that it is about a specific property P. The principle of induction makes a general statement about all properties. In other words, we would like to use a predicate variable, say Φ (ranging over all possible subsets of $\mathbb{N}$) instead of the predicate constant P. This gives the second-order wff
>
> $$\forall\Phi((\Phi(0) \wedge \forall x(\Phi(x) \rightarrow \Phi(succ(x)))) \rightarrow \forall x(NatNum(x) \rightarrow \Phi(x)))$$

A second-order language is specified by an alphabet of symbols as well as rules for forming expressions in the form of strings of symbols.

An *alphabet* of a second-order language consists of the union of three disjoint sets:

- A set of logical symbols, usually containing
  - Parentheses: (and)
  - Sentential connective symbols: $\rightarrow$, $\neg$, $\wedge$, $\vee$ and $\leftrightarrow$
  - Quantifier symbols: $\forall$ and $\exists$

- A set of constant symbols, comprising
  - · Zero or more individual constants
  - · Zero or more predicate constants
  - · Zero or more function constants
- A set of variable symbols, comprising
  - · A countably infinite number of individual variables
  - · A countably infinite number of predicate variables
  - · A countably infinite number of function variables.

A distinguished predicate constant symbol, $=$, called *equality*, will be included in every second-order alphabet.

The predicate variable symbols may be taken to be uppercase letters of the Greek alphabet, e.g. Φ and Ψ. Subscripts can be appended to distinguish different predicate variables, e.g. Φ_1 and Φ_{113}.

The function variable symbols will be taken to be lowercase letters of the Greek alphabet, e.g. ϕ and ψ. (Lowercase letters at the beginning of the Greek alphabet, e.g. α and β, are sometimes used to refer to arbitrary wffs). Subscripts can be appended to distinguish different function variables, e.g. ϕ_1 and ϕ_{113}.

Every predicate and function constant or variable has a non-negative integer associated with it, denoting its *arity*. The arity of a predicate or function constant or variable represents the number of arguments it takes.

An alphabet S uniquely determines a second-order language by the following two-stage process: Firstly, a subset of S* (the set of strings over S), whose members are called *terms*, is defined, and then a subset of S*, whose members are called *well-formed formulas* (*wffs*), is defined. The set of wffs is the *second-order language*.

- Terms:

  The set of terms over the alphabet S is the smallest subset of S* such that
  - · if c is an individual constant then c is a term
  - · if x is an individual variable then x is a term
  - · if $t_1,..,t_n$ are terms and f is an n-ary function constant then $f(t_1,..,t_n)$ is a term
  - · if $t_1,..,t_n$ are terms and ϕ is an n-ary function variable then $\phi(t_1,..,t_n)$ is a term.

- Well-formed formulas:

  The set of wffs is the smallest subset of S* such that

  · if $t_1,..,t_n$ are terms and P is an n-ary predicate constant then $P(t_1,..,t_n)$ is a wff

  · if $t_1,..,t_n$ are terms and Φ is an n-ary predicate variable then $\Phi(t_1,..,t_n)$ is a wff

  · if α and β are wffs and X is any individual, predicate or function variable, then the following are all wffs

$(\neg\alpha)$	the negation of α
$(\alpha \rightarrow \beta)$	the conditional with antecedent α and consequent β
$(\alpha \wedge \beta)$	the conjunction of α and β
$(\alpha \vee \beta)$	the disjunction of α and β
$(\alpha \leftrightarrow \beta)$	the biconditional involving α and β
$\forall X(\alpha)$	the universal quantification of X in α
$\exists X(\alpha)$	the existential quantification of X in α.

Terms which contain no variables are called *ground terms*. Wffs which contain no sentential connective or quantifier symbols are called *atomic formulae* or simply *atoms*. Atoms which contain no variables are called *ground atoms*.

The *scope* of a quantifier is the shortest substring following it which is itself a wff. An occurrence of a (individual, predicate or function) variable X in a wff is *bound* iff it is in the scope of a $\forall X$ or $\exists X$, otherwise it is *free*. Wffs which contain no free occurrences of variables are called *sentences*.

As an example of a set of second-order axioms, consider the Peano axioms for the natural numbers [Enderton 1977] in the language whose alphabet contains the individual constant c (intended to represent zero) and the function constant *succ* (intended to represent the injective successor function):

$$\forall x(\neg succ(x) = c)$$

$$\forall xy(succ(x) = succ(y) \rightarrow x = y)$$

$$\forall\Phi((\Phi(c) \wedge \forall x(\Phi(x) \rightarrow \Phi(succ(x)))) \rightarrow \forall x(\Phi(x)))$$

Standard semantics of a second-order language

To formally describe the meaning of the symbols of a second-order language, the concept of an interpretation must be made precise.

A *standard interpretation* I of a second-order language consists of

- A *domain*: a non-empty set of "objects", denoted by $|\mathrm{I}|$
- A function from the constant symbols to their *denotations*, namely to elements of, relations on and functions over the domain. More precisely
  - Individual constants, c_i, are mapped to specific elements of $|\mathrm{I}|$, namely c_i^{I}
  - Predicate constants, P_i, are mapped to specific n-ary relations on $|\mathrm{I}|$, i.e. sets of n-tuples of $|\mathrm{I}|^n$, namely P_i^{I}
  - Function constants, f_i, are mapped to specific n-ary functions from $|\mathrm{I}|^n$ to $|\mathrm{I}|$, namely f_i^{I}.

One interpretation I of the language (indeed, the intended interpretation) in which the above axiomatisation of the natural numbers is formulated, is

- $|\mathrm{I}| = \mathbb{N}$
- $c^{\mathrm{I}} = 0$
- $=^{\mathrm{I}}$ is the identity relation on $\mathbb{N}$ (i.e. $\{(x,y) \in \mathbb{N} \mid x = y\}$)
- $succ^{\mathrm{I}}$ is the successor function on $\mathbb{N}$.

Another interpretation J of the same language is

- $|\mathrm{J}| = \{0,1,2\}$
- $c^{\mathrm{I}} = 0$
- $=^{\mathrm{J}}$ is the identity relation on $\{0,1,2\}$
- $succ^{\mathrm{J}}$ is the successor function for modulo 3 arithmetic on $\{0,1,2\}$, i.e. the successor of 2 is 0.

A *valuation* in a standard interpretation I is a function v which assigns elements of, relations on, or functions over the domain to each of the individual, predicate and function variables. More precisely

- Individual variables: an element of $|\mathrm{I}|$ is assigned to each individual variable x_i, namely $v(x_i)$

- Predicate variables: an n-ary relation on $|I|$ is assigned to each n-ary predicate variable Φ_i, namely $v(\Phi_i)$
- Function variables: an n-ary function from $|I|^n$ to $|I|$ is assigned to each n-ary function variable ϕ_i, namely $v(\phi_i)$.

Note that there are many possible valuations for any interpretation.

Given a valuation v in an interpretation I we define the term-value function ν from the set of terms to the domain of I as follows:

- $\nu(c_i) = c_i^I$, for every individual constant c_i
- $\nu(x_i) = v(x_i)$, for every individual variable x_i
- $\nu(f_i(t_1,..,t_n)) = f_i^I(\nu(t_1),..,\nu(t_n))$, for every function constant f_i, where $t_1,..,t_n$ are terms of the language
- $\nu(\phi_i(t_1,..,t_n)) = v(\phi_i)(\nu(t_1),..,\nu(t_n))$, for every function variable ϕ_i, where $t_1,..,t_n$ are terms of the language.

A valuation v is said to *satisfy* a wff in an interpretation I under the following conditions:

- if α is an atomic formula, i.e. either $P_k(t_1,..,t_n)$ or $\Phi_k(t_1,..,t_n)$, then v satisfies α iff the n-tuple $\langle\nu(t_1),..,\nu(t_n)\rangle$ is in the relation P_k^I or $v(\Phi_k)$, respectively
- v satisfies $(\neg\beta)$ iff v does not satisfy β
- v satisfies $(\alpha \rightarrow \beta)$ iff v does not satisfy α or v satisfies β
- v satisfies $(\alpha \wedge \beta)$ iff v satisfies α and v satisfies β
- v satisfies $(\alpha \vee \beta)$ iff v satisfies α or v satisfies β or both
- v satisfies $(\alpha \leftrightarrow \beta)$ iff v satisfies both α and β or neither α nor β
- v satisfies $\forall X(\beta)$, where X is any individual, predicate or function variable, iff every valuation v' which differs from v at most in the value it assigns to X satisfies β
- v satisfies $\exists X(\beta)$, where X is any individual, predicate or function variable, iff there is some valuation v' which differs from v at most in the value it assigns to X which satisfies β.

A wff is *true* in an interpretation I iff it is satisfied by all possible valuations in I. A wff is *false* in an interpretation I iff it is not satisfied by any valuations in I. A wff is *logically valid* iff it is true in every interpretation. A wff is *contradictory* iff it is false in every interpretation.

Given a set of wffs A then a standard interpretation M is called a *standard model* of A iff all wffs of A are true under the interpretation M. A is said to be *satisfiable* iff it has a standard model, otherwise it is *unsatisfiable.*

If A is a set of wffs and α is a single wff, then A *semantically entails* α *with respect to the standard semantics* iff every valuation in every standard model of A satisfies α with respect to the standard semantics. We write $A \models \alpha$ to indicate that A semantically entails α. We write $\models \alpha$ iff α is true in every standard interpretation. Just as in the case of first-order logic, $\models$ is monotonic.

The interpretation I given above is a model of the Peano axioms. It is easy to see that the first two axioms are true in I. The third axiom states that any subset of the domain which contains 0 and is closed under the successor function must contain all objects in the domain. This axiom is true in I - recall that $\mathbb{N}$ is the smallest set containing 0 and closed under the formation of successors.

The interpretation J is not a model of the axioms, since the first axiom is false in J. To see this, note that 0 is the successor of 2 in J.

Consider the interpretation J' of the same set of axioms:

- $|J'| = \mathbb{R}^+$ (the set of non-negative reals)
- $c^{J'} = 0$
- $=^{J'}$ is the identity relation on $\mathbb{R}^+$
- $succ^{J'}$ is the function which maps every element x of $\mathbb{R}^+$ to $x+1$

J' is not a model of the set of axioms, since the last axiom is not true in J'. This is because $\{0,1,2,..\}$ is a subset of $\mathbb{R}^+$ which is closed under the successor function but which is not equal to $\mathbb{R}^+$.

Note furthermore that J' is a model of the first two axioms. The class of models of the first two axioms is thus larger than the class comprising the models of all three axioms. This illustrates that axioms have the effect of incrementally reducing the size of the class of models.

Soundness, completeness and consistency of first- and second-order logic

We now give some important theorems (without proof) concerning what we may hope to achieve with theorem-proving algorithms for first- and second-order logic. For a more detailed exposition of the various types of theorem-proving algorithms for first-order logic, see [Genesereth & Nilsson 1988]. For a discussion of restrictions on second-order theorem-provers and alternatives to the standard semantics, see [Enderton 1972] and [Shapiro 1991].

By a theorem-proving algorithm (or deductive system) we understand an algorithm that uses rewriting rules (i.e. syntactic transformations on strings of the language) to establish connections between formulas. A theorem-proving algorithm is said to be *sound* iff every wff which can be deduced by it from a set of axioms is semantically entailed by the set of axioms. A theorem-proving algorithm is said to be *complete* iff every wff which is entailed by a set of axioms can be deduced from the set of axioms by the algorithm. (We write $A \vdash^D \alpha$ if the wff α can be deduced by algorithm D from the set of axioms A).

Theorem 1.1 (Soundness and completeness of first-order logic)
There exists a theorem-proving algorithm D such that for any set A of first-order axioms and any wff α

- If $A \vdash^D \alpha$ then $A \models \alpha$ (Soundness)
- If $A \models \alpha$ then $A \vdash^D \alpha$ (Completeness).

■

Remark: There are many sound and complete theorem-proving algorithms for first-order logic, ranging from Hilbert-style systems that employ rules like Modus Ponens to refutation-style systems that use Robinson's resolution rule.

Given a sound and complete theorem-proving algorithm D, a set of axioms A is said to be *consistent* iff there is a wff α such that α cannot be deduced from A by D.

Theorem 1.2 (Consistency and satisfiability of first-order logic)
A first-order set of axioms is consistent iff it is satisfiable.

■

This means that all that is needed to show that a first-order set of axioms is consistent, is a model of its axioms.

Things are not so good for second-order logic, however.

Theorem 1.3 (Soundness of second-order logic)
There exists a theorem-proving algorithm D such that for any set A of second-order axioms and any wff α, $A \vdash^{D} \alpha$ implies that $A \models \alpha$.
■

One of the most important limitations of the standard semantics for second-order logic, however, is

Theorem 1.4 (Incompleteness of second-order logic)
No theorem-proving algorithm that is sound with respect to the standard semantics for second-order logic, is complete. In other words, $A \models \alpha$ does not imply that $A \vdash^{D} \alpha$.
■

Corollary 1.4
In second-order logic (with the standard semantics) the semantic notion of satisfiability lacks a syntactic analogue (i.e. there is no equivalent notion of consistency, as there is in the case of first-order logic).
■

In view of Theorem 1.4, it is sometimes preferable to employ an alternative to the standard semantics for second-order logic. We take up this matter in Chapter Four.

CHAPTER TWO
CIRCUMSCRIPTION

Suppose we represent our knowledge of some system in a first-order language. If the constants have been selected in a sensible way, there will be, among the many possible interpretations of the language, at least one that bears a close resemblance to the system of interest. This we call the intended interpretation. The purpose of the axioms that represent our knowledge of the system is to select, from the class of all interpretations, a (hopefully much smaller) subclass of models which contains the intended interpretation and, preferably, nothing else. In general, however, it is difficult (and in some cases impossible) to incorporate sufficient knowledge about a system to get rid of all unwanted models, i.e. to give a complete axiomatisation of the system.

One of the earliest attempts to devise a general method for completing a set of axioms involved the **closed world assumption** [Reiter 1978]. The closed world assumption is a meta-theoretical postulate to the effect that the denotations of all predicate constants should be as small as possible. As an example of the application of the closed world assumption, consider a database used by an airline booking system.

In the database, we wish to represent facts about which cities the airline provides flights between (connections). This can be achieved by a first-order language consisting of individual constants representing all the relevant cities and a 2-ary predicate constant *IsConnectedTo*. For instance, the atom *IsConnectedTo*(*London*,*NewYork*) would be included in the set of axioms (call it DB, for database) to indicate that the airline provides a flight between London and New York.

A *query* on the database is answered by determining whether a wff representing the query is entailed by the set of axioms. For instance, the query of whether Paris is connected to Quebec would consist of determining whether DB $\models$ *IsConnectedTo*(*Paris*,*Quebec*).

However, a problem arises when no specific atom is included in the database representing the connection being queried, e.g. if *IsConnectedTo*(*Paris*,*Quebec*) is not included in the set of axioms. In such a case, neither *IsConnectedTo*(*Paris*,*Quebec*) nor its negation is entailed by DB because there are some models of DB which include the ordered pair (PARIS,QUEBEC) in the denotation of *IsConnectedTo* and there are models which do not.

The closed world assumption is the assumption that DB contains all relevant knowledge about the connections that exist, so that the query whether Paris is connected to Quebec should in this case be answered negatively. In effect, the closed world assumption excludes all the models of DB having denotations of the predicate constant *IsConnectedTo* that are larger than absolutely necessary. In fact, subject to certain mild restrictions (namely that DB consist of definite Horn clauses) the closed world assumption excludes all except a unique model (known as the least Herbrand model).

Circumscription may be viewed as a generalisation of the closed world assumption, which allows us to focus on some rather than all of the predicate constants.

Circumscription achieves this by adding new axioms which talk about alternative ways to interpret selected predicate constants. In fact, a variety of circumscriptive techniques have been developed, each of which has its own distinctive way of selecting out a class of 'minimal' models, i.e. models in which the denotations of one or more predicate constants are 'minimised'. The key to all these techniques is to use wffs of the form $\forall x(P(x) \rightarrow Q(x))$ to express the idea that the denotation of P is a subset (not necessarily proper) of the denotation of Q. It will be convenient to introduce a special notation to abbreviate sequences of such wffs.

Notation Tuples of predicate constants or variables are indicated in boldface. For example, $\boldsymbol{P}$ represents the tuple of predicate constants $\langle P_1,..,P_n\rangle$ and $\boldsymbol{\Phi}$ represents the tuple of predicate variables $\langle \Phi_1,..,\Phi_m\rangle$.

Let P and Q be two arbitrary predicate constants or variables (for simplicity, let them be unary) and let x be an arbitrary individual variable:

$\forall x(P(x) \rightarrow Q(x))$ is abbreviated to $P \leq Q$

and

$(P \leq Q) \wedge \neg(Q \leq P)$ is abbreviated to $P < Q$.

Similarly for two n-tuples of predicate constants or variables $\boldsymbol{P}$ and $\boldsymbol{Q}$:

$\forall x(P_1(x) \rightarrow Q_1(x)) \wedge .. \wedge \forall x(P_n(x) \rightarrow Q_n(x))$ is abbreviated to $\boldsymbol{P} \leq \boldsymbol{Q}$

and

$(\boldsymbol{P} \leq \boldsymbol{Q}) \wedge \neg(\boldsymbol{Q} \leq \boldsymbol{P})$ is abbreviated to $\boldsymbol{P} < \boldsymbol{Q}$.

■

In subsequent sections we shall explore increasingly powerful forms of circumscription. The particular application that we have in mind for each of these forms of circumscription is to formalise the kind of common-sense reasoning that employs default rules of the form "Normally, such and such is the case". Probably the most famous example found in texts on common-sense reasoning is one about a bird called Tweety. Using the fact that normally birds can fly, we wish to be able to 'infer' (more in the sense of making a plausible conjecture than a logical deduction) that Tweety can fly. A set of axioms (call the set A) representing our knowledge is

$$Bird(Tweety)$$
$$\forall x(Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x))$$

(The predicate constant *Ab*, representing the predicate 'is abnormal', was first introduced by McCarthy [1980] as a device that would enable one to represent rules of the form "normally such and such is the case" in the formal language.)

The first thing to note about the set of axioms A is that, as it stands, it does not entail the wff *Flies*(*Tweety*) since no statement affirming or denying Tweety's abnormality is included in A. In accordance with the intuition that most birds are normal, and that one should regard an entity as being abnormal only if forced to do so, one may decide to discard all models of A except those in which the denotation of *Ab* is minimal. Circumscription embodies the attempt to achieve the exclusion of unwanted models in a syntactic way (by adding certain new axioms to A). Circumscription will achieve our goal of formalising the 'inference' that Tweety can fly if it eliminates all models of A except those in which Tweety is normal, for then *Flies*(*Tweety*) is true in each remaining model. In other words, the circumscription of *Ab* will achieve our goal if the resulting expanded set of axioms, denoted by CIRC(A;*Ab*), entails *Flies*(*Tweety*).

Of course, one could eliminate all the undesirable models quite straightforwardly by simply adding the wff $\neg Ab(Tweety)$ to the original set of axioms. But then *Flies*(*Tweety*) is classically entailed by our axioms which means that it may be asserted as definite knowledge. While the axioms from which we begin represent our definite knowledge, circumscription is intended to lead us some distance beyond what is definite, namely to plausible conjectures. For instance, it is a plausible conjecture that Tweety can fly, since there is no information to the contrary and the greater proportion of birds do fly. There is a subtle but important difference between *knowing* that Tweety is a normal bird (and hence a flying bird) and making a plausible conjecture to this effect.

