TM-0963

Set CAL- a solver of set constraint in CAL system

by
Y. Sato, K. Sakai & S. Menju

October, 1990

ICOT
Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan
(03)3456-3191 ~ 5
Telex ICOT J32964

Institute for New Generation Computer Technology

SetCAL - a solver of set constraint in CAL system

Yosuke Sato, Kô Sakai, and Satoshi Menju

ICOT Research Center
1-4-28,Mita,Minano-ku,Tokyo 108,JAPAN

ABSTRACT

We give a short introduction to SetCAL that is a solver of constraint about sets in CAL system being developed in ICOT.

1. Introduction

In constraint logic programming, there are often applications in which we want to write membership or inclusion of sets such as $\in$ and $\subseteq$. SetCAL is a solver in CAL system, developed in order to deal with such constraint. A set which computer can handle naively is finite or co-finite(i.e. complement of a finite set). It is important that a class of finite and co-finite subsets of a fixed ground set forms a Boolean ring. Any constraint which we want to write can be expressed in terms of a polynomial ring over this Boolean ring which is called a Boolean polynomial ring. For example $a \in X, b \notin Y, X \subseteq Y$ are expressed as $\{a\}X = \{a\}$, $\{b\}Y = 0$, $XY = X$. We first show how constraint written in the language of SetCAL is expressed in terms of a Boolean polynomial ring. Then give some important theories of Boolean polynomial ring such as Boolean Gröbner base, and show how they are applied to give a complete decision procedure for solving constraint of SetCAL.

2. Language of SetCAL

$a, b, c, \ldots$: first-order constant symbols for elements of sets

$x, y, z, \ldots$: first-order variables for elements of sets

$X, Y, Z, \ldots$: second-order variables for finite sets or co-finite sets

$\in, \subseteq, \{\cdot\}, \cap, \cup, \cdot^c, \ldots$: predicate and function symbols for sets

$=$: equality

$\bigvee, \bigwedge, \neg, \rightarrow, \ldots$: logical symbols

$\forall_0, \exists_0$: first-order quantifiers for elements of sets

$\forall_1, \exists_1$: second-order quantifiers for finite or co-finite sets

3. Expression of set constraint by Boolean polynomial ring

For a Boolean algebra $< \mathbf{B}, \vee, \wedge, \neg, 0, 1 >$, define

$x + y =_{def} (x \wedge \neg y) \vee (\neg x \wedge y), \quad x \cdot y =_{def} x \wedge y,$

then $< \mathbf{B}, +, \cdot, 0, 1 >$ becomes a commutative ring with unit. This ring has the following two properties.

(i) $\forall x \in \mathbf{B} \quad x^2 = x$

(ii) $\forall x \in \mathbf{B} \quad x + x = 0$

Conversely, for a commutative ring with unit if we define $\vee, \wedge, \neg$ by

$x \vee y =_{def} x + y + x \cdot y, \quad x \wedge y =_{def} x \cdot y, \quad \neg x =_{def} 1 + x,$

it becomes a Boolean algebra.

Therefore we can treat a commutative ring with unit which has the above two properties as a Boolean algebra. We call such a ring a **Boolean ring**. For a Boolean ring $\mathbf{B}$, a polynomial f of a polynomial ring $\mathbf{B}[X_1, X_2, \ldots, X_n]$ is called a **Boolean polynomial** if the degree of each variable of f is at most 1. Using a rule $X^2 = X$ for each variable, a set of all Boolean polynomials forms a Boolean ring. This ring is called a **Boolean polynomial ring** and denoted by $\mathbf{B}(X_1, X_2, \ldots, X_n)$. In other words a Boolean polynomial ring $\mathbf{B}(X_1, X_2, \ldots, X_n)$ is a quotient ring $\mathbf{B}[X_1, X_2, \ldots, X_n]/I$, where I is an ideal generated by $\{X_1^2 + X_1, X_2^2 + X_2, \ldots, X_n^2 + X_n\}$. A Boolean polynomial is considered as a representative of an equivalent class.

