scription for a propositional logic or a first-order logic with domain closure axioms and unique name axioms. Our method can compute minimal models for this class of axioms not only with fixed predicates, but also with varied predicates. Moreover, (Nerode et al., 1995) gives a checking method of circumscriptive entailment for a limited class of formulas, whereas we give a complete checking method. Then, we extend our method to apply for prioritized circumscription as well.

(Cadoli et al., 1992) propose a method of eliminating varied predicates in circumscription by translating inference problem of a formula under circumscription with varied predicates and fixed predicates into another inference problem under circumscription without varied predicates nor fixed predicates. So, readers might think that methods of (Bell et al., 1992; 1996) which compute all the minimal models without varied nor fixed propositions are sufficient for computing minimal models. However, it is not clear how to apply the method proposed by (Cadoli et al., 1992) to computing minimal models since the relationship between a model of the original circumscription and a model of the translated circumscription is not known.

Preliminaries

We restrict our attention to propositional circumscription. For the first-order case with domain closure axioms and unique name axioms, we can translate each ground atom into a distinct proposition.

We assume that all propositional formulas are translated into a set of clauses of the form $L_1 \vee L_2 \vee ... \vee L_n$ where L_i is a positive literal p_i or a negative literal $\neg p_i$.

We associate each propositional symbol p with 0-1 variable X_p which represents the truth value of p; If $X_p = 1$, p is true and if $X_p = 0$, p is false. We also use an interpretation I to represent a solution of the assignments to variables from integer programming. If $p \in I$, it represents $X_p = 1$ and if $p \notin I$, it represents $X_p = 0$.

Let F and G be tuples of formulas, $\langle F_1, F_2, ..., F_n\rangle$ and $\langle G_1, G_2, ..., G_n\rangle$. We define $F \leq G$ as $\bigwedge_{i=1}^{n} F_i \supset G_i$. We define $F < G$ as $F \leq G$ and $G \not\leq F$, and $F \approx G$ as $F \leq G$ and $G \leq F$.

Let A be a conjunction of formulas and $\mathcal{P}$ be a set of propositional symbols used in A. We divide $\mathcal{P}$ into disjoint three tuples of propositions P, Z, Q which are called *minimized propositions*, *varied propositions*, and *fixed propositions*.

Circumscription of P for A with Z varied is defined as follows.

$$Circum(A; P; Z) = A(P, Z) \wedge \neg\exists p \exists z (A(p, z) \wedge p < P).$$

For a model theory of circumscription, we define an order of interpretations to minimize P with Z varied as follows. Let I be an interpretation and Φ be a tuple of propositional symbols. We define $I[\Phi]$ as $\{p \in \Phi | I \models p\}$ or, equivalently, $I \cap \Phi$.

Let I_1 and I_2 be interpretations.

$I_1 \leq^{P;Z} I_2$ if

1. $I_1[Q] = I_2[Q]$.
2. $I_1[P] \subseteq I_2[P]$.

We define $I_1 <^{P;Z} I_2$ as $I_1 \leq^{P;Z} I_2$ and $I_2 \not\leq^{P;Z} I_1$. A minimal model M of $A(P, Z)$ w.r.t. P with Z varied is defined as follows.

1. M is a model of $A(P, Z)$.
2. There is no model M' of $A(P, Z)$ such that $M' <^{P;Z} M$.

According to (Lifschitz, 1985), I is a minimal model of $A(P, Z)$ w.r.t. P with Z varied if and only if I is a model of $Circum(A; P; Z)$.

Step 1: Let $AC := \emptyset$ and $SS := \emptyset$.

Step 2: Minimizing $\sum_{p \in P} X_p$ under $Tr(A) \cup AC$ using 0-1 integer programming.

Step 3: If there is no solution for the above minimization, output SS

Step 4: Otherwise,

1. Let M be a solution of the above minimization.
2. Add $M[P]$ to SS.
3. Add $\sum_{p \in M[P]} X_p \leq |M[P]| - 1$ to AC.
4. Go to Step 2.

Figure 1: The algorithm of Nerode et al.