Wffs entailed by CIRC(A;*Ab*) but not by A represent these plausible conjectures.

The second thing to note is that this type of common-sense reasoning is non-monotonic. In other words, plausible conjectures which are made by common-sense reasoning might have to be retracted in the light of new information. For example, if an axiom were added affirming the abnormality of Tweety, i.e. *Ab*(*Tweety*), the circumscription of *Ab* would (once again) select out the models in which the denotation of *Ab* is as small as possible. In this case, every such model would have to include the denotation of *Tweety* in the denotation of *Ab*, and in such models there is no need for the denotation of *Tweety* to be in *Flies*. In other words, a conjecture (namely *Flies*(*Tweety*)) that was plausible with regard to the set of axioms A (plausible in the sense that *Flies*(*Tweety*) was entailed by CIRC(A;*Ab*)) is not plausible with regard to the expanded set of axioms A $\cup$ {*Ab*(*Tweety*)} because CIRC(A $\cup$ {*Ab*(*Tweety*)};*Ab*) does not entail *Flies*(*Tweety*).

Naïve circumscription

We begin by describing a form of circumscription, which we call naïve circumscription, for which the reader will search the literature in vain. We have invented the concept because it enables us to see the principle forms of circumscription as variations on a single theme.

Definition Suppose that A is a finite set of axioms in a first-order language and $\boldsymbol{P}$ is the tuple of predicate constants $\langle P_1,..,P_n\rangle$. The *naïve circumscription of* $\boldsymbol{P}$ *in* A, denoted by CIRC(A;$\boldsymbol{P}$), is the second-order formula

$$A \wedge \neg\exists\Phi(A[\Phi] \wedge \Phi < \boldsymbol{P})$$

where Φ is an n-tuple of predicate variables $\langle\Phi_1,..,\Phi_n\rangle$ whose arities correspond to those of the predicate constants in $\boldsymbol{P}$ and A[Φ] is the formula obtained from A by substituting $\Phi_1,..,\Phi_n$ for $P_1,..,P_n$ in A. Thus CIRC(A;$\boldsymbol{P}$) is the conjunction of the original set of axioms A and the *circumscription axiom* $\neg\exists\Phi(A[\Phi] \wedge \Phi < \boldsymbol{P})$.

■

If we only circumscribe a single predicate P in a set of axioms A the naïve circumscription of P in A is

$$A \wedge \neg\exists\Phi(A[\Phi] \wedge \Phi < P)$$

where Φ is a predicate variable of the same arity as the predicate constant P.

Expressed intuitively, this states that the denotation of P is as small as possible; i.e. the denotation of P satisfies A but there is no proper subset of the denotation of P which satisfies A.

In the full definition of naïve circumscription given above, the circumscription axiom is represented by $\neg\exists\Phi(A[\Phi] \wedge \Phi < P)$. In view of the equivalence between $\exists\Phi(\alpha)$ and $\neg\forall\Phi(\neg\alpha)$, the circumscription axiom can be rewritten as

$$\forall\Phi((A[\Phi] \wedge \Phi \leq P) \rightarrow P \leq \Phi).$$

This form of the circumscription axiom is used in many of the examples and theorems which follow.

> Say we have a (simplified) blocks world consisting of two blocks which can be stacked on each other. Suppose we want to describe the situation(s) in which a certain block is red and it is stacked on top of another block. A language for such a blocks world should have separate individual constants representing the two blocks, say a and b, a unary predicate constant *Red* representing the colour property of a block and a binary predicate constant *On* representing the relation between two stacked blocks. A set of axioms, A, describing this system might be:
>
> $$Red(a) \wedge On(a,b)$$
>
> If we want to limit ourselves to models of A in which only a is red, we could either add the axiom $\forall x(Red(x) \rightarrow x = a)$ or we could use circumscription to circumscribe the predicate constant *Red*. The benefit of the latter is that in complex situations in which we don't know exactly how few objects must have the property *Red*, just that it should be as few as possible, the first option does not work but circumscription does.

CIRC(A;*Red*) is given by

$$Red(a) \wedge On(a,b) \wedge \neg\exists\Phi(\Phi(a) \wedge On(a,b) \wedge \Phi < Red)$$

Expressed intuitively, this states that the denotation of a is a member of the denotation of *Red,* the denotation of *On* contains the ordered pair of objects denoted by a and b respectively, and that there is no proper subset of the denotation of *Red* that contains the object denoted by a.

We now show that there are interpretations of the language which are models of the set of axioms, A, but which are not models of CIRC(A;*Red*).

Say I is the following interpretation:

- $|\,I\,|$ = {BLOCKA,BLOCKB}
- a^I = BLOCKA, b^I = BLOCKB
- Red^I = {BLOCKA,BLOCKB}

  On^I = {(BLOCKA,BLOCKB)}.

A is true in I because $a^I \in Red^I$ and $(a^I,b^I) \in On^I$, i.e. I is a model of A.

We rewrite CIRC(A;*Red*) in expanded form:

$$Red(a) \wedge On(a,b) \wedge \forall\Phi((\Phi(a) \wedge On(a,b) \wedge \forall x(\Phi(x) \rightarrow Red(x))) \rightarrow \forall x(Red(x) \rightarrow \Phi(x)))$$

To show that I is not a model of CIRC(A;*Red*), we show that the wff $\forall x(Red(x) \rightarrow x = a)$ is true in every model of CIRC(A;*Red*). (This wff is clearly false in I, so if the wff is entailed by CIRC(A;*Red*), I cannot be a model of CIRC(A;*Red*)).

Let M be any model of CIRC(A;*Red*). We wish to show that any valuation in M, say v, satisfies $\forall x(Red(x) \rightarrow x = a)$. This will be the case if every valuation in M which differs from v at most in the value it assigns to x, satisfies $Red(x) \rightarrow x = a$. Let w be such a valuation in M, but assume that w does not satisfy $Red(x) \rightarrow x = a$, i.e. w satisfies $Red(x)$ but it does not satisfy $x = a$, or in other words, $w(x) \in Red^M$ and $w(x) \neq a^M$.

We know that w satisfies CIRC(A;*Red*) and so w also satisfies the circumscription axiom. So any valuation which differs from w at most on Φ will satisfy $(\Phi(a) \wedge On(a,b) \wedge \forall x(\Phi(x) \rightarrow Red(x))) \rightarrow \forall x(Red(x) \rightarrow \Phi(x))$. Let u be the valuation which differs from w only in as much as $u(\Phi) = Red^M - \{w(x)\}$. Then u satisfies $\Phi(a)$ since $a^M \in Red^M$ and $a^M \neq w(x)$. Also u satisfies $\forall x(\Phi(x) \rightarrow Red(x))$. Hence u must satisfy $\forall x(Red(x) \rightarrow \Phi(x))$. So any valuation differing from u at most on x must satisfy $Red(x) \rightarrow \Phi(x)$. But u itself is such a valuation. In particular then, it must be the case that if $u(x) \in Red^M$ then $u(x) \in u(\Phi)$. But $u(x) = w(x)$ (since u differs from w only in the value it assigns to Φ), and $w(x) \in Red^M$. Therefore $u(x) \in u(\Phi)$, contradicting the choice of u. Hence we reject the assumption that $w(x) \neq a^M$. So w must satisfy $Red(x) \rightarrow x = a$.

The above example shows that circumscribing a predicate P in a set of axioms has the effect of discarding some of the models of the original set of axioms, namely all except those models that have the fewest possible individuals that satisfy P. In order to make the phrase "the fewest possible" precise, we need the notion of P-minimality.

Definition Let A be a finite set of axioms and $\boldsymbol{P}$ a tuple of predicate constants $\langle P_1,..,P_n\rangle$. Suppose M and N are models of A, then M is a ***P*-submodel** of N, written $M \leq^P N$, iff

- $|M| = |N|$
- $K^M = K^N$ for every function constant or predicate constant K not in $\boldsymbol{P}$
- $K^M \subseteq K^N$ for every predicate constant K in $\boldsymbol{P}$.

M is a ***P*-minimal model** of A iff every $\boldsymbol{P}$-submodel of M is identical to M (in other words, iff M is minimal in the usual sense, relative to the partial order $\leq^P$, of having no proper submodels).
■

The theorem below shows that naïve circumscription does in fact reduce the class of models of a set of axioms to precisely the class of minimal models. In other words, all $\boldsymbol{P}$-minimal models of A are models of CIRC(A;$\boldsymbol{P}$) and vice versa.

Theorem 2.1

Let A be any finite set of axioms of a first-order language. Suppose that $\boldsymbol{P}$ is a tuple of predicate constants and let M be a model of A.

Then M is a model of CIRC(A;$\boldsymbol{P}$) iff M is a $\boldsymbol{P}$-minimal model of A.

Proof

Without loss of generality we consider the naïve circumscription of a single predicate P (rather than a tuple of predicates) in a set of axioms, A.

($\Leftarrow$ part) Assume that there is a P-minimal model M of A which is not a model of CIRC(A;P). Since M is a model of A, all the axioms of A are true in M. Consequently, the circumscription axiom, namely

$$\neg\exists\Phi(A[\Phi] \wedge \Phi < P)$$

cannot be true in M, otherwise M would be a model of CIRC(A;P). There must therefore be some valuation in M which does not satisfy the circumscription axiom. In other words, there must be some valuation, say v, that satisfies $A[\Phi] \wedge \Phi < P$, and hence a subset of $|M|^n$ (where n is the arity of P), namely $v(\Phi)$, which is a proper subset of P^M and which is such that if P were reinterpreted as precisely $v(\Phi)$ while all other constants are interpreted as in M, then the resulting interpretation would still be a model of the set of axioms, A.

More precisely, let N be the interpretation constructed as follows:

- $|N| = |M|$
- $K^N = K^M$ for every function or predicate constant K other than P
- $P^N = v(\Phi)$.

Since the valuation v satisfies $A(\Phi)$ and $P^N = v(\Phi)$, N is a model of A. Furthermore, $N \leq^P M$, by construction. However, $N \neq M$ because $P^N = v(\Phi)$ and $P^M \neq v(\Phi)$. This contradicts the assumption that M is a P-minimal model of A.

($\Rightarrow$ part) Assume, on the other hand, that there is a model M of CIRC(A;P) which is not a P-minimal model of A. Since M is not a P-minimal model, there must be another model of A, say M', such that $|M'| = |M|$, $K^{M'} = K^M$ for every function or predicate constant K other than P, and $P^{M'} \subset P^M$ (i.e. $P^{M'}$ is a proper subset of P^M). However, this means that M cannot be a model of the circumscription of P in A, namely $A \wedge \neg\exists\Phi(A[\Phi] \wedge \Phi < P)$, which states that no proper subset of P^M exists which satisfies A. To see this, let v be any valuation in M such that $v(\Phi) = P^{M'}$. Then v satisfies $A[\Phi] \wedge \Phi < P$ and therefore fails to satisfy $\neg\exists\Phi(A[\Phi] \wedge \Phi < P)$.

■

Remark: The 'if' part of the above proof is based on the proof of Theorem 6.6 in [Łukaszewicz 1990] which in turn is based on the sketchy proof in [McCarthy 1980].

Corollary 2.1

If a set of axioms, A, has no ***P***-minimal models, then CIRC(A,***P***) is not satisfiable (i.e. it has no models). In other words, naïve circumscription does not necessarily preserve satisfiability - it depends on whether A has ***P***-minimal models or not.

■

This corollary states a serious weakness of naïve circumscription, since there are satisfiable sets of axioms which have no minimal models. This is shown in the following example [Etherington, Mercer & Reiter 1985]:

> Suppose that A is the set of axioms
>
> $$\exists x(P(x) \wedge \forall y(P(y) \rightarrow \neg(x = s(y))))$$
> $$\forall x(P(x) \rightarrow P(s(x)))$$
> $$\forall xy(s(x) = s(y) \rightarrow x = y)$$
>
> Consider an interpretation, M, where $|\,M\,| = \mathbb{N} = \{0,1,2,\ldots\}$, $P^M = \mathbb{N}$ and s^M is the successor function on $\mathbb{N}$. M is a model of A, so A is satisfiable.
>
> Let R be any model of A and let b^R be any member of P^R. By the second axiom of A, P^R must also contain $s^R(b^R)$, $s^R(s^R(b^R))$, $s^R(s^R(s^R(b^R)))$, etc., and by the third axiom these are all distinct. In other words, the denotation of P in any model of A contains an infinite sequence of elements (in effect, a copy of $\mathbb{N}$).
>
> No matter what model R is, a *P*-submodel of R can be constructed by leaving out a finite initial sequence of the denotation of *P*. For instance, from M we may construct a new model of A, say M', by taking $P^{M'}$ to be the subset $\mathbb{N}^+ = \{1,2,3,\ldots\}$. Therefore every model of A has at least one proper *P*-submodel and consequently A has no *P*-minimal models.
>
> Therefore, by Corollary 2.1, CIRC(A;*P*) is not satisfiable.

If, however, we restrict our attention to sets of axioms which do have minimal models (the so-called well-founded sets of axioms), naïve circumscription does preserve satisfiability. The notion of well-foundedness is interesting because analogous ideas are to be found in the theories of deductive databases and logic programming - see [Apt, Blair & Walker 1988].

Definition Suppose A is a finite set of axioms and ***P*** is a tuple of predicate constants. A is *well-founded with respect to* ***P*** iff each model of A has a ***P***-minimal submodel. A set of axioms A is *well-founded* iff A is well-founded with respect to every ***P***.

■

Theorem 2.2 (Preservation of satisfiability for well-founded theories)
Suppose that a set of axioms, A, is satisfiable. If A is well-founded with respect to a tuple of predicate constants ***P***, then CIRC(A,***P***) is satisfiable.

Proof
If A is satisfiable, then it has a model, say M. Furthermore, if A is well-founded with respect to a tuple of predicate constants ***P***, then M has a ***P***-minimal submodel, say N. By Theorem 2.1, N is also a model of CIRC(A;***P***), therefore CIRC(A;***P***) is satisfiable.

■

No syntactic characterisation of well-foundedness is known. In other words, the definition of well-foundedness is not very functional in the sense that there is no known means of syntactic inspection by which to decide whether an arbitrary set of axioms is well-founded or not. There are, however, procedures to decide whether a set of axioms is universal.

Definition A finite set of first-order axioms is *universal* iff it is logically equivalent to a wff that consists of a prefix comprising zero, one or more universal quantifiers followed by a quantifier-free wff.

■

Following Lifschitz [1986], a larger class of wffs called the almost-universal wffs, for which it is also possible to determine membership algorithmically, can be defined as follows:

Definition A finite set of first-order axioms is *almost-universal with respect to a tuple* ***P*** *of predicate constants* iff it is logically equivalent to a wff that consists of a prefix comprising zero, one or more universal quantifiers followed by a wff whose unnegated atomic formulae involving predicate constants in ***P*** do not contain variables that fall in the scope of any quantifiers other than those in the prefix.

■

Clearly, all universal sets of axioms are also almost-universal with respect to any ***P***.

Theorem 2.3 [Lifschitz 1986]
All sets of axioms which are almost-universal with respect to a tuple of predicate constants ***P*** are well-founded with respect to ***P***.

Proof
Suppose A is a set of axioms which is almost-universal with respect to a tuple of predicate constants ***P***. If A is not satisfiable then A is (trivially) well-founded since every model of A (of which none exist) has a ***P***-minimal submodel.

So let A be a satisfiable set of axioms which is almost-universal with respect to an n-ary predicate constant P. (We consider a single predicate constant P rather than a tuple of predicate constants, for the sake of simplicity). Let M be a model of A.

Let $SUB^P(A,M)$ be the set of all models of A which are P-submodels of M. We will use Zorn's lemma to show that M has a minimal submodel. Let L be any subset of $SUB^P(A,M)$ such that L is linearly ordered with respect to P (i.e. for any two distinct models in L, the denotation of P in one of them is a proper subset of the denotation of P in the other).

Let M' be the interpretation defined by

- $|M'| = |M|$
- $K^{M'} = K^M$ for all constants K besides the predicate constant P
- $P^{M'} = \bigcap_{N \in L} P^N$.

We now wish to show that M' is a lower bound of L. The first step is to show that M' is model

of A. A is almost-universal with respect to P, so it is equivalent to a conjunction of wffs of the form

$$\forall x_1..x_m(N \vee P(t_1) \vee .. \vee P(t_k))$$

where N is a wff not containing any unnegated atoms involving P whose free variables are in $x_1,..,x_m$, and $t_1,..,t_k$ are n-tuples of terms containing no variables beside $x_1,..,x_m$. (Since A is of the form $\forall x_1..x_m(\alpha)$, the reformulation of A is achieved by applying a standard algorithm for writing propositions in conjunctive normal form to α).

Let v be any valuation in M' and let $N \vee P(t_1) \vee .. \vee P(t_k)$ be the unquantified part of one of the conjuncts of A, such that v does not satisfy any of $P(t_1),..,P(t_k)$. (Note that if no such wff exists then M' is immediately a model of A).

Since, by construction, the denotation of P in M' is the intersection of the denotations of P in all the models in L, there must be, for every $i = 1..k$, a model M_i in L such that $P(t_i)$ is not satisfied by v in M_i. (Note that v is a valuation in all of the models in L, since the domain of M' is identical to the domain of each of the models in L). Since L is linearly ordered with respect to the denotation of P, $\{M_1,..,M_k\}$ is also linearly ordered with respect to P, and since $\{M_1,..,M_k\}$ is a finite subset of L, one of $M_1..M_k$ must be the "smallest" with respect to its denotation of P, say M_s. By the choice of $M_1..M_k$, none of $P(t_i)$ are satisfied by v in M_s. Since M_s is a model of A, N must be satisfied by v in M_s. Since the only occurrences of P in N are negated atoms, and $P^{M'} \subseteq P^{M_s}$, N must be satisfied by v in M'.

Therefore M' is a model of A. Moreover, by construction M' is a P-submodel of every model of L.

Zorn's lemma states that if a lower bound can be found for every linearly ordered subset of a given set, the set itself has a minimal element. We have shown that, for every linearly ordered subset L of $SUB^P(A,M)$, we can find a model of A, namely M', such that M' is a P-submodel of every model in L. Therefore a P-minimal model of A exists. Since A has a P-minimal model, it is well-founded with respect to P.

■

Corollary 2.3

Naïve circumscription of a set of axioms which is almost-universal with respect to a tuple of predicate constants, preserves satisfiability.

■

Remark: The significance of this result resides in the fact that a finite set of clauses is a universal wff, and so the class of universal axioms includes logic programs and deductive databases.

The preservation of satisfiability is a problem for all forms of circumscription. A particular weakness of naïve circumscription is its lack of expressive power. We measure the expressive power of circumscription by the new ground atoms entailed by CIRC(A;***P***) but not by A. Consider firstly the case of a ground atom involving a circumscribed predicate constant.

Theorem 2.4

Suppose that a finite set of axioms A is well-founded, $P \in \boldsymbol{P}$ is an n-ary predicate constant and **t** is an n-tuple of ground terms, then

$$\text{CIRC}(A;\boldsymbol{P}) \models P(\mathbf{t}) \text{ iff } A \models P(\mathbf{t}).$$

Proof [Etherington, Mercer & Reiter 1985]

If $A \models P(\mathbf{t})$, then $\text{CIRC}(A;\boldsymbol{P}) \models P(\mathbf{t})$. This follows directly from the definition of CIRC(A;***P***).

