ICOT Technical Report: TR-680

TR-680

Solving constraints over sets by Boolean Gröbner bases

by

Y. Sato, S. Menju & K. Sakai

August, 1991

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

Solving constraints over sets by Boolean Gröbner bases

Yosuke Sato, Satoshi Menju, and Kô Sakai

ICOT Research Center
Mita Kokusai Bldg.21F
1-4-28, Mita, Minano-ku, Tokyo 108, JAPAN
email: ysato@icot.or.jp

ABSTRACT

Any constraint over sets can be represented in terms of a Boolean polynomial ring whenever the family of sets that we consider forms a Boolean ring. In this paper we give a complete solution method for such constraints using Boolean Gröbner bases. A Boolean Gröbner base is a modification of a standard Gröbner base which we developed to solve constraints of general Boolean polynomial rings.

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 $\subseteq$. A set which a computer can handle naively is a finite set or a co-finite set(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. Almost 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 $\{a\}X = \{a\}$, $\{b\}Y = 0$, $XY = X$. In order to deal with set constraints represented in this form, we devised a Boolean Gröbner base as a canonical form of a given finite set of Boolean equations and developed a modified Buchberger algorithm to calculate it. This algorithm gives a complete decision procedure. That is, for a constraint given in a form of a finite set of Boolean equations, the algorithm calculates the Boolean Gröbner base as its canonical form whenever it is satisfiable, otherwise it tells the constraint is unsatisfiable. One of the most important properties of a Boolean Gröbner base is that it does not include any variable which is not included in the given equations, which does not hold in the other existing methods([2],[3]). Especially in the case of set constraints, this property makes solutions very easy to read. We first give a language which is sufficient to express our constraints over sets, and show how constraints written in this language are expressed in terms of a Boolean polynomial ring. Then we give several important our results of a Boolean polynomial ring such as Boolean Gröbner bases, and show how they are applied to give a complete decision procedure for solving constraint written in our language.

2. A language for set constraints

Suppose we are given a fixed ground set U of elements, and we want to write constraints over (finite or co-finite)subsets of U and elements of U. In order to represent these constraints, let us suppose that we can use the following symbols.

$a_1, a_2, a_3, \ldots$: first-order constant symbols for elements

$x_1, x_2, x_3, \ldots$: first-order variables for elements

$f_1, f_2, f_3, \ldots$: function symbols for functions from elements to elements

$X_1, X_2, X_3, \ldots$: 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 for sets

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

(The symbol $\cdot^c$ is a function symbol which gives a complement of a set.)

Since we want to deal with just two types of objects, elements and sets, we assign exactly one type for each symbol. For example, we can not have an expression such as $\{a\} \in X$. The left side of $\in$ must be an element, however, $\{a\}$ is not an element but a set.

3. Expression of set constraints by a Boolean polynomial ring

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

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

Then $\langle \mathbf{B}, +, \cdot, 0, 1\rangle$ becomes a commutative ring with a 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 a unit if we define $\vee, \wedge, \neg$ by

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

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 is 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 defined as 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 ground terms using $a_1, a_2, a_3, \ldots$ and $f_1, f_2, f_3, \ldots$. The set of all finite or co-finite subsets of U denoted by $P^{FC}(U)$ forms a Boolean algebra $\langle P^{FC}(U), \vee, \wedge, \neg, 0, 1\rangle$ taking a set union $\cup$ for $\vee$, a set intersection $\cap$ 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 $\bigvee, \bigwedge, \neg, \to, \ldots$. For example, $a \in X \cap (Y^c \cup Z)$ is expressed as follows.

$$a \in X \cap (Y^c \cup Z) \Leftrightarrow a \in X((Y+1)Z + (Y+1) + Z) \Leftrightarrow a \in XYZ + XY + X$$
$$\Leftrightarrow \{a\} \cap (XYZ + XY + X) = \{a\} \Leftrightarrow \{a\}XYZ + \{a\}XY + \{a\}X = \{a\}$$

($\cdot$ is abbreviated, as is customary.)

4. Boolean Gröbner bases

We outline the method to solve equations of a general Boolean polynomial ring by Boolean Gröbner bases, which is a base of our solution method for set constraints represented in terms of $P^{FC}(U)(X, Y, Z, \ldots)$. The reader can be referred to [4],[7] for more detailed description.

Finite multi-sets of variables are called **power products**, which are denoted by meta-symbols $\alpha, \beta, \gamma, \ldots$. Note that in any polynomial ring a polynomial can be represented as

$$a_0 + a_1\alpha_1 + \ldots + a_k\alpha_k,$$

with elements $a_0, a_1, \ldots, a_k$ of a coefficient ring and power products $\alpha_1, \ldots, \alpha_k$.

An ordering $\geq$ over power products is called **admissible** if it satisfies the following properties.