Computing Minimal Models without Varied Propositions

Let A be a set of clauses. Then, a set of inequalities, $Tr(A)$, translated from A is defined as follows.

$$Tr(A) = \{X_{p_1} + ... + X_{p_n} + (1 - X_{q_1}) + ... + (1 - X_{q_m}) \geq 1 | p_1 \vee ... \vee p_n \vee \neg q_1 \vee ... \vee \neg q_n \in A\}$$

Let Z be empty. Then, the algorithm proposed in (Nerode et al., 1995) is in Figure 1. We adapt their algorithm for propositional circumscription. Note that the algorithm is an extended version of the algorithm of (Bell et al., 1992; 1996) to minimize all propositions. The algorithm of (Bell et al., 1992; 1996) is a special case of the algorithm of (Nerode et al., 1995) where a set of fixed propositions, Q, is empty.

The algorithm works as follows. We start with $Tr(A)$ as the initial constraints and minimize an objective function corresponding with minimized propositions under $Tr(A)$. If we do not obtain any solution, we are done. Otherwise, we add a constraint AC which excludes non-minimal models larger than the obtained solution.

(Nerode et al., 1995) claims the following on the correctness and completeness of the above algorithm.

Claim(Nerode et al., 1995, Theorem 1) *Output SS from the algorithm in Figure 1 is equivalent to*

$\{M[P]|M$ *is a minimal model of* $A(P)$ *with respect to* P *with no propositions varied* $\}$.

Unfortunately, this claim is not correct in general as the following example shows.

Example 1 *Let* $A(ab)$ *be the following set of clauses.*

$\neg bird \vee ab \vee fly$.
$bird$.

Then, the minimal models of $A(ab)$ *with respect to* $\langle ab\rangle$ *are* $M_1 = \{bird, fly\}$ *and* $M_2 = \{bird, ab\}$. *Note that fly is a fixed proposition and so, the two models are incomparable since interpretations of fly are different in these two models.*

However, from the algorithm in Figure 1, we cannot obtain M_2 *as follows.*

$Tr(A)$ *is*
$1 - X_{bird} + X_{ab} + X_{fly} \geq 1$
$X_{bird} \geq 1$
By minimizing X_{ab} *using 0-1 integer programming under* $Tr(A)$, *we obtain a solution* $X_{ab} = 0, X_{bird} = 1, X_{fly} = 1$ *which corresponds with* M_1.

Then, we add $M_1[\langle ab\rangle] = \emptyset$ *to SS. Since* $M_1[\langle ab\rangle] = \emptyset$, $\sum_{p\in M[\langle ab\rangle]} X_p = 0$ *and* $|M_1[\langle ab\rangle]| = 0$. *Thus, we add the following constraint to AC.*

$0 \leq -1$.

Obviously, we cannot get any further solution. This means that we cannot obtain a minimal model M_2.

Therefore, the above claim does not work in general if there is a fixed proposition. Although their method is not correct with circumscription with fixed propositions, we later show that their method actually works for circumscription with varied propositions without fixed propositions.

Now, we give an algorithm which works correctly for circumscription with fixed propositions in Figure 2. Let I be an interpretation and Φ be a tuple of propositional symbols. We define $\overline{I}[\Phi]$ used in Figure 2 as $\{p \in \Phi | I \not\models p\}$ or equivalently, $\Phi - I$.

Theorem 1 *Output SS from the algorithm in Figure 2 is equivalent to*

$\{M|M$ *is a minimal model of* $A(P)$ *with respect to* P *with no propositions varied* $\}$.

Proof: Let α be a formula which consists of logical connectives and propositional symbols in P. Then, according to (de Kleer and Konolige, 1989), $Circum(A; P) \models \alpha$ if and only if $Circum(A \wedge (R \equiv \neg \cdot Q); P, Q, R) \models \alpha$ where R is a tuple of new propositions not in A and $R \equiv \neg \cdot Q$ is $\bigwedge_{i=1}^{m} r_i \equiv \neg q_i$ for $R = \langle r_1, ..., r_m\rangle$ and $Q = \langle q_1, ..., q_m\rangle$. Then, we use the algorithm of (Bell et al., 1992) to minimize all propositions and replace every occurrence of variables X_{r_i} for a proposition r_i in R by $1 - X_{q_i}$. □

Example 2 *Let* $A(ab)$ *be the following set of clauses as in Example 1*

Step 1: Let $AC := \emptyset$ and $SS := \emptyset$.