If $\text{CIRC}(A;\boldsymbol{P}) \models P(\mathbf{t})$, then $A \models P(\mathbf{t})$, for if not, there is a model M of A in which $P(\mathbf{t})$ is false, but whose P-minimal sub-models must, by Theorem 2.1, satisfy $P(\mathbf{t})$. Suppose M' is a P-minimal sub-model of M. Then $P^{M'} \subseteq P^{M}$. If the n-tuple of objects denoted by **t** belongs to P^{M}, that same n-tuple will also belong to $P^{M'}$. Hence if $P(\mathbf{t})$ is true in M', $P(\mathbf{t})$ will also be true in M, contradicting the choice of M. Therefore M has no P-minimal sub-model, contradicting the well-foundedness of A.

■

This theorem states that no new ground atoms can be entailed by circumscribing the relevant predicate constant in a well-founded theory. This is not a very surprising result since circumscription of a predicate constant is meant to minimize its denotation in the models of the theory. More distressing is the following:

Theorem 2.5

Suppose A is a well-founded set of axioms, $Q \notin P$ is an n-ary predicate constant and t is an n-tuple of ground terms, then

(i) CIRC(A;P) $\models Q$(t) iff A $\models Q$(t)

(ii) CIRC(A;P) $\models \neg Q$(t) iff A $\models \neg Q$(t).

Proof [Etherington, Mercer & Reiter 1985]

(i) If A $\models Q$(t) then CIRC(A;P) $\models Q$(t) follows directly from the definition of CIRC(A;P).

Assume that Q(t) is not semantically entailed by A. Then there is a model of A, say M, in which Q(t) is false. Since A is well-founded, M has a P-minimal submodel, say M'. By the definition of a P-submodel, the interpretation of Q is the same in M and M', since $Q \notin P$. Hence Q(t) is also false in M'. Then, by Theorem 2.1, Q(t) is not semantically entailed by CIRC(A;P).

(ii) The proof is similar to (i).

■

The above theorem states that no new plain or negated ground atoms involving uncircumscribed predicate constants can be entailed by naïve circumscription of well-founded sets of axioms. This is a serious flaw, since it proves that naïve circumscription is inadequate for the formulation of simple common-sense arguments.

As an example, consider the set of axioms A given earlier for the Tweety problem:

$$Bird(Tweety)$$
$$\forall x(Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x))$$

We want to be able to conjecture that Tweety can fly by circumscribing Ab in A, i.e. we want to show that CIRC(A;Ab) $\models Flies(Tweety)$.

However, Theorem 2.5 states that no new positive new ground instances of atomic formulae involving *Flies* can be derived by naïvely circumscribing Ab.

To analyse this problem, consider the following two models of A, namely M and N:

- $|M| = \{TWEETY\}$
- $Tweety^M = TWEETY$
- $Bird^M = \{TWEETY\}$

  $Ab^M = \{\}$

  $Flies^M = \{TWEETY\}$,

- $|N| = \{TWEETY\}$
- $Tweety^N = TWEETY$
- $Bird^N = \{TWEETY\}$

  $Ab^N = \{TWEETY\}$

  $Flies^N = \{\}$.

M and N are both models of A because all the axioms of A are true in both M and N. Furthermore, M and N are both *Ab*-minimal models of A because the denotation of *Ab* cannot be made any smaller in either of them without violating the axioms in A. M is an "intended" *Ab*-minimal model because, since there is no information stating that any objects are abnormal with regard to being able to fly, we want to entertain the possibility that there are none, i.e. that Ab^M is empty. However, our definition of minimality (which we have shown in Theorem 2.1 to be captured precisely by our definition of naïve circumscription) also allows N to be an *Ab*-minimal model of A. Intuitively, one would imagine that one could construct a proper submodel N' of N simply by taking $Ab^{N'} = \{\}$. However, such a re-interpretation of *Ab* violates the second axiom unless *Flies* is also re-interpreted, specifically as $Flies^{N'} = \{TWEETY\}$. Our current version of minimality does not permit this: the interpretation of all constants other than *Ab* must remain fixed.

The reader may feel that the model N (and indeed also the model M) is unnatural in that it has a domain containing the single member TWEETY. After all, it hardly makes sense to use rules of the form "Normally, such and such is the case" for domains containing just one object. One might wonder whether an alternative to allowing *Flies* to vary might be to add an axiom like $\exists x(\neg(x = Tweety))$ to the original set of axioms A. It is indeed quite reasonable to work with such a set of axioms. However, the conjecture *Flies*(*Tweety*) is still blocked - consider the minimal model N' with $|N'| = \{TWEETY, CHARLIE\}$, $Tweety^{N'} = TWEETY$, $Bird^{N'} = \{TWEETY, CHARLIE\}$, $Ab^{N'} = \{TWEETY\}$ and $Flies^N = \{CHARLIE\}$. In order to shrink $Ab^{N'}$ it is still necessary to enlarge $Flies^{N'}$.

The above example suggests that we should allow the denotations of (some selected) predicate constants to vary in the process of minimisation so as to allow the denotation of the predicate constant(s) which are being circumscribed to really be as small as possible. This will require adaptations to our definition of circumscription as well as to our definition of minimality.

Circumscription allowing predicates to vary

We now present a new formulation of circumscription to address the principal flaw of our first definition of circumscription, namely that no new ground atoms involving uncircumscribed predicates are obtainable. This problem is overcome by allowing the denotations of some (other) predicate constants to vary in the process of minimisation.

The new definition of circumscription is expressed, as before, as the conjunction of a second-order circumscription axiom and a given set of first-order axioms:

Definition Suppose that A is a finite set of axioms in a first-order language and the predicate constants in the tuples $\boldsymbol{P} = \langle P_1,..,P_m\rangle$ and $\boldsymbol{Q} = \langle Q_1,..,Q_n\rangle$ are disjoint. The *circumscription of* $\boldsymbol{P}$ *in* A *allowing predicates* $\boldsymbol{Q}$ *to vary*, denoted by CIRC(A;$\boldsymbol{P}$;$\boldsymbol{Q}$), is the second-order formula

$$A \wedge \neg\exists\Phi\Psi(A[\Phi,\Psi] \wedge \Phi < \boldsymbol{P})$$

where Φ is an m-tuple of predicate variables $\langle\Phi_1,..,\Phi_m\rangle$ whose arities correspond to those of the constants in $\boldsymbol{P}$, Ψ is an n-tuple of predicate variables $\langle\Psi_1,..,\Psi_n\rangle$ whose arities correspond to those of the constants in $\boldsymbol{Q}$, and A[Φ,Ψ] is the formula obtained by substituting $\Phi_1,..,\Phi_m$ for $P_1,..,P_m$ and $\Psi_1,..,\Psi_n$ for $Q_1,..,Q_n$ in A. CIRC(A;$\boldsymbol{P}$;$\boldsymbol{Q}$) is therefore the conjunction of the original set of axioms A with the *circumscription axiom* $\neg\exists\Phi\Psi(A[\Phi,\Psi] \wedge \Phi < \boldsymbol{P})$.

■

The circumscription axiom may equivalently be written as

$$\forall\Phi\Psi((A[\Phi,\Psi] \wedge \Phi \leq \boldsymbol{P}) \rightarrow \boldsymbol{P} \leq \Phi).$$

To describe the semantics of circumscription allowing predicates to vary, the previous definitions of submodels and minimality must be adapted slightly:

Definition Let A be a finite set of axioms and P and Q two disjoint tuples of predicate constants. Say M and N are models of A, then M is a *$P;Q$-submodel* of N, written $M \leq^{P;Q} N$, iff

- $|M| = |N|$
- $K^M = K^N$, for every function constant K and predicate constant K not in P or Q
- $K^M \subseteq K^N$, for every predicate constant K in P.

We write $M <^{P;Q} N$ iff $M \leq^{P;Q} N$ but not $N \leq^{P;Q} M$.
M is a *$P;Q$-minimal model* of A iff there is no model N such that $N <^{P;Q} M$.
■

It is interesting to note that if M and N differ only in their denotations of Q, M and N are both $P;Q$-submodels of one another but are not identical to one another, i.e. $M \leq^{P;Q} N$ and $N \leq^{P;Q} M$ but $M \neq N$. In other words, the relation $\leq^{P;Q}$ is not necessarily antisymmetric, and therefore does not necessarily form a partial ordering on the set of models, but rather a *pre-order*, i.e. it is a reflexive, transitive relation.

Like the previous definition of circumscription, the present version of circumscription also picks out precisely the minimal models of a set of axioms. Minimality just happens to have changed its meaning slightly. (The proof is analogous to Theorem 2.1.)

Allowing the denotations of certain predicate constants to vary in the process of minimisation does not solve the problem of the preservation of satisfiability. More precisely, even if A is a satisfiable set of axioms, CIRC(A;P;Q) is not necessarily satisfiable. The problem is exactly the same as for the case where no predicates are allowed to vary. Since the models of CIRC(A;P;Q) are precisely the $P;Q$-minimal models, if we add the circumscription axiom to a set of axioms which has no minimal model, the resulting set of axioms will not have a model, i.e. it will be unsatisfiable. Since there are satisfiable sets of axioms which do not have a minimal model, circumscription does not preserve satisfiability in such cases.

If, however, we confine ourselves to sets of axioms which do have minimal models, circumscription (allowing predicates to vary) does preserve satisfiability. The definition of well-foundedness is adapted accordingly:

Definition Suppose A is a set of axioms and $\boldsymbol{P}$ and $\boldsymbol{Q}$ are two disjoint, finite tuples of predicate constants. A is *well-founded with respect to* $\boldsymbol{P;Q}$ iff each model of A has a $\boldsymbol{P;Q}$-minimal submodel. A is *well-founded* iff A is well-founded with respect to every pair of finite tuples. ■

The example below shows that new ground atoms involving uncircumscribed predicates can be entailed by circumscription if the denotation of such predicate constants are allowed to vary in the process of minimisation.

Say A is the set of axioms given previously for the Tweety example, namely:

$$Bird(Tweety)$$
$$\forall x(Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x))$$

We would like to show that Tweety can fly by circumscribing *Ab* in A. We should allow *Flies* to vary, since the predicate constant *Ab* (whose denotation we wish to minimise) represents abnormality with regard to being able to fly. (In fact, the denotation of *Ab* is intended to be precisely the complement of the denotation of *Flies*. Hence it does not seem sensible to attempt to minimise *Ab* without recognising that the process should affect *Flies*). We therefore want to show that CIRC(A;*Ab*;*Flies*) $\models$ *Flies*(*Tweety*).

Consider all the *Ab*;*Flies*-minimal models of A. One such model is M:

- $|\,M\,| = \{TWEETY\}$
- $Tweety^M = TWEETY$
- $Bird^M = \{TWEETY\}$

$Ab^M = \{\}$

$Flies^M = \{TWEETY\}$.

M is a model of A because all the axioms of A are true in M. M is a *Ab*;*Flies*-minimal model of A because no *Ab*;*Flies*-submodel of M can be found whose denotation of *Ab* is a proper subset of Ab^M, since Ab^M is already as small as it can be.

The question is whether every *Ab*;*Flies*-minimal model of A must have the object denoted by *Tweety* in the denotation of *Flies*. Suppose there is an *Ab*;*Flies*-minimal model of A,

say N, which does not have the object denoted by *Tweety* in the denotation of *Flies*. Then Ab^N must contain the object denoted by *Tweety*, otherwise N would not satisfy the second axiom of A. But then we can form an interpretation N' from N by removing the object denoted by *Tweety* from the denotation of *Ab* and adding it to the denotation of *Flies*. We note that N' is a model of A because N is a model of A and N' is identical to N except for the denotations of *Ab* and *Flies* which differ only in that the object denoted by *Tweety* is a member of one and not the other, ensuring that both axioms of A are true in N'. More importantly, however, we note that N' is an *Ab*;*Flies*-submodel of N, since its domain is the same as that of N, the denotations of all its constants besides *Ab* and *Flies* are exactly the same as those of N. Furthermore, N is not an *Ab*;*Flies*-submodel of N' because its denotation of *Ab* is not a subset of the denotation of *Ab* in N'. This contradicts the assumption that N is an *Ab*;*Flies*-minimal model of A.

So we see that every *Ab*;*Flies*-minimal model of A must have the object denoted by *Tweety* in the denotation of *Flies*. Since the *Ab*;*Flies*-minimal models of A are precisely the models of CIRC(A;*Ab*;*Flies*), we conclude that CIRC(A;*Ab*;*Flies*) $\models$ *Flies*(*Tweety*).

The following is another example of circumscribing an *Ab* predicate, but in this case, a 2-ary *Ab* predicate:

Consider the default rule that, normally, we give gifts to friends on their birthdays. This can be expressed in the following wff:

$$\forall xy(Birthday(y) \wedge Friend(x,y) \wedge \neg Ab(x,y) \rightarrow GivesGift(x,y))$$

If Anne's friend Alice has a birthday, can we use this rule to conjecture that Anne will give a gift to Alice? We firstly need to represent this further knowledge by means of axioms:

Birthday(*Alice*)

Friend(*Anne*,*Alice*)

Naïve circumscription of *Ab* in the set of axioms (call the set A) will not entail *GivesGift*(*Anne*,*Alice*), since Theorem 2.5 states that no new atoms involving uncircumscribed predicate constants can be entailed by naïve circumscription.

We therefore allow *GivesGift* to vary while circumscribing *Ab* in A, not only to get around Theorem 2.5, but also because a change in the denotation of *Ab* may be expected to affect the denotation of *GivesGift*. We therefore want to show that CIRC(A;*Ab*;*GivesGift*) $\models$ *GivesGift*(*Anne*,*Alice*), or, in other words, that *GivesGift*(*Anne*,*Alice*) is true in every *Ab*;*GivesGift*-minimal model of A.

Let M be an *Ab*;*GivesGift*-minimal model of A, and assume that *GivesGift*(*Anne*,*Alice*) is not true in M.

Now construct the interpretation M' of A as follows:

- $|M'| = |M|$
- $Alice^{M'} = Alice^{M}$, $Anne^{M'} = Anne^{M}$
- $Birthday^{M'} = Birthday^{M}$, $Friend^{M'} = Friend^{M}$

$Ab^{M'} = Ab^{M} - \{(Anne^{M},Alice^{M})\}$

$GivesGift^{M'} = GivesGift^{M} \cup \{(Anne^{M},Alice^{M})\}$.

To see that M' is a model of A, consider just the first axiom (the other two axioms of A are patently true in M'). The denotations of *Birthday* and *Friend* are the same in M' as in M. We may therefore confine our attention to the possible violation of the axioms by the change to the denotations of *Ab* and *GivesGift*. The only problem which could occur by removing the ordered pair $(Anne^{M},Alice^{M})$ from the denotation of *Ab* is that this might allow valuations assigning $Anne^{M}$ to x and $Alice^{M}$ to y to satisfy the antecedent of the (unquantified) axiom. However, even if this is the case, the ordered pair $(Anne^{M},Alice^{M})$ has been added to the denotation of *GivesGift* to ensure that the same valuation also satisfies the consequent of the axiom.

Since the domains of M and M' are the same, as well as the denotations of all constants (besides *Ab* and *GivesGift*), and since $Ab^{M'}$ is a subset of Ab^{M}, M' is a *Ab*;*GivesGift*-submodel of M. However, Ab^{M} is not a subset of $Ab^{M'}$ (because the ordered pair $(Anne^{M},Alice^{M})$ must be in Ab^{M} to satisfy the first axiom in M), M is not a *Ab*;*GivesGift*-submodel of M', which contradicts the supposition that M is an *Ab*;*GivesGift*-minimal model of A. Therefore, *GivesGift*(*Anne*,*Alice*) must be true in every *Ab*;*GivesGift*-minimal model of A, i.e. CIRC(A;*Ab*;*GivesGift*) $\models$ *GivesGift*(*Anne*,*Alice*).

The final examples illustrate circumstances in which more than one predicate is permitted to vary and in which more than one predicate is minimised simultaneously.

Assume that birds normally fly and build nests and that we are interested in the capabilities of a bird called Tweety. Consider two alternative sets of axioms to represent this knowledge:

A: $\forall x(Bird(x) \wedge \neg Ab_1(x) \rightarrow Flies(x))$
$\forall x(Bird(x) \wedge \neg Ab_2(x) \rightarrow BuildsNests(x))$
$Bird(Tweety)$

B: $\forall x(Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x) \wedge BuildsNests(x))$
$Bird(Tweety)$

Suppose that we wish to formalise the common-sense inference that Tweety can fly. From the set of axioms A this can be achieved (as done previously) by forming CIRC(A;Ab_1;*Flies*). From B, however, the same strategy does not work, i.e. it is not true that CIRC(B;*Ab*;*Flies*) $\models$ *Flies*(*Tweety*). Consider the interpretation N of the language for B:

- $|N| = \{TWEETY\}$
- $Tweety^N$ = TWEETY
- $Bird^N = \{TWEETY\}$

$Ab^N = \{TWEETY\}$

$Flies^N = BuildsNests^N = \{\}$.

It is easy to confirm that N is a model of B. Furthermore, *Flies*(*Tweety*) is not true in N. We assert that N is a *Ab*;*Flies*-minimal model of B. To see this, assume the existence of a proper *Ab*;*Flies*-submodel of N, say N'. $Ab^{N'}$ must be empty for N' to be a proper submodel of N, but the only way to prevent N' from violating the default axiom would be to remove TWEETY from $Bird^{N'}$ (which would violate the axiom *Bird*(*Tweety*)) or to add TWEETY to $Flies^{N'}$ and $BuildsNests^{N'}$ (but we may not change the denotation of *BuildsNests* since the denotations of all constants besides *Ab* and *Flies* must be the same in N' as in N for N' to be a submodel of N). So the existence of a proper submodel of N is impossible, confirming that N is an *Ab*;*Flies*-minimal model of B.

The above examination of the existence of a model of B in which the denotation of *Ab* is empty suggests that *BuildsNests* should also be allowed to vary in the circumscription of *Ab*, i.e. to form CIRC(B;*Ab*;*Flies*,*BuildsNests*). This is not really surprising since the *Ab* predicate in B represents abnormality with regard to flying and building nests, so minimising the number of abnormal individuals can be expected to have an effect on the number of individuals that can fly and build nests.

The reader will find it a useful exercise to check that CIRC(B;*Ab*;*Flies*,*BuildsNests*) $\models$ *Flies*(*Tweety*).

Suppose that we now wish to formalise the common-sense argument that Tweety can fly and build nests. To do so from A, we need to circumscribe both Ab_1 and Ab_2 allowing *Flies* and *BuildsNests* to vary. To do so from B, we need to circumscribe *Ab* allowing *Flies* and *BuildsNests* to vary. We will show how it works in the case of A, i.e. we will show that CIRC(A;Ab_1,Ab_2;*Flies*,*BuildNests*) $\models$ *Flies*(*Tweety*) $\wedge$ *BuildsNests*(*Tweety*).

Let M be any Ab_1,Ab_2;*Flies*,*BuildNests*-minimal model of A. Suppose, however, that *Flies*(*Tweety*) $\wedge$ *BuildsNests*(*Tweety*) is not true in M, i.e. that $Tweety^M \notin Flies^M$ or $Tweety^M \notin BuildsNests^M$. As a consequence, note that in order not to violate either of the default axioms of A, $Tweety^M$ must be in Ab_1^M or in Ab_2^M.