(i) If $\alpha \supseteq \beta$, then $\alpha \geq \beta$.

(ii) If $\alpha \geq \beta$, then $\alpha\gamma \geq \beta\gamma$ for any power product γ, where $\alpha\gamma$, $\beta\gamma$ are power products defined by multi-set unions of α and γ, β and γ respectively.

There are several examples of **total** admissible orderings over power products, based on a lexicographic ordering or total degree ordering. From now on, we fix a total admissible ordering $\geq$ over power products.

$a\alpha \oplus \phi$ is an expression of a Boolean polynomial whose greatest power product is 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 $\varphi = \psi + b\alpha\beta$, 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 $b\alpha\beta = b(1 + a)\alpha\beta + ba\alpha\beta$. Secondly, since $a\alpha \oplus \phi = 0$ implies $a\alpha = \phi$, multiplying $ab\beta$ from both sides, we have $ba\alpha\beta = ab\beta\phi$. Therefore under the condition $a\alpha \oplus \phi = 0$, we have $b\alpha\beta = b(1 + a)\alpha\beta + ab\beta\phi$. The reader who is familiar with the standard Gröbner base (for a polynomial ring over field or Euclidean ring)[1], should notice that our rewriting rule is much different from its rewriting rule. This is explained as follows. 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 cannot 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 $dX \Rightarrow_{(1+c)X} 0$.

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 $\overset{*}{\Rightarrow}_R$. That is $\phi \overset{*}{\Rightarrow}_R \psi$ means ϕ is rewritten to ψ by applying $\Rightarrow_R$ finitely many(possibly 0) times.

Theorem 4.1

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

Definition 4.2 Boolean Gröbner base

Let I be an ideal of a Boolean polynomial ring $\mathbf{B}(X_1, \ldots, X_n)$. A 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 $g \in G$ cannot 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.

In the definition of a standard Gröbner base, usually the property (iii) is not included. We require it in order to have the following property 4.3(iii). The property (iv) is also needed for this, although which is a direct conclusion from the property (iii) in case of a standard Gröbner base.

Properties 4.3

(i) The ideal generated by G is I.

(ii) The constraint given by I, that is 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(a non-zero element of $\mathbf{B}$).

(iii) 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 several definitions needed to describe an algorithm to get a Boolean Gröbner base. For a Boolean polynomial $a\alpha \oplus \phi$, a Boolean polynomial $a\phi + \phi$ is called its **coefficient self-critical pair** denoted by $csc(a\alpha \oplus \phi)$ and a Boolean polynomial $X\phi + \phi$ for each variable X in α is called its **variable self-critical pair**. 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**. For example, a coefficient self critical pair of $aXYZ + bYW$ is $(ab+b)YW$. There are two non-zero variable self-critical pairs, namely $bXYW + bYW$ and $bYZW + bYW$. A critical pair of $aXYZ + bZ$ and $cXZW + Y$ such that $ac \neq 0$ is $aY + bcZW$. 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 $CP(\phi, R)$.

For a finite set of Boolean polynomials R , add all polynomials whose greatest power products are same and put them together to form a set of Boolean polynomials denoted by $Glue(R)$. For example, let

$$R = \{aXY \oplus X, bXY \oplus Y, bXZ \oplus X, XZ \oplus Z\}, \quad \text{then}$$

$$Glue(R) = \{(a+b)XY \oplus (X+Y), (b+1)XZ \oplus (X+Z)\}.$$

For a set of Boolean polynomials R and a Boolean polynomial ϕ, $\phi{\downarrow}_R$ denotes one of irreducible forms of ϕ by $\Rightarrow_R$.

Theorem 4.4
For a given finite set E_0 of Boolean polynomials, an algorithm to get a Boolean Gröbner base for an ideal generated by E_0 is given as follows.

input $E \longleftarrow E_0, R \longleftarrow \emptyset$
while $E \neq \emptyset$
  choose $\phi \in E$ and $\phi' \longleftarrow \phi{\downarrow}_R$ ......(a)
  if $\phi' \neq 0$ **then** $E \longleftarrow (E - \{\phi\}) \cup \{csc(\phi')\}$
    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 $R \longleftarrow R \cup \{\phi'\}$
  else $E \longleftarrow (E - \{\phi\})$
  end-if
end-while
output $Glue(R)$ ......(b)

The output $Glue(R)$ of (b) is the desired Boolean Gröbner base. The choice of an element of E at (a) must be fair, that is any element must be picked up at some stage of (a).