Step 2: Minimizing $\sum_{p\in P} X_p$ under $Tr(A) \cup AC$ using 0-1 integer programming.

Step 3: If there is no solution for the above minimization, output SS.

Step 4: Otherwise,

1. Let M be a solution of the above minimization.
2. Add M to SS.
3. Add $\sum_{p\in M[P]} X_p + \sum_{q\in M[Q]} X_q + \sum_{q'\in \overline{M}[Q]} (1 - X_{q'}) \leq |M[P]| + |Q| - 1$ to AC.
4. Go to Step 2.

Figure 2: Algorithm for circumscription with fixed propositions

$\neg bird \vee ab \vee fly$.
$bird$.

Then, the minimal models of $A(ab)$ *with respect to* $\langle ab\rangle$ *are* $M_1 = \{bird, fly\}$ *and* $M_2 = \{bird, ab\}$.

By minimizing X_{ab} *under* $Tr(A)$, *we obtain a solution* $X_{ab} = 0, X_{bird} = 1, X_{fly} = 1$ *which corresponds with a minimal model* M_1.

Then, we add M_1 *to SS and we add the following constraint to AC.*

$X_{bird} + X_{fly} \leq 1$.

Then, minimizing X_{ab} *under* $Tr(A) \cup AC$, *we obtain a solution* $X_{ab} = 1, X_{bird} = 1, X_{fly} = 0$ *which corresponds with a minimal model* M_2.

Then, we add M_2 *to SS and we add the following constraint to AC.*

$X_{ab} + X_{bird} + (1 - X_{fly}) \leq 2$

Then, minimizing X_{ab} *under* $Tr(A) \cup AC$, *we no longer obtain any solution and therefore,* $SS = \{\{bird, fly\}, \{bird, ab\}\}$ *is obtained.*

Computing Minimal Models with Varied Propositions

As shown in Introduction, we need varied proposition to perform commonsense reasoning. We give a computation method of handling varied propositions in Figure 3.

Let F, G be disjoint sets of propositions. We define $\mathcal{F}(F, G)$ as $\bigwedge_{f\in F} f \wedge \bigwedge_{f\in G} \neg f$.

Theorem 2 *Output MS from the algorithm in Figure 3 is equivalent to*

$\{M|M$ *is a minimal model of* $A(P, Z)$ *with respect to* P *with* Z *varied* $\}$.

Proof: At Step 4, every $M \in SS$ is a minimal model but there might be alternative models such that the

Step 1: Let $AC := \emptyset$ and $SS := \emptyset$.

Step 2: Minimizing $\sum_{p \in P} X_p$ under $Tr(A) \cup AC$ using 0-1 integer programming.

Step 3: If there is no solution for the above minimization, go to Step 5.

Step 4: Otherwise,

1. Let M be a solution of the above minimization.
2. Add $M[P \cup Q]$ to SS
3. Add $\sum_{p \in M[P]} X_p + \sum_{q \in M[Q]} X_q + \sum_{q' \in \overline{M}[Q]} (1 - X_{q'}) \leq |M[P]| + |Q| - 1$ to AC.
4. Go to Step 2.

Step 5: Let MS be $\emptyset$ and for every $S \in SS$ do the following.

1. $A' := A \wedge \mathcal{F}(S, (P \cup Q) - S)$.
2. Compute all the models of A' and add these models to MS.

Output MS.

Figure 3: Algorithm for circumscription with varied propositions

interpretations of $P \cup Q$ are the same but the interpretations of Z are different. At Step 5, we compute such alternative models. □

Example 3 *Let $A(ab)$ be the following set of clauses.*

$\neg bird \vee ab \vee fly.$
$bird.$

Then, the minimal model of $A(ab)$ with respect to $\langle ab \rangle$ with $\langle fly \rangle$ varied is $M_1 = \{bird, fly\}$.

