On Theorem Proving in Annotated Logics

Mi Lu & Jinzhao Wu

To cite this article: Mi Lu & Jinzhao Wu (2000) On Theorem Proving in Annotated Logics, Journal of Applied Non-Classical Logics, 10:2, 121-143, DOI: 10.1080/11663081.2000.10510993

To link to this article: https://doi.org/10.1080/11663081.2000.10510993

Published online: 30 May 2012.

Submit your article to this journal

Article views: 15

View related articles
On Theorem Proving in Annotated Logics

Mi Lu
Department of Electrical Engineering
Texas A&M University, College Station, TX 77843, USA
E-mail: mlu@ee.tamu.edu

Jinzhao Wu*
Max-Planck-Institut für Informatik
Im Stadtwald, D-66123 Saarbrücken, Germany
E-mail: wu@mpi-sb.mpg.de

ABSTRACT. We are concerned with the theorem proving in annotated logics. By using annotated polynomials to express knowledge, we develop an inference rule superposition. A proof procedure is thus presented, and an improvement named M-strategy is mainly described. This proof procedure uses single overlaps instead of multiple overlaps, and above all, both the proof procedure and M-strategy are refutationally complete.

KEYWORDS: Theorem proving, Annotated logics, Annotated polynomials, Superposition.

1 Introduction

Annotated logics are a class of paraconsistent logics[Co 74] that have been used to provide the semantical foundations of various reasoning systems which may contain useful inconsistent or imprecise information. They were initially proposed by Subrahmanian based on a specific set, viz. \([0, 1] \cup \{T\}\), of truth values[Su 87], and subsequently studied from the view points of foundation and logic programming[BS 89][CSV 91][KL 89]. Moreover, Kifer and Subrahmanian developed further improvements on the syntactic nature of annotation based program, and showed how annotated programs subsumed quantitative reasoning methods in logic programming, as well as various existing formalisms for temporal logic programming [KS 89][KS 92]. Ng and Subrahmanian investigated the theory of probabilistic reasoning based on another generalization of annotated logics[NS 93]. In addition, annotated logics have also been applied to the development of a declarative semantics for inheritance networks[KK 89].
and object-oriented databases [KW 89]. In view of the applicability of annotated logics to these differing formalisms, it's necessary to study these logics carefully.

As is known, it is necessary and important to establish the resolution theory in various non-classical logics. Many works have been done in this field, and there is no exception for annotated logics. In 1990, da Costa, Henschen, Lu and Subrahmanian developed a whole theory of resolution in annotated logics, which is not only sound, but also complete [CLSH 90] [LSC 91] [Lu 92].

On the other hand, just like classical logic, more general theorem proving methods are also of importance. For instance, In 1982, Murray presented a non-clausal proof procedure which is referred as NC-resolution in classical logic [Mu 82]. In 1982, Hsiang extended the idea of using Boolean polynomials to represent propositional formulae to the case of first-order predicate calculus. Based on Knuth-Bendix completion procedure, he proposed term-rewriting method [Hs 85]. Later, in 1985, Kapur and Narendran developed another approach along this line [KN 85]. One obvious advantage to do so is that every propositional formula has a unique representation. Stimulated by them, some approaches have been reported. Most of them use multiple overlaps [Zh 94].

We investigate the theorem proving in annotated logics in this paper. We choose annotated logics not only because they are widely used as we have stated and much richer than classical logic, but also because the model theory is essentially two-valued, and the truth values are used only for evaluation of atoms [CSV 91]. Actually, some suggestions have been provided by Hähnle. For the clue to justify the method used in this paper rather than a tableaux-based one, the reader is referred to Sections 5.5 and 8.1.5 of [Ha 93].

By employing the annotated polynomials defined in this paper to express knowledge, we mainly present a proof procedure for annotated logics which is based on Hsiang's idea [Hs 85], and can be seen as an extension of the system given by Zhang [Zh 94]. We utilize single overlaps instead of multiple overlaps, because multiple overlaps generate more inferences that are not possible for single overlaps, thus the searching space is increased, and multiple overlaps are much more expensive to compute than single overlaps since the operation mentioned is associative and commutative. For other information about single overlaps and multiple overlaps, the reader consults [WL 96] [Zh 94].

Although resolution based approach already exists, there is still a need to develop such a method. An obvious advantage, for instance, is that each propositional formula has a unique representation. A reduction procedure thus goes on naturally, in which some redundant inferences are avoided. Above all, in this way various algebraic and rewriting techniques may be conveniently applied, and we can also consider defining more general notions of redundancy and introducing more powerful simplification rules to speed up the proof. This outline has been shown to be successful in classical first-order theorem proving [BG 97] [CLO 97] [Hs 85] [KN 85]. We believe that it is true for some non-classical cases (e.g. annotated logics), too. This paper is an initial try on this. We haven't done the experimental work so far, which should be done
after a further investigation, to indicate at least in which sense the method is well-suited for mechanizing annotated logics. In this paper, we only show that in theoretical viewpoint it is basically available along the above line.

We do not deal with all the logics considered in da Costa et al's paper [CLSH 90][CSV 91]. In this paper, we discuss the case that the annotation set is a total ordering set. This is not meaningless. It is very useful in dealing with imprecise information or uncertainty. If the annotation set, for example, is the unit interval $[0, 1]$ with $\neg x = 1 - x$, this logic facilitates quantitative or fuzzy reasoning. For more information, the reader is referred to [CSV 91][KS 89]. Additionally, the truth values in two-valued classical logic can be viewed as being ordered by $false < true$ in one sense although the semantics do not coincide with each other[Su 92]. Also, gp-resolution and p-resolution are the same for this case, we cannot embed parallel searching within the inference[LSC 91][Lu 92].

This paper is arranged as follows: In Section 1, we present some necessary concepts and results in annotated logics; We define annotated polynomials in Section 2, and develop the proof procedure and show that it is complete in Section 3; In Section 4, we describe several strategies, and mainly prove that M-strategy, one of these strategies, is also complete. We have a brief conclusion in the last section.

2 A brief review of annotated logics

In this section, we give some basic concepts and results in annotated logics, which are necessary for the following discussions.

We first consider the formal syntax of annotated logics. In this paper, $T = (T, \leq)$ denotes a total ordering set, we call it annotation set. The language $L$ consists of finite sets of constant symbols, function symbols(with associated arities), and predicate symbols(with associated arities). In addition, $L$ contains infinitely many variable symbols, binary connectives $\land, \lor, \rightarrow$, unary connectives $\lnot$, and $\exists$, as well as two quantifiers $\forall$ and $\exists$. Terms and atoms are defined in the same way as classical logic.

If $p$ is an atom, and $\mu \in T$, then $p : \mu, \sim^k (P : \mu)$ and $\neg^k (P : \mu)(k \geq 1)$ are called annotated atom, mega-literal of order $k$ and hyper-literal of order $k$, respectively.

An annotated atom or a hyper-literal is a formula; If $\varphi, \varphi_i(i = 1, 2)$ are formulae and $x$ is a variable symbol, then $\sim (\varphi), \varphi_1 \land \varphi_2, \varphi_1 \lor \varphi_2, \varphi_1 \rightarrow \varphi_2, \varphi_1 \leftrightarrow \varphi_2, (\forall x)\varphi$ and $(\exists x)\varphi$ are formulae; And all formulae are formed through this way in finitely many steps.

To be convenient, we often omit parentheses if possible. Now, we give the semantics of annotated logics by defining interpretations.

An interpretation $I$ (for $L$) consists of a non-empty domain $D(I)$, and (1) a function $C_I$ that maps constants in $L$ to $D(I)$; (2) a function $F_I$ that maps each n-ary function symbol in $L$ to a function from $D(I)^n$ to $D(I)$; (3) a function $P_I$ that maps each n-ary predicate symbol in $L$ to a function from $D(I)^n$ to $T$. 

Suppose now \( I \) is an interpretation. A variable assignment \( \nu_I \) w.r.t. \( I \) is a map from the variable symbol set of \( L \) to \( D(I) \). It can be extended to be \( \nu_I \), a map from the term set of \( L \) to \( D(I) \): (1). If \( t \) is a constant symbol, then \( \nu_I(t) = c_I(t) \); (2). If \( t \) is a variable symbol, then \( \nu_I(t) = \nu_I(t) \); (3). If \( t = f(t_1, \ldots, t_n) \), then \( \nu_I(t) = \mathcal{F}(f)(\nu_I(t_1), \ldots, \nu_I(t_n)) \), where \( f \) is a \( n \)-ary function symbol in \( L \).

Let \( I \) be an interpretation, \( \nu_I \) a variable assignment w.r.t. \( I \). In the sequel, \( p(t_1, \ldots, t_n) \) is an atom, \( \mu \in T \), and \( \varphi \), \( \varphi_i \) \( (i = 1, 2) \) are formulae. When it is written in front of an element of \( T \), \( \neg \) represents some arbitrary, but fixed, unary function from \( T \) to \( T \).