Now construct an interpretation M' of the language of A as follows:

- $|M'| = |M|$
- $Tweety^{M'} = Tweety^M$
- $Bird^{M'} = Bird^N$

$Ab_1^{M'} = Ab_1^M - \{Tweety^M\}$

$Ab_2^{M'} = Ab_2^M - \{Tweety^M\}$

$Flies^{M'} = Flies^M \cup \{Tweety^M\}$

$BuildsNests^{M'} = BuildsNests^M \cup \{Tweety^M\}$.

To see that M' is a model of A, note that the denotation of *Bird* is the same in M' as in M. The only way that $\forall x(Bird(x) \wedge \neg Ab_1(x) \rightarrow Flies(x))$ could be violated in M' would be for $Flies^{M'}$ to be a proper subset of $Flies^M$ (which is not the case) or for $Ab_1^{M'}$ to be a proper subset of Ab_1^M (which could be the case). Suppose $Ab_1^{M'} \subset Ab_1^M$, i.e. that $Tweety^M$

$\in Ab_1^M$. Then M' still satisfies the axiom since $Tweety^M$ is added to the denotation of *Flies*. A similar argument shows that the axiom $\forall x(Bird(x) \wedge \neg Ab_2(x) \rightarrow BuildsNests(x))$ is also true in M'. *Bird*(*Tweety*) must also be true in M' since it is true in M and the denotations of *Bird* and *Tweety* are the same in M and M'.

By construction, M' is an $Ab_1,Ab_2;Flies,BuildNests$-submodel of M, since $Ab_1^{M'} \subseteq Ab_1^M$ and $Ab_2^{M'} \subseteq Ab_2^M$ and the denotations of all other constants (besides *Flies* and *BuildsNests*) are the same. However, M is not an $Ab_1,Ab_2;Flies,BuildNests$-submodel of M', since both $Ab_1^M \subseteq Ab_1^{M'}$ and $Ab_2^M \subseteq Ab_2^{M'}$ cannot be true due to the fact that $Tweety^M$ must be in Ab_1^M or in Ab_2^M but is not in either of $Ab_1^{M'}$ or $Ab_2^{M'}$. This contradicts the supposition that M is an $Ab_1,Ab_2;Flies,BuildNests$-minimal model of A in which *Flies*(*Tweety*) $\wedge$ *BuildsNests*(*Tweety*) is not true. So CIRC(A;$Ab_1,Ab_2;Flies,BuildNests$) $\models$ *Flies*(*Tweety*) $\wedge$ *BuildsNests*(*Tweety*).

We have seen in the examples above that circumscription allowing predicates to vary can give new ground atoms involving those predicate constants whose denotations are allowed to vary in the process of minimisation. However, no new (plain or negated) ground atoms involving uncircumscribed predicate constants whose denotations are *not* allowed to vary in the process of minimisation, can be reached, as shown in the following theorem.

Theorem 2.6

Let A be a finite set of axioms of a first-order language and $\boldsymbol{P}$ and $\boldsymbol{Q}$ two disjoint tuples of predicate constants, let R be a k-ary predicate constant not in $\boldsymbol{P}$ or $\boldsymbol{Q}$ and let $\mathbf{t}$ be a k-tuple of ground terms in the language. If A is well-founded with respect to $\boldsymbol{P;Q}$ then

(i) CIRC(A;$\boldsymbol{P}$;$\boldsymbol{Q}$) $\models R(\mathbf{t})$ iff A $\models R(\mathbf{t})$

(ii) CIRC(A;$\boldsymbol{P}$;$\boldsymbol{Q}$) $\models \neg R(\mathbf{t})$ iff A $\models \neg R(\mathbf{t})$.

Proof [Etherington 1988]

The proof is essentially an alphabetic variant of Theorem 2.5, and is therefore not repeated.

■

However, new ground atoms involving uncircumscribed, unvarying predicates can be obtained if the denotations of other constants, specifically function constants, are allowed to vary in the process of minimisation. This brings us to the next formulation of circumscription.

Standard circumscription

As a motivation for a further adaptation of our definition of circumscription, consider the following example due to Reiter [1980]:

> Bill is married and lives in Vancouver. Normally, spouses live in the same town. Common-sense would then suggest that Bill's wife also lives in Vancouver. A language in which to formalise this common-sense argument could contain individual constants *Bill* and *Vancouver*, and two function constants *livesin* and *wife*. (Strictly speaking, we should use a many-sorted language to avoid meaningless terms such as *wife*(*Vancouver*), but, in the interest of readability, we ignore the technical complications). The following axioms are a rather naïve attempt to express the system in first-order logic:
>
> $\forall x(\neg Ab(x) \rightarrow livesin(x) = livesin(wife(x)))$
>
> $livesin(Bill) = Vancouver$

An example of an intended interpretation M is:

- $|M| = \{BILL, BILLSWIFE, VANCOUVER\}$
- $Bill^M = BILL$, $Vancouver^M = VANCOUVER$
- $Ab^M = \{\}$

  $=^M$ is the identity relation on M
- $livesin^M(BILL) = VANCOUVER$

  $livesin^M(BILLSWIFE) = VANCOUVER$

  $wife^M(BILL) = BILLSWIFE$.

It is clear that M is a model of the axioms. However, there are other models of this set of axioms in which Bill's wife does not live in Vancouver, for example, the interpretation N:

- $|N| = \{BILL, BILLSWIFE, VANCOUVER, SASKATOON\}$
- $Bill^N = BILL$, $Vancouver^N = VANCOUVER$
- $Ab^N = \{BILL\}$

  $=^N$ is the identity relation on N
- $livesin^N(BILL) = VANCOUVER$

  $livesin^N(BILLSWIFE) = SASKATOON$

  $wife^N(BILL) = BILLSWIFE$.

This is the kind of model which circumscription is intended to eliminate. We therefore circumscribe *Ab* in the set of axioms. Since we don't have any information to say that Bill is abnormal with regard to where he and his wife live, the resulting axioms should entail that Bill's wife also lives in Vancouver. In other words, we form CIRC(A;*Ab*) so as to entail *livesin*(*wife*(*Bill*)) = *Vancouver*. However, Theorem 2.5 states that no new ground atoms involving uncircumscribed predicate constants (in this case $=$) can be obtained by circumscription. We may consider allowing $=$ to vary in the process of minimisation, i.e. by forming CIRC(A;*Ab*; $=$). However, this would be counter-productive, since it would allow models where anything is equal to anything else. We specifically want the denotation of $=$ to be the identity relation on the domain.

The most obvious solution is to allow the denotation of the function constant *livesin* to vary in the process of minimisation, i.e. by forming CIRC(A;*Ab*;*livesin*). This is a reasonable thing to do since the *Ab* predicate constant is intended to apply to those individuals who do not live in the same city as their wives. Consequently, reducing the number of abnormal individuals implicitly influences the function *livesin*.

To make the need for this innovation even clearer, here is another example:

Consider the default rule that the biological father of a brother and sister are normally the same person:

$$\forall xy(Siblings(x,y) \wedge \neg Ab(x,y) \rightarrow fatherof(x) = fatherof(y))$$
$$\forall xy(Siblings(x,y) \rightarrow Siblings(y,x))$$

(The second axiom is included to cut out a class of nonsense models.)

Say we want to use this default rule to be able to conjecture that James, who is the father of Jack, is also the father of Jack's sister Jill. To do this we first need to add some axioms affirming the necessary relationships, forming the complete set of axioms A:

$$Siblings(Jack,Jill)$$
$$fatherof(Jack) = James$$

We might hope that the desired result could be achieved by circumscribing the predicate constant *Ab* in A, i.e. that CIRC(A;*Ab*) $\models$ *fatherof(Jill)* = *James*. However, an *Ab*-minimal model of A can be constructed with some other object (eg. JOE) playing the role of Jill's father. (This model must necessarily have (JACK,JILL) in the denotation of *Ab*.) Once again, allowing the denotation of = to vary would be counter-productive.

Since the *Ab* predicate constant is specifically meant to represent "abnormality with respect to having the same father", the most obvious solution is to allow the denotation of the function constant *fatherof* to vary in the process of minimisation, i.e. by forming CIRC(A;*Ab*;*fatherof*).

Instead of only allowing the denotation of certain *predicate* constants to vary in the process of minimisation, we therefore also wish to allow the denotations of selected *function* constants to vary (and, since individual constants may be viewed as 0-ary function constants, it follows that we will permit the denotations of individual constants to vary as well). The idea is due to Lifschitz, who in [1985] called this form 'second-order parallel circumscription'. We shall refer to it as standard circumscription.

The definition of standard circumscription is expressed, once again, as the conjunction of a second-order circumscription axiom and a given set of axioms.

Definition Suppose that A is a finite set of axioms in a first-order language, $\boldsymbol{P} = \langle P_1,..,P_m\rangle$ is a tuple of predicate constants and $\boldsymbol{S} = \langle S_1,..,S_n\rangle$ is a tuple of individual, predicate or function constants disjoint from $\boldsymbol{P}$. The *circumscription of* $\boldsymbol{P}$ *in A allowing the constants* $\boldsymbol{S}$ *to vary*, denoted by CIRC(A;$\boldsymbol{P}$;$\boldsymbol{S}$), is the second-order formula

$$A \wedge \neg\exists\Phi\boldsymbol{X}(A[\Phi,\boldsymbol{X}] \wedge \Phi < \boldsymbol{P})$$

where Φ is an m-tuple of predicate variables $\langle\Phi_1,..,\Phi_m\rangle$ whose arities correspond to those of the constants in $\boldsymbol{P}$, $\boldsymbol{X}$ is an n-tuple of individual, predicate or function variables $\langle X_1,..,X_n\rangle$ similar to $S_1,..,S_n$, and A[Φ,$\boldsymbol{X}$] is the formula obtained by substituting $\Phi_1,..,\Phi_m$ for $P_1,..,P_m$ and $X_1,..,X_n$ for $S_1,..,S_n$ in A. CIRC(A;$\boldsymbol{P}$;$\boldsymbol{S}$) is therefore the conjunction of the original set of axioms A with the *circumscription axiom* $\neg\exists\Phi\boldsymbol{X}(A[\Phi,\boldsymbol{X}] \wedge \Phi < \boldsymbol{P})$.

■

To describe the semantics of standard circumscription, the definition of minimality must be adapted accordingly.

Definition Let A be a finite set of axioms, $\boldsymbol{P}$ a tuple of predicate constants and $\boldsymbol{S}$ a tuple of individual, predicate or function constants. Suppose M and N are models of A, then M is a ***P;S-submodel*** of N, written $M \leq^{P;S} N$, iff

- $|M| = |N|$
- $K^M = K^N$, for every (individual, function or predicate) constant K not in $\boldsymbol{P}$ or $\boldsymbol{S}$
- $K^M \subseteq K^N$, for every predicate constant K in $\boldsymbol{P}$.

We write $M <^{P;S} N$ iff $M \leq^{P;S} N$ but not $N \leq^{P;S} M$.
M is a ***P;S-minimal model*** of A iff there is no model N such that $N <^{P;S} M$.
■

Like the previous versions of circumscription, the latest version also picks out precisely the corresponding minimal models. (The proof of this is analogous to that of Theorem 2.1.) This fact can be used to show that circumscription with varying function constants does support the desired conjectures in the above examples:

> To show that CIRC(A;*Ab*;*livesin*) $\models$ *livesin*(*wife*(*Bill*)) = *Vancouver*, consider the *Ab*;*livesin*-minimal models of A. Suppose M to be any *Ab*;*livesin*-minimal model of A and assume that $livesin^M$(BILLSWIFE) = SASKATOON (or any object in the domain besides the denotation of *Vancouver*). For M to be a model of A, the denotation of *Bill* must be in the denotation of *Ab*.
>
> Construct an interpretation N as follows:
>
> - $|N| = |M|$
> - $Bill^N = Bill^M$ and $Vancouver^N = Vancouver^M$
> - $Ab^N = \{\}$
> - $wife^N = wife^M$
>
> $livesin^N = livesin^M$ except that $livesin^N$(BILLSWIFE) = $Vancouver^N$.
>
> It is easy to see that N is a model of A because all the axioms of A are true in N.

Furthermore, by construction, N is an *Ab;livesin*-submodel of M, since N only differs from M in its denotations of *Ab* and *livesin*, and since $Ab^N \subseteq Ab^M$. However, M is not an *Ab;livesin*-submodel of N, since $Ab^M \not\subseteq Ab^N$. This contradicts the supposition that M is an *Ab;livesin*-minimal model of A and hence also the assumption that $livesin^M$(BILLSWIFE) can be anything besides the denotation of *Vancouver*. In other words, *livesin*(*wife*(*Bill*)) = *Vancouver* is entailed by every *Ab;livesin*-minimal model of A, and therefore by CIRC(A;*Ab;livesin*).

The new version of circumscription also works for the example about Jack and Jill's father:

To show that CIRC(A;*Ab;fatherof*) $\models$ *fatherof*(*Jill*) = *James*, consider the *Ab;fatherof*-minimal models of A. Suppose M to be any *Ab;fatherof*-minimal model of A and assume that $fatherof^M(Jill^M)$ is some object in the domain, say s, other than the denotation of *James*.

Construct an interpretation M' as follows: let $|M'| = |M|$, let the denotations of all individual constants be the same in M' as in M and let the denotation of *Siblings* be the same too. However, let $Ab^{M'} = Ab^M - \{(Jack^M,Jill^M),(Jill^M,Jack^M)\}$ and let $fatherof^{M'} = fatherof^M$ except that $fatherof^{M'}(Jill^{M'}) = James^{M'}$. (Note that both ordered pairs $(Jack^M,Jill^M)$ and $(Jill^M,Jack^M)$ must be in Ab^M to satisfy the first axiom since both ordered pairs are in $Siblings^M$ (in order to satisfy the fourth and third axioms) and $fatherof^M(Jill^M) = s \neq fatherof^M(Jack^M) = James^M$.)

M' is a model of A because all axioms of A are true in M': To confirm that the first axiom is true in M', consider a valuation v in M'. If (v(*x*),v(*y*)) is anything besides $(Jack^{M'},Jill^{M'})$ or $(Jill^{M'},Jack^{M'})$, v must satisfy the first axiom, since any valuation w in M satisfies the first axiom, and M' is identical to M in its denotations of all individual constants as well as that of *Siblings*, and the denotation of *Ab* in M' only differs from that of M in that it does not contain the ordered pairs $(Jack^M,Jill^M)$ and $(Jill^M,Jack^M)$. Let (v(*x*),v(*y*)) therefore be $(Jack^{M'},Jill^{M'})$. Well $(Jack^{M'},Jill^{M'}) \in Siblings^M$, $(Jack^{M'},Jill^{M'}) \notin Ab^{M'}$ and $fatherof^{M'}(Jill^{M'}) = fatherof^{M'}(Jack^{M'}) = James^{M'}$. Therefore v satisfies the first axiom. A similar argument holds if (v(*x*),v(*y*)) is $(Jill^M,Jack^M)$. The first axiom is therefore true in M' since it is satisfied by all valuations in M'.

It is straightforward to check that the final three axioms of A are also true in M'. M' is therefore a model of A.

Furthermore, by construction, M' is an *Ab;fatherof*-submodel of M, since M' only differs from M in its denotations of *Ab* and *fatherof*, and since $Ab^{M'} \subseteq Ab^{M}$. However, M is not an *Ab;fatherof*-submodel of M', since $Ab^{M} \not\subseteq Ab^{M'}$. This contradicts the supposition that M is an *Ab;fatherof*-minimal model of A and hence also the assumption that $fatherof^{M}(Jill^{M})$ can be anything besides the denotation of *James*. In other words, *fatherof(Jill)* = *James* is entailed by every *Ab;fatherof*-minimal model of A, and therefore by CIRC(A;*Ab;fatherof*).

Standard circumscription shares a flaw with the previous two versions, namely that they do not necessarily preserve satisfiability. (This can be shown with the same example as given for naïve circumscription.) As with the previous versions, standard circumscription can be shown to be satisfiable for the well-founded sets of axioms, which are defined in this case as follows:

Definition Suppose A is a finite set of axioms, $\boldsymbol{P}$ is a tuple of predicate constants and $\boldsymbol{S}$ is a tuple of individual, predicate or function constants disjoint from $\boldsymbol{P}$. A is *well-founded with respect to* $\boldsymbol{P;S}$ iff each model of A has a $\boldsymbol{P;S}$-minimal submodel. A is *well-founded* iff A is well-founded with respect to every pair of finite tuples $\boldsymbol{P}$ and $\boldsymbol{S}$.
■

We have seen in the previous section that circumscription allowing only predicates to vary does not allow new positive ground atoms to be entailed for those predicates which are not allowed to vary in the process of minimisation. However, for standard circumscription, new ground atoms can be entailed for uncircumscribed predicates which are not allowed to vary in the process of minimisation. This is shown in the following example from [Etherington 1988].

Suppose we have the set of axioms A given by $P(a) \wedge P(b) \wedge Q(b)$ and we circumscribe P in A allowing the individual constant a to vary.

$$\text{CIRC}(A;P;a) = A \wedge \forall\Phi x((A[\Phi,x] \wedge \forall y(\Phi(y) \rightarrow P(y))) \rightarrow \forall z(P(z) \rightarrow \Phi(z)))$$
$$= A \wedge \forall\Phi x((\Phi(x) \wedge \Phi(b) \wedge Q(b) \wedge \forall y(\Phi(y) \rightarrow P(y))) \rightarrow \forall z(P(z) \rightarrow \Phi(z)))$$

We show that CIRC(A;P;a) $\models Q(a)$ as follows:

Let M be any $P;a$-minimal model of A such that the denotation of a is not in the denotation of Q. We note that to be a model of A, the denotation of Q must contain the denotation of b. It follows from the assumption above that the denotations of a and b in M are therefore distinct. The denotation of P in M must therefore contain (at least) two objects, namely a^M and b^M, for M to be a model of A.

Now construct an interpretation M' of A which is identical to M except in its denotation of a: Let $a^{M'} = b^M$. It is easy to check that M' is a model of A. We note that M' is a $P;a$-submodel of M, since all the constants in M' (beside P and a) have the same denotations as in M. However, M is not a $P;a$-submodel of M' because the denotations of a and b in M' are one and the same object, requiring the denotation of P in M' to only contain one object, namely $a^{M'}$, i.e. $P^{M'}$ is a proper subset of P^M. This contradicts the assumption that M is a $P;a$-minimal model of A and proves that every $P;a$-minimal model of A must have the denotation of a in the denotation of Q.

Since the $P;a$-minimal models of A are precisely the models of CIRC(A;$P;a$), every model of CIRC(A;$P;a$) must have the denotation of a in the denotation of Q. In other words, CIRC(A;$P;a$) $\models Q(a)$.

The Yale shooting problem

A recent and famous example intended to reveal limitations in the forms of circumscription discussed thus far is the Yale shooting problem, constructed by Hanks and McDermott [1987]. Before stating the problem itself, some background explanation is required concerning how knowledge about systems in which changes occur can be expressed by means of the situation calculus.

All the examples we have considered so far, eg. the Blocks World and Tweety, have only involved reasoning about states of the system which do not change. In other words, we have never attempted to reason about the effects of actions like moving a block or clipping Tweety's wings.

The situation calculus is a fairly standard way, invented by McCarthy, of expressing (in a non-modal first-order language) knowledge about the changes caused to the state of a system by actions. For our purposes, a situation calculus language contains individual constants of three

types, namely **situations** (which each represent the state of the system at a particular time), **actions** (which represent certain actions performed or events that occur in the system) and **properties** (which represent certain properties which hold in a situation), a 2-ary function constant *result* which is used to express the change of situation caused by an action, eg. *result*(a,s) represents the resulting situation produced by performing action a in situation s, and finally, a 2-ary predicate constant *Holds* which is used to express what properties hold in specific situations, eg. *Holds*(p,s) states that property p holds in situation s.