Let U be the set of all constant symbols defined in section 2. The set of all finite or co-finite subsets of U denoted by $P^{FC}(U)$ forms a Boolean algebra $< P^{FC}(U), \vee, \wedge, \neg, 0, 1 >$ taking a set union $\vee$ for $\vee$, a set intersection $\wedge$ for $\wedge$,a complement operation $\cdot^c$ for $\neg$, the empty set $\emptyset$ as 0, and U as 1. Any constraint described by the language defined in section 2, can be expressed in terms of equations of $P^{FC}(U)(X, Y, Z, \ldots)$ using $\bigvee, \bigwedge, \neg, \rightarrow, \ldots, \forall_0, \exists_0, \forall_1, \exists_1$.

Example 3.1
$a \in X \cap (Y \cup Z) \Leftrightarrow a \in X(YZ + Y + Z) \Leftrightarrow a \in XYZ + XY + XZ \Leftrightarrow \{a\} \cap (XYZ + XY + XZ) = \{a\} \Leftrightarrow \{a\}XYZ + \{a\}XY + \{a\}XZ = \{a\}$ (We abbreviate $\cdot$)

4. Solution of Boolean equations

We give an outline of the method to solve equations of general Boolean polynomial ring by using a Gröbner base technique. SetCAL solves constraint given by the language of section 2 based on this method. The detail is given in [1],[2].

Definition 4.1
A monomial in a Boolean polynomial, i.e. a monomial the degree of each variable of which is at most 1, is called a **Boolean monomial** and denoted by meta symbols $\alpha, \beta, \gamma, \ldots$.

Definition 4.2
An ordering $\geq$ over Boolean monomials is called **admissible** if it satisfies the following properties.

(i) If $\alpha \supset \beta$ considering monomials as sets of variables, then $\alpha \geq \beta$.

(ii) If $\alpha \geq \beta$, then $\alpha\gamma \geq \beta\gamma$ for any Boolean monomial γ. such that γ does not include a common variable with α or β.

From now on, we fix an admissible total ordering $\geq$ over Boolean monomials.

Definition 4.3
$a\alpha \oplus \phi$ is an expression of a Boolean polynomial with the greatest Boolean monomial α. For each Boolean polynomial $a\alpha \oplus \phi$, we define a rewriting rule $\Rightarrow_{a\alpha\oplus\phi}$ over Boolean polynomials as follows.

For a Boolean polynomial $\varphi = \psi + ba\beta$, if $ab \neq 0$, then $\varphi \Rightarrow_{a\alpha\oplus\phi} \varphi'$, where φ' is a Boolean polynomial given by $\psi + b(1+a)\alpha\beta + ab\beta\phi$.

The soundness of this rewriting rule is explained as follows. Firstly, note that $ba\beta = b(1+a)\alpha\beta + ba\alpha\beta$. Secondly, since $a\alpha \oplus \phi = 0$ implies $a\alpha = \phi$, multiplying $ab\beta$ from both sides, we have $ba\alpha\beta = ba\beta\phi$. Therefore under the condition $a\alpha \oplus \phi = 0$, we have $ba\beta = b(1+a)\alpha\beta + ab\beta\phi$.

Example 4.4
Let c, d be elements of $\mathbf{B}$ such that $cd = 0, c \neq 0, d \neq 0$. If $(1+c)X = 0$, then $dX = d(1+c)X = 0$. However since $\mathbf{B}$ is not a field, we can not rewrite dX to 0 by a standard rewrite rule by a substitution. Under the assumption $cd = 0$, $(1+c)d = d \neq 0$. Hence, we can apply our rewriting rule to get $dX \Rightarrow_{(1+c)X} 0$.

Definition 4.5

Let R be a set of Boolean polynomials. If there exists a Boolean polynomial φ such that $\phi \Rightarrow_{\varphi} \psi$, we write $\phi \Rightarrow_R \psi$. The transitive reflexive closure of $\Rightarrow_R$ is denoted by $\overset{*}{\Rightarrow}_R$.