By minimizing X_{ab} under $Tr(A)$, we obtain a solution where $X_{ab} = 0, X_{bird} = 1$ and $X_{fly} = 1$. We add $M_1[\langle ab \rangle \cup \langle bird \rangle] = \{bird\}$ to SS and the following constraint to AC.

$X_{bird} \leq 0.$

Obviously, there is no solution for $Tr(A) \cup AC$ and therefore, $SS = \{\{bird\}\}$ is obtained.

Then, we add $\mathcal{F}(\{bird\}, \{ab, bird\} - \{bird\}) = bird \wedge \neg ab$ to A to obtain A' and compute all the models of A'. We obtain $MS = \{\{bird, fly\}\}$.

Actually, in the algorithm in Figure 3, if Q is empty and we output SS at Step 3 instead of going to Step 5, then this is equivalent to the algorithm of Nerode et al. In other words, the correct claim for (Nerode et al., 1995) is as follows.

Corollary 1 *Let $\mathcal{P}$ be $P \cup Z$ and Q be empty. Final SS in the algorithm in Figure 3 is equivalent to*

$\{M[P] | M$ is a minimal model of $A(P, Z)$ with respect to P with Z varied $\}$.

If we only concern about circumscriptive entailment discussed in (Nerode et al., 1995), that is, whether $Circum(A; P; Z) \models \alpha$ or not, we do not need Step 5. Instead, we check whether $A \wedge \mathcal{F}(S, (P \cup Q) - S) \wedge \neg \alpha$ for every $S \in SS$ has any models or not. This can be done by checking whether $Tr(A \wedge \mathcal{F}(S, (P \cup Q) - S) \wedge \neg \alpha)$ does not have any solution when minimizing any arbitrary objective function. Note that in (Nerode et al., 1995), they use "upper and lower fringes" to compute circumscriptive entailment for a restricted class of formulas, but actually, such "fringes" are not necessary.

Computing Minimal Models in Prioritized Circumscription

We firstly give a definition of prioritized circumscription. We divide a set of propositions into n partitions and give an order over partitions. Suppose that this is $P_1 > P_2 > ... > P_n$. Intended meaning of this order is that we firstly minimize P_1, then P_2 .... ,then P_n. Let P and Q be a tuple of propositions which have orders $P_1 > P_2 > ... > P_n$ and $Q_1 > Q_2 > ... > Q_n$. We define $P \preceq^i Q$ as follows. If $i = 1$, $P \preceq^i Q$ is $P_1 \leq Q_1$ and if $i > 1$, $(\bigwedge_{j=1}^{i-1} P_j \approx Q_j) \supset P_i \leq Q_i$. We define $P \preceq Q$ as $\bigwedge_{i=1}^{n} P \preceq^i Q$ and $P \prec Q$ as $P \preceq Q$ and $Q \not\preceq P$.

Prioritized circumscription of $P_1 > P_2 > ... > P_n$ for A with Z varied is defined as follows.

$$Circum(A; P_1 > P_2 > ... > P_n; Z) = A(P, Z) \wedge \neg \exists p \exists z (A(p, z) \wedge p \prec P).$$

In a model theory of prioritized circumscription, we define an order over interpretations as follows.

Let I_1 and I_2 be interpretations and let $\mathcal{P}$ consist of disjoint sets $P_1, P_2, ..., P_n, Q, Z$.

$I_1 \preceq^{P_1 > P_2 > ... > P_n; Z} I_2$ if

1. $I_1[Q] = I_2[Q]$.
2. $I_1[P_1] \subseteq I_2[P_1]$.
3. For every i, if for every $1 \leq j \leq i-1$, $I_1[P_j] = I_2[P_j]$, then $I_1[P_i] \subseteq I_2[P_i]$.

We define $I_1 \prec^{P_1 > P_2 > ... > P_n; Z} I_2$ as $I_1 \preceq^{P_1 > P_2 > ... > P_n; Z} I_2$ and $I_2 \not\preceq^{P_1 > P_2 > ... > P_n; Z} I_1$.