Formalising knowledge about the changes brought about by actions in a system has turned out to be particularly tricky. Normally, properties are unaffected by actions, i.e. a given action changes a small part of the system and leaves the rest unchanged. For example, if I pick up a block, the positions and colours of all the other blocks remain the same. The problem is that models of the new state of the system can exist in which all sorts of things have changed. One way to remedy the situation is to add so-called frame axioms, eg.

$$\forall x(Holds(ColourOfBlockA,x) \rightarrow Holds(ColourOfBlockA,result(MoveBlockA,x)))$$

and

$$\forall x(Holds(ColourOfBlockA,x) \rightarrow Holds(ColourOfBlockA,result(PaintBlockB,x))).$$

This strategy requires a potentially enormous number of frame axioms of this type specifying all the properties that do not change from one situation to another. The question whether there is a more satisfactory strategy than the inclusion of frame axioms is known as the frame problem. One solution to the frame problem is to use a single general axiom which specifies that most things do not change, eg.

$$\forall psa(Holds(p,s) \wedge \neg Ab(p,s,a) \rightarrow Holds(p,result(a,s)))$$

Expressed intuitively, this axiom (which we shall call a persistence axiom) states that, normally, for any property p which holds in situation s, property p will still hold in the situation resulting from the action a. The idea is then to circumscribe *Ab* in this axiom, to make its denotation only contain those combinations of properties, situations and actions for which the action in a specific situation is known to change a property.

We are now ready to deal with the Yale shooting problem. The story is simply that when a gun is fired while it is loaded, a person (commonly referred to as Fred) dies.

For this system, the relevant language consists of a situation *s0*, two actions, *wait* and *shoot*, and two properties, *loaded* and *alive* (together with the other constant symbols needed to employ the situation calculus, namely *result*, *Holds* and *Ab*). Notice that the individual constants represent abstract concepts such as the property of being loaded, rather than concrete objects like birds, blocks or a gun.

In the original situation (*s0*) the gun is loaded and Fred is alive. Let A be the set of axioms consisting of the persistence axiom (given above) together with the following:

$$\forall x(Holds(loaded,x) \rightarrow \neg Holds(alive,result(shoot,x)) \wedge \neg Holds(loaded,result(shoot,x)))$$
$$Holds(loaded,s0)$$
$$Holds(alive,s0)$$

With these axioms we can conclude (without using circumscription) that $\neg Holds(alive,result(shoot,s0))$. However, if the actions *wait* and then *shoot* are performed in succession, can we still conclude that Fred dies, without using circumscription, i.e. is $\neg Holds(alive,result(shoot,result(wait,s0)))$ entailed by A? There are no axioms in A involving the *wait* action, so only the persistence axiom is of relevance. Since no axioms confirm or deny the abnormality of the result of waiting, obviously the wff $\neg Holds(alive,result(shoot,result(wait,s0)))$ is not entailed by A without using circumscription.

This is a scenario in which we would like to apply circumscription to minimise the *Ab* predicate and hence conjecture that the gun is still loaded after waiting. (We should then be able to conclude that Fred will not be alive after the gun is subsequently discharged.) In other words, we want to determine whether $CIRC(A;Ab) \models Holds(loaded,result(wait,s0))$. However, Theorem 2.5 states that no new ground atoms involving uncircumscribed predicate constants can be entailed by naïve circumscription. This is because there are models of A in which the gun (mysteriously) becomes unloaded during the waiting process (and even models in which Fred is no longer alive!), preventing the required atom from being entailed.

Consider the following two interpretations M and N:

- $|M| = \{S0,S1,S2\} \cup \{WAIT,SHOOT\} \cup \{LOADED,ALIVE\}$
- $s0^M = S0$

  $wait^M = WAIT$, $shoot^M = SHOOT$

  $loaded^M = LOADED$, $alive^M = ALIVE$
- $Holds^M = \{(LOADED,S0),(ALIVE,S0),(LOADED,S1),(ALIVE,S1)\}$

  $Ab^M = \{(LOADED,S0,SHOOT),(ALIVE,S0,SHOOT),$
  $(LOADED,S1,SHOOT),(ALIVE,S1,SHOOT)\}$
- $result^M(WAIT,S0) = S1$

  $result^M(WAIT,S1) = S1$

  $result^M(WAIT,S2) = S2$

  $result^M(SHOOT,S0) = S2$

  $result^M(SHOOT,S1) = S2$

  $result^M(SHOOT,S2) = S2,$

- $|N| = \{S0,S1,S2\} \cup \{WAIT,SHOOT\} \cup \{LOADED,ALIVE\}$
- $s0^N = S0$

  $wait^N = WAIT$, $shoot^N = SHOOT$

  $loaded^N = LOADED$, $alive^N = ALIVE$
- $Holds^N = \{(LOADED,S0),(ALIVE,S0),(ALIVE,S1)\}$

  $Ab^N = \{(LOADED,S0,SHOOT),(ALIVE,S0,SHOOT),(LOADED,S0,WAIT)\}$
- $result^N(WAIT,S0) = S1$

  $result^N(WAIT,S1) = S1$

  $result^N(WAIT,S2) = S2$

  $result^N(SHOOT,S0) = S2$

  $result^N(SHOOT,S1) = S1$

  $result^N(SHOOT,S2) = S2.$

To see that M and N are both models of A, we consider only the persistence axiom (it is easy to check that the remaining three axioms are true in both M and N). All the property-situation combinations in the denotation of *Holds* (in M and N) for which an action would change the property are listed in the denotation of *Ab*. This ensures that the persistence axiom is true in both M and N.

To see that M and N are both models of CIRC(A;*Ab*), i.e. that they are both *Ab*-minimal models of A, all we need to do is to check whether each ordered triplet in the denotation of *Ab* is really necessary. In M, removal of any of these triplets will violate the persistence axiom since the necessary ordered pairs (required to satisfy the consequent of the persistence axiom) are not in $Holds^M$. For example, (LOADED,S0,SHOOT) is necessary because (LOADED,S0) is in $Holds^N$ but (LOADED,S2) is not. In N, all triplets are essential for the same reason. For example, the triplet (LOADED,S0,WAIT) is necessary because (LOADED,S0) is in $Holds^N$ but (LOADED,S1) is not. Therefore M and N are both *Ab*-minimal models of A.

The wff *Holds*(*loaded*,*result*(*wait*,*s0*)) is thus true in M but not in N, which explains why it is not entailed by CIRC(A;*Ab*).

This is similar to that which occurred in the Tweety problem. In the case of Tweety, there were some models which had the denotation of *Tweety* in the denotation of *Ab* and others which did not. The solution was to allow the denotation of *Flies* to vary in the process of minimisation. The reason why we allowed the denotation of *Flies* to vary (as opposed to other predicate or individual constants in the language) was because the abnormality predicate specifically represented abnormality with respect to flying, so we should expect that minimising the denotation of *Ab* would have an effect on the denotation of *Flies*.

But what should we allow to vary in the case of the Yale shooting problem; *Holds*, *result*, the individual constants, or some combination of these? In this case, the *Ab* predicate constant is meant to denote all the combinations of properties, situations and actions for which an action causes a property which holds in a specific situation to no longer hold in the resulting situation.

The most obvious choice is to let *Holds* vary. This was the choice which was tried when circumscription was first applied to the Yale shooting problem, but it did not work. The reason for this can be seen in the following two interpretations, K and L, of A:

- $|K| = \{S0,S1\} \cup \{WAIT,SHOOT\} \cup \{LOADED,ALIVE\}$
- $s0^K = S0$

  $wait^K = WAIT$, $shoot^K = SHOOT$

  $loaded^K = LOADED$, $alive^K = ALIVE$
- $Holds^K = \{(LOADED,S0),(ALIVE,S0)\}$

Ab^K = {(LOADED,S0,SHOOT),(ALIVE,S0,SHOOT)}

$result^K$(WAIT,S0) = S0

$result^K$(WAIT,S1) = S1

$result^K$(SHOOT,S0) = S1

$result^K$(SHOOT,S1) = S1,

| L | = {S0,S1} ∪ {WAIT,SHOOT} ∪ {LOADED,ALIVE}

$s0^L$ = S0

$wait^L$ = WAIT, $shoot^L$ = SHOOT

$loaded^L$ = LOADED, $alive^L$ = ALIVE

$Holds^L$ = {(LOADED,S0),(ALIVE,S0)}

Ab^L = {(LOADED,S0,SHOOT),(ALIVE,S0,SHOOT),
(LOADED,S0,WAIT),(ALIVE,S0,WAIT)}

$result^L$(WAIT,S0) = S1

$result^L$(WAIT,S1) = S1

$result^L$(SHOOT,S0) = S1

$result^L$(SHOOT,S1) = S1.

To see that K and L are both models of A, consider only the persistence axiom (it is easy to check that the remaining three axioms are true in both K and L). All the property-situation combinations in the denotation of *Holds* (in K and L) for which an action would change the property are listed in the denotation of *Ab*. This ensures that the persistence axiom is true in both K and L.

To see that K and L are both models of CIRC(A;*Ab*;*Holds*), i.e. that L is an *Ab*;*Holds*-minimal model of A, note that the denotation of *Holds* in both K and L cannot be changed at all without violating some axiom of A. (If any one of the ordered pairs is removed from the denotation of *Holds*, one of the two final axioms of A will be violated. If any ordered pair is added to the denotation of *Holds* the second axiom of A will be violated.) In K, removal of any of the triplets in the denotation of *Ab* will violate the persistence axiom since the necessary ordered pairs (required to satisfy the consequent of the persistence axiom) are not in $Holds^K$. For example, (LOADED,S0,SHOOT) is necessary because (LOADED,S0) is in $Holds^K$ but (LOADED,S1) is not. In L, all triplets of Ab^L are essential for the same reason. For example, (ALIVE,S0,WAIT) is necessary because (ALIVE,S0) is in $Holds^L$ but (ALIVE,S1) is not. Therefore K and L are both *Ab*;*Holds*-minimal models of A.

These two *Ab*;*Holds*-minimal models show that it is not good enough to just allow the denotation of *Holds* to vary in the process of minimisation. Allowing *result* to vary is also necessary.

Consider therefore an *Ab*;*Holds*,*result*-minimal model of A, say M, and assume that the wff *Holds*(*loaded*,*result*(*wait*,*s0*)) is not true in M. Note firstly that the ordered pairs $(loaded^M,s0^M)$ and $(alive^M,s0^M)$ must be in the denotation of *Holds*, in order that M might be a model of A. Consequently, the triplets $(loaded^M,s0^M,shoot^M)$ and $(alive^M,s0^M,shoot^M)$ must be in the denotation of *Ab* in order that the second axiom might be true in M. Secondly, the denotation of *Ab* must contain the triplet $(loaded^M,s0^M,wait^M)$ for the wff *Holds*(*loaded*,*result*(*wait*,*s0*)) not to be true in M.

Now construct an interpretation M' as follows:

- $|M'| = |M|$
- $s0^{M'} = s0^M$

  $wait^{M'} = wait^M$, $shoot^{M'} = shoot^M$

  $loaded^{M'} = loaded^M$, $alive^{M'} = alive^M$
- $Holds^{M'} = \{(loaded^{M'},s0^{M'}),(alive^{M'},s0^{M'})\}$

  $Ab^{M'} = \{(loaded^{M'},s0^{M'},shoot^{M'}),(alive^{M'},s0^{M'},shoot^{M'})\}$
- $result^{M'}(wait^{M'},SX) = SX$

  $result^{M'}(shoot^{M'},SX) = result^{M'}(shoot^{M'},s0^{M'})$

  for every situation object $SX \in |M'|$.

To see that M' is a model of A, we show that each of the axioms of A is true in M'. Consider firstly the first axiom of A (the persistence axiom). The only valuations assigning values to the variables *p*, *s* and *a* that need to be checked are $(loaded^{M'},s0^{M'},shoot^{M'})$, $(alive^{M'},s0^{M'},shoot^{M'})$, $(loaded^{M'},s0^{M'},wait^{M'})$ and $(alive^{M'},s0^{M'},wait^{M'})$ since the only two ordered pairs in $Holds^{M'}$ are $(loaded^{M'},s0^{M'})$ and $(alive^{M'},s0^{M'})$. The first two valuations satisfy the persistence axiom since these two triplets are included in $Ab^{M'}$. The last two valuations also satisfy the persistence axiom since $result^{M'}(wait^{M'},s0^{M'}) = s0^{M'}$ and $(loaded^{M'},s0^{M'})$ and $(alive^{M'},s0^{M'})$ are both in $Holds^{M'}$. To see that the second axiom is true in M', consider the following argument: In M, the value of $result^M(shoot^M,s0^M)$ cannot be $s0^M$, otherwise the second axiom would be violated, preventing M from being a model of A. Say $result^M(shoot^M,s0^M) = SD \neq s0^M$ for some situation object $SD \in |M|$. Consequently, for every situation object $SX \in |M'|$, $result^{M'}(shoot^{M'},SX) = SD$. Since $(loaded^{M'},SD)$ and $(alive^{M'},SD)$ are not in $Holds^{M'}$, the axiom is true in M'. The last two

axioms are patently true in M'.

Since all the axioms of A are true in M', M' is a model of A.

But by construction, M' is an *Ab*;*Holds*,*result*-submodel of M, since the denotation of all constants (besides *Ab*, *Holds* and *result*) are the same. However, M is not an *Ab*;*Holds*,*result*-submodel of M', since $Ab^{M'}$ is a proper subset of Ab^{M}. This contradicts the choice of M as being an *Ab*;*Holds*,*result*-minimal model of A. The wff *Holds*(*loaded*,*result*(*wait*,*s0*)) must therefore be true in every *Ab*;*Holds*,*result*-minimal model of A, i.e. CIRC(A;*Ab*;*Holds*,*result*) $\models$ *Holds*(*loaded*,*result*(*wait*,*s0*)). So, in every *Ab*;*Holds*,*result*-minimal model of A, Fred will be dead in the situation *result*(*shoot*,*result*(*wait*,*s0*)).

Remarks: The Yale shooting problem illustrates the sensitivity of circumscription to the axiomatisation of a system. For example, if the second conjunct of the consequent of the second axiom, namely $\neg$*Holds*(*loaded*,*result*(*shoot*,*x*)), is left out, the interpretation M' (which incidentally is probably the closest we can get to the class of intended interpretations) would no longer be a model of A, destroying the whole argument. (Indeed, our solution to the Yale shooting problem, while based on that in [Baker 1989], is markedly simpler because Baker did not include the relevant conjunct in his axiomatisation). The lesson to be learnt from this is that we can't expect that the use of circumscription will always give the results we are wanting. It depends critically on the choice of axioms to represent our knowledge about the system.

Secondly, the decision of which constants to allow to vary is often difficult. The problem of choosing which constants to allow to vary should be seen as part of the problem of axiomatisation, since the choice depends on the axioms themselves, on what common-sense conjectures we want to be able to make, as well as on the complexity of the problem at hand. There is no known algorithm for deciding which constants to allow to vary.

In the case of the Yale shooting problem, the detailed exploration of various *Ab*-minimal models was necessary to come to a decision of which constants to allow to vary. This brings us to the final comment, namely that a semantic approach is very helpful when working with logic-based formalisms of common-sense reasoning. All the articles on circumscription on which our versions of circumscription have been based ([McCarthy 1980 & 1986] and [Lifschitz 1985]) as well as

most other articles on topics connected to circumscription have had a proof-theoretic bias, i.e. they work from the basis of the deduction of wffs from a set of axioms rather than the semantic entailment of wffs by a set of axioms. It appears to be particularly helpful to always view a set of axioms in terms of its models and hence to determine the wffs which are true in all models of the set of axioms. For example, this emphasis enabled us to get a clearer view of exactly what the problem with naïve circumscription was and also how to rectify the problem. Furthermore, the use of semantical arguments enabled us to simplify some long-winded proofs given in the literature and to replace others which consist of nothing more than a few informal sentences.

CHAPTER THREE
PRIORITIES AND THE SCOPE OF REASONING

In this chapter we consider some common-sense reasoning problems for which the versions of circumscription given in Chapter Two are inadequate. Certain technical adjustments to the definition of circumscription are given to cope with these problems.

Prioritised circumscription

The following example was first given in [Reiter & Criscuolo 1985]:

> Suppose we are given the following information: Normally, Quakers are pacifists and, normally, Republicans are not pacifists. Nixon is a Quaker and a Republican. A set of axioms A describing this system is
>
> $$\forall x(Quaker(x) \wedge \neg Ab_1(x) \rightarrow Pacifist(x))$$
> $$\forall x(Republican(x) \wedge \neg Ab_2(x) \rightarrow \neg Pacifist(x))$$
> $$Quaker(Nixon) \wedge Republican(Nixon)$$
>
> The question is whether Nixon is a pacifist or not. Before deciding what we would like the answer to be, let us examine how standard circumscription operates on this question.
>
> Firstly, consider any model M of A in which $Nixon^M \in Pacifist^M$. For the second axiom of A to be true in M, Ab_2^M must contain $Nixon^M$. By a similar argument, in any model N of A in which $Nixon^N \notin Pacifist^N$, Ab_1^N must contain $Nixon^N$.
>
> As normally done, we circumscribe the two abnormality predicates. We allow the denotation of *Pacifist* to vary in the process of minimisation since, as the denotations of the *Ab* predicate constants change, we expect the number of pacifists to change. We therefore form CIRC(A;Ab_1,Ab_2;*Pacifist*) which is equivalent to restricting the semantics to the Ab_1,Ab_2;*Pacifist*-minimal models of A.
>
> The class of Ab_1,Ab_2;*Pacifist*-minimal models of A are partitioned into two subclasses: those whose denotations of Ab_1 (but not Ab_2) contain the denotation of *Nixon* and those

whose denotations of Ab_2 (but not Ab_1) contain the denotation of *Nixon*. In fact, in one subclass, the minimal models of A have the denotation of *Nixon* as the only member of the denotation of Ab_1 and the denotation of Ab_2 empty, and in the other subclass, the denotation of Ab_1 is empty and the denotation of *Nixon* is the only element of the denotation of Ab_2. These two subclasses of the class of minimal models of A have the denotation of *Nixon* in and not in the denotation of *Pacifist*, respectively. Since circumscription entails what is true in *all* minimal models, neither *Pacifist(Nixon)* nor $\neg$*Pacifist(Nixon)* is entailed. The most that we can hope to entail is *Pacifist(Nixon)* $\vee$ $\neg$*Pacifist(Nixon)*.

This should not surprise us, however, since reading the description given at the beginning of the example does not give us any clue as to what aspect of Nixon's convictions we should consider as being of more importance.

We have seen that when more than one predicate is circumscribed "in parallel" in a set of axioms, we can find ourselves in a situation where the minimality of one of the predicates requires some of the other predicates to contain more in their denotations, and vice versa. This causes the minimal models to be divided into sub-classes in which the denotations of each of the predicates being circumscribed are "truly" as small as possible. Furthermore, this allows no new ground atoms involving these predicates to be entailed by circumscription. What we require is a way to specify which predicates' denotations should be minimised "first", or, in other words, which predicates should take priority in the process of minimisation. This is what prioritised circumscription attempts to do. Prioritised circumscription was first suggested by McCarthy in a 1984 draft of [McCarthy 1986], and applied to circumscription allowing (any) constants to vary by Lifschitz [1985].