Theorem 4.6

For each set of Boolean polynomials R, $\Rightarrow_R$ has a termination property. ∎

Definition 4.7 Boolean Gröbner base

Let I be a finitely generated ideal of a Boolean polynomial ring $\mathbf{B}(X_1, \ldots, X_n)$. A finite set G of Boolean polynomials is called a Boolean Gröbner base of I, if it satisfies the following properties.

(i) $G \subseteq I$

(ii) If $f \equiv g (mod I)$(i.e. $f + g \in I$), there exists a Boolean polynomial h such that $f \overset{*}{\Rightarrow}_G h$ and $g \overset{*}{\Rightarrow}_G h$.

(iii) Each $g \in G$ can not be rewritten by $\Rightarrow_{g'}$ for any $g' \in G$ which is distinct from g.

(iv) The greatest monomial of a Boolean polynomial of G, is distinct each other.

4.8 Some properties of Boolean Gröbner base

(i) The ideal generated by G is I. Hence, I and G forms the same constraint by Zero-point theorem [3].

(ii) The constraint given by I, i.e. a set of equations $\{f = 0 | f \in I\}$ is unsolvable if and only if the Gröbner base of I includes a non-zero constant, i.e a non-zero element of $\mathbf{B}$.

(ii) The Gröbner base G of I is unique. Hence, we can consider G as a canonical form of the constraint given by I.

We give some definitions needed to describe an algorithm to get a Boolean Gröbner base.

Definition 4.9

For a Boolean polynomial $a\alpha \oplus \phi$, a Boolean polynomial $a\phi + \phi$ is called its **coefficient self-critical pair** and denoted by $csc(a\alpha \oplus \phi)$. For each variable X in α, a Boolean polynomial $X\phi + \phi$ is called its **variable self-critical pair**.

Example 4.10

A coefficient self-critical pair of $aXYZ \oplus bYW$ is $(ab + b)YW$. There are two variable self-critical pairs, namely $bXYW + bXW$ and $bYZW + bYZ$.

Definition 4.11

For Boolean polynomials $a\alpha\gamma \oplus \phi$ and $b\beta\gamma \oplus \psi$ such that $ab \neq 0, \gamma \neq 1$, α and β do not include common variables, a Boolean polynomial $b\beta\phi + a\alpha\psi$ is called their **critical pair**.

Example 4.12

A critical pair of $aXY^{*}Z + bZ$ and $cXZW + Y^{*}$ such that $ac \neq 0$ is $aY + bcZW$.

Definition 4.13

For a finite set of Boolean polynomials R and a Boolean polynomial ϕ, the set of all possible critical pairs between ϕ and elements of R and variable self-critical pairs of ϕ is denoted by $CP(\phi, R)$.

Definition 4.14

For a finite set of Boolean polynomials R. Let $\{a_1\alpha \oplus \phi_1, \ldots, a_n\alpha \oplus \phi_n\}$ be a set of all polynomials in R such that the greatest monomial is α. Then, $(a_1 + \ldots + a_n)\alpha \oplus (\phi_1 + \ldots + \phi_n) \in Glue(R)$. $Glue(R)$ is a set of such Boolean polynomials.

Example 4.15

Let $R = \{aXY^{*} \oplus X, bXY^{*} \oplus Y, bXZ \oplus X, XZ \oplus Z\}$, then $Glue(R) = \{(a+b)XY^{*} \oplus (X + Y^{*}), (b+1)XZ \oplus (X + Z)\}$.

Definition 4.16

For a set of Boolean polynomials R and a Boolean polynomial ϕ, $\phi\downarrow_R$ denotes an irreducible form of ϕ by $\Rightarrow_R$.