A minimal model M of $A(P, Z)$ w.r.t. $P_1 > P_2 > ... > P_n$ with Z varied is defined as follows.

1. M is a model of $A(P, Z)$.
2. There is no model M' of $A(P, Z)$ such that $M' \prec^{P_1 > P_2 > ... > P_n; Z} M$.

According to (Lifschitz, 1985), I is a minimal model of $A(P, Z)$ w.r.t. $P_1 > P_2 > ... > P_n$ with Z varied iff I is a model of $Circum(A; P_1 > P_2 > ... > P_n; Z)$.

Similar to the problem in non-prioritized circumscription, the method proposed in (Nerode et al., 1995) of computing prioritized circumscription is correct if there are no fixed propositions.

To manipulate fixed propositions in prioritized circumscription, we need the following theorem which is a generalization of the result of (de Kleer and Konolige, 1989).

Theorem 3 *Let a set of propositions $\mathcal{P}$ consist of disjoint sets $P_1, P_2, ..., P_n, Q, Z$ and $P = P_1 \cup P_2 \cup ... \cup P_n$ and α be a formula which consists of logical connectives and propositional symbols in P. Then,*

$Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z) \models \alpha$ if and only if $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P_1 > P_2 > ... > P_n; Z) \models \alpha$.

Proof: See appendix.

This theorem means that we can translate prioritized circumscription with fixed propositions to prioritized circumscription without fixed propositions. Moreover, we extend the method so that it is applicable even if there are varied propositions. We show the algorithm in Figure 4.

Theorem 4 *Output MS from the algorithm in Figure 4 is equivalent to*

$\{M | M$ is a minimal model of $A(P,Z)$ with respect to $P_1 > ... > P_n$ with Z varied$\}$.

Proof: According to (Lifschitz, 1985),

$Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z)$ is equivalent to

$Circum(A(P,Z); P_1; P_2, ..., P_n, Z) \wedge$
$Circum(A(P,Z); P_2; P_3, ..., P_n, Z) \wedge ...$
$Circum(A(P,Z); P_n; Z)$.

At **Step 1**,...,**Step 4**, we compute interpretation of P_1 and Q for every model of

$Circum(A(P,Z); P_1; P_2, ..., P_n, Z)$.

At the iteration where $i = 2...n$ in **Step 5**, we compute interpretation of $P_1, ..., P_i$ and Q for every model of

$Circum(A(P,Z); P_1; P_2, ..., P_n, Z) \wedge$
$Circum(A(P,Z); P_2; P_3, ..., P_n, Z) \wedge ...$
$Circum(A(P,Z); P_i; P_{i+1}, ..., P_n, Z)$

since we reflect interpretation of propositions $P_1 \cup ... \cup P_{i-1} \cup Q$ which are obtained up to the latest iteration. Therefore, at n-th iteration, we obtain interpretation of $P_1, ..., P_n$ for every model of $Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z)$. □

Example 4 *Consider the following axioms.*

$ab_1 \vee \neg fly$
$\neg bird \vee ab_2 \vee fly$.

We compute minimal models of $Circum(A; \langle ab_2 \rangle > \langle ab_1 \rangle; \langle fly \rangle)$ meaning that we minimize ab_1 and ab_2 with fly varied (and $bird$ fixed) and ab_2 is preferably minimized than ab_1. The minimal models are $\{bird, fly, ab_1\}$ and $\emptyset$. We have two minimal models since the interpretations of bird in these models are different from each other.

Step 1: *$AC := \emptyset$ and $SS := \emptyset$.*

Step 2(1): *Minimize X_{ab_2} under the following constraints:*

$X_{ab_1} + 1 - X_{fly} \geq 1$
$1 - X_{bird} + X_{ab_2} + X_{fly} \geq 1$

Step 3(1): *Then, there are three solutions for this minimization:*

$S_1 = \{X_{ab_2} = 0, X_{bird} = 1, X_{fly} = 1, X_{ab_1} = 1\}$,

---

Step 1: Let $AC := \emptyset$ and $SS := \emptyset$.