The notation $P_1 \succ P_2$ is used to indicate that predicate P_1 takes priority over predicate P_2. The notation $\mathbf{P}^1 \succ \mathbf{P}^2$ is used to indicate that the tuple of predicates $\mathbf{P}^1$ (whose individual predicates all have the same priority) take priority over the tuple of predicates $\mathbf{P}^2$.

Definition Suppose that A is a finite set of axioms in a first-order language, $\boldsymbol{P} = \langle P_1,..,P_n \rangle$ is a tuple of predicate constants and $\boldsymbol{S} = \langle S_1,..,S_m \rangle$ is a tuple of individual, predicate or function constants disjoint from $\boldsymbol{P}$. Then the *circumscription of* $\boldsymbol{P}$ *in* A *with respect to priorities* $\boldsymbol{P}^1 \succ .. \succ \boldsymbol{P}^n$ *allowing* $\boldsymbol{S}$ *to vary*, is denoted by CIRC(A;$\boldsymbol{P}^1 \succ .. \succ \boldsymbol{P}^n$;**S**) and is the second-order formula

$$A \wedge \neg \exists \Phi \boldsymbol{X}(A[\Phi,\boldsymbol{X}] \wedge \Phi \lll \boldsymbol{P})$$

where $\Phi \lll \boldsymbol{P}$ is an abbreviation for

$$(\Phi^1,..,\Phi^n) \lll (\boldsymbol{P}^1,..,\boldsymbol{P}^n)$$

where $(\Phi^1,..,\Phi^n) \lll (\boldsymbol{P}^1,..,\boldsymbol{P}^n)$ is defined as

$$(\Phi^1,..,\Phi^n) \trianglelefteq (\boldsymbol{P}^1,..,\boldsymbol{P}^n) \wedge \neg((\boldsymbol{P}^1,..,\boldsymbol{P}^n) \trianglelefteq (\Phi^1,..,\Phi^n))$$

where $(\Phi^1,..,\Phi^n) \trianglelefteq (\boldsymbol{P}^1,..,\boldsymbol{P}^n)$ is defined as

$$\Phi^1 \leq \boldsymbol{P}^1$$
$$\wedge\ (\Phi^1 = \boldsymbol{P}^1 \rightarrow \Phi^2 \leq \boldsymbol{P}^2)$$
$$\wedge\ (\Phi^1 = \boldsymbol{P}^1 \wedge \Phi^2 = \boldsymbol{P}^2 \rightarrow \Phi^3 \leq \boldsymbol{P}^3)$$
$$\wedge$$
$$:$$
$$\wedge\ (\Phi^1 = \boldsymbol{P}^1 \wedge .. \wedge \Phi^{n-1} = \boldsymbol{P}^{n-1} \rightarrow \Phi^n \leq \boldsymbol{P}^n)$$

where $\Phi^i = \boldsymbol{P}^i$ is defined as

$$\Phi^i \leq \boldsymbol{P}^i \wedge \boldsymbol{P}^i \leq \Phi^i$$

and $\Phi^i \leq \boldsymbol{P}^i$ is defined in the normal way as

$$(\Phi_1^i \leq P_1^i) \wedge .. \wedge (\Phi_m^i \leq P_m^i).$$

■

For example, consider the Nixon-Pacifist set of axioms given above. To resolve the problem, we need to ascribe a priority to the abnormality predicates. This is equivalent to deciding which of Nixon's convictions would be more likely to take precedence. Suppose we decide that Nixon's convictions as a Republican will take precedence over the fact that he is a Quaker. In other words, Ab_1 must be of higher priority than Ab_2. We therefore determine CIRC(A;$Ab_1 \succ Ab_2$;*Pacifist*):

$$A \wedge \forall\Phi_1\Phi_2\Psi(A[\Phi_1,\Phi_2,\Psi] \wedge (\Phi_1 \leq Ab_1 \wedge (\Phi_1 = Ab_1 \rightarrow \Phi_2 \leq Ab_2)) \wedge \neg(Ab_1 \leq \Phi_1 \wedge (Ab_1 = \Phi_1 \rightarrow Ab_2 \leq \Phi_2)))$$

which is equivalent to

$$A \wedge \forall\Phi_1\Phi_2\Psi(A[\Phi_1,\Phi_2,\Psi] \wedge (\Phi_1 \leq Ab_1 \wedge ((\Phi_1 \leq Ab_1 \wedge Ab_1 \leq \Phi_1) \rightarrow \Phi_2 \leq Ab_2)) \wedge \neg(Ab_1 \leq \Phi_1 \wedge ((Ab_1 \leq \Phi_1 \wedge \Phi_1 \leq Ab_1) \rightarrow Ab_2 \leq \Phi_2)))$$

This states that .. (what does this state?)

Fortunately there is an alternative to the above definition of prioritised circumscription. It consists of rewriting the prioritised circumscription as a conjunction of parallel circumscriptions:

Theorem 3.1 [Lifschitz 1985]
CIRC(A;$\boldsymbol{P^1} \succ .. \succ \boldsymbol{P^n}$;**S**) is equivalent to
CIRC(A;$\boldsymbol{P^1}$,..,$\boldsymbol{P^n}$;**S**) $\wedge$ CIRC(A;$\boldsymbol{P^2}$,..,$\boldsymbol{P^n}$;**S**) $\wedge$ .. $\wedge$ CIRC(A;$\boldsymbol{P^n}$;**S**).
■

This theorem considerably simplifies the prioritised circumscription of a set of axioms.

For example, CIRC(A;$Ab_1 \succ Ab_2$;*Pacifist*)

$$= \text{CIRC}(A;Ab_1,Ab_2;Pacifist) \wedge \text{CIRC}(A;Ab_2;Pacifist)$$

$$= A \wedge \neg\exists\Phi_1\Phi_2\Psi(A[\Phi_1,\Phi_2,\Psi] \wedge \Phi_1 < Ab_1 \wedge \Phi_2 < Ab_2) \wedge A \wedge \neg\exists\Phi_2\Psi(A[\Phi_2,\Psi] \wedge \Phi_2 < Ab_2)$$

Expressed intuitively, this states that the denotation of Ab_2 must be as small as possible, i.e. empty in this case (which consequently forces the denotation of *Nixon* to be in the denotation of Ab_1), and then that the denotation of Ab_1 must be as small as possible, i.e. that it must not contain anything besides the denotation of *Nixon*. The denotation of *Nixon* is therefore not in the denotation of *Pacifist* in every model of the prioritised circumscription of the set of axioms, and therefore the wff $\neg$*Pacifist*(*Nixon*) is entailed thereby.

Scoped circumscription

A special kind of problem in the application of circumscription occurs when an axiom asserting the existence of an anonymous counter-example to a default rule is included in a set of axioms [Etherington, Kraus & Perlis 1991]. For example:

> Consider adding a counter-example axiom to our set of axioms for the Tweety example:
>
> $Bird(Tweety)$
> $\forall x(Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x))$
> $\exists x(Bird(x) \wedge \neg Flies(x))$
>
> If we circumscribe Ab in this set of axioms we get no conjectures about Tweety other than those entailed by the set of axioms themselves. More specifically, $Flies(Tweety)$ is not entailed because there are minimal models in which Tweety is the only bird, (i.e. she plays the role of the flightless bird required by the counter-example axiom). Even if we add axioms asserting the existence of birds other than Tweety, circumscription has no way of preferring Tweety's flying to that of any other bird.
>
> Note that not all counter-example axioms give this problem. For example, if we add $\exists x(Bird(x) \wedge \neg Flies(x) \wedge \neg(x = Tweety))$ instead of the counter-example axiom used above, $Flies(Tweety)$ is entailed by the circumscription of Ab. But the second counter-example can replace the first only in a context in which we have the definite knowledge that a non-flying bird other than Tweety exists. This is more than simply affirming the existence of a non-flying bird, since Tweety might just have been the bird on which our knowledge is based.

Here is another example:

> Consider the so-called *lottery-paradox*: It is usually safe to assume that any particular ticket in a lottery will not win, given the overwhelming odds against winning. However, the conjunction of such conclusions for each ticket in the lottery will allow us to conclude that no ticket will win.

More formally, the circumscription of *Ab* in the axiom

$$\forall x(\neg Ab(x) \rightarrow \neg Wins(x))$$

entails $\forall x(\neg Wins(x))$. This conclusion is not what we want, however, since some ticket must win. However, the fact that some ticket must win is a piece of knowledge which is not included in this set of axioms. We therefore add a counter-example axiom asserting the existence of a winning ticket, to form the set of axioms A:

$$\forall x(\neg Ab(x) \rightarrow \neg Wins(x))$$
$$\exists x(Wins(x))$$

However, there are as many minimal models of A as there are tickets (each with a different ticket as the winner). Since circumscription describes what is true in all minimal models, nothing can be assumed about any individual ticket. In particular, if I were considering purchasing ticket number 113, the common-sense conjecture that ticket number 113 will not win is blocked. The most that can be assumed is that if some particular ticket wins, it will be the only one.

In general then, circumscription does not always allow us to make the expected conjectures from default rules when the existence of counter-examples is asserted in the set of axioms. However, human common-sense reasoning can generally cope with the existence of counter-examples. This seems to suggest that "normality" is applied in a more limited way in human common-sense reasoning than is done by circumscription. Generally, the intention of common-sense reasoning is not to determine the properties of every individual in the domain, but rather those of some particular individual(s) of interest. Common-sense reasoning says that we should only take the possibility that something is abnormal into account if forced to. Circumscription (as defined so far) forces us to consider as much as possible to be normal. This is a subtly different matter.

In both the examples discussed above, the problem arises because the existence of something abnormal is entailed, but in some models the individual(s) we want to reason about play the role of the counter-example. Consider the Tweety example augmented by a counter-example axiom: If there were reason to believe that the anonymous counter-example was likely to be among the individuals of interest, one could safely make conjectures about the individuals of interest (i.e.

Tweety) without wrestling with the identity of the counter-example. The same thing happens in the case of the lottery paradox. If we could consider only the small set of tickets that we might consider buying, there should be no problem conjecturing that none of them would win. The underlying intuition is probabilistic: if there are n tickets each with an equal probability of winning, namely $\frac{1}{n}$, then the probability that one of the tickets in a set X is directly proportional to the size of X. The set X can therefore be thought of as representing the scope of our interest.

As stated in [Etherington, Kraus & Perlis 1988], in human common-sense reasoning we assess whether our reasoning has appropriately narrow scope, at least when challenged by evidence of exceptions. When trying to decide about an entire population of individuals, there may well be concern about making default conjectures. In other words, restricting the scope of our reasoning should provide a solution to the existence of counter-examples. However, we shall have to engineer such a restriction of scope without the explicit use of probabilities, because in many applications of common-sense reasoning no probability measure is known - even the precise cardinalities of the population and of the scope of interest may be unknown.

Scope can be accommodated in circumscription by introducing a new predicate constant called *Scope* and then minimising the intersection of *Ab* and *Scope*. To represent this intersection, a new predicate constant D is introduced into the language and the wff $\forall x(D(x) \leftrightarrow Ab(x) \wedge Scope(x))$ is added to the set of axioms. More generally, if we wish to minimise the intersection of the denotations of an arbitrary predicate constant P (i.e. not necessarily *Ab*) and the predicate constant *Scope*, then the wff defining the new predicate constant D will be $\forall x(D(x) \leftrightarrow P(x) \wedge Scope(x))$.

To determine the scoped circumscription of P in a set of axioms A, we firstly form the *scoped set of axioms* A' by adding to A one or more *scope axioms* (consisting of ground atoms involving the predicate constant *Scope* which specify the individuals of interest) and the wff defining the new predicate constant D. The scoped circumscription of P in A' then minimises the new predicate constant D while permitting P and *Scope* (amongst others) to vary.

Definition Let A' be a scoped set of axioms, P a predicate constant and $S = \langle S_1,\ldots,S_m\rangle$ an m-tuple of constants not containing P. Let *Scope* be the scope predicate and D the predicate defining the intersection between P and *Scope*. Then the *scoped circumscription of P in* A' *allowing S to vary*, namely CIRC(A';D;P,*Scope*,S), is the conjunction of the scoped set of axioms A' with a circumscription axiom to form the second-order sentence

$$A' \wedge \neg\exists\Phi\boldsymbol{X}(A'[\Phi,\boldsymbol{X}] \wedge \Phi < D)$$

where $\boldsymbol{X}$ is an (m+2)-tuple of individual, predicate or functions variables corresponding to the individual, predicate and function constants in P, *Scope* and S, and A'[Φ,$\boldsymbol{X}$] is the wff obtained by replacing all occurrences of D, P, *Scope* and S in A' by the variables Φ and $\boldsymbol{X}$, respectively.
■

The concept of minimal models needs to be adjusted for scoped circumscription:

Definition Let A' be a scoped set of axioms, let P be a predicate constant and S a tuple of individual, predicate or function constants not containing P. Let *Scope* be the scope predicate and D the new predicate constant. If M and N are models of A', then M is a *scoped-P;S-submodel* of N, written $M \leq^{D;P,Scope,S} N$, iff

- $|M| = |N|$
- $K^M = K^N$, for every (individual, function or predicate) constant K not in P, S, *Scope* or D
- $D^M \subseteq D^N$.

We write $M <^{D;P,Scope,S} N$ iff $M \leq^{D;P,Scope,S} N$ but not $N \leq^{D;P,Scope,S} M$.

M is a *scoped-P;S-minimal model* of A' iff there is no model N such that $N <^{D;P,Scope,S} M$.
■

Theorem 3.2
Given a scoped set of axioms A', a predicate constant P and a tuple of constants S not containing P, then the models of CIRC(A';D;P,*Scope*,S) are precisely the scoped-P;S-minimal models of A'.

Proof Analogous to Theorem 2.1.
■

To see how scoped circumscription works, we apply it to both of the examples given above:

To be able to use scoped circumscription on the Tweety problem augmented by a counter-example axiom, we must first form the scoped set of axioms A' by adding a scope axiom and an axiom defining D. It is also necessary to add an axiom asserting the existence of an object other than Tweety:

$$Bird(Tweety)$$
$$\forall x(Bird(x) \wedge \neg Ab(x) \rightarrow Flies(x))$$
$$\exists x(Bird(x) \wedge \neg Flies(x))$$
$$Scope(Tweety)$$
$$\forall x(D(x) \leftrightarrow Ab(x) \wedge Scope(x))$$
$$\exists x(\neg(x = Tweety))$$

(The reason why the extra axiom asserting the existence of an object other than Tweety is added is to make the set of axioms rich enough to allow the formation of the models we are interested in. In particular, we need an object other than Tweety that we can at least imagine to be a flightless bird to let Tweety off the hook. Without this axiom, models with domains consisting of a single object would exist. The counter-example axiom would then force Tweety to be abnormal. While we are willing to entertain the possibility that Tweety is indeed abnormal, we should not allow situations in which Tweety's abnormality is forced. Another way to think of it is that the scope of our interest should be narrow in proportion to the size of the total population, and this would not be the case in models whose domains consist only of Tweety.)

We wish to show that CIRC(A';*D*;*Ab*,*Scope*,*Flies*) $\models$ *Flies*(*Tweety*). To do this, we show that for any model of CIRC(A';*D*;*Ab*,*Scope*,*Flies*) (i.e. for any scoped-*Ab*;*Flies*-minimal model of A'), the denotation of *Tweety* must be a member of the denotation of *Flies*.

Assume that there is a scoped-*Ab*;*Flies*-minimal model of A', namely M, for which the denotation of *Tweety* is not a member of the denotation of *Flies*. Then to satisfy the second axiom, $Tweety^M$ must be in Ab^M and to satisfy the axiom defining D, $Tweety^M$ must also be in D^M. The domain of M must contain an element distinct from $Tweety^M$. Suppose s is such an element. Now construct an interpretation N as follows:

Let $|N| = |M|$ and let the denotations of all individual constants be the same in N as in M. Let the denotations of the predicate constants be adapted so that $Ab^N = Ab^M - \{Tweety^M\}$, $Flies^N = Flies^M \cup \{Tweety^M\} - \{s\}$, $D^N = D^M - \{Tweety^M\}$ but let $Bird^N = Bird^M$.

It is easy to check that N is a model of A'.

By construction, D^N is a proper subset of D^M since the only difference between D^M and D^N is that D^M contains the denotation of *Tweety* and D^N does not. In other words, N is a scoped-*Ab*;*Flies*-submodel of M but M is not a scoped-*Ab*;*Flies*-submodel of N. This contradicts the assumption that M is a scoped-*Ab*;*Flies*-minimal model of A'. To sum up then, every scoped-*Ab*;*Flies*-minimal model of A' must have the denotation of *Tweety* in the denotation of *Flies*. In other words, CIRC(A';*D*;*Ab*,*Scope*,*Flies*) $\models$ *Flies*(*Tweety*).

To use scoped circumscription to solve the lottery paradox, we must add a scope axiom specifying the individuals of interest (in this case, the tickets I might consider buying) and an axiom defining *D* to form a scoped set of axioms A', and an axiom ensuring that our scope of interest is not the whole population:

$$\forall x(\neg Ab(x) \rightarrow \neg Wins(x))$$
$$\exists x(Wins(x))$$
$$Scope(Ticket2468) \wedge Scope(Ticket2469) \wedge .. \wedge Scope(Ticket3011)$$
$$\forall x(D(x) \leftrightarrow Ab(x) \wedge Scope(x))$$
$$\exists x(\neg(x = Ticket2468) \wedge \neg(x = Ticket2469) \wedge .. \wedge \neg(x = Ticket3011))$$

We wish to show that for any of the tickets I would consider buying, common-sense reasoning says that I would not win. In other words, we wish to show that CIRC(A';*D*;*Ab*,*Scope*,*Wins*) $\models \neg\exists x(Scope(x) \wedge Wins(x))$. (We allow the denotation of *Wins* to vary in the process of minimisation because abnormality of a ticket has specifically to do with its being a winning ticket). Let M be a scoped-*Ab*;*Wins*-minimal model of A' and assume that there is one ticket among those which I would consider buying which will win in the model M. For argument's sake, say the denotation of *Ticket2999* (namely T2999) is in $Wins^M$ and therefore also in Ab^M. (Obviously T2999 must

be in $Scope^M$ for M to be a model of the axioms.) Furthermore, T2999 must also be in D for the axioms defining D to be true in M. Also, the domain of M must contain an element distinct from T2468,..,T3011, say s, to satisfy the last axiom.

Now construct an interpretation N as follows: Let $|N| = |M|$ and let the denotations of all the individual constants representing the tickets be the same in N as in M. Let $Scope^N = Scope^M$, but let $Ab^N = Wins^N = \{s\}$ and $D^N = \{\}$.

N is a model of A' because the first axiom is true in N (since all tickets not in Ab^N, i.e. $|N| - \{s\}$, are not in $Wins^N$), the second axiom is true in N ($Wins^N$ contains an object), the axiom defining D is true in N (since the intersection of Ab^N and $Scope^N$ is the empty set) and so is the last axiom (since the denotation of *Scope* is the same in N and M, and M is a model of A').

By construction, $D^N = \{\}$ is a proper subset of $D^M = \{T2999\}$. In other words, N is a scoped-*Ab*;*Wins*-submodel of M but M is not a scoped-*Ab*;*Wins*-submodel of N. This contradicts the assumption that M is a scoped-*Ab*;*Wins*-minimal model of A'. Therefore the denotation of *Ticket2999* (and any other ticket in my scope of interest, for that matter) cannot be in the denotation of *Wins*, for any scoped-*Ab*;*Wins*-minimal model of A'. In other words, $CIRC(A';D;Ab,Scope,Wins) \models \neg\exists x(Scope(x) \wedge Wins(x))$.