Theorem 4.17

An algorithm to get a Boolean Gröbner base for an ideal generated by a given finite set E of Boolean polynomials, is given as follows.

input E
$R \longleftarrow \emptyset$
while $E \neq \emptyset$
  choose $\phi \in E$
  $E \longleftarrow (E - \{\phi\}) \cup \{csc(\phi)\}$ and $\phi' \longleftarrow \phi\downarrow_R$
  if $\phi' \neq 0$ **then**
    for every $a\alpha \oplus \psi \in R$
      if $a\alpha \Rightarrow_{\phi'} \varphi$
      then $E \longleftarrow E \cup \{\varphi + \psi\}$ and $R \longleftarrow R - \{a\alpha \oplus \psi\}$
      else $R \longleftarrow (R - \{a\alpha \oplus \psi\}) \cup \{a\alpha \oplus (\psi\downarrow_{R\cup\{\phi'\}})\}$
      end-if
    end-for
    $E \longleftarrow E \cup CP(\phi', R)$ and $R \longleftarrow R \cup \{\phi'\}$
  end-if
end-while
output $Glue(R)$ ($Glue(R)$ is the desired Boolean Gröbner base)

(The choice of an element of E must be fair, i.e. any element must be picked up somewhere in the outermost loop.)

Proof:

A detailed proof is given in [1]. ∎

An element of a polynomial ring $\mathbf{B}(X_1, \ldots, X_m, Y_1, \ldots, Y_n)$ can also be considered as an element of a Boolean polynomial ring $(\mathbf{B}(X_1, \ldots, X_m))(Y_1, \ldots, Y_n)$ with variables $Y_1, \ldots, Y_m$ and a coefficient Boolean ring $\mathbf{B}(X_1, \ldots, X_m)$.

Example 4.18
A Boolean polynomial $aXYZ + XZ + bYW + XY + X$ in a Boolean polynomial ring $\mathbf{B}(X, Y, Z, W)$, where a, b are elements of $\mathbf{B}$, is represented as $(aXY + X)Z + (bY)W + (XY + X)$ in $(\mathbf{B}(X, Y))(Z, W)$, and as $(aZ+1)XY+(Z+1)X+(bW)Y$ in $(\mathbf{B}(Z, W))(X, Y)$.

In the following $(\mathbf{B}(X_1, \ldots, X_m))(Y_1, \ldots, Y_n)$ is abbreviated by $\mathbf{B}(X_1, \ldots, X_m)(Y_1, \ldots, Y_n)$.

The next result is very important to solve set constraint.

Theorem 4.19

Let G be a Boolean Gröbner base of a finitely generated ideal I in $\mathbf{B}(X_1, \ldots, X_m)(Y_1, \ldots, Y_n)$. Let θ be a substitution of elements of $\mathbf{B}$ for variables $X_1, \ldots, X_m$. $G\theta$ denotes the set of $g\theta$ for $g \in G$ such that $(ag_1)\theta \neq 0$ where g_1 is the greatest monomial in g and a its coefficient. Then $G\theta$ forms a Boolean Gröbner base of a finitely generated ideal $I\theta$ in a Boolean polynomial ring $\mathbf{B}(Y_1, \ldots, Y_n)$. Moreover, for any Boolean polynomial $f \in \mathbf{B}(X_1, \ldots, X_m, Y_1, \ldots, Y_n)$, we have $(f\theta)\downarrow_{G\theta} = (f\downarrow_G)\theta$.

Proof:

A detailed proof is given in [4]. ∎

5. How to solve set constraint

Any constraint described by the language of section 1 can be solved by expressing it in terms of a Boolean polynomial ring. That is there exists a complete decision procedure to decide whether a constraint given by the language of section 1 is satisfiable. We first explain the base of the general case (Case 1), then describe the special case (Case 2) which is very important for application, and finally give a brief sketch of the decision procedure. The detail is given in [5].

