TM-0488

Boolean Groebner Bases

by
K. Sakai & Y. Sato

March, 1988

©1988, ICOT

ICOT
Mita Kokusai Bldg. 21F
1-28 Mita 1-Chome
Minato ku Tokyo 108 Japan

(03) 456-3191～5
Telex ICOT J32964

Boolean Gröbner Bases

Kô Sakai and Yosuke Sato

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

(April 11, 1988)

ABSTRACT

This paper proposes an algebraic tool to solve the constraints of propositional calculus. Any constraint of propositional calculus can be described in terms of equations in a Boolean algebra, and also translated into equations of a Boolean ring where the constraint is represented in the form of a finite set of polynomials. We give an algorithm which produces a rewriting system for a given Boolean constraint, which reduces Boolean polynomials equivalent under the constraint to the same normal form. The algorithm is a central part of the constraint solver in CAL (Contrainte avec Logique), which is a constraint logic programming language being developed at ICOT.

1. Boolean ring

We assume that the reader is familiar with elementary algebraic notions such as ring and ideal (see [Waerden 37, 40], for example), and the terminology of rewriting systems (see [Huet 80], for example).

For a Boolean algebra $< B, \vee, \wedge, \neg >$, define $x + y =_{def} (x \wedge \neg y) \vee (\neg x \wedge y)$ and $x \times y =_{def} x \wedge y$ for each x, y in B, then $< B, \times, + >$ is known to be a commutative ring with a unit with the following properties.

(i) $\forall x \in B \quad x + x = 0$

(ii) $\forall x \in B \quad x \times x = x$

A ring with these properties is called a Boolean ring. Here, we define a Boolean ring of polynomials specifically, as used in this paper

Definition 1.1

Let there be countably many *Boolean variables*, which are denoted by metasymbols $a, b, c, \ldots$. A *Boolean monomial* is a (finite) multiset of Boolean variables. It is denoted by using $\times$. For example, $a \times a \times b \times c$ denotes multiset $\{a, a, b, c\}$. The empty Boolean

monomial, $\{\}$, is denoted 1. We use metasymbols $A, B, C, \ldots$ for Boolean monomials. $A \times B$ is defined as the multiset union of A and B. For example, $A \times B = \{a, a, b, c\}$ when $A = \{a, b\}$ and $B = \{a, c\}$. A *Boolean polynomial* is a (finite) multiset of Boolean monomials. It is denoted by using $+$. For example, $A + A + B + C$ denotes multiset $\{A, A, B, C\}$. The empty Boolean polynomial, $\{\}$, is denoted 0. We use metasymbols $X, Y, Z, \ldots$ for Boolean polynomials. $X + Y$ is defined as the multiset union of X and Y. Binary function $\times$ is extended to Boolean polynomials in a natural way. For example,

$$(A + B) \times (A + C + D) = A \times A + A \times C + A \times D + B \times A + B \times C + B \times D.$$

Note that both $\times$ and $+$ are associative and commutative, and are also distributive, i.e., $X \times (Y + Z) = X \times Y + X \times Z$ for each Boolean polynomial, X, Y, and Z. We abuse metasymbols $a, b, c, \ldots$ to denote Boolean monomials $\{a\}, \{b\}, \{c\}, \ldots$ and $A, B, C, \ldots$ for Boolean polynomials $\{A\}, \{B\}, \{C\}, \ldots$, which will be clear from the context.

In this paper, we omit $\times$. For example, we write aab instead of $a \times a \times b$.

Definition 1.2

The rewriting rule, $\rightarrow_\times$, on Boolean monomials is defined as $X + aaA \rightarrow_\times X + aA$ for each variable a, monomial A, and polynomial X. The rewriting rule, $\rightarrow_+$, on Boolean polynomials is defined as $X + A + A \rightarrow_+ X$ for each monomial A and polynomial X.

It is easy to show the following:

Proposition 1.3

The rewriting system of rules $\{\rightarrow_\times, \rightarrow_+\}$ is confluent and terminating. ∎

Definition 1.4

For a Boolean polynomial, X, the normal form of X by $\{\rightarrow_\times, \rightarrow_+\}$ is denoted $X{\downarrow}$ and called a *Boolean normal polynomial*. For example, $aabbccc{\downarrow} = abc$, $(aabc + abcc + bc + cdd){\downarrow} = bc + cd$. A Boolean normal polynomial is the sum of different Boolean monomials, each of which is the product of different Boolean variables.