1. \((I, \nu_I) \) satisfies \( p(t_1, \ldots, t_n) \) : \( \mu \) iff \( P_I(p)(\nu_I(t_1), \ldots, \nu_I(t_n)) \geq \mu \);
2. \((I, \nu_I) \) satisfies \( \neg^k (p(t_1, \ldots, t_n) : \mu) \) iff \( (I, \nu_I) \) satisfies \( \neg^{k-1} (p(t_1, \ldots, t_n) : \mu) \) \( (k \geq 1) \);
3. \((I, \nu_I) \) satisfies \( \varphi_1 \land \varphi_2 \) iff \( (I, \nu_I) \) satisfies \( \varphi_1 \) and \( (I, \nu_I) \) satisfies \( \varphi_2 \);
4. \((I, \nu_I) \) satisfies \( \varphi_1 \lor \varphi_2 \) iff \( (I, \nu_I) \) satisfies \( \varphi_1 \) or \( (I, \nu_I) \) satisfies \( \varphi_2 \);
5. \((I, \nu_I) \) satisfies \( \varphi_1 \rightarrow \varphi_2 \) iff \( (I, \nu_I) \) satisfies \( \varphi_2 \) or \( (I, \nu_I) \) does not satisfy \( \varphi_1 \);
6. \((I, \nu_I) \) satisfies \( \varphi_1 \leftrightarrow \varphi_2 \) iff \( (I, \nu_I) \) satisfies both \( \varphi_1 \) and \( \varphi_2 \), or \( (I, \nu_I) \) satisfies neither \( \varphi_1 \) nor \( \varphi_2 \);
7. \((I, \nu_I) \) satisfies \( \neg \varphi \) iff \( (I, \nu_I) \) does not satisfy \( \varphi \);
8. \((I, \nu_I) \) satisfies \( (\forall x) \varphi \) iff for all variable assignment \( \nu_I' \) w.r.t. \( I \) such that for all variable symbols \( y \) different from \( x \), \( \nu_I'(y) = \nu_I'(y) \), it is the case that \( (I, \nu_I') \) satisfies \( \varphi \);
9. \((I, \nu_I) \) satisfies \( (\exists x) \varphi \) iff for some variable assignment \( \nu_I' \) w.r.t. \( I \) such that for all variable symbols \( y \) different from \( x \), \( \nu_I'(y) = \nu_I'(y) \), it is the case that \( (I, \nu_I') \) satisfies \( \varphi \).

We say that \( I \) satisfies \( \varphi \) if for any variable assignment \( \nu_I \) w.r.t. \( I \), \( (I, \nu_I) \) satisfies \( \varphi \). A formula set \( FS \) is called unsatisfiable if for any interpretation \( I \), \( I \) does not satisfy an element of \( FS \).

We remark that the symbol \( \neg \) is important in the view point of the expressive power of annotated logics, the philosophical motivation behind it can be found in [CSV 91][KL 89]. However, just like the resolution theory in annotated logics[CLSH 90][LSC 91][Lu 92], \( \neg \) is unnecessary for our study because (2) holds. Obviously, all the interpretations satisfy the annotated atom \( p : \mu \) if \( \mu \) is the smallest element in \( T \).

Of course, we only investigate closed formula set, i.e., all the variable symbols occurring in the formulae are quantified. Clearly, refutational proof procedure is applicable, we can define and compute the skolem standard form \( s\varphi \) of a formula \( \varphi \) just like what we did in classical logic case, and we have

**Theorem 1** A formula set \( FS \) is unsatisfiable iff \( \{ s\varphi | \varphi \in FS \} \) is unsatisfiable.
3 Annotated polynomials

For the concepts we use directly without giving the definitions in the following, the reader consults [CL 73][CLO 97][LSC 91].

3.1 Basic concepts and results

Let $Z_2$ be the residue class ring modulo 2. An annotated polynomial is a polynomial over $Z_2$ with annotated atoms as indeterminates. For the definition of polynomials over $Z_2$, please see [CLO 97]. As a matter of fact, the reason why we employ such polynomials is that the annotated atoms always have value 1 or 0 though the annotation sets may be different. An annotated polynomial is ground if the atoms in it are ground. A product of some annotated atoms is called a monomial. We still call the variable symbols in atoms variable symbols.

For a given annotated polynomial $c$, suppose $\mu_1, \cdots, \mu_n$ ($\mu_1 > \cdots > \mu_n$) are all the annotations which appear in $c$, and $p_1, \cdots, p_k$ are all the atoms occurring in $c$. Let $R$ be the polynomial ring over $Z_2$ with $p_1 : \mu_1, \cdots, p_k$ as indeterminates,

$$G = \{(p_s : \mu_i)(p_s : \mu_j) + p_s : \mu_i | 1 \leq s \leq k, 1 \leq i \leq j \leq n\} \subseteq R,$$

and $(G)$ the ideal generated by $G$ in $R$. Also, assume "<" is the lex order determined by the following variable ordering:

$$p_s : \mu_i > p_t : \mu_j \text{ if } s < t, \text{ or } s = t \text{ and } i < j.$$

It is not hard to see that $G$ is indeed the Groebner basis of $(G)$ under the lex order "<". Hereafter, we use $\Phi(c)$ to denote the unique normal form of $c$ w.r.t. $G$.

In fact, $\Phi(c)$ may be defined without making reference to such a $G$. It is just used to prove that $\Phi(c)$ is unique and follows from $c$. In practical we can obtain $\Phi(c)$ from $c$ very easily. If an atom $p$ appears more than once in a monomial in $c$, then for the annotated atoms in which $p$ appears, we only keep the one with the largest annotation and discard the remaining ones in this monomial, the annotated polynomial we get finally is just $\Phi(c)$. Obviously, for annotated polynomials $c, c_1, c_2$ and substitution $\theta$, we have $\Phi(c^2) = \Phi(c)$, $\Phi(c_1 + c_2) = \Phi(c_1) + \Phi(c_2)$, $\Phi(c_1c_2) = \Phi(\Phi(c_1)\Phi(c_2))$, and $\Phi(c\theta) = \Phi(\Phi(c)\theta)$.

Throughout this paper, we always use $a, b, c, d, e, r$ to denote annotated polynomials, $PS$ represents a finite non-zero annotated polynomial set, An atom or annotated atom occurs(appears) in $c$ iff it occurs(appears) in $\Phi(c)$, and we say that $PS$ is ground if every element of $PS$ is ground. Also, $\mu \in T$, and $ap : \mu + b$ represents an annotated polynomial, in which $\Phi(a) = a \neq 0$, $\Phi(b) = b$, atom $p$ doesn’t appear in $a$, and annotated atom $p : \mu$ doesn’t appear in $b$. By $c(p \leftarrow \mu)$ we denote the annotated polynomial obtained from $c$ by substituting atom $p$ in $c$ by $\mu$, in which $\mu_i : \mu_j$ is replaced by 1 if $\mu_i \geq \mu_j$, otherwise by 0 for $\mu_i, \mu_j \in T$.

For example, let $T = \{0.1, 0.2\}(0.1 < 0.2)$, and $c = (p : 0.1)(q : 0.2) + r : 0.1$. Then $c(p \leftarrow 0.2) = (0.2 : 0.1)(q : 0.2) + r : 0.1 = q : 0.2 + r : 0.1$ due to...
0.2 : 0.1 = 1.

From the definition of $\Phi(c)$, we know $\Phi(c(p \leftarrow \mu)) = \Phi(c)(p \leftarrow \mu)$. In addition, $\Phi(c\theta)$ is called an instance of $c$, where $\theta$ is a substitution.

For an arbitrary formula $\varphi$, we call the annotated polynomial $c\varphi$ defined below the one associated with $\varphi$:

\[
c\varphi = \begin{cases} 
\varphi, & \text{if } \varphi \text{ is an annotated atom; } \\
\Phi(c\varphi) + 1, & \text{if } \varphi = \neg \phi; \\
\Phi((c\phi_1)(c\phi_2)), & \text{if } \varphi = \phi_1 \land \phi_2; \\
\Phi(c\phi_1) + \Phi(c\phi_2) + \Phi((c\phi_1)(c\phi_2)), & \text{if } \varphi = \phi_1 \lor \phi_2; \\
\Phi(c\phi_1) + \Phi((c\phi_1)(c\phi_2)) + 1, & \text{if } \varphi = \phi_1 \rightarrow \phi_2; \\
\Phi(c\phi_1) + \Phi(c\phi_2) + 1, & \text{if } \varphi = \phi_1 \leftrightarrow \phi_2,
\end{cases}
\]

where $\phi, \phi_1, \phi_2$ are formulae.

We remark that we define $c\varphi = 1$ for the formula $\varphi = p : \mu$, if $p$ is an atom, $\mu$ is the smallest element in $T$, and "+" and "*" (we omit it as usual) here in fact stand for exclusive or "V" and conjunction "\land" respectively. Also, we see that $\Phi(c(\varphi\theta)) = \Phi((c\varphi)\theta)$ holds for any substitution $\theta$.

Considered as a propositional formula, formula $\varphi$ is called a tautology if all the interpretations satisfy it. For example, if $\mu_1 \geq \mu_2$, then $\neg (p(x) : \mu_1) \lor p(x) : \mu_2$ is a tautology. We say that two formulae $\varphi_1$ and $\varphi_2$ are equivalent if formula $\varphi_1 \leftrightarrow \varphi_2$ is a tautology. Thus, it is not hard to prove that the annotated polynomials associated with tautologies are 1, and the annotated polynomials associated with two equivalent formulae are the same. This is really one of the reasons why we define and use annotated polynomials. Especially, similar to the analogous result in [Hs 85], we have

**Theorem 2** The annotated polynomials associated with a formula are unique.

Let $I$ be an interpretation (for $L$), $v_I$ be a variable assignment w.r.t. $I$. Suppose

\[c = \sum_i \prod_j p_{ij}(t_{1ij}, \ldots, t_{kij}) : \mu_{ij} + r\]

is an annotated polynomial, where $p_{ij}(t_{1ij}, \ldots, t_{kij})$ are atoms, $r \in \{0, 1\}$.

We say that $(I, v_I)$ satisfies $c$ if $\sum_i \prod_j P(p_{ij})(\overline{v_I}(t_{1ij}), \ldots, \overline{v_I}(t_{kij})): \mu_{ij} + r = 0$. Similarly, $I$ satisfies $c$ iff for an arbitrary variable assignment $v_I$ w.r.t. $I$, $(I, v_I)$ satisfies $c$. An annotated polynomial set $PS$ is called unsatisfiable if for any interpretation $I$, $I$ does not satisfy an element of $PS$.