Case 1

$$\begin{cases} f_1(\vec{x}, \vec{X}) = 0 \\ \quad \vdots \\ f_n(\vec{x}, \vec{X}) = 0 \\ g_1(\vec{x}, \vec{X}) \neq 0 \\ \quad \vdots \\ g_m(\vec{x}, \vec{X}) \neq 0 \end{cases}$$

($\vec{x}$, $\vec{X}$ denote finite number of first-order variables and second-order variables respectively)

$\Longrightarrow$

$$\begin{cases} f_1(\vec{x}, \vec{X}) = 0 \\ \quad \vdots \\ f_n(\vec{x}, \vec{X}) = 0 \\ y_1 \in g_1(\vec{x}, \vec{X}) \\ \quad \vdots \\ y_m \in g_m(\vec{x}, \vec{X}) \end{cases}$$

($y_1, \ldots, y_m$ are first-order variables different from $\vec{x}$)

$\Longrightarrow$

$$\begin{cases} f'_1(\vec{S}, \vec{X}) = 0 \\ \quad \vdots \\ f'_n(\vec{S}, \vec{X}) = 0 \\ g'_1(\vec{S}, \vec{T}, \vec{X}) = 0 \\ \quad \vdots \\ g'_m(\vec{S}, \vec{T}, \vec{X}) = 0 \end{cases}$$

($\vec{S}, \vec{T}$ are second-order variables different from $\vec{X}$. Substituting $\{x_i\}$ by S_i and $\{y_j\}$ by T_j, $f'_1, \ldots, f'_n$ is given from $f_1, \ldots, f_n$ and $g'_1, \ldots, g'_m$ is given from $\{y_1\}g_1 + \{y_1\}, \ldots, \{y_m\}g_m + \{y_m\}$.)

$\Longrightarrow$

(In a polynomial ring $P^{FC}(U)(\vec{S}, \vec{T})(\vec{X})$, calculate a Boolean Gröbner base $h_1(\vec{S}, \vec{T}, \vec{X})$, $\ldots$, $h_l(\vec{S}, \vec{T}, \vec{X})$, $h(\vec{S}, \vec{T})$ of the ideal generated by $f'_1, \ldots, f'_n, g'_1, \ldots, g'_m$.)

$$\begin{cases} h_1(\vec{S}, \vec{T}, \vec{X}) = 0 \\ \quad \vdots \\ h_l(\vec{S}, \vec{T}, \vec{X}) = 0 \\ h(\vec{S}, \vec{T}) = 0 \end{cases}$$

$\Longrightarrow$

$$\begin{cases} h'_1(\vec{x}, \vec{y}, \vec{X}) = 0 \\ \quad \vdots \\ h'_l(\vec{x}, \vec{y}, \vec{X}) = 0 \\ h'(\vec{x}, \vec{y}) = 0 \end{cases}$$

($h'_1, \ldots, h'_l, h'$ are given from $h_1, \ldots, h_l, h$ by substituting $\{x_i\}$ to S_i and $\{y_j\}$ to T_j.)

Then, it is a necessary and sufficient condition for the given constraint to have solutions that there exist elements $\vec{x}, \vec{y}$ such that $h'(\vec{x}, \vec{y}) = 0$. Moreover, for any $\vec{a}, \vec{b}$ satisfying $h'(\vec{a}, \vec{b}) = 0$, $\begin{cases} h'_1(\vec{a}, \vec{b}, \vec{X}) = 0 \\ \quad \vdots \\ h'_l(\vec{a}, \vec{b}, \vec{X}) = 0 \end{cases}$ is a canonical form of the given constraint, i.e. a solution.

The solution of $h'(\vec{x}, \vec{y}) = 0$ essentially belongs to the problem of a finite domain. Let $a_1, \ldots, a_s$ be constant symbols occurring in h' and b, c be constant symbols distinct from them. Then it is known to be sufficient to check $h'(\vec{x}, \vec{y}) = 0$ for all substitution from $\vec{a}, b, c$ to $\vec{x}, \vec{y}$. (See [5].)

Correctness of the algorithm can be shown easily by using Theorem 4.14.