Step 2: Minimizing $\sum_{p \in P_1} X_p$ under $Tr(A) \cup AC$ using 0-1 integer programming.

Step 3: If there is no solution for the above minimization, go to Step 5.

Step 4: Otherwise,

1. Let M be a solution of the above minimization.
2. Add $M[P_1 \cup Q]$ to SS
3. Add $\sum_{p \in M[P_1]} X_p + \sum_{q \in M[Q]} X_q + \sum_{q' \in \overline{M}[Q]} (1 - X_{q'}) \leq |M[P_1]| + |Q| - 1$ to AC.
4. Go to Step 2.

Step 5: For $i := 2$ to n do the following.

1. $SS' := \emptyset$.
2. For every $S \in SS$ do

   Step 5-1: Let $AC := \emptyset$.

   Step 5-2: Minimizing $\sum_{p \in P_i} X_p$ under $Tr(A \wedge \mathcal{F}(S, (P_1 \cup ... \cup P_{i-1} \cup Q) - S)) \cup AC$ using 0-1 integer programming.

   Step 5-3: If there is no solution for the above minimization, process the next S.

   Step 5-4: Otherwise,

   (a) Let M be a solution of the above minimization.
   (b) Add $M[P_1 \cup ... \cup P_i \cup Q]$ to SS'.
   (c) Add $\sum_{p \in M[P_i]} X_p \leq |M[P_i]| - 1$ to AC.
   (d) Go to Step 5-2.
3. $SS := SS'$ and do the next iteration for i.

If iteration stops then let MS be $\emptyset$ and for every $S \in SS$ do the following.

1. Let $A' := A \wedge \mathcal{F}(S, (P_1 \cup ... \cup P_n \cup Q) - S)$.
2. Compute all the models of A' and add these models to MS.

Output MS.

Figure 4: Algorithm for prioritized circumscription

$S_2 = \{X_{ab_2} = 0, X_{bird} = 0, X_{fly} = 1, X_{ab_1} = 1\}$,
$S_3 = \{X_{ab_2} = 0, X_{bird} = 0, X_{fly} = 0, X_{ab_1} = 0\}$.

Step 4(1): *Suppose that we obtain* S_1.

1. $M_1 = \{bird, fly, ab_1\}$
2. *Add* $M_1[\langle ab_2\rangle \cup \langle bird\rangle] = \{bird\}$ *to* SS. *(*SS *becomes* $\{\{bird\}\}$*.)*
3. *Add* $X_{bird} \leq 0$ *to* AC.

Step 2(2): *Minimize* X_{ab_2} *under new* AC.

Step 3(2): *Then, there are two solutions for this minimization,* S_2 *and* S_3.

Step 4(2): *Suppose that we obtain* S_2.

1. $M_2 = \{fly, ab_1\}$.
2. *Add* $M_2[\langle ab_2\rangle \cup \langle bird\rangle] = \emptyset$ *to* SS. *(*SS *becomes* $\{\{bird\}, \emptyset\}$*.)*
3. *Add* $1 - X_{bird} \leq 0$ *to* AC.

Step 2(3): *Minimize* X_{ab_2} *under new* AC.

Step 3(3): *Then, we no longer obtain any solutions, and go to Step 5.*

Step 5:
$i := 2$ *and* $SS' := \emptyset$.

1. $S := \{bird\}$.

Step 5-1(1): $AC := \emptyset$.

Step 5-2(1): *Minimize* X_{ab_1} *under the following constraints:*

$X_{ab_1} + 1 - X_{fly} \geq 1$
$1 - X_{bird} + X_{ab_2} + X_{fly} \geq 1$
$X_{ab_2} \leq 0$
$X_{bird} \geq 1$

Step 5-3(1): *Then, we obtain the solution* S_1 *again.*

Step 5-4(1):

(a) $M_1 = \{bird, fly, ab_1\}$
(b) *Add* $M_1[\langle ab_2\rangle \cup \langle ab_1\rangle \cup \langle bird\rangle] = \{ab_1, bird\}$ *to* SS'. *(*SS' *becomes* $\{\{ab_1, bird\}\}$*.)*
(c) *Add* $X_{ab_1} \leq 0$ *to* AC.