By this definition, we can easily prove

**Theorem 3** $FS$ is unsatisfiable iff $PS$ is unsatisfiable, where $FS$ is a formula set, $PS = \{c\varphi + 1 | \varphi \in FS\}$.

This theorem is similar to that of [KN 85], by which we know that our task from now on is to decide whether a given annotated polynomial set is unsatisfiable.
3.2 Herbrand theorem

An appealing feature of annotated polynomials is that Herbrand Theorem is still applicable. Herbrand interpretation is defined in the usual way: the $i$-level constants $H_i$, Herbrand universe $H$, the assignment to constant and function symbols are the same as the classical logic case. However, $n$-ary predicate symbols are mapped to functions from $H^n$ to $T$. Similar to the proof in classical logic case [CL 73], we have

**Theorem 4** $PS$ is unsatisfiable iff all the Herbrand interpretations do not satisfy $PS$.

Suppose that $PS|H_i$ is the set which is constructed from $PS$ by replacing each variable symbol in the atoms in each annotated polynomial of $PS$ by all the elements of $H_i$. Then we have the following Herbrand theorem.

**Theorem 5** $PS$ is unsatisfiable iff there exists an $i \in \{0, 1, \ldots\}$, such that $PS|H_i$ is unsatisfiable.

By using Theorem 4 and the same technique to prove Herbrand Theorem in [CSV 91][KL 89], we can prove Theorem 5.

Note that we may define annotated polynomials to be polynomials over the rational number field $Q$ instead of $Z_2$, and define an interpretation $I$ to satisfy such an annotated polynomial in a similar way. Further, we can derive the annotated polynomial $c\varphi$ associated with formula $\varphi$ by employing Lagrange interpolation formula. Now, if we let $PS = \{1 - c\varphi|\varphi \in FS\}$, then Theorem 3 and 5 still hold. we can therefore reduce the theorem proving in annotated logics to an algebraic problem:

Let $T'$ be the set of all the annotations occurring in $PS$. We might as well suppose $T' \subset Q$. Let $p_j (j = 1, \ldots, n)$ be all the atoms which appear in $PS|H_i$. For any annotated polynomial $c$ in $PS|H_i$, we use the polynomial

$$\sum_{s \in T', s \geq \mu, t \in T', t \neq s} \prod (p_j - t)/(s - t)$$

in the polynomial ring $Q[p_1, \ldots, p_n]$ to substitute the annotated atom $p_j : \mu_j$ if $p_j : \mu_j$ appears in $c$. We thus obtain a polynomial, say $P(c)$, in $Q[p_1, \ldots, p_n]$ from $c$. Let $QPS_i = \{P(c)|c \in PS|H_i\}$, $V(QPS_i)$ denote the zero set of $QPS_i$ (over the complex number field). Then the elements of $V(QPS_i) \cap (T' \cup \{\mu\})^n$ can represent all the Herbrand interpretations satisfying $PS|H_i$, where $\mu$ is an element of $T$ and $\mu < \mu_i$ holds for any $\mu_i \in T'$. However, $(T' \cup \{\mu\})^n$ is the zero set of $A_i = \prod_{t \in T' \cup \{\mu\}} (p_j - t)|j = 1, \ldots, n \subset Q[p_1, \ldots, p_n]$. Hence, $PS|H_i$ is unsatisfiable iff the zero set of $QPS_i \cup A_i$ is empty. Therefore, we can use Groebner basis method or Wu-Ritt’s method [CLO 97] to decide whether $PS|H_i$ is unsatisfiable.

This method is indeed a little similar to that of [WT 94]. For more details, the reader is referred to this paper. In the next section, we develop a proof procedure by using substitution and unification.
4 The proof procedure

As usual we first present the inference rules, and then consider the completeness.

4.1 The inference rules

The inference rules of the proof procedure consist of the following two rules:

Superposition:

\[ a_1 p_1 : \mu_1 + b_1, \]
\[ a_2 p_2 : \mu_2 + b_2, \mu_1 \geq \mu_2, \theta \text{ is an mgu of atoms } p_1 \text{ and } p_2 \]
\[ \Phi(b_1 \theta(a_2 + b_2)\theta) \]

We call \( \Phi(b_1 \theta(a_2 + b_2)\theta) \) a superposant of \( c_1 \) and \( c_2 \) (upon \( p_1 : \mu_1 \) and \( p_2 : \mu_2 \)), where \( \Phi(c_i) = a_i p_i : \mu_i + b_i (i = 1, 2) \).

Just as mega-resolution is not a simple copy of the resolution in classical logic case, we remark that the superposition defined above is not a simple copy of that of [Zh 94], either. That is impossible since first of all some rules in the rewriting system of the Boolean ring should be changed completely. For instance, we must find out a suitable substitution for the idempotency rule \( B_2 \). It is clearly not sufficient to keep it in the original form, to guarantee not only the soundness but also the completeness. In classical logic case we know that \( \mu_1 = \mu_2 \). So roughly speaking, the two involved polynomials have the same status, the inference rule is symmetric. However, for general annotated logics, all these facts making things simple do not hold any more, the situation becomes more complicated. Up till now, we still don't know whether the semantic strategy given in [Zh 94] is applicable to annotated logics. The technical difficulty, we will see later, generally lies on two aspects. Firstly, it's impossible for us to utilize the techniques of shrinking semantic trees as most other methods (including Zhang's odd-strategy [Zh 94]) did. We have to look for new ways to do that. Secondly, when an atom is assigned to a special value, some other atoms may disappear at the same time. We have to know how to track them especially in the case that a partial ordering between atoms is introduced.

Like most of theorem proving systems [BG 97] [CL 73] [Hs 85] [KN 85] [Zh 94], the following factor computation is necessary.

Factoring:

\[ c, \theta \text{ is an mgu of atoms } p_i(i \geq 1) \text{ occurring in } c, \]
\[ \Phi(c \theta) \]

We call \( \Phi(c \theta) \) a factor of \( c \).

We see that single overlaps are employed in superposition instead of multiple overlaps. Furthermore, we have

Theorem 6 Superposition is sound, i.e., any interpretations satisfying \( c_1 \) and \( c_2 \) satisfy the superposants of \( c_1 \) and \( c_2 \).

Proof: Certainly this theorem can be treated as a consequence of the completeness of mega-resolution [CLSH 90]. However, we can prove it directly.
Since an interpretation satisfies \( c \) iff it satisfies all the ground instances of \( c \). So, without loss of generality, we assume that \( \Phi(c_1) = a_1 p : \mu_1 + b_1 \) and \( \Phi(c_2) = a_2 p : \mu_2 + b_2 (\mu_1 \geq \mu_2) \) are ground. Let \( I \) be an interpretation satisfying \( c_1 \) and \( c_2 \), and \( I \) maps atom \( p \) to \( \mu \). If \( \mu \geq \mu_1 \), then \( \mu \geq \mu_2 \), \( I \) satisfies \( a_2 + b_2 \); If \( \mu < \mu_1 \), then \( I \) satisfies \( b_1 \). Thus, no matter which case holds, \( I \) always satisfies \( \Phi(b_1 (a_2 + b_2)) \), the superposant of \( c_1 \) and \( c_2 \) upon \( p : \mu_1 \) and \( p : \mu_2 \).

Now, we consider the relationship between resolution [CLSH 90] [LSC 91] [Lu 92] and superposition. Before doing so, we first recall two related concepts.

A mega-clause is of the form \( L_1 \lor \cdots \lor L_n \), where each \( L_i \) is a mega-literal of order 0 or 1.

Suppose that \( \varphi_1 \) and \( \varphi_2 \) are two mega-clauses,
\[
\varphi_1 = L_1 \lor \cdots \lor L_i \lor \cdots \lor L_n,
\]
\[
\varphi_2 = L'_1 \lor \cdots \lor L'_j \lor \cdots \lor L'_m,
\]
\[
L_i = p_1 : \mu_1, \quad L'_j = \sim p_2 : \mu_2, \quad \mu_1 \geq \mu_2, \quad \text{and } \theta \text{ is an mgu of atoms } p_1 \text{ and } p_2.
\]

Then
\[
(L_1 \lor \cdots \lor L_{i-1} \lor L_{i+1} \lor \cdots \lor L_n \lor L'_1 \lor \cdots \lor L'_{j-1} \lor L'_{j+1} \lor \cdots \lor L'_m) \theta
\]
is called the binary mega-resolvent of \( \varphi_1 \) and \( \varphi_2 \) upon \( p_1 : \mu_1 \) and \( p_2 : \mu_2 \).

We define a mega-clause \( \varphi \) to be regular if for any atom \( p \) in \( \varphi \), either \( \varphi = p : \mu \lor \varphi' \), or \( \varphi = \sim (p : \mu) \lor \varphi' \), or \( \varphi = p : \mu_1 \lor \sim (p : \mu_2) \lor \varphi' \), where \( \varphi' \) is a mega-clause in which \( p \) doesn't appear.

Since \( p : \mu_1 \lor p : \mu_2 \) and \( \sim (p : \mu_1) \lor \sim (p : \mu_2) \) are equivalent to \( p : \mu_2 \) and \( \sim (p : \mu_1) \) respectively if \( \mu_1 \geq \mu_2 \), any mega-clause is equivalent to a regular one. Thus, it is sufficient, in fact, for us to consider regular mega-clauses.

**Theorem 7** Let \( \varphi_1 \), \( \varphi_2 \) be two mega-clauses, \( \theta \) be an mgu of atom \( p_1 \) and \( p_2 \), where \( p_1 \) and \( p_2 \) are two atoms occurring in \( \varphi_1 \) and \( \varphi_2 \) respectively. Suppose \( \varphi_1 \theta = p_1 \theta : \mu_1 \lor \varphi'_1 \) and \( \varphi_2 \theta = \sim (p_2 \theta : \mu_2) \lor \varphi'_2 (\mu_1 \geq \mu_2) \) are regular, and \( \varphi'_1 \lor \varphi'_2 \) is the binary mega-resolvent of \( \varphi_1 \) and \( \varphi_2 \) upon \( p_1 : \mu_1 \) and \( p_2 : \mu_2 \). If \( \varphi_1 \theta \) and \( \varphi_2 \theta \) are not tautologies, then \( \Phi(c(\varphi'_1 \lor \varphi'_2)) + 1 \) is the superposant of \( c p_1 + 1 \) and \( c p_2 + 1 \) upon \( p_1 : \mu_1 \) and \( p_2 : \mu_2 \).