Remark 5.1
The above method can be applied to remove quantifiers $\exists_0 \exists_1$ from $\exists_0 \vec{x} \exists_1 \vec{X}\ f_1(\vec{x}, \vec{y}, \vec{X}, \vec{Y}) = 0 \bigwedge \ldots \bigwedge f_n(\vec{x}, \vec{y}, \vec{X}, \vec{Y}) = 0 \bigwedge g_1(\vec{x}, \vec{y}, \vec{X}, \vec{Y}) \neq 0 \bigwedge \ldots \bigwedge g_m(\vec{x}, \vec{y}, \vec{X}, \vec{Y}) \neq 0$ to get a quantifier-free formula $h_1(\vec{y}, \vec{Y}) = 0 \bigvee \ldots \bigvee h_l(\vec{y}, \vec{Y}) = 0$.

Remark 5.2
Boolean Gröbner base is not the only tool to solve general Boolean constraint. There are some other known methods to handle Boolean equations. We adopted Boolean Gröbner base because of the following two reason, which does not hold in the other existing methods.

(i) There does not appear any new variable in the Boolean Gröbner base which is not in the given equations.

(ii) For a fixed ordering of variables, Boolean Gröbner base is determined unique.

By (i), the solution looks clear especially in the case of $P^{FC}(U)$. (ii) guarantees the paradigm of Constraint Logic Programming ,that is any constraint must have its normal form as the solution.

Example 5.3
$\begin{cases} \{a\} \cap X = \{x\} \cap X \\ \quad X \neq 0 \end{cases}$

$\Longrightarrow$

$\begin{cases} \{a\}X = \{x\}X \\ \{y\}X = \{y\} \end{cases}$

$\Longrightarrow$

$(\{x\} \leftarrow S, \{y\} \leftarrow T)$

$$\begin{cases} \{a\}X + SX = 0 \\ \quad TX + T = 0 \end{cases}$$

$\Longrightarrow$

(Calculate the Boolean Gröbner base in $P^{FC}(U)(S,T)(X)$, and put it equal to 0.)

$$\begin{cases} (\{a\} + S + T)X + T(1 + \{a\} + S) = 0 \\ \qquad\qquad\qquad\qquad T(S + \{a\}) = 0 \end{cases}$$

$\Longrightarrow$

$(S \leftarrow \{x\}, T \leftarrow \{y\})$

$$\begin{cases} (\{a\} + \{x\} + \{y\})X + \{y\}(1 + \{a\} + \{x\}) = 0 \\ \qquad\qquad\qquad\qquad \{y\}(\{x\} + \{a\}) = 0 \end{cases}$$

Solve $\{y\}(\{x\} + \{a\}) = 0$.

(i) $x = a$

In this case, the equation instantiated is $\{y\}X + \{y\} = 0$, which means $y \in X$ for arbitrary y.

(ii) $x = b, y = c$

In this case, the equation instantiated is $\{a, b, c\}X + \{c\} = 0$, which means $c \in X \bigwedge a, b \notin X$.

Case 2

$$\forall_1 \vec{X} \left(\begin{cases} f_1(\vec{x}, \vec{X}) = 0 \\ \quad \vdots \\ f_n(\vec{x}, \vec{X}) = 0 \end{cases} \rightarrow \quad h(\vec{x}, \vec{X}) = 0 \right).$$

$\Longrightarrow$

$$\begin{cases} f_1'(\vec{S}, \vec{X}) = 0 \\ \quad \vdots \\ f_n'(\vec{S}, \vec{X}) = 0 \end{cases}$$

($\vec{S}$ are second-ordered variables different from $\vec{X}$, f_i' is given from f_i by substituting $\{x_i\}$ by S_i)

$\Longrightarrow$

(Calculate the Boolean Gröbner base of the ideal generated by $f_1', \ldots, f_n'$ in $P^{FC}(U)(\vec{S})(\vec{X})$, and put it equal to 0.)

$$\begin{cases} g_1(\vec{S}, \vec{X}) = 0 \\ \quad \vdots \\ g_l(\vec{S}, \vec{X}) = 0 \\ g(\vec{S}) = 0 \end{cases}$$

$\Longrightarrow$

Transform $g_1, \ldots, g_l, g$ to $g_1', \ldots, g_l', g'$ by substituting $\{x_i\}$ by S_i, and let $G' = \{g_1', \ldots, g_l'\}$.