An element of a polynomial ring $\mathbf{B}(X_1, \ldots, X_m, Y_1, \ldots, Y_n)$ can be also 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 $\mathbf{B}(X_1, \ldots, X_m)$. For example, a Boolean polynomial

$$aXYZ + XZ + bYW + XY + X$$

in a Boolean polynomial ring $\mathbf{B}(X, Y, Z, W)$, is represented

$$(aXY + X)Z + (bY)W + (XY + X)$$

as an element of $(\mathbf{B}(X, Y))(Z, W)$, and

$$(aZ + 1)XY + (Z + 1)X + (bW)Y$$

as an element of $(\mathbf{B}(Z, W))(X, Y)$, where a, b are elements of $\mathbf{B}$. 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 for our solution method for set constraints.

Theorem 4.5

Let G be a Boolean Gröbner base of a finitely generated ideal I in $\mathbf{B}(X_1, \ldots, X_m)(Y_1, \ldots, Y_n)$. For a substitution θ of elements of $\mathbf{B}$ to variables $X_1, \ldots, X_m$, $G\theta$ denotes the set of all $g\theta$ for $g \in G$ such that $(a\alpha)\theta \neq 0$ where α is the greatest power product in g and a is 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 $(f\theta)\downarrow_{G\theta} = (f\downarrow_G)\theta$.

5. How to solve set constraints

For constraints described by the language of section 2, we give a complete solution method using Boolean Gröbner bases. Note that any constraint can be represented as the following form.

$$\bigvee_{i=1}^{n} H_i \qquad (H_i = \bigwedge_{j=1}^{n_i} H_i^j, \quad H_i^j \text{ is an equation or disequation.})$$

It suffices to consider solving the following constraints consisting of equations and disequations.

$$\begin{cases} F_1(\vec{x}, \vec{f}, \vec{X}) = 0 \\ \quad\vdots \\ F_n(\vec{x}, \vec{f}, \vec{X}) = 0 \\ G_1(\vec{x}, \vec{f}, \vec{X}) \neq 0 \\ \quad\vdots \\ G_m(\vec{x}, \vec{f}, \vec{X}) \neq 0 \end{cases}$$

($\vec{x}$, $\vec{f}$ and $\vec{X}$ denote a finite number of first-order variables, terms which begin with function symbols and second-order variables respectively.)

This is transformed to the following using first-ordered variables $y_1, \ldots, y_m$ which are distinct from $\vec{x}$.

$$\begin{cases} F_1(\vec{x}, \vec{f}, \vec{X}) = 0 \\ \quad \vdots \\ F_n(\vec{x}, \vec{f}, \vec{X}) = 0 \\ \quad \{y_1\} = \{y_1\}G_1(\vec{x}, \vec{f}, \vec{X}) \\ \quad \vdots \\ \quad \{y_m\} = \{y_m\}G_m(\vec{x}, \vec{f}, \vec{X}) \end{cases}$$

Substitute $\{x_i\}, \{f_j\}$ and $\{y_k\}$ by distinct second-ordered variables S_i, T_j and Y_k which are different from $\vec{X}$. We get the following.

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

In the Boolean polynomial ring $P^{FC}(U)(\vec{S}, \vec{T}, \vec{Y})(\vec{X})$, calculate its Boolean Gröbner base

$$\{H_1(\vec{S}, \vec{T}, \vec{Y}, \vec{X}), \ldots, H_l(\vec{S}, \vec{T}, \vec{Y}, \vec{X}), H(\vec{S}, \vec{T}, \vec{Y})\}.$$

By resubstituting S_i, T_j, Y_k by $\{x_i\}, \{f_j\}, \{y_k\}$ we get the following equations.

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

By Theorem 4.5, it is satisfiable if and only if $H'(\vec{x}, \vec{f}, \vec{y}) = 0$ has a solution for the first-order variables $\vec{x}, \vec{y}$. Moreover, for any such solution $\vec{\alpha}, \vec{\beta}$

$$\begin{cases} H'_1(\vec{\alpha}, \vec{f}(\vec{\alpha}, \vec{\beta}), \vec{\beta}, \vec{X}) = 0 \\ \quad \vdots \\ H'_l(\vec{\alpha}, \vec{f}(\vec{\alpha}, \vec{\beta}), \vec{\beta}, \vec{X}) = 0 \end{cases}$$

is a canonical form(i.e. a solution) of the originally given constraint. Before describing how to solve $H'(\vec{x}, \vec{f}, \vec{y}) = 0$, consider the equation, $\{x\}\{a\} + \{y\}\{a\} = 0$. For the variables x, y, there are the following 5 possibilities, (i) $x = y = a$ (ii) $x = y \neq a$ (iii) $a = x \neq y$ (iv) $x \neq y = a$ (v) $x \neq y, x \neq a, y \neq a$. $\{x\}\{a\} + \{y\}\{a\}$ is equal to 0 only in the cases (i),(ii) and (v). It is equal to $\{a\}$ in (iii) and (iv). That is, $\{x\}\{a\} + \{y\}\{a\} = 0$ is equivalent to (i) or (ii) or (v). It is easy to generalize this obervation to get the following formulation of $H'(\vec{x}, \vec{f}, \vec{y}) = 0$.