**Proof:** If \( p_1 \theta \) doesn't appear in both \( \varphi'_1 \) and \( \varphi'_2 \), then
\[
\Phi((c \varphi'_1 + 1)(c \varphi'_2 + 1)) = \Phi(\varphi'_1 \lor \varphi'_2) + 1
\]
is the superposant of \( c p_1 + 1 \) and \( c p_2 + 1 \) upon \( p_1 : \mu_1 \) and \( p_2 : \mu_2 \).

If \( p_1 \theta \) appears in \( \varphi'_1 \) but doesn't appear in \( \varphi'_2 \), then, let \( \varphi'_1 = \sim (p_1 \theta : \mu'_1) \lor \varphi''_1 \), where \( \varphi''_1 \) is a mega-clause in which \( p_1 \theta \) doesn't appear. Since \( \varphi_1 \theta \) is not a tautology, we have \( \mu_1 > \mu'_1 \). Thus,
\[
c(\varphi_1 \theta) + 1 = \Phi((c \varphi''_1 + 1)(p_1 \theta : \mu_1 + 1)(c \varphi'_1 + 1)(p_1 \theta : \mu'_1)).
\]
Because $\Phi((p_1: \mu_1 + 1)p_1: \mu_1) = p_1: \mu_1 + p_1: \mu_1$, we know that

$$\Phi((c\varphi'_1 + 1)(p_1: \mu_1 + 1)(c\varphi'_2 + 1)) = \Phi((c\varphi'_1 + 1)(c\varphi'_2 + 1)) = \Phi(\varphi'_1 \lor \varphi'_2 + 1)$$

is the superposant of $c\varphi'_1 + 1$ and $c\varphi'_2 + 1$ upon $p_1: \mu_1$ and $p_2: \mu_2$.

If $p_1: \mu_1$ appears in $\varphi'_2$ but doesn't appear in $\varphi'_1$, then, let $\varphi'_2 = p_1: \mu_2 \lor \varphi''_2$, where $\varphi''_2$ is a mega-clause in which $p_1: \mu_1$ doesn't appear. Since $\varphi_2: \mu_1$ is not a tautology, we have $\mu_2 > \mu_2$. Similarly, we can find that

$$\Phi((c\varphi'_1 + 1)(p_2: \mu'_2 + 1)(c\varphi''_2 + 1)) = \Phi((c\varphi'_1 + 1)(c\varphi'_2 + 1)) = \Phi(\varphi'_1 \lor \varphi'_2 + 1)$$

is the superposant of $c\varphi'_1 + 1$ and $c\varphi'_2 + 1$ upon $p_1: \mu_1$ and $p_2: \mu_2$.

If $p_1: \mu_1$ appears in both $\varphi'_1$ and $\varphi'_2$, then, let $\varphi'_1 = (p_1: \mu'_1) \lor \varphi''_1, \varphi'_2 = p_1: \mu'_2 \lor \varphi''_2$, where $\varphi''_1, \varphi''_2$ are mega-clauses in which $p_1: \mu_1$ doesn't appear. And we have $\mu_1 > \mu'_1, \mu'_2 > \mu_2$. Therefore, we can find that

$$\Phi((p_1: \mu'_1)(c\varphi''_1 + 1)(p_2: \mu'_2 + 1)(c\varphi''_2 + 1))$$

$$= \Phi((c\varphi'_1 + 1)(c\varphi'_2 + 1))$$

$$= \Phi(\varphi'_1 \lor \varphi'_2 + 1)$$

is the superposant of $c\varphi'_1 + 1$ and $c\varphi'_2 + 1$ upon $p_1: \mu_1$ and $p_2: \mu_2$. ■

This theorem shows that if we are restricted to clausal theorem proving, then, roughly speaking our inference rules are the same as resolution, the main difference in this case is the knowledge expression. We use annotated polynomials, one advantage to do so is that the tautologies are deleted naturally, and the equivalent clauses correspond to the same annotated polynomial. Also, we want to pay more attention to non-clausal theorem proving.

**Example** For a fuzzy graph, define the weight of a path to be the minimum of the weights of the edges composing it. The least unsure path between two vertices is a path between these two vertices with the maximal weight. Show that in the following fuzzy graph, the weight of the least unsure paths between $a$ and $e$ is not less than 0.5.

We only need to show that the following $PS$ is unsatisfiable.

$$PS = \left\{ \begin{array}{l}
(edge(x, y): \mu)(path(x, y): \mu + 1),
(path(x, y): \mu_1)(edge(y, z): \mu_2)(path(x, z): \min\{\mu_1, \mu_2\} + 1),
edge(a, b): 0.4 + 1, edge(a, c): 0.6 + 1, edge(c, d): 0.6 + 1,
edge(b, d): 0.5 + 1, edge(d, b): 0.5 + 1, edge(b, e): 0.6 + 1,
edge(d, e): 0.4 + 1, path(a, e): 0.5
\end{array} \right\}$$
where $\mu, \mu_1, \mu_2 \in \{0.4, 0.5, 0.6\}$. In fact, 
$(\text{edge}(x, y) : \mu)(\text{path}(x, y) : \mu + 1)$ 
states that if the weight of edge $(x, y)$ is not less than $\mu$ then, being a path, 
it has a weight not less than $\mu$. 
$(\text{path}(x, y) : \mu_1)(\text{edge}(y, z) : \mu_2)(\text{path}(x, z) : \min\{\mu_1, \mu_2\} + 1)$ represents that if the weight of a path between $x$ and $y$ is not less than $\mu_1$, and there is an edge $(y, z)$ whose weight is not less than $\mu_2$, then the weight of the path between $x$ and $z$ is not less than the minimum of $\mu_1$ and $\mu_2$. Each $\text{edge}(x, y) : \mu + 1$ stands for the statement that the weight of the edge $(x, y)$ is not less than $\mu$. $\text{path}(a, e) : 0.5$ represents that the weights of the paths between $a$ and $e$ are less than 0.5. $\text{path}(a, c) : 0.5 + 1$ is a superposant of $\text{edge}(a, c) : 0.6 + 1$ and $\text{edge}(b, e) : 0.6 + 1$. 
$\text{path}(a, d) : 0.5 + 1$ is a superposant of $\text{path}(a, c) : 0.5 + 1$ and $\text{edge}(d, b) : 0.5 + 1$. $\text{path}(a, e) : 0.5$ is a superposant of $\text{path}(a, c) : 0.5$ and $\text{path}(a, d) : 0.5 + 1$. 
$\text{path}(a, b) : 0.5 + 1$ is a superposant of $\text{path}(a, c) : 0.5$ and $\text{path}(a, e) : 0.5 + 1$. $\text{path}(a, e) : 0.5 + 1$ is a superposant of $\text{path}(a, b) : 0.5 + 1$ and $\text{path}(a, e) : 0.5 + 1$. $1$ is a superposant of $\text{path}(a, c) : 0.5 + 1$ and $\text{path}(a, e) : 0.5 + 1$. Thus $PS$ is unsatisfiable.

4.2 Completeness

Hereafter, we show that the given proof procedure is refutationally complete. It is difficult to apply the well-known Anderson and Bledsoe technique or the method of shrinking semantic trees [CL 73][LSC 91] just as [Hs 85][Zh 94] did. We first begin with several necessary concepts.

As usual, a deduction from $PS$ to $C_n$ is defined to be a sequence $c_1, \ldots, c_n$, where either there exists a $c \in PS$, such that $c_1 = \Phi(c)$, or $c_1$ is a factor of $c_k(k < i)$, or $c_1$ is a superposant of $c_1$ and $c_s(t, s < i)$ for $i \in \{1, \ldots, n\}$.

Suppose $\Phi(c) \neq 0$, $m$ is the number of the monomials occurring in $\Phi(c)$ which is not 1. If 1 doesn’t appear in $\Phi(c)$, we say that the length of $c$, denoted by $|c|$, is $m$; Otherwise, $|c|$ equals $m + 0.5$.

(1) The ground case

In what follows, we use $r(c_1, c_2)$ to represent the superposant of $c_1$ and $c_2$ upon two annotated atoms, say $p_1 : \mu_1$ and $p_2 : \mu_2$, in $c_1$ and $c_2$ respectively. Whereas $r(c_1(p \leftarrow \mu), c_2(p \leftarrow \mu))$ represents the superposant of $c_1(p \leftarrow \mu)$ and $c_2(p \leftarrow \mu)$ upon $p_1 : \mu_1$ and $p_2 : \mu_2$ if $p_1 : \mu_1$ and $p_2 : \mu_2$ appear in $\Phi(c_1(p \leftarrow \mu))$ and $\Phi(c_2(p \leftarrow \mu))$, respectively.

Lemma 1 Suppose that $c_1, c_2$ are ground. Then $r(c_1(p \leftarrow \mu), c_2(p \leftarrow \mu)) = r(c_1, c_2)(p \leftarrow \mu)$. 