Step 5-2(2): *Minimize* X_{ab_1} *under new* AC.

Step 5-3(2): *Then, we no longer obtain any solutions.*

2. $S := \emptyset$.

Step 5-1(1): $AC := \emptyset$.

Step 5-2(1): *Minimize* X_{ab_1} *under the following constraints:*

$X_{ab_1} + 1 - X_{fly} \geq 1$
$1 - X_{bird} + X_{ab_2} + X_{fly} \geq 1$
$X_{ab_2} \leq 0$
$X_{bird} \leq 0$

Step 5-3(1): *Then, we obtain the solution* S_3 *only.*

Step 5-4(1):

(a) $M_3 = \emptyset$
(b) *Add* $M_3[\langle ab_2\rangle \cup \langle ab_1\rangle \cup \langle bird\rangle] = \emptyset$ *to* SS'. *(*SS' *becomes* $\{\{ab_1, bird\}, \emptyset\}$*.)*
(c) *Add* $0 \leq -1$ *to* AC.

Step 5-2(2): *Minimize* X_{ab_1} *under new* AC.

Step 5-3(2): *Then, we no longer obtain any solutions.*

Iteration stops and by calculation of MS *from* SS', *we obtain* $\{\{bird, fly, ab_1\}, \emptyset\}$.

We can also give a method of circumscriptive entailment in prioritized circumscription as in ordinary circumscription. After iteration stops, we check for every $A' \in SS$, $A' \wedge \neg\alpha$ does not have any models to check whether α is consequence of the prioritized circumscription or not.

Conclusion

Contributions of this paper are as follows.

1. We correctly give the method of computing all the models of circumscription not only with fixed propositions, but also with varied propositions.
2. We give a complete method of computing circumscriptive entailment for propositional logic.
3. We also extend the method of computing minimal models to include varied propositions in prioritized circumscription.

Acknowledgements This research is partly supported by Grant-in-Aid for Scientific Research (Project No. 11878067), The Ministry of Education, Japan. We also thank the anonymous referees for comments on this paper.

References

Bell, C., Nerode, A., Ng, R., and Subrahmanian, V. S. 1992. Implementing Deductive Database by Linear Programming. *Proc. of PODS92*, pages 283 – 292.

Bell, C., Nerode, A., Ng, R., and Subrahmanian, V. S. 1996. Implementing Deductive Database by Mixed Integer Programming. *ACM Transactions on Database Systems*, **21**, pages 238 – 269.

Cadoli, M., Eiter, T., and Gottlob, G. 1992 An Efficient Method for Eliminating Varying Predicates from a Circumscription. *Artificial Intelligence*, **54**, pages 397 – 410.

de Kleer, J., and Konolige, K. 1989. Eliminating the Fixed Predicates from a Circumscription. *Artificial Intelligence*, **39**, pages 391 – 398.

Etherington, D., Mercer, R., and Reiter, R. 1985. On the Adequacy of Predicate Circumscription for Closed-world Reasoning. *Computational Intelligence*, **1**, pages 11 – 15.

Ginsberg, M. 1989. Circumscriptive Theorem Prover. *Artificial Intelligence*, **39**, pages 209 – 230.

Lifschitz, V. 1985. Computing Circumscription. *Proc. of IJCAI-85*, pages 121 – 127.

McCarthy, J. 1986. Applications of Circumscription to Formalizing Common-Sense Knowledge. *Artificial Intelligence*, **28**, pages 89 – 116.

Nerode, A., Ng, R. T., and Subrahmanian, V. S. 1995. Computing Circumscriptive Databases, I. Theory and Algorithms. *Information and Computation*, **116**, pages 58–80.

Przymusinski, T. 1989. An algorithm to compute circumscription. *Artificial Intelligence*, **38**, pages 49 – 73.