$\Longrightarrow$

The given constraint is equivalent to

$g'(\vec{x}) \neq 0 \bigvee h(\vec{x}, \vec{X})\downarrow_{G'} = 0.$

Correctness of the algorithm is also easily shown by Theorem 4.14.

5.4 Application

It can be considered as unification to solve this kind of constraint. We can treat some unifications such as order sorted unification in this framework uniformly.

General Case

Note that any constraint can be represented as one of the following prenex normal forms.

(i) $(quantifiers)\exists_0 \exists_1 \bigvee_{i=1}^{n} G_i$

$G_i = \bigwedge_{j=1}^{n_i} G_i^j$, G_i^j is an equation or disequation

(ii) $(quantifiers)\forall_0 \forall_1 \bigwedge_{i=1}^{n} F_i$

$F_i = \bigvee_{j=1}^{n_i} F_i^j$, F_i^j is an equation or disequation

$(quantifiers)$ denotes a finite sequence of quantifiers.

The innermost quantifiers are removed as follows.

Solution of (i).

$$
\begin{aligned}
&\exists_0 \exists_1 \bigvee_{i=1}^{n} G_i \\
&\Longrightarrow \\
&\bigvee_{i=1}^{n} \exists_0 \exists_1 G_i \\
&\Longrightarrow \\
&\bigvee_{i=1}^{n} \exists_0 \exists_1 \bigwedge_{j=1}^{n_i} G_i^j \\
&\Longrightarrow \\
&\bigvee_{i=1}^{n} (g_1^i = 0 \bigvee \ldots \bigvee g_{l_i}^i = 0 \text{ (by the remark of Case 1)}
\end{aligned}
$$

Solution of (ii).

$$
\begin{aligned}
&\forall_0 \forall_1 \bigwedge_{i=1}^{n} F_i \\
&\Longrightarrow \\
&\bigwedge_{i=1}^{n} \forall_0 \forall_1 F_i \\
&\Longrightarrow \\
&\bigwedge_{i=1}^{n} \forall_0 \forall_1 \bigvee_{j=1}^{n_i} F_i^j \\
&\Longrightarrow \\
&\bigwedge_{i=1}^{n} \neg(\exists_0 \exists_1 \bigwedge_{j=1}^{n_i} (\neg F_i^j)) \\
&\Longrightarrow \\
&\bigwedge_{i=1}^{n} (f_1^i \neq 0 \bigwedge \ldots \bigwedge f_{l_i}^i \neq 0) \text{ (by the remark of Case 1)}
\end{aligned}
$$

For a given constraint, applying either solution of (i) or (ii) repeatedly, we can finally get a quantifier-free formula. Since any quantifier-free formula is represented as

$\bigvee_{i=1}^{n} \bigwedge_{j=1}^{n_i} H_i^j$,where H_i^j is an equation or disequation,

we can use the solution method of Case 1 to decide if it is satisfiable.

Remark 5.5
The above method works not only for $P^{FC}(U)$ but also for any atomic Boolean algebra. The detail is given in [5].

ACKNOWLEDGEMENTS

We would like to express our gratitude to R. Hasegawa, Fourth and Fifth Laboratory Chief, and A. Aiba, Fourth Laboratory Deputy Chief, of ICOT for providing the opportunity to conduct this research. Special thanks are also due to K. Mukai, Theme Leader, of ICOT for many worthwhile suggestions.

REFERENCES

[1] Sakai,K. Sato,Y. Menju,S.: *Boolean Gröbner base(revised)*, to appear (1990)

[2] Sakai,K. Sato,Y.: *A note on solvability of Boolean equations* , IEICE Technical Report, Vol.89,No.276,pp.41-44 (1989)

[3] Sakai, K. and Sato, Y.: *Zero-point theorem for Boolean polynomial ring*, to appear (1990)

[4] Sato, Y.: *Universal Boolean Gröbner base*, to appear (1990)

[5] Sato, Y.: *Quantifier elimination for atomic Boolean constraint*, to appear (1990)