Proof: Let
\[ \Phi(c_i) = \left( \sum_k a_{ik} p : \mu_{ik} + b_i \right) \mu_i + \sum_l d_{il} p : \mu_{il} + e_i, \]
where atom \( p \) does not appear in \( a_{ik}, b_i, d_{il} \) and \( e_i \), atom \( p_1 \) doesn’t appear in \( a_{ik} \) and \( b_i \), runed atom \( p_1 : \mu_i \) doesn’t appear in \( d_{il} \) and \( e_i (i = 1, 2) \), and
\[ r(c_1, c_2) = \Phi(\sum_l d_{1l} p : \mu_{1l} + e_1) (\sum_k a_{2k} p : \mu_{2k} + b_2 + \sum_l d_{2l} p : \mu_{2l} + e_2)). \]
Therefore,
\[ r(c_1, c_2)(p \leftarrow \mu) = \Phi(\sum_l d_{1l} p : \mu_{1l} + e_1) (\sum_k a_{2k} p : \mu_{2k} + b_2 + \sum_l d_{2l} p : \mu_{2l} + e_2)) = r(c_1 (p \leftarrow \mu), c_2 (p \leftarrow \mu)). \]

Lemma 2 Let \( c \) be ground, and \( \Phi(c)(p \leftarrow \mu) = 1 \). Then there are a \( c_0 \) and a deduction from \( \{ c \} \) to \( c_0 \), such that either \( c_0 = 1 \), or \( p \) is the unique atom occurring in \( c_0 \) and \( c_0(p \leftarrow \mu) = 1 \).

Proof: If \( \Phi(c) = 1 \), we are done. Otherwise, we can write \( \Phi(c) \) in the form
\[ \Phi(c) = \sum_{k=1}^n a_k (p : \mu_k + i_k) + 1, \]
where \( \mu_t \neq \mu_s \) if \( t \neq s \), \( i_k = 0 \) if \( \mu_k > \mu \); otherwise \( i_k = 1 \), \( a_k \) are non-zero, and atom \( p \) does not appear in \( a_k \). Let \( m \) be the number of the annotated atoms occurring in \( a_1, \ldots, a_n \). We want to use mathematical induction on \( m \) to complete the proof.

If \( m = 0 \), then
\[ \Phi(c) = \sum_{k=1}^n (p : \mu_k + i_k) + 1. \]
Let \( c_0 = \Phi(c) \). The conclusion clearly holds. Suppose now the conclusion holds for \( m(m \geq 1) \). Let \( p' : \mu' \) be an annotated atom occurring in \( a_{tj} (j = 1, \ldots, s, tj \in \{1, \ldots, n\}) \), and \( p' : \mu' \) doesn’t appear in \( a_q (q \in \{1, \ldots, n\} \{t1, \ldots, ts\}) \). Let \( a_{tj} = d_{tj} p' : \mu' + e_{tj} \). Then
\[ \Phi(c) = \sum_{j=1}^s d_{tj} (p : \mu_{tj} + i_{tj}) p' : \mu' + \sum_{j=1}^s e_{tj} (p : \mu_{tj} + i_{tj}) + \sum_{k \notin \{t1, \ldots, ts\}} a_k (p : \mu_k + i_k) + 1. \]
Let \( a = \sum_{j=1}^s d_{tj} (p : \mu_{tj} + i_{tj}), b = \sum_{j=1}^s e_{tj} (p : \mu_{tj} + i_{tj}) + \sum_{k \notin \{t1, \ldots, ts\}} a_k (p : \mu_k + i_k) + 1. \) Since \( a \neq 0 \), \( \Phi(b(a + b)) \) is the superposant of \( c \) and itself upon \( p' : \mu' \) and \( p' : \mu' \). Because \( a(p \leftarrow \mu) = 0, b(p \leftarrow \mu) = 1 \), we have \( \Phi(b(a + b))(p \leftarrow \mu) = 1 \). Since the number of the annotated atoms occurring in \( \Phi(b(a + b)) \) in which the atoms are not \( p \) is less than \( m \), by the induction hypothesis, we know that there are a \( c_0 \) and a deduction \( D_1 \) from \( \{ \Phi(b(a + b)) \} \) to
c₀, such that c₀ = 1 or p is the unique atom occurring in c₀ and c₀(p ↷ μ) = 1. Thus, Φ(c), D₁ is what we want.

Lemma 3 Suppose c is ground, in which only one atom p appears, c(p ↷ μ) = 1, and |c| ≥ 2.5. Then there exist a c₀ and a deduction from {c} to c₀, such that only p appears in c₀, c₀(p ↷ μ) = 1, and |c₀| ≤ 2.5.

Proof: If |c| ≤ 2.5, the conclusion holds obviously. Now, suppose |c| ≥ 3, and

\[ \Phi(c) = \sum_{i=1}^{n} p : \mu_i + r, \mu_1 > \cdots > \mu_n, r \in \{0, 1\}, n \geq 3. \]

Let m be the number of the elements of the set \{μᵢ ∈ \{μ₁, ..., μₙ\} | μᵢ > μ\}.

Case 1: r = 1.

Case 1.1: n is odd.

In this case, n ≥ 3, and m is odd. If m ≠ n, then, since n ≠ 1, we can compute that p : μₙ + p : μₙ₊₁ is the superposant of c and c upon p : μₙ and p : μₙ₊₁. Clearly, (p : μₙ + p : μₙ₊₁)(p ↷ μ) = 1, and |p : μₙ + p : μₙ₊₁| = 2.

So we can choose c₀ = p : μₙ + p : μₙ₊₁. If n = m, then for any i = 1, ..., n, (p : μᵢ)(p ↷ μ) = 0. However, p : μ₁ + p : μ₂ is the superposant of c and c upon p : μ₁ and p : μ₂, c₁ = ∑ᵢ₌₃ p : μᵢ₊₁ is the superposition of c and p grave 1 + p grave 2 upon p grave 1 and p grave 2, the number of the annotated atoms occurring in c₁ is n − 2, it's odd and for arbitrary i = 3, ..., n, (p : μᵢ)(p ↷ μ) = 0. So we can repeat the above procedure from c₁ and obtain p : μₙ + 1 eventually. Thus we may choose c₀ = p : μₙ + 1.

Case 1.2: n is even.

In this case, n ≥ 4, and m is even. If 0 < m < n, then, since n ≠ 2, p : μₙ + p : μₙ₊₁ is the superposant of c and c upon p : μₙ and p : μₙ₊₁. So we can choose c₀ = p : μₙ + p : μₙ₊₁. If m = 0, then for any i = 1, ..., n, (p : μᵢ)(p ↷ μ) = 1. Since p : μ₂ + p : μ₃ is the superposant of c and c upon p : μ₂ and p : μ₃, p : μ₁ is the superposant of p : μ₂ + p : μ₃ and c upon p : μ₂ and p : μ₂, we can choose c₀ = p : μ₁. If m = n, then for any i = 1, ..., n, (p : μᵢ)(p ↷ μ) = 0. However, p : μ₂ + p : μ₃ is the superposant of c and c upon p : μ₂ and p : μ₃, c₁ = ∑ᵢ₌₄ p : μᵢ₊₁ is the superposant of c and p : μ₂ + p : μ₃ upon p : μ₂ and p : μ₂, and the number of the annotated atoms occurring in c₁ is n − 3, it’s odd. Like Case 1.1, we can eventually get p : μₙ + 1, and choose c₀ = p : μₙ + 1.

Case 2: r = 0.

Case 2.1: n is odd.

In this case, n ≥ 3 and m is even. If m ≠ 0, then, because n ≠ 1, p : μₙ + p : μₙ₊₁ is the superposant of c and c upon p : μₙ and p : μₙ₊₁. We can choose c₀ = p : μₙ + p : μₙ₊₁. If m = 0, since p : μ₂ + p : μ₃ is the superposant of c and c upon p : μ₂ and p : μ₃, p : μ₁ is the superposant of p : μ₂ + p : μ₃ and c upon p : μ₂ and p : μ₂, and (p : μ₁)(p ↷ μ) = 1, we may choose c₀ = p : μ₁.
Case 2.2: $n$ is even.

In this case, $n \geq 4$, $m$ is odd, and $0 < m < n$. Thus, because $n \neq 2$, $p : \mu_m + p : \mu_{m+1}$ is the superposant of $c$ and $c$ upon $p : \mu_m$ and $p : \mu_{m+1}$, and we can choose $c_0 = p : \mu_m + p : \mu_{m+1}$.

Lemma 4 Let $PS$ be ground, in which only one atom appears, and for each $c \in PS$, $|c| \leq 2.5$. If $PS$ is unsatisfiable, then there exist a $PS_0$ and deductions from $PS$ to each $c_0 \in PS_0$, such that $PS_0$ is unsatisfiable, and $|c_0| \leq 2$ for every $c_0 \in PS_0$.

Proof: Suppose in $PS$ there are $m$ annotated polynomials whose length are 2.5. We use mathematical induction on $m$.

When $m = 0$, we can choose $PS_0 = PS$. The conclusion holds clearly. Now suppose it is true for $< m$. Choose a $c \in PS$, $\Phi(c) = p : \mu_1 + p : \mu_2 + 1$, where $\mu_1 > \mu_2$. Choose $\mu$ such that $\mu_1 < \mu_2$. Since $PS$ is unsatisfiable, there is a $c_1 \in PS$, such that $c_1 (p \leftarrow \mu) = 1$.

Case 1: $|c_1| = 0.5$.

In this case, $\Phi(c_1) = 1$, we are done.

Case 2: $|c_1| = 1.5$.

Assume $\Phi(c_1) = p : \mu_1 + 1$, then $\mu < \mu_1$. Thus $\mu_1 > \mu_2$. For arbitrary $\mu'$, $c(p \leftarrow \mu') = 1$, we have $\mu' \geq \mu_1$ or $\mu' < \mu_2$. For the former case, $(p : \mu_1)(p \leftarrow \mu') = 1$, for the latter case, $(p : \mu_2 + 1)(p \leftarrow \mu') = 1$. Thus $PS_1 = (PS - \{c\}) \cup \{p : \mu_1, p : \mu_2\}$ is unsatisfiable. Clearly, the number of the elements of $PS_1$ whose length are 2.5 is less than $m$. By the induction hypothesis, the conclusion holds for $PS_1$. However, $p : \mu_2 + 1$ is the superposant of $c$ and $c_1$ upon $p : \mu_1$ and $p : \mu_2$, $p : \mu_1$ is the superposant of $p : \mu_2 + 1$ and $c$ upon $p : \mu_2$ and $p : \mu_2$. We thus get the result as required.