Previous versions of circumscription have suffered from various weaknesses, one of which is the failure to preserve satisfiability. In the corollary to Theorem 2.3 it was stated that the satisfiability of an (almost-) universal set of axioms is preserved by circumscription. The counter-example problem given above specifically involves the application of the existential quantifier to an atom involving a relevant predicate constant, which makes the resulting set of axioms, by definition, neither universal nor almost-universal. However, as shown in the theorem below, the satisfiability of any set of axioms A is preserved as long as our scope of interest is not forced to be infinite.

Theorem 3.2 (Satisfiability of scoped circumscription)
If the scoped set of axioms A' has a model in which the predicate constant *Scope* has a finite denotation, then $CIRC(A';D;P,Scope,S)$ is satisfiable.

Proof

Assume that A' has a model M in which $Scope^M$ is finite, but that CIRC(A';*D*;*P*,*Scope*,*S*) is not satisfiable. Then every submodel of M has a proper submodel. (Otherwise A' has a scoped minimal model, and then by Theorem 3.2, CIRC(A';*D*;*P*,*Scope*,*S*) has a model, contradicting the assumption that it is unsatisfiable). This means that D^M must be an infinite set, but this requires both P^M and $Scope^M$ to be infinite, since *D* is defined by the wff $\forall x(D(x) \leftrightarrow P(x) \wedge Scope(x))$, which contradicts the original assumption.

Therefore, if there is a model of A' in which *Scope* has a finite denotation, then CIRC(A';*D*;*P*,*Scope*,*S*) is satisfiable.

■

Corollary 3.2

If the denotation of *Scope* is finite in every model of A', then A' is well-founded (in the sense that every model of A' has a scoped-minimal submodel).

■

This result has an interesting implication, namely that a set of axioms need not be well-founded for scoped circumscription to be able to work. All that is required is that the scoped set of axioms must have a model in which the denotation of *Scope* is finite. In the example following Corollary 2.1 in Chapter Two (which showed that circumscription only preserves satisfiability for well-founded sets of axioms), the denotation of *P* was forced to be an infinite set in every model of the circumscription of *P*, preventing any *P*-minimal models. However, scoping *P* so that *Scope* has a finite denotation will force the denotation of *D* to be finite, ensuring that a *D*-minimal model can always be found and thus ensuring the satisfiability of the scoped circumscription of *P* in the set of axioms.

We have now completed our survey of five versions of circumscription. In each case we have illustrated, by means of simple examples, the weakness or inadequacy of the formalism to cope with some specific type of common-sense inference. We then attempted to show how the definition of circumscription can be adapted accordingly. Although there are other problems with circumscription which have not been not addressed in this text, an attempt has been made to give an up-to-date overview of the most important versions of circumscription and the most important results pertaining to them.

In the next chapter, we consider the feasibility of using theorem-proving algorithms to deduce the wffs which are entailed by the circumscription of predicate constants in a set of axioms. In this process, we will present some of the original versions of circumscription, namely predicate and variable circumscription. It seems as if it was partly the intention of these original versions, [McCarthy 1980 & 1986] and [Perlis & Minker 1986], to be able to employ theorem-proving algorithms.

CHAPTER FOUR
THEOREM-PROVING ALGORITHMS AND CIRCUMSCRIPTION

An implementation for knowledge representation based on circumscription would, according to Lifschitz [1985], include a database A, a metamathematical statement describing how circumscription should be performed, and a theorem-prover capable of deriving logical consequences from the result of circumscribing some predicates in A.

The design of such an implementation has to deal with a major difficulty: the definition of circumscription involves quantification over second-order variables. As pointed out by Theorem 1.4, no complete theorem-proving algorithm (with respect to the standard semantics) exists for second-order logic. In other words, no theorem-proving algorithm exists which is able, for all sets of axioms, to deduce all the wffs entailed by the set of axioms.

Strategies to deal with this problem include the following:

- Be satisfied with an incomplete but sound algorithm.
- Use an appropriate alternative to the standard semantics.
- Examine the feasibility of replacing the second-order wff produced by circumscription with a logically equivalent first-order wff.

In the remainder of this chapter we discuss the second and third alternatives in greater detail.

Predicate circumscription

Predicate circumscription is the original form of circumscription introduced in [McCarthy 1980]. This form of circumscription was devised specifically in order to sidestep the limitations on theorem-proving algorithms imposed by the standard semantics of second-order languages. More precisely, predicate circumscription amounts (although McCarthy gives no indication that he is aware of this) to replacing the standard semantics by a restricted form of the Henkin semantics (see [Shapiro 1991]). The result, as we shall see, is a form different from and indeed weaker than all the forms discussed previously.

In the standard semantics of second-order logic, a unary predicate variable Φ ranges over *all* subsets of the domain $|M|$ of an interpretation M. (More generally, an n-ary predicate variable

ranges over all subsets of $|M|^n$.) In other words, by fixing a domain, one fixes the range of both the first-order variables and the second-order variables; the only further things that need to be interpreted are the constants. This is not the case with the Henkin semantics, in which the ranges of the first- and second-order variables are determined separately. In the Henkin semantics, the predicate variables range over a fixed collection of relations on the domain, which need not include all the relations.

The crucial idea behind predicate circumscription is to adopt a Henkin semantics in which the predicate variables range only over the *definable* relations. (The notion of definability will be made precise in the paragraphs which follow, with the help of the notion of a lambda-expression). The effect of restricting the range to definable relations is dramatic: the second-order circumscription axiom may be replaced by a semantically equivalent set of first-order axioms (infinitely many of them), and so it becomes possible, at least in principle, to employ a sound and complete first-order theorem proving-algorithm.

Definition If α is a first-order wff and $x_1,..,x_n$ are individual variables occurring free in α, then the string $\lambda x_1..x_n(\alpha)$ is called an n-ary *lambda-expression.*

■

Note that λ-expressions belong to the metalanguage - they will not usually be wffs of the object language.

Definition Let M be an interpretation of a first-order language and let $\lambda x_1..x_n(\alpha)$ be a λ-expression formed from a wff α of the same language. A relation $R \subseteq |M|^n$ is *defined* by $\lambda x_1..x_n(\alpha)$ with respect to a valuation v in M iff R is the set of all n-tuples $\langle s_1,..,s_n\rangle$ such that, if the valuation v' differs from v at most in assigning s_i to x_i, then v' satisfies α in M, i.e. $s_i = v'(x_i)$.

■

Now instead of working with the second-order circumscription axiom $\forall\Phi((A[\Phi] \wedge \Phi \leq P) \rightarrow P \leq \Phi)$ and restricting the values of Φ to definable relations, the same effect can be achieved by working with first-order wffs obtained by the following process:

- Pick any λ-expression, $\lambda x_1..x_n(\alpha)$, defining a relation of the same arity as Φ.
- Replace each occurrence of the symbol Φ in the unquantified wff $((A[\Phi] \wedge \Phi \leq P) \rightarrow P \leq \Phi)$ by the λ-expression $\lambda x_1..x_n(\alpha)$.

Replace each subexpression of the form $\lambda x_1..x_n(\alpha)(t_1,..,t_n)$ by the *application* of $\lambda x_1..x_n(\alpha)$ to $(t_1,..,t_n)$, i.e. by the wff $\alpha[t_1,..,t_n]$ which is the result of replacing in α the free occurrences of each x_i by the corresponding term t_i (using the usual technical devices such as renaming bound variables to ensure that no variables free in $t_1,..,t_n$ are bound by quantifiers in α).

To illustrate the relevant concepts, let A be the axiom $Red(a) \wedge (Blue(b) \vee Red(b))$. Let M be the interpretation with $|\,M\,| = \{1,2,3\}$, $a^M = 1$, $b^M = 2$, $Red^M = \{1,3\}$, $Blue^M = \{2\}$ and $=^M$ as usual the identity relation on $|\,M\,|$. The relation $\{1\}$ is definable by $\lambda x(x = a)$ because $\{1\}$ is the set of all values $v(x)$ where v is a valuation satisfying the wff $(x = a)$ in M.

Taking *Red* to be the predicate constant P that is to be circumscribed, the relevant unquantified second-order wff is $(\Phi(a) \wedge (Blue(b) \vee \Phi(b)) \wedge \forall x(\Phi(x) \rightarrow Red(x))) \rightarrow \forall x(Red(x) \rightarrow \Phi(x))$. Substituting $\lambda x(x = a)$ for Φ in the manner described above delivers the first-order wff

$$((a = a) \wedge (Blue(b) \vee (b = a)) \wedge \forall x((x = a) \rightarrow Red(x))) \rightarrow \forall x(Red(x) \rightarrow (x = a))$$

which expresses the claim that b is blue and a is the only thing that is red. (This claim is of course false in M, since the denotation of *Red* in M is not minimal - the definable relation $\{1\}$ is a proper subset of Red^M that would function adequately as an alternative denotation of *Red*).

Definition Suppose that A is a finite set of axioms in a first-order language and $\boldsymbol{P}$ is a tuple of predicate constants. The *predicate circumscription* of $\boldsymbol{P}$ in A, denoted by $CIRC_{PR}(A;\boldsymbol{P})$ is the set of first-order wffs $A \cup \underline{CS}(A;\boldsymbol{P})$, where $\underline{CS}(A;\boldsymbol{P})$ is the set of all wffs obtainable from the unquantified second-order wff $((A[\Phi] \wedge \Phi \leq \boldsymbol{P}) \rightarrow \boldsymbol{P} \leq \Phi)$ by substituting λ-expressions for the predicate variables of Φ.

■

Semantics of predicate circumscription

Definition Let A be a finite set of axioms and let $\boldsymbol{P}$ be a tuple of predicate constants. If M and N are models of A, then M is a ***P-submodel*** of N, written $M \leq^{P} N$, iff

- $|M| = |N|$
- $K^M = K^N$, for every function constant or predicate constant not in $\boldsymbol{P}$
- $K^M \subseteq K^N$, for every predicate constant K in $\boldsymbol{P}$.

M is a ***P-minimal model*** of A iff every $\boldsymbol{P}$-submodel of M is identical to M.

■

Theorem 4.1 (Soundness of predicate circumscription)
If $\text{CIRC}_{\text{PR}}(A;\boldsymbol{P}) \models \alpha$, then α is true in every $\boldsymbol{P}$-minimal model of A.

Proof

Without loss of generality, we consider the circumscription of a single predicate constant P in a set of axioms A.

Suppose α is a wff which is entailed by the predicate circumscription of P in A. If $\alpha \in A$, then the theorem is trivially true. Assume, therefore, that $\alpha \in \underline{CS}(A;P)$, i.e. α is an instance of $((A[\Phi] \wedge \Phi \leq P) \rightarrow P \leq \Phi)$. Assume that there is a P-minimal model of A, say M, in which α is not true. Therefore there must be a λ-expression, say U, which can be substituted for Φ in $((A[\Phi] \wedge \Phi \leq P) \rightarrow P \leq \Phi)$ so that the antecedent is satisfied in M, but the consequent is not satisfied in M, i.e. so that $(A[U] \wedge U \leq P)$ is true in M but $P \leq U$ is not true in M.

Now define an interpretation N of A as follows:

- $|N| = |M|$
- $K^N = K^M$ for every (predicate or function) constant K not equal to P
- $P^N = U^M$.

By construction $N \leq^{P} M$. Also, $N \neq M$, because $P^N = U^M$ but $P^M \neq U^M$. This contradicts the assumption that M is a P-minimal model of A. Therefore α must be true in every P-minimal model of A.

■

Corollary 4.1

If M is a *P*-minimal model of A, then M is a model of $\mathrm{CIRC}_{\mathrm{PR}}(\mathrm{A};P)$.

■

The converse of this theorem, namely that the semantics for predicate circumscription is complete, does not hold in general. This can be shown with the following example:

In order to feel comfortable with this example, imagine that someone has presented you with a somewhat inelegant axiomatisation of the set $\mathbb{N}$ of non-negative integers. The set of axioms A employs a language containing the individual constant b (intended to represent zero), the function constant s (intended to represent the injective successor function) and a predicate constant P (intended to represent the predicate 'is a natural number'):

$$P(b)$$
$$\forall xy(P(x) \wedge (s(x) = y \vee s(y) = x) \rightarrow P(y))$$
$$\forall x(\neg(b = s(x)))$$
$$\forall xyz((s(x) = z \wedge s(y) = z) \rightarrow x = y)$$
$$\forall xyz((s(x) = y \wedge s(x) = z) \rightarrow z = y)$$

Consider the wff

$$\alpha \equiv \forall x(P(x) \rightarrow (x = b \vee \exists y(s(y) = x))).$$

Let R be any *P*-minimal model of A. In other words, let R be a model with the extension of P as small as possible. P^R must contain b^R to satisfy the first axiom of A. But then, to satisfy the second axiom of A, P^R must contain $s^R(b^R)$ as well as $s^R(s^R(b^R))$ and so on and so on. The last two axioms force s^R to be a one-one function, in other words, $s^R(b^R)$, $s^R(s^R(b^R))$, ... etc. must all be distinct values of the domain. Any interpretation of P must therefore contain a subset which could be mapped bijectively to the set of natural numbers. Since R is a *P*-minimal model of A, P^R must be isomorphic (precisely) to the set of natural numbers, $\mathbb{N}$. α is a statement which is obviously true in any such interpretation. In other words, α is true in any *P*-minimal model of A.

We now show that α is not entailed by the predicate circumscription of P in A.

Let N = {0,1,2,...} and N' = {0',1',2',...}. Suppose that M is the interpretation with $|M| = N \cup N'$; $b^M = 0$; $P^M = |M|$ and s^M is the successor function in N and N'.

We first prove that M is a model of A: The first axiom of A is true because $b^M \in P^M$. Since the only values that a valuation can assign variable y to are elements of the domain, namely $|M|$, the consequence of the second axiom will be satisfied by any valuation in M. The second axiom is therefore true in M. 0 is not the successor of any element of $|M|$, therefore the third axiom is true in M. The final two axioms of A follow from the notion of a successor function.

Observe, however, that a valuation in M exists which maps x to 0', making α not true in M.

We shall define below the notion of weak P-minimality and show that the wffs entailed by $CIRC_{PR}(A;P)$ are precisely those true in all weakly P-minimal models of A. M is an example of such a weakly P-minimal model. Since α is not true in M, α cannot be one of the wffs entailed by $CIRC_{PR}(A;P)$.

Definition Suppose A is a set of axioms and $\boldsymbol{P} = \langle P_1,..,P_n \rangle$ is a tuple of predicate constants, and M is a model of A. Then M is a *weakly **P**-minimal model* of A iff there is no model N of a such that $N <^{P} M$ and all $P_1^N,..,P_n^N$ are definable in M.

■

Theorem 4.3 (Soundness and completeness of predicate circumscription)
$CIRC_{PR}(A,\boldsymbol{P}) \models \alpha$ iff α is true in every weakly ***P***-minimal model of A.

Proof [Besnard, Moinard & Mercer 1989]
It suffices to show that a model M of A is a model of $\underline{CS}(A;\boldsymbol{P})$ iff M is a weakly ***P***-minimal model of A. For simplicity, we consider the predicate circumscription of a single predicate P in a set of axioms A.

(⇒ part) Let M be a model of A ∪ $\underline{CS}$(A;*P*) but assume that M is not weakly *P*-minimal. Then there is some model N of A such that N $<^P$ M. In other words,

- $|N| = |M|$
- $K^N = K^M$ for all constants *K* other than *P*
- $P^N \subset P^M$

where P^N is definable in M with respect to some valuation v.

Therefore, there is a wff β, with free variables $x_1,..,x_m$, such that the set of all tuples $\langle s_1,..,s_m\rangle$ which satisfy β are precisely P^N.

Now M is a model of $\underline{CS}$(A;*P*) = $((A[\Phi] \wedge \Phi \leq P) \rightarrow P \leq \Phi)$. Let *U* be the λ-expression $\lambda x_1..x_m(\beta)$, and substitute *U* for Φ in $\underline{CS}$(A;*P*). This will give $((A[\beta] \wedge \beta \leq P) \rightarrow P \leq \beta)$, which must be true in M, since Φ can be substituted by any λ-expression. However, while $(A[\beta] \wedge \beta \leq P)$ is true by construction, $P \leq \beta$ is not true in M, since the tuples satisfying β are a proper subset of the denotation of *P* in M (again by construction). Therefore, since M is a model of A ∪ $\underline{CS}$(A;*P*), the construction of N must have been illegitimate, i.e. M must by weakly *P*-minimal.

Therefore, every model of A ∪ $\underline{CS}$(A;*P*) must be a weakly *P*-minimal model of A.

(⇐ part) Similar to the proof of Theorem 4.1.

■

As with naïve circumscription, predicate circumscription can be adapted to allow predicates to vary. This version was called *variable circumscription* by Perlis and Minker [1986]. Variable circumscription suffers from the same weakness as predicate circumscription, namely that the models of the variable circumscription of *P* in A allowing ***Q*** to vary are precisely the weakly ***P;Q***-minimal models of A.

The real problem with the strategy of expressing circumscription as a schema of first-order wffs, as is done in predicate and variable circumscription, is to decide what λ-expression(s) should be used to substitute the predicate variable(s). The choice of the defining wff is crucial for making the required deductions by a theorem-proving algorithm.

Separable formulas

We have seen that predicate circumscription, although succeeding in delivering first-order wffs, is weaker than standard circumscription. Lifschitz [1985] examined the question of whether some instances of standard circumscription could be described in which the second-order formula is semantically equivalent to a set of first-order wffs. (This is no longer a question of replacing the standard semantics for second-order logic by a form of Henkin semantics; it is a matter of finding some syntactical description of those second-order wffs for which the set of models can be given a first-order characterisation. That such syntactical criteria exist is clear - one example is provided by those second-order wffs in which no second-order variables occur. Lifschitz's achievement lies in finding a class of second-order wffs that is broad enough to be useful.)

Lifschitz defines a notion of separability and shows that the circumscription of a separable set of axioms is equivalent to a finite set of first-order wffs. The advantage of this is that once the circumscription of the set of axioms is given as a set of first-order wffs, existing first-order theorem-proving algorithms can be used, for example the *resolution* theorem-proving algorithm.

Definition Given a finite set of axioms A and a tuple of predicate constants $\boldsymbol{P} = \langle P_1,..,P_m\rangle$, A is said to be *solitary with respect to* $\boldsymbol{P}$ if A is a conjunction of

- wffs containing no positive occurrences of $P_1,..,P_m$, and
- wffs of the form $\forall x(U_i(x) \rightarrow P_i(x))$, where each U_i is a λ-expression of the form $\lambda x(\alpha)$ such that the wff α does not contain any of $P_1,..,P_m$.

■

Note that the above definition assumes that the arities of $P_1,..,P_m$ are all 1. This need not be so. The arities of each $P_1,..,P_m$ can differ, with the proviso that the λ-expressions $U_1,..,U_m$ have corresponding arities.

Any solitary formula can be written equivalently in the form

$$N \wedge (\boldsymbol{U} \leq \boldsymbol{P})$$

where N is a wff which contains no positive occurrences of $P_1,..,P_m$ and $\boldsymbol{U}$ is a tuple of λ-expressions not involving $P_1,..,P_m$.