$$H'(\vec{x}, \vec{f}, \vec{y}) = 0 \Longleftrightarrow \bigvee_{i=1}^{r} \bigwedge_{j=1}^{n_i} G_i^j.$$

For each j, i, G_i^j is an equation or a disequation. Each side of the equation or disequation is either a variable among $\vec{x}, \vec{y}$, a term among $\vec{f}$ or a first-order constant symbol which appears in $H'(\vec{x}, \vec{f}, \vec{y}) = 0$. Hence, a solution of $H'(\vec{x}, \vec{f}, \vec{y}) = 0$ is nothing but a unifier of $\bigwedge_{j=1}^{n_i} G_i^j$ for some i.

The above method is the most naive approach. In our implementation, instead of solving $H'(\vec{x}, \vec{f}, \vec{y}) = 0$ directly as above we calculate a Boolean Gröbner base of $H(\vec{S}, \vec{T}, \vec{Y})$ in the Boolean polynomial ring $P^{FC}(U)(\vec{S}, \vec{T}, \vec{Y})$, and use it to simpify $H_1(\vec{S}, \vec{T}, \vec{Y}, \vec{X}), \ldots, H_l(\vec{S}, \vec{T}, \vec{Y}, \vec{X})$ before resubstituting S_i, T_j, Y_k by $\{x_i\}, \{f_j\}, \{y_k\}$.

In the example of our program in the appendix, include({s},a\/c),member(f(x1,x2),a/\~b), member(f(z1,g(z2)),c/\~b) and member(f(e,y),b) mean $\{s\} \supseteq a \cup c$, $f(x_1, x_2) \in a \cap b^c$, $f(z_1, g(z_2)) \in c \cap b^c$ and $f(e, y) \in b$, respectively, [s,x1,x2,y,z1,z2] is a list of first-order variables and [a,b,c] is a list of second order variables. The program outputs a Boolean Gröbner base and conditions to be unified. In the example, the unifier is $\{ x_1 = z_1, x_2 = g(z2), s = f(z_1, g(z_2)), (e, y) \neq (z_1, g(z_2))\}$. The instanciated Boolean Gröbner base by this unifier is then $\{c = \{f(z_1, g(z_2))\}, a = \{f(z_1, g(z_2))\}, (\{f(e, y)\} + \{f(z_1, g(z_2))\}b = \{f(e, y)\}\}$. The last equation means $f(e, y) \in b$ and $f(z_1, g(z_2)) \notin b$.

We can extend our method for more complicated constraint using quantifiers, see [8].

REFERENCES

[1] Buchberger, B.: *Gröbner Bases an Algebraic method in Polynomial Ideal Theory*, Technical Report,CAMP LINZ (1983)

[2] Buttner, W. Simonis, Helmut.: *Embedding Boolean Expressions into Logic Programming*, J.Symbolic Computation, pp.191-205 (1987)

[3] Martin, U. Nipkow, T.: *Unification in Boolean Rings,* Proceedings of the Eighth International Conference on Automated Deduction CADE-8, Oxford(England) (1986)

[4] Sakai, K. Sato, Y. Menju, S.: *Boolean Gröbner base(revised),* ICOT Technical Report (1990)

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

[6] Sakai, K. and Sato, Y.: *Zero-point theorem for Boolean polynomial ring,* ICOT Technical Memorandum (1990)

[7] Sato, Y.: *Universal Boolean Gröbner base,* ICOT Technical Memorandum (1990)

[8] Sato, Y.: *Quantifier elimination for Boolean constraint,* to appear (1991)

APPENDIX

```
:- setgb([include({s},a\/c),member(f(x1,x2),a/\~b),
    member(f(z1,g(z2)),c/\~b),
    member(f(e,y),b)],[s,x1,x2,y,z1,z2],[a,b,c]).

GB =
  (1+{s}+{f(z1,g(z2))})/\c = {f(z1,g(z2))}
  (1+{s}+{f(x1,x2)})/\a = {f(x1,x2)}
  ({f(e,y)}+{f(x1,x2)}+{f(z1,g(z2))}+{f(x1,x2)}/\{f(z1,g(z2))})/\b = {f(e,y)}
  -------------
  {f(e,y)}/\{f(z1,g(z2))} = 0
  {s}/\{f(z1,g(z2))} = {f(z1,g(z2))}
  {f(x1,x2)}/\{f(e,y)} = 0
  {s}/\{f(x1,x2)} = {f(x1,x2)}

Unify
  s = f(z1,g(z2))
  s = f(x1,x2)
  f(e,y) =\= f(z1,g(z2))
  f(x1,x2) =\= f(e,y)
```