Case 3: $|c_1| = 1.5$.

Assume $\Phi(c_1) = p : \mu_1 + 1$, then $\mu < \mu_1$. Thus $\mu_1 > \mu_2$. Let $PS_1$ be the same as that of Case 2, then the conclusion holds for $PS_1$. However, $p : \mu_1$ is the superposant of $c_1$ and $c$ upon $p : \mu_1$ and $p : \mu_2$, $p : \mu_2 + 1$ is the superposant of $c$ and $p : \mu_2$ upon $p : \mu_1$ and $p : \mu_2$. We are done.

Case 4: $|c_1| = 2$.

Assume $\Phi(c_1) = p : \mu_1 + p : \mu_2$, then $\mu_1 < \mu_2$. Thus $\mu_1 > \mu_2$. Let $PS_1$ be the same as that of Case 2, then the conclusion holds for $PS_1$. However, $p : \mu_1$ is the superposant of $c_1$ and $c$ upon $p : \mu_1$ and $p : \mu_2$, $p : \mu_2 + 1$ is the superposant of $c$ and $c_1$ upon $p : \mu_1$ and $p : \mu_2$. We are done.

Case 5: $|c_1| = 2.5$.

Assume $\Phi(c_1) = p : \mu_1 + p : \mu_2 + 1(\mu_1 > \mu_2)$, then $\mu_1 < \mu_2$. Thus $\mu_1 > \mu_2$. Since $c(p \leftarrow \mu) = 0, \Phi(c) \neq \Phi(c_1)$. If $\mu \geq \mu_1$, then, if $\mu_2 \geq \mu_1$, we have $\mu_1 < \mu_2 < \mu_1$. Hence, for arbitrary $\mu': c(p \leftarrow \mu') = 1$, we have $c_1(p \leftarrow \mu') = 1$. Thus $PS - \{c\}$ is unsatisfiable, and the number of the elements in $PS - \{c\}$ whose length are 2.5 is less than $m$. By induction hypothesis, the conclusion holds for $PS - \{c\}$. Of course, it holds for
Theorem 8 Suppose $PS$ is ground. If $PS$ is unsatisfiable, then there exists a deduction from $PS$ to $P S'$. 

Proof: We want to use mathematical induction on the number $m$ of the atoms occurring in $PS$. Let $\mu_1, \cdots, \mu_n (\mu_1 > \cdots > \mu_{n-1})$ be all the annotations occurring in $PS$, and $\mu_n < \mu_{n-1}$. We want to use mathematical induction on the number $m$ of the atoms occurring in $PS$, and $\mu_n < \mu_{n-1}$. We can obtain the result as required. So now we can assume for any $c' \in PS$, we can get the result as required. If $\mu < \mu_s$, then, for the two cases $\mu < \mu_t$ and $\mu_t \geq \mu_i$, we may prove the conclusion similarly. 

Lemma 5 Let $PS$ be ground, in which only one atom appears, and for each $c \in PS$, $|c| \leq 2$. If $PS$ is unsatisfiable, then there exist a $PS_0$ and deductions from $PS$ to each $c_0 \in PS_0$, such that $PS_0$ is unsatisfiable, and $|c_0| \leq 1.5$ for every $c_0 \in PS_0$. 

Proof: Suppose $\mu_1, \cdots, \mu_{n-1} (\mu_1 > \cdots > \mu_{n-1})$ are all the annotations occurring in $PS$, and $\mu_n < \mu_{n-1}$. Then $PS$ is unsatisfiable if for arbitrary $\mu_k$, there exists $c \in PS$ such that $c(p \leftarrow \mu_k) = 1$. We use mathematical induction on $m$, the number of the elements of $PS$ whose length are 2. 

When $m = 0$, we can choose $PS_0 = PS$. Now suppose it is true for $m$. Choose a $c_0 \in PS$, $P \leftarrow c_0 = p : \mu_i + p : \mu_j (\mu_i > \mu_j)$. For $c \in PS$, we see that $c(p \leftarrow \mu_i) = 0$ if $|c| \geq 1.5$. So there is at least a $c_1 \in PS$ such that $|c_1| = 1$. Suppose $\Phi(c_1) = p : \mu_r$.

If $\mu_i < \mu_j$, then $p : \mu_j$ is the superposant of $c_1$ and $c_0$ upon $p : \mu_i$ and $p : \mu_j$, and for arbitrary $\mu_k$, if $c_0(p \leftarrow \mu_k) = 1$ then $(p : \mu_j)(p \leftarrow \mu_k) = 1$. Thus, $PS_1 = (PS - \{c_0\}) \cup \{p : \mu_j\}$ is unsatisfiable. The number of the elements of $PS_1$ whose length is 2 is less than $m$. By the induction hypothesis, the conclusion holds for $PS_1$. Hence we are done. So now we can suppose for any $\mu_k < \mu_i$ and $c \in PS$, if $c(p \leftarrow \mu_k) = 1$, then $|c| \neq 1$. Hence, there exist $\mu_k \geq \mu_i$, and $c \in PS$, $|c| = 1$ such that $c(p \leftarrow \mu_k) = 1$. If there are $\mu_k = \mu_i$, and $c_2 \in PS$, $|c_2| = 1.5$ such that $c_2(p \leftarrow \mu_k) = 1$, let $\Phi(c_2) = p : \mu_i + 1$. Then $\mu_i > \mu_j > \mu_k$. However, for arbitrary $\mu_k$, if $c_0(p \leftarrow \mu_k) = 1$ then $(p : \mu_k + 1)(p \leftarrow \mu_k) = 1$. Thus, $PS_1 = PS - \{c_0\}$ is unsatisfiable, and we can obtain the result as required. So now we can assume for any $\mu_k \geq \mu_i$ and $c \in PS$, if $c(p \leftarrow \mu_k) = 1$ then $|c| = 1$ or $|c| = 2$.

Clearly, there exist $\mu_s \geq \mu_i$, and $c_3, c_4 \in PS$, with $|c_3| = 1, |c_4| = 2$ such that $c_3(p \leftarrow \mu_s) = 1, c_4(p \leftarrow \mu_{i+1}) = 1$. Let $\Phi(c_3) = p : \mu_i, \Phi(c_4) = p : \mu_g + p : \mu_h (\mu_g > \mu_h)$. Then $\mu_s \geq \mu_i, \mu_h \leq \mu_{i+1} < \mu_g$. Thus $\mu_s \leq \mu_g, p : \mu_h$ is the superposant of $c_3$ and $c_4$ upon $p : \mu_g$ and $p : \mu_i$. For arbitrary $\mu_k$, if $c_4(p \leftarrow \mu_k) = 1$ then $(p : \mu_k)(p \leftarrow \mu_k) = 1$. Thus, for any $\mu_k$, there exists $c' \in PS' = (PS - \{c_4\}) \cup \{p : \mu_h\}$, such that $c'(p \leftarrow \mu_k) = 1$. So $PS_1$ is unsatisfiable. We can thus get the result by the induction hypothesis. 

Theorem 8 Suppose $PS$ is ground. If $PS$ is unsatisfiable, then there exists a deduction from $PS$ to $P S'$. 

Proof: We want to use mathematical induction on the number $m$ of the atoms occurring in $PS$. Let $\mu_1, \cdots, \mu_n (\mu_1 > \cdots > \mu_{n-1})$ be all the annotations occurring in $PS$, and $\mu_n < \mu_{n-1}$. 

…
If \( m = 1 \), according to Lemma 3, 4 and 5, we know that there exist a \( PS_0 \) and deductions from \( PS \) to each \( c_0 \in PS_0 \), such that \( PS_0 \) is unsatisfiable, and \(|c_0| \leq 1.5 \) for every \( c_0 \in PS_0 \). Therefore, there must be \( c_1, c_2 \in PS_0 \) such that \( \Phi(c_1) = p : \mu_i + 1, \Phi(c_2) = p : \mu_j, \) and \( \mu_i > \mu_j \). Since 1 is the superposant of \( c_1 \) and \( c_2 \) upon \( p : \mu_i \) and \( p : \mu_j \), there is a deduction from \( PS \) to 1.

Now suppose the conclusion holds for \( m > 1 \). Let \( p \) be an atom occurring in \( PS \). Then \( PS_k = \{ (p \leftarrow \mu_k) | c \in PS \} \) is unsatisfiable. Since the number of the atoms in each \( PS_k \) is less than \( m \), from the induction hypothesis, we know that for arbitrary \( k \in \{ 1, \ldots, n \} \), there is a deduction from \( PS_k \) to 1. According to Lemma 1 and Lemma 2, there are deductions from \( PS \) to \( c_k \), where \( c_k(p \leftarrow \mu_k) = 1 \) for \( k = 1, \ldots, n \). Let \( PS_0 = \{ c_k | k = 1, \ldots, n \} \), then \( PS_0 \) is unsatisfiable. From the case of \( m = 1 \), we know that there is a deduction from \( PS_0 \) to 1. Thus there exists a deduction from \( PS \) to 1.

(2) Lifting

**Lemma 6** Let \( c_1' \) and \( c_2' \) be ground instances of \( c_1 \) and \( c_2 \) respectively, and \( c' \) a superposant of \( c_1' \) and \( c_2' \). Then there exist a superposant \( c \) of a factor of \( c_1 \) and a factor of \( c_2 \), such that \( c' \) is an instance of \( c \).