Theorem 4.4 [Lifschitz 1985]
The naïve circumscription of $\boldsymbol{P}$ in a finite set of axioms which is solitary with respect to $\boldsymbol{P}$, namely

$$\mathrm{CIRC}(\mathrm{N} \wedge (\boldsymbol{U} \leq \boldsymbol{P}); \boldsymbol{P})$$

is given by

$$\mathrm{N}[\boldsymbol{U}] \wedge (\boldsymbol{U} = \boldsymbol{P})$$

where N[$\boldsymbol{U}$] is the result of replacing all occurrences of predicate constants of $\boldsymbol{P}$ with the corresponding λ-expressions of $\boldsymbol{U}$ in N.

■

Let A consist of the axiom $Red(a) \wedge On(a,b)$. The reader will recall that the wff $\forall x(Red(x) \rightarrow x = a)$, although not a consequence of A, is entailed by the naïve circumscription of *Red* in A.

A is solitary with respect to *Red* since A is a conjunction of wffs containing no positive occurrences of *Red*, namely $On(a,b)$, and wffs of the form $\forall x(U(x) \rightarrow Red(x))$, namely $\forall x(\lambda x(x = a)(x) \rightarrow Red(x))$.

$$\mathrm{CIRC}(\mathrm{A};Red) = \mathrm{CIRC}(Red(a) \wedge On(a,b);Red)$$

$$= \mathrm{CIRC}(On(a,b) \wedge \forall x(x = a \rightarrow Red(x));Red)$$

which is equivalent to

$$On(a;b) \wedge \forall x(x = a \leftrightarrow Red(x)).$$

From this first-order wff equivalent to CIRC(A;*Red*), a theorem-proving algorithm would be able to deduce $\forall x(Red(x) \rightarrow x = a)$.

Definition Given a finite set of axioms A and a tuple of predicate constants $\boldsymbol{P} = \langle P_1,..,P_m\rangle$, A is said to be *separable with respect to* $\boldsymbol{P}$ if it is constructed by conjunctions and disjunctions of

- wffs containing no positive occurrences of $P_1,..,P_m$, and
- wffs of the form $\forall x(U_i(x) \rightarrow P_i(x))$, where each U_i is a λ-expression of the form $\lambda x(\alpha)$ such that the wff α does not contain any of $P_1,..,P_m$.

■

Any separable formula can be written as a disjunction of solitary formulas:

$$(N_1 \wedge (\boldsymbol{U}^1 \leq \boldsymbol{P})) \vee (N_2 \wedge (\boldsymbol{U}^2 \leq \boldsymbol{P})) \vee .. \vee (N_n \wedge (\boldsymbol{U}^n \leq \boldsymbol{P}))$$

where each N_i contains no positive occurrences of $P_1,..,P_m$ and each $\boldsymbol{U}^i$ is a tuple of λ-expressions not involving $P_1,..,P_m$.

Theorem 4.5 [Lifschitz 1985]
The naïve circumscription of $\boldsymbol{P}$ in a set of axioms A which is separable with respect to $\boldsymbol{P}$ is given by

$$(D_1 \wedge (\boldsymbol{U}^1 = \boldsymbol{P})) \vee (D_2 \wedge (\boldsymbol{U}^2 = \boldsymbol{P})) \vee .. \vee (D_n \wedge (\boldsymbol{U}^n = \boldsymbol{P}))$$

where D_i is

$$N_i[\boldsymbol{U}^i] \wedge \bigwedge_{j \neq i} \neg(N_j[\boldsymbol{U}^j] \wedge (\boldsymbol{U}^j < \boldsymbol{U}^i)).$$

■

Let A be the set of axioms

$$(P(a) \vee Q(a)) \wedge \neg(P(a) \wedge Q(a))$$

Intuitively, A says 'a is in P or in Q but not both'. Suppose we minimise P. We should then be able to infer that a is in Q. In fact, we should also be able to infer that P is empty.

Now A can be rewritten equivalently as follows

$$(Q(a) \wedge \neg P(a)) \vee (P(a) \wedge \neg Q(a)).$$

A can now be written as a disjunction of two solitary formulas, namely

$$Q(a) \wedge (P \leq R) \wedge (S \leq P)$$

and

$$\neg Q(a) \wedge (P \leq S) \wedge (R \leq P),$$

where

$$R \equiv \lambda x(x = a)$$

and

$S \equiv \lambda x$(false), where 'false' denotes your favourite contradictory wff, eg. $(x = x) \wedge \neg(x = x)$.

Then CIRC(A;P) is equivalent to

$$(Q(a) \wedge (S \leq R) \wedge \neg(\neg Q(a) \wedge (R \leq S) \wedge (R < S)) \wedge (S = P)) \vee$$
$$(\neg Q(a) \wedge (R \leq S) \wedge \neg(Q(a) \wedge (S \leq R) \wedge (S < R)) \wedge (R = P))$$

which is equivalent to

$$(Q(a) \wedge (S \leq R) \wedge (S = P)) \vee$$
$$(\neg Q(a) \wedge (R \leq S) \wedge (R = P))$$

which is equivalent to

$$(Q(a) \wedge \forall x(\text{false} \rightarrow x = a) \wedge \forall x(\text{false} \leftrightarrow P(x))) \vee$$
$$(\neg Q(a) \wedge \forall x(x = a \rightarrow \text{false}) \wedge \forall x(x = a \leftrightarrow P(x))).$$

A theorem-prover would deduce $Q(a) \wedge \forall x(\neg P(x))$ from the above.

The above result cannot, however, be applied directly to circumscription if we want to allow a tuple $\boldsymbol{S}$ of (individual, predicate and/or function) constants to vary. However, every circumscription with non-empty $\boldsymbol{S}$ can be reduced to circumscription with empty $\boldsymbol{S}$ as follows:

$$\mathrm{CIRC}(\mathrm{A};\boldsymbol{P};\boldsymbol{S}) \equiv \mathrm{A} \wedge \mathrm{CIRC}(\exists \boldsymbol{X}(\mathrm{A}[\boldsymbol{X}]);\boldsymbol{P})$$

where $\mathrm{A}[\boldsymbol{X}]$ is the result of replacing all occurrences of constants in $\boldsymbol{S}$ with variables $\boldsymbol{X}$ of corresponding type and arity.

The problem with this trick is that the set of axioms being circumscribed now contains second-order quantifiers. Such quantifiers can sometimes be eliminated as follows: if Φ is a tuple of predicate variables and A is separable with respect to Φ, i.e. A can be written in the form

$$(\mathrm{N}_1 \wedge (\boldsymbol{U}^1 \leq \Phi)) \vee (\mathrm{N}_2 \wedge (\boldsymbol{U}^2 \leq \Phi)) \vee .. \vee (\mathrm{N}_m \wedge (\boldsymbol{U}^m \leq \Phi))$$

where the N_i are wffs not containing any unnegated atoms involving Φ, then $\exists\Phi(\mathrm{A}[\Phi])$ is equivalent to

$$\mathrm{N}_1[\boldsymbol{U}^1] \vee \mathrm{N}_2[\boldsymbol{U}^2] \vee .. \vee \mathrm{N}_m[\boldsymbol{U}^m]$$

where each $\mathrm{N}_i[\boldsymbol{U}^i]$ is the result of replacing all occurrences of variables in Φ with the corresponding λ-expressions in $\boldsymbol{U}^i$.

As an example, consider the following set of axioms, A:

$$\forall x(\mathit{Bird}(x) \wedge \neg \mathit{Ab}(x) \rightarrow \mathit{Flies}(x))$$
$$\forall x(\mathit{Ostrich}(x) \rightarrow \mathit{Bird}(x))$$
$$\forall x(\mathit{Ostrich}(x) \rightarrow \neg \mathit{Flies}(x))$$

From A it is easily shown that all ostriches are abnormal. If we minimise *Ab* we would expect to be able to show that only ostriches are abnormal.

Using the trick explained above, $\mathrm{CIRC}(\mathrm{A};\mathit{Ab};\mathit{Flies})$ can be rewritten as $\mathrm{A} \wedge \mathrm{CIRC}(\exists\Phi(\mathrm{A}[\Phi]);\mathit{Ab})$.

Fortunately, A[Φ] is separable with respect to Φ, since it can be rearranged into solitary form:

$$\forall x(Ostrich(x) \rightarrow Bird(x)) \wedge$$
$$\forall x(Ostrich(x) \rightarrow \neg\Phi(x)) \wedge$$
$$\forall x(Bird(x) \wedge \neg Ab(x) \rightarrow \Phi(x))$$

The first two axioms contain no positive occurrences of Φ, so N is

$$\forall x(Ostrich(x) \rightarrow Bird(x)) \wedge$$
$$\forall x(Ostrich(x) \rightarrow \neg\Phi(x))$$

and the last axiom is of the form $U \leq \Phi$, where U is $\lambda x(Bird(x) \wedge \neg Ab(x))$

We can therefore rewrite $\exists\Phi(A[\Phi])$ as $N[U]$, namely

$$\forall x(Ostrich(x) \rightarrow Bird(x)) \wedge$$
$$\forall x(Ostrich(x) \rightarrow \neg(Bird(x) \wedge \neg Ab(x)))$$

which simplifies to

$$\forall x(Ostrich(x) \rightarrow Bird(x)) \wedge$$
$$\forall x(Ostrich(x) \rightarrow Ab(x)).$$

Substituting this in CIRC($\exists\Phi(A[\Phi]);Ab$) and using Theorem 4.4 we get that CIRC(A;*Ab*;*Flies*) is equivalent to

$$A \wedge \forall x(Ostrich(x) \rightarrow Bird(x)) \wedge \forall x(Ostrich(x) \leftrightarrow Ab(x))$$

which is equivalent to

$$A \wedge \forall x(Ostrich(x) \leftrightarrow Ab(x)).$$

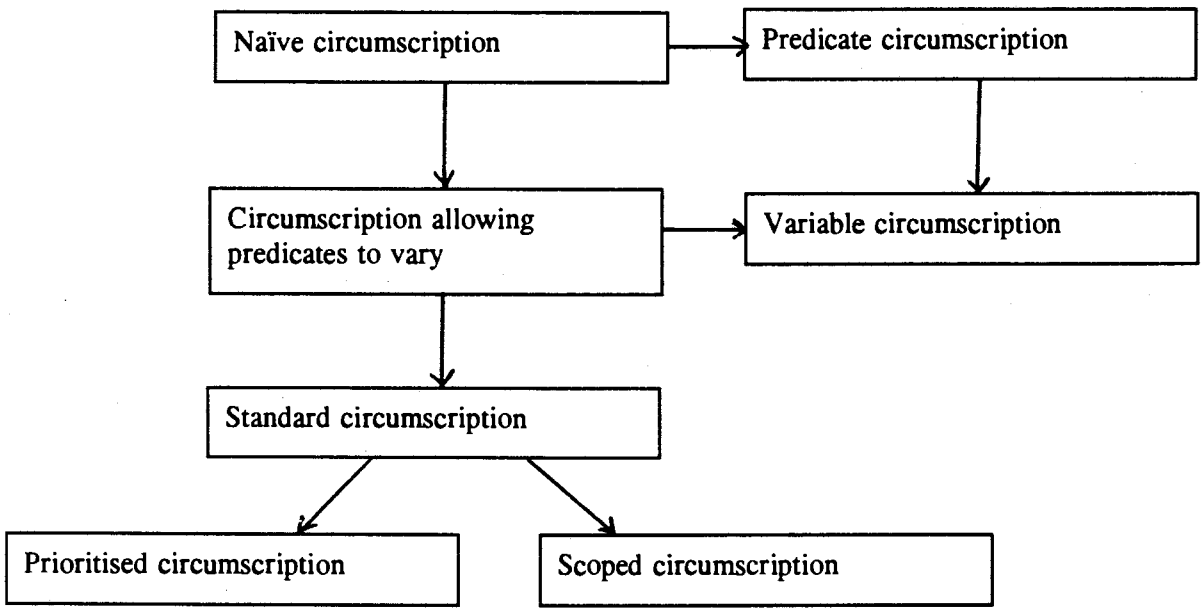
From this first-order wff a theorem prover would be able to deduce that all abnormal things are ostriches.

CHAPTER FIVE
CONCLUSION

We have seen how circumscription may be used to formalise certain types of common-sense argument. The types of argument for which circumscription is particularly suited are those which are based on rules of the form "Normally, such and such is the case". The underlying intuition would be something like the following: since we have no information to lead us to believe that the case in hand is abnormal, we assume that such and such is indeed the case. We have noted that this type of inference is non-monotonic - if information is gained affirming the abnormality of the case in hand, the inference can no longer be made.

Both first- and second-order logic are monotonic, i.e. adding a wff to a set of axioms does not invalidate previous conclusions, or in other words, if $A \models \alpha$ then $A \cup \{\beta\} \models \alpha$. Non-monotonicity is attained by circumscription in the following interesting way: consequences of the circumscription of a set of axioms may not be consequences of the circumscription of an augmented version of the set of axioms. In other words, if $CIRC(A;\boldsymbol{P};\boldsymbol{Q}) \models \alpha$ then it is not necessarily true that $CIRC(A \cup \{\beta\};\boldsymbol{P};\boldsymbol{Q}) \models \alpha$.

We have considered various forms of circumscription. They differ in simplicity and power - in fact there is often a trade-off between simplicity and power. The relationships between the various forms of circumscription discussed in this dissertation can be illustrated as follows:



The central and most generally useful form of circumscription is standard circumscription. Naïve circumscription and circumscription allowing predicates to vary are special cases of this general form. Prioritised circumscription and scoped circumscription are elaborations of standard circumscription, which are intended to deal with certain types of problematic examples. Predicate circumscription and variable circumscription were presented as a way to express circumscription in first-order logic, to be able to use first-order theorem-provers to make inferences from circumscription.

Interestingly enough, the historical order of development of the above forms of circumscription was somewhat different. The order was: predicate circumscription [McCarthy 1980], standard and prioritised circumscription [Lifschitz 1985] (which Lifschitz collectively called second-order circumscription), variable circumscription [Perlis & Minker 1986] and scoped circumscription [Etherington, Kraus & Perlis 1991].

Naïve circumscription and circumscription allowing predicates to vary are in fact our own invention. They are reverse extrapolations of standard circumscription, matching predicate circumscription and variable circumscription, but keeping the definitions in second-order logic.

In fact, another more famous form of circumscription, namely formula circumscription [McCarthy 1986] predates variable circumscription. Formula circumscription is just a slightly different formulation of variable circumscription allowing a wff (i.e. a formula) to be circumscribed rather than a tuple of predicate constants.

Furthermore, scoped circumscription was presented in [Etherington, Kraus & Perlis 1991] as an adaption of formula circumscription, rather than of standard circumscription. In keeping with our pattern of controverting the attempts to express circumscription without second-order logic, we formulated scoped circumscription in second-order logic.

We have been very careful to maintain a model-theoretic view of each presentation of circumscription. In other words, we have gone to pains to consider the models of each formalisation of circumscription. In this way we have been able to consider precisely whether the formalisation of a common-sense argument does in fact entail the inferences we want to make in the argument. In fact, only in the fourth chapter, where we considered how a theorem-proving algorithm could be used to implement circumscription, did we consider what inferences could be

deduced from the circumscription of a (tuple of) predicate constant(s) in a set of axioms. Otherwise, we have concentrated on what wffs are semantically entailed by the circumscription of the predicate constant(s) in the set of axioms. Even in the consideration of the use of theorem-proving algorithms, a model-theoretic view of the substitution of λ-expressions in the first-order wff schema was invaluable in showing the pitfalls and limitations of predicate circumscription.

The big problem with circumscription (as with all formalisations of common-sense reasoning) is to represent our knowledge about the system we want to reason about. The problem isn't so much being able to find *a* representation of our knowledge (this is usually not too difficult), but rather to represent it in a way which is suitable for circumscription to be used to make the inferences we want. If suitable knowledge representation was only difficult for obscure and complicated arguments, we wouldn't mind so much. However, many simple common-sense arguments (like the Tweety examples) require a tremendous amount of time and effort to represent the knowledge about the system in a way which allows circumscription to precisely entail the common-sense inferences we would like to make. Part of the problem of representing our knowledge about a system of interest is deciding what predicates to circumscribe, and what constants to allow to vary. Although some rules of thumb can be given, there is no step by step method or algorithm for making this decision.

BIBLIOGRAPHY

Apt, K.R., Blair, H. and Walker, A. 1988. Towards a Theory of declarative knowledge, in *Foundations of Deductive Databases and Logic Programming* (ed. J. Minker) Morgan Kaufmann, San Mateo CA, p89-148

Baker, A.B. 1989. A Simple Solution to the Yale Shooting Problem, in *Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning*, Toronto, p11-19

Besnard, P., Moinard, Y. and Mercer, R.E. 1989. The Importance of Open and Recursive Circumscription, *Artificial Intelligence* vol. 39, p251-262

Bossu, G. and Siegel, P. 1985. Saturation, Nonmonotonic Reasoning and the Closed-World Assumption, *Artificial Intelligence* vol. 25, p13-63

Enderton, H.B. 1972. *A Mathematical Introduction to Logic*, Academic Press, New York, p??

Enderton, H.B. 1977. *Elements of Set Theory*, Academic Press, New York, p??

Etherington, D.W., Mercer, R.E. and Reiter, R. 1985. On the adequacy of predicate circumscription for closed-world reasoning, *Computing Intelligence* vol. 1, p174-179

Etherington, D.W. 1988. *Reasoning with Incomplete Information*, Pitman, London, ch. 5 and 6

Etherington, D.W., Kraus, S. and Perlis, D. 1991. Nonmonotonicity and the scope of reasoning, *Artificial Intelligence* vol. 52, p221-261

Genesereth, M.R. and Nilsson, N.J. 1988. *Logical Foundations of Artificial Intelligence*, Morgan Kaufmann, Palo Alto CA, ch. 3-5

Hanks, S. and McDermott, D. 1987. Nonmonotonic Logic and Temporal Projection, *Artificial Intelligence* vol. 33, p379-412

Lifschitz, V. 1984. Some Results on Circumscription, in *Non-monotonic Reasoning - Proceedings of the AAAI Workshop,* New Paltz, New York, p151-164

Lifschitz, V. 1985. Computing Circumscription, Reprinted in *Readings in Nonmonotonic Reasoning* (ed. M.L. Ginsberg) 1987, Morgan Kaufmann, San Mateo CA, p167-173

Lifschitz, V. 1986. On the Satisfiability of Circumscription, *Artificial Intelligence* vol. 28, p17-27

Łukaszewicz, W. 1990. *Non-monotonic Reasoning - Formalizations of commonsense reasoning*, Ellis Horwood, New York, ch. 1 and 6

McCarthy, J. 1980. Circumscription - A Form of Non-Monotonic Reasoning, *Artificial Intelligence* vol. 13, p27-39

McCarthy, J. 1986. Applications of Circumscription to Formalizing Common-Sense Knowledge, *Artificial Intelligence* vol. 28, p89-116

Perlis, D. and Minker, J. 1986. Completeness Results for Circumscription, *Artificial Intelligence* vol. 28, p29-42

Reiter, R. 1978. On Closed-World Data Bases, in *Logic and Data Bases* (eds. H. Gallaire and J. Minker), Plenum Press, New York, p55-76

Reiter, R. 1980. A Logic for Default Reasoning, *Artificial Intelligence* vol. 13, p81-132

Reiter, R. and Criscuolo, G. 1983. Some Representational Issues in Default Reasoning , *International Journal of Computers and Mathematics* vol. 9, p1-13

Shapiro, S. 1991. *Foundations without Foundationalism - A Case for Second-order Logic*, Clarendon Press, Oxford, ch. 3