Appendix (Proof of Theorem 3)

Lemma 1 *Let a set of propositions $\mathcal{P}$ consist of disjoint sets P, Q, Z. $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P; Z)$ is equivalent to $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P; Z)$.*

Proof: Suppose that M is a model of $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P; Z)$, but M is not a model of $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P; Z)$. Then, there is a model M' s.t.

- The interpretations of R and Q are same in M and M'.
- $M' <^{P;Z} M$.

Then, this means that $M' <^{P,Q,R;Z} M$ and it contradicts the assumption that M is a model of $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P; Z)$.

On the other hand, Suppose that M is a model of $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P; Z)$, but M is not a model of $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P; Z)$. Then, there is a model M' s.t. $M' <^{P,Q,R;Z} M$. If the interpretations of Q and R were same in M and M', it would mean that $M' <^{P;Z} M$. Therefore, there must be some difference in the interpretations of Q and R between M and M'. Let s be proposition in Q or R s.t. the interpretation of s is different in M and M'. Suppose that s is in Q. Then, there is some proposition t in R s.t. $s \equiv \neg t$. This means that $M' \not< ^{P,Q,R;Z} M$. The same argument applies if s is in R. Therefore, it leads to contradiction. □

Lemma 2 *Let a set of propositions $\mathcal{P}$ consist of disjoint sets $P_1, P_2, ..., P_n, Q, Z$ and $P = P_1 \cup P_2 \cup ... \cup P_n$.*

$Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P_1 > P_2 > ... > P_n; Z)$

is equivalent to

$Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P_1 > P_2 > ... > P_n; Z)$.

Proof: According to (Lifschitz, 1985),

$Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P_1 > P_2 > ... > P_n; Z)$ is equivalent to

$Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P_1; P_2, ..., P_n, Z)$
$\wedge Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P_2 > ... > P_n; Z)$.

Then, use Lemma 1. □

We also need the following lemma.

Lemma 3 *Let a set of propositions $\mathcal{P}$ consist of disjoint sets $P_1, P_2, ..., P_n, Q, Z$ and $P = P_1 \cup P_2 \cup ... \cup P_n$.*

$Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P_1 > P_2 > ... > P_n; Z)$

is equivalent to

$Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z) \wedge (R \equiv \neg \cdot Q)$.

Proof: Suppose that M is a model of $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P_1 > P_2 > ... > P_n; Z)$, but M is not a model of $Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z) \wedge (R \equiv \neg \cdot Q)$. M is a model of $(R \equiv \neg \cdot Q)$ and therefore, M must not be a model of $Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z)$. Then, there is a model M' of $A(P,Z)$ s.t. $M' \prec^{P_1 > P_2 > ... > P_n; Z} M$. Since R and Q are fixed propositions, $M' \models (R \equiv \neg \cdot Q)$ since $M \models (R \equiv \neg \cdot Q)$. This contradicts the assumption that M is a minimal model of $A(P,Z) \wedge (R \equiv \neg \cdot Q)$ w.r.t. $P_1 > P_2 > ... > P_n$ with Z varied.

On the other hand, suppose that M is a model of $Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z) \wedge (R \equiv \neg \cdot Q)$, but M is not a model of $Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P_1 > P_2 > ... > P_n; Z)$. Since M is a model of $A(P,Z) \wedge (R \equiv \neg \cdot Q)$, there is a model M' of $A(P,Z) \wedge (R \equiv \neg \cdot Q)$ s.t. $M' \prec^{P_1 > P_2 > ... > P_n; Z} M$. However, this contradicts with the assumption that M is a minimal model of $A(P,Z)$ w.r.t. $P_1 > P_2 > ... > P_n$ with Z varied since the interpretations of R and Q are same in M and M'. □

Proof of Theorem 3 By Lemma 2,

$Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); Q, R, P_1 > P_2 > ... > P_n; Z)$

is equivalent to

$Circum(A(P,Z) \wedge (R \equiv \neg \cdot Q); P_1 > P_2 > ... > P_n; Z)$.

Then, by Lemma 3, the above is equivalent to

$Circum(A(P,Z); P_1 > P_2 > ... > P_n; Z) \wedge (R \equiv \neg \cdot Q)$. □