**Proof:** For \( i = 1, 2 \), let \( c_i' = a_i p' : \mu_i' + b_i' \), \( \mu_1 \geq \mu_2 \), and \( c' = \Phi(b_1'(a_2' + b_2')) \). Suppose \( c' = \Phi(c\theta) \), where \( \theta \) is a substitution, and \( p_{ki} \) are all the atoms occurring in \( c_i \) such that \( p_{ki}\theta = p' \). Let

\[
\Phi(c_i) = \sum_t d_{it}U_{it} + \sum_s e_{is}V_{is} + r_i,
\]

where each atom \( p_{ki} \) doesn't appear in \( d_{it}, e_{is}, r_i \), if some atoms \( p_{ki} \) appear in \( r_i \), then the annotations of \( p_{ki} \) are less than \( \mu_i' \), and \( U_{it}, V_{is} \) are monomials in which only some atoms \( p_{ki} \) appear, and in \( U_{it} \) there is an annotation which is \( \mu_i' \) and the remaining annotations are not greater than \( \mu_i' \), and in \( V_{is} \) there is an annotation which is greater than \( \mu_i' \). Let \( \lambda_i \) be an mgu of all these \( p_{ki} \). Without loss of generality, we assume that \( c_1 \) and \( c_2 \) have no common variable symbols. Let \( \lambda = \lambda_1 \cup \lambda_2 \), and \( \alpha \) be an mgu of \( p_{k1}\lambda \) and \( p_{k2}\lambda \), then \( \lambda \circ \alpha \) is an mgu of all the atoms \( p_{k1}, p_{k2} \). Thus there exists a substitution \( \sigma \) such that \( \theta = \lambda \circ \alpha \circ \sigma \). Obviously,

\[
\Phi(c_i\lambda) = \Phi((\sum_t d_{it}\lambda)_{pi1\lambda} : \mu_i' + (\sum_s e_{is}V_{is} + r_i)\lambda).
\]

Let \( b_i = (\sum_s e_{is}V_{is} + r_i)\lambda \), \( a_i = \sum_t d_{it}\lambda \). Since \( p_{i1}\lambda : \mu_i' \) doesn't appear in \( b_i \), and \( p_{i1}\lambda \) doesn't appear in \( a_i \), \( c = \Phi(b_1(\alpha(a_2 + b_2)\alpha) \) is a superposant of \( c_1\lambda \) and \( c_2\lambda \). Because \( a_i' = \Phi(a_i\sigma), b_i' = \Phi(b_i\sigma) \), we have \( c' = \Phi(c\sigma) \).

**Lemma 7** If 1 is an instance of \( c \), then there is a deduction from \( \{ c \} \) to 1.
Proof: We use mathematical induction on \( m \), the number of the atoms occurring in \( c \). Suppose \( \Phi(c\theta) = 1 \), \( \theta \) is a substitution.

If \( m = 0 \), then \( \Phi(c) = 1 \). We are done. Suppose the conclusion holds for \(< m(m \geq 1) \). Clearly, \( m \neq 1 \). Therefore, there must be two atoms \( p_1 \) and \( p_2 \) occurring in \( c \), such that \( p_1\theta = p_2\theta \). Let \( \lambda \) be an mgu of \( p_1 \) and \( p_2 \), then \( \Phi(c\lambda) \) is a factor of \( c \), and 1 is still an instance of \( c\lambda \). Since the number of the atoms occurring in \( c\lambda \) is less than \( m \), by induction hypothesis, we know that there is a deduction \( D_1 \) from \( \{ c\lambda \} \) to 1. So the sequence \( c, D_1 \) is what we want.

Theorem 9 If \( PS \) is unsatisfiable, then there exists a deduction from \( PS \) to 1.

Proof: From Theorem 5, we know that there is an \( i \in \{ 0, 1, \cdots \} \) such that \( PS|H_i \) is unsatisfiable. So there exists a deduction from \( PS|H_i \) to 1 according to Theorem 8. By Lemma 6, we know that there are \( c \) and a deduction from \( PS \) to \( c \), such that 1 is an instance of \( c \). However, according to Lemma 7, there exists a deduction from \( \{ c \} \) to 1. We thus get the result as required.

This theorem tells us that the proof procedure we presented is refutationally complete.

5 Strategies

It is well known that two major works in theorem proving are how to express knowledge and how to establish inference rules. On the other hand, developing various strategies is another important job, because efficiency is a major concern in any proof procedures.

5.1 Subsumption

 Unlimited applications of the inference rules may produce many redundant annotated polynomials. In this subsection, we introduce a rule to delete some of such annotated polynomials.

For two given \( c_1 c_2 \), if there exists a substitution \( \theta \) such that \( \Phi(c_2(1+c_1\theta)) = 0 \), then we say that \( c_1 \) subsumes \( c_2 \). For example, suppose \( c_1 = p(x) : \mu_1 + 1 \), \( c_2 = p(a) : \mu_2 + p(a) : \mu_3(\mu_1 \geq \mu_2 > \mu_3) \). Then \( c_1 \) subsumes \( c_2 \), where \( \theta = \{ a/x \} \).

In the proof procedure, for any two annotated polynomials in \( PS \) or produced by the inference rules, say \( c_1 \) and \( c_2 \), if \( c_1 \) subsumes \( c_2 \), then we delete \( c_2 \). Since for an arbitrary interpretation \( I \), if \( c_1 \) subsumes \( c_2 \) and \( I \) doesn’t satisfy \( c_2 \), then \( I \) doesn’t satisfy \( c_1 \). So the deletion strategy does not affect the unsatisfiability. As usual, the completeness of our deletion strategy depends upon how the annotated polynomials are deleted.

Now, the problem is how to decide whether there exists a substitution \( \theta \) such that \( \Phi(c_2(1+c_1\theta)) = 0 \). As a matter of fact, without loss of generality,
we assume that $c_1$ and $c_2$ have no common variable symbols, then this problem is equivalent to finding out a unifier of two monomials. However, the latter problem is not hard to solve.

Before we end this subsection, we remark that the subsumption defined here includes that of [LSC 91].

5.2 M-strategy: introducing an order on atoms

We develop here a strategy which is called M-strategy. One advantage, besides restricting superposition, and the reader can see later, is that it is symmetric.

By $>_A$ we denote a partial ordering (a transitive and irreflexive binary relation) on atoms, $p >_A 1 >_A 0$ holds for any atom $p$ and $>_A$ is stable, namely, for any atoms $p_1, p_2$ and any substitution $\theta$, $p_1 >_A p_2$ implies $p_1\theta >_A p_2\theta$. We say that atom $p$ is maximal (minimal) in $c$, if for any $p_1$ occurring in $c$, $p_1 >_A p$ ($p <_A p_1$).

In the definition of factors, if $p_i\theta$ is maximal in $c_\theta$, we call $\Phi(c\theta)$ an M-factor of $c$. In the definition of superposants, if $p_i\theta$ is maximal in $c_1$ and $c_2$ (upon $p_1 : \mu_1$ and $p_2 : \mu_2$), Similarly, we derive the definition of an M-deduction from $P_S$ to $C_n$ if superposant and factor are replaced by M-superposant and M-factor respectively in the definition of a deduction from $P_S$ to $C_n$. We call the proof procedure using M-deduction M-strategy. Now we show the completeness of M-strategy. We use $r(c_1, c_2)$ to represent an M-superposant of $c_1$ and $c_2$, and assume that atom $p$ is minimal in both $c_1$ and $c_2$.

**Lemma 8** Suppose that $c_1, c_2$ are ground. Then there are an $r$ and an M-deduction from \( \{c_1, c_2\} \) to $r$ such that $r(p \leftarrow \mu) = r(c_1(p \leftarrow \mu), c_2(p \leftarrow \mu))$.

**Proof:** Suppose $r(c_1(p \leftarrow \mu), c_2(p \leftarrow \mu))$ is the M-superposant of $c_1(p \leftarrow \mu)$ and $c_2(p \leftarrow \mu)$ upon $p_1 : \mu_1$ and $p_1 : \mu_2$. Then for $i = 1, 2$, we claim that there exist $d_i$ and an M-deduction from $\{c_i\}$ to $d_i$ such that $p_1$ is maximal in $d_i$, and $d_i(p \leftarrow \mu) = c_i(p \leftarrow \mu)$. In fact, Let $M_i = \{p_2 | p_2$ occurs in $c_i$, and $p_2 >_A p_1\}$. If $M_i = \emptyset$, we are done. Otherwise, let $p_2$ be a maximal atom in $M_i$, $p_2 : \mu'$ is an annotated atom occurring in $c_i$, and

$$
\Phi(c_i) = \sum_j (a_j p_2 : \mu' + b_j(p : \mu_j + l_j) + \Phi(c_i(p \leftarrow \mu)),
$$

where $p_2$ doesn’t appear in $a_j \neq 0$, $p_2 : \mu'$ doesn’t appear in $b_j$, $l_j = 0$ if $\mu_j > \mu$, otherwise $l_j = 1$. Now, let $a = \sum_j a_j p : \mu_j + l_j$, $b = \sum_j b_j(p : \mu_j + l_j) + \Phi(c_i(p \leftarrow \mu))$. Then $d = \Phi(b(a + b))$ is the M-superposant of $c_i$ and itself upon $p_3 : \mu'$ and $p_2 : \mu'$. Since $a(p \leftarrow \mu) = 0$, $d(p \leftarrow \mu) = \Phi(c_i(p \leftarrow \mu))$. Thus, if $p_1$ is maximal in $d$, then we can choose $d_i = d$, and we get the result as required. Otherwise, we continue this procedure repeatedly, we eventually get the $d_i$ satisfying the condition and there is an M-deduction from $\{c_i\}$ to $d_i$. However, by Lemma 1, we know that $r(d_1, d_2)(p \leftarrow \mu) = r(c_1(p \leftarrow \mu), c_2(p \leftarrow \mu))$. ■
In the proof of Lemma 2, we can choose $p'$ to be a maximal atom in $c$. So the conclusion still holds if in Lemma 2 the word "deduction" is replaced by "M-deduction". By this fact, Lemma 8, and Lemma 3, 4 and 5, we have

**Theorem 10** Suppose $PS$ is ground and unsatisfiable. Then there exists an $M$-deduction from $PS$ to 1.

**Lemma 9** Let $c'$ be an instance of $c$, $p'$ appear and be maximal in $c'$. Then, there exist $d$, a substitution $\theta$ and an $M$-deduction from $\{c\}$ to $d$, such that $c' = \Phi (d \theta)$, and there is a unique atom $p$ occurring in $d$, $p \theta = p'$ and $p \theta$ is maximal in $d \theta$.

**Proof:** Let $c' = \Phi (c \lambda)$, where $\lambda$ is a substitution, and $M_c = \{ p_1 \mid p_1$ occurs in $c$ and $p_1 \lambda > A p' \}$.

If $M_c = \emptyset$, let $M'_c = \{ p_1 \mid p_1$ occurs in $c$ and $p_1 \lambda = p' \}$. If $M'_c$ contains only one element $p_1$, we can choose $d = c$, $p = p_1$, $\theta = \lambda$. Otherwise choose two different $p_1, p_2 \in M'_c$. Since $p_1 \lambda = p_2 \lambda = p'$, there exists an mgu $\sigma$ of $p_1$ and $p_2$. Suppose $\lambda = \sigma \circ \alpha$, where $\alpha$ is a substitution. Then $p_1 \sigma$ must be maximal in $c \sigma$. Otherwise, there exists an $p_3$ occurring in $c$, such that $p_3 \sigma > A p_1 \sigma$. Hence, $p_3 \sigma \circ \alpha > A p_1 \sigma \circ \alpha$, i.e., $p_3 \lambda > A p_1 \lambda = p'$, $p_3 \in M_c = \emptyset$. This is impossible. At the same time, we know $c' = \Phi (c \lambda) = \Phi (c \sigma \circ \alpha)$, $c'$ is an instance of $c \sigma$. $\Phi (c \sigma)$ is an M-factor of $c$. If there is a unique atom $p$ occurring in $c \sigma$, such that $p \theta = p'$ and $p \theta$ is maximal in $d \theta$, we can choose $d = c \sigma$, $\theta = \sigma$, and we are done. Otherwise, continue this procedure from $c \sigma$ and $c'$, we eventually get the result as required.

If $M_c \neq \emptyset$, let $N_c = \{ p_1 \mid p_1$ occurs in $c$, $p_1 \lambda$ is maximal in $c \lambda \}$. Then $N_c \cap M_c \neq \emptyset$. Indeed, choose an $p_1 \in M_c$, if $p_1 \notin N_c$, there is an $p_2 \in N_c$, such that $p_2 \lambda > A p_1 \lambda$. Since $p_1 \lambda > A p', p_2 \lambda > A p'$, so $p_2 \in M_c$. Now choose some $p_1 \in N_c \cap M_c$. If for any $p_i (i \neq 1)$ occurring in $c$, we have $p_i \lambda \neq p_1 \lambda$, then if we let $\Phi (c) = \sum_{j=1} \kappa_j p_1 : \mu_j + b$, where $p_1$ does not appear in $a_j \neq 0$, $p_1 : \mu_j$ does not appear in $b$, then for arbitrary atom $p_i$ occurring in $a_j$ or $b$, $p_i \lambda \neq p_1 \lambda$. Thus $\Phi (a_j \lambda) = 0$ holds for any $j \geq 1$, and $c' = \Phi (b \lambda)$ since $p_1 \lambda > A p'$ and $p'$ is maximal in $c$. We claim that $p_1$ is maximal in $c$. Otherwise, there is an $p_2$ occurring in $c$, such that $p_2 > A p_1$. Therefore $p_2 \lambda > A p_1 \lambda$. But this is impossible because $p_1 \in N_c$. Let

$$c_1 = \Phi \left( \sum_{j=2} a_j p_1 : \mu_j + b \right) \left( a_1 + \sum_{j=2} a_j p_1 : \mu_j + b \right),$$

then $c_1$ is an M-superposant of $c$ and $c$ upon $p_1 : \mu_1$ and $p_1 : \mu_1$. Clearly, $\Phi (c_1 \lambda) = \Phi (b \lambda) = c'$. So $c'$ is still an instance of $c_1$. Continue this produce from $c_1$, until we obtain a $c_k$ such that $p_1$ does not appear in $c_k$, and $c'$ is an instance of $c_k$. If there is $p_i (i \neq 1)$ occurring in $c$, such that $p_i \lambda = p_1 \lambda$, then let $\sigma$ be an mgu of $p_1$ and $p_i$. Similar to the proof of the case of $M_c = \emptyset$, $c_k = \Phi (c \sigma)$ is an M-factor of $c$. If $c_k$ satisfies the condition required, we are done. Otherwise, continue this procedure, we can eventually get what we want.
According to Lemma 9 and 6, we have lifting lemma of the following form.

**Lemma 10** Let $c'_1$ and $c'_2$ be ground instances of $c_1$ and $c_2$, respectively. Suppose $c'$ is an M-superposant of $c'_1$ and $c'_2$. Then there exist a $c$ and an M-deduction from $\{c_1, c_2\}$ to $c$, such that $c'$ is an instance of $c$.

As a corollary of Lemma 9, we can prove that there is an M-deduction from $\{c\}$ to 1 if 1 is an instance of $c$. From this fact, Theorem 5, Theorem 10 and Lemma 9, we have the following theorem, which states that M-strategy is refutationally complete in fact.

**Theorem 11** If $PS$ is unsatisfiable, then there exists an M-deduction from $PS$ to 1.

Note that this theorem also demonstrates that M-strategy is refutationally complete for classical logic, because classical logic may be viewed as the special case that the set of the annotations occurring in $PS$ is a singleton. Compared with the approach given in [KN 85], M-strategy uses single overlaps.

### 5.3 A remark

It is very interesting to consider introducing semantic strategy in the proof procedure. Let $\mu$ be the largest annotation occurring in $PS$, $I$ be the Herbrand interpretation that maps each atom to $\mu$, then $I$ does not satisfy $c$ iff the number of the monomials in $\Phi(c)$ is odd. We say $c$ is odd if this number is odd. Thus, $I$ divides $PS$ into two parts: $PS_1 = \{c \in PS \mid c$ is odd$\}$ and $PS - PS_1$. And if $PS$ is unsatisfiable, then $PS_1 \neq \emptyset$.

In the definition of superposants, if we require $c_2 \in PS_1$, and $p_2 \theta$ is maximal in $c_2 \theta$, then, combining with the computation of M-factors, we derive a strategy which is likely a positive resolution. However, so far we don't know whether this strategy is complete. The proof system presented in [Zh 94] can be viewed as a special case, its completeness demonstrates that the strategy is complete when the set of the annotations occurring in $PS$ contains only one element.

Dually, if $\mu$ is less than any annotations occurring in $PS$, then $I$ does not satisfy $c$ iff 1 appears in $\Phi(c)$. We call such $c$ tail. Let $PS_2 = \{c \in PS \mid c$ is tail$\}$. In the definition of superposants, we require $c_1 \in PS_1$, and $p_1 \theta$ is maximal in $c_1 \theta$. We thus have a strategy like negative resolution.

### 5.4 Parallel processing

We see that superposition has natural parallel sources. We only present here a rough description of parallelizing the proof procedure since this is not the core of this paper. It is also available for M-strategy.

The computations in our proof procedure are factoring and superposition. Like resolution, we can combine factoring into superposition. Of course, if
1 is derived by factoring, we stop because we have got what we want. So parallelizing superposition is the key step to parallelize the proof procedure.

We employ a message-passing model in a master/slaves paradigm. The master process coordinates the work of the slave processes. All the slave processes perform the same task, namely, computing superposants.

The work of the master process consists of maintaining the annotated polynomial set, sending necessary pairs of the annotated polynomials maintained to the slave processes, and receiving annotated polynomials from slave processes. When 1 is maintained in the master process, the work of the master process terminates. The slave processes perform the actual computation of the superposants of each pair of the annotated polynomials received from the master process, and send the superposants to the master process. It terminates when the master process has no more computation to farm out. Communication is always between a slave process and the master process in a star-like topology. Slave and master processes can communicate data via PVM[Wa 96].

Indeed, there are still a lot of work to do. For example, it’s a problem how to deal with the load balancing. In addition, If we use shared-memory models, basically, we can adopt the parallel algorithm presented in [SL 90].

6 Conclusion

In this paper, we dealt with the annotated logics, in which the annotation sets are totally ordered. We presented a proof procedure by introducing the concept of annotated polynomials, and developed several strategies. Besides implementing the proof procedure, the future work is, in one word, to increase the expressive power of annotated polynomials. This includes three useful extensions:

a) Allowing the annotations to be variables or functions of such annotation variables. We can thus at least save the space consuming in such a way. From the example in the fourth section we can see this point. \( path(x, y) : \mu_1, edge(y, z) : \mu_2 \leftarrow path(x, z) : min\{\mu_1, \mu_2\} \) stands for several formulae.

b) Allowing the formulae to be annotated. This is natural for the reasoning with uncertainty because probabilistic data may be available for compound events, but not for simple events individually.

c) Extending \( T \) to be a complete lattice. We may thus deal with the inconsistent information naturally. In this case, we can define annotated polynomials in the same way, \( (p : \mu_1)(p : \mu_2) = p : \cup\{\mu_1, \mu_2\} \), and the inference rules we developed in this paper are clearly sound. However it is not complete. We have to give an inference rule like cloning[LSC 91][Lu 92]. We guess the proof system may be complete if we introduce the computation of the production of two annotated polynomials to be an inference rule.
Acknowledgment

We would like to thank the anonymous referees for their constructive comments and suggestions.

References