Definition 1.5

Define the product, $\times'$, and the sum, $+'$, of Boolean normal polynomials as follows:

$$X \times' Y =_{def} (X \times Y){\downarrow} \qquad X +' Y =_{def} (X + Y){\downarrow}$$

The set of all Boolean normal polynomials with operations $\times'$ and $+'$, defined above, forms a Boolean ring.

2. Boolean Gröbner base

Definition 2.1

Let $\geq$ be an ordering on Boolean monomials. The ordering is said to be *admissible* if the following hold:

(i) $A \geq B$ for any monomials A and B such that $A \subseteq B$ in the sense of multiset inclusion.

(ii) If $A \geq B$, then $AC \geq BC$ for any monomials A, B, and C.

Let V be a fixed finite set of Boolean variables. The fact that an admissible ordering on monomials consisting only of the variables in V is well-founded is well known as Dickson's lemma [Dickson 13], or easily proven as its corollary. An admissible ordering on monomials is extended to polynomials by employing induced multiset ordering [Dershowitz 79]. Since induced multiset ordering is well-founded if the base ordering is well-founded, the extension is well-founded on polynomials consisting only of the variables in V. Moreover, induced multiset ordering is total if the base ordering is total. In what follows let $\geq$ be a fixed admissible total ordering on Boolean monomials.

Definition 2.2

Let $A \oplus X$ denote $A + X$ but also mean that Boolean monomial A is greater than any Boolean monomial in X with respect to $\geq$. If Y is a Boolean polynomial such that $Y = S + AB$ and Z is a Boolean polynomial such that $Z = S + XB$, then we write $Y \rightarrow_{A \oplus X} Z$. Similarly, if V is a Boolean normal polynomial such that $V = T + AC$ and W is a Boolean normal polynomial such that $W = (T + XC)\downarrow$, we write $V \Rightarrow_{A \oplus X} W$.

This means that Z or W is obtained from Y or V by substituting A for X by using the rule $A = X$ which is equivalent to $A + X = 0$.

Example 2.3

Let $Y = abc + bc$. Then $Y \rightarrow_{ab \oplus c} cc + bc$ and $Y \Rightarrow_{ab \oplus c} c + bc$, since $(cc + bc)\downarrow = c + bc$.

Lemma 2.4

Let $A \oplus X$ be a Boolean normal polynomial. If $Y \rightarrow_{A \oplus X} Z$, then $Y > Z$ for any Boolean polynomials Y and Z. If $V \Rightarrow_{A \oplus X} W$, then $V > W$ for any Boolean normal polynomials V and W.

Proof: Easy to check. ∎

Corollary 2.5

For any set of Boolean normal polynomials $\{X_1, X_2, \ldots, X_n\}$, the rewriting systems $\{\rightarrow_\times, \rightarrow_+, \rightarrow_{X_1}, \rightarrow_{X_2}, \ldots, \rightarrow_{X_n}\}$ and $\{\Rightarrow_{X_1}, \Rightarrow_{X_2}, \ldots, \Rightarrow_{X_n}\}$ are terminating. ∎

Definition 2.6

Let R be a finite set of Boolean normal polynomials. We write $Y \Rightarrow_R Z$ if there exists $X \in R$ such that $Y \Rightarrow_X Z$, and $Y \overset{*}{\Rightarrow}_R Z$ if $Y = Z$ or there exists a possibly empty sequence $Y_1, Y_2, \ldots, Y_m$ of polynomials such that $Y \Rightarrow_R Y_1, Y_1 \Rightarrow_R Y_2, \ldots, Y_{m-1} \Rightarrow_R Y_m, Y_m \Rightarrow_R Z$. That is $\overset{*}{\Rightarrow}_R$ is the transitive reflexive closure of $\Rightarrow_R$.

In what follows, we will discuss ideals in the ring of Boolean normal polynomials. Intuitively, an ideal can be regarded as the set of all normal polynomials of value 0 under a certain constraint.

Definition 2.7

Let I be an ideal of the ring of Boolean normal polynomials. A *Gröbner base* for I is a finite set of Boolean normal polynomials R such that $\Rightarrow_R$ is confluent and terminating, and moreover, the following two conditions are equivalent for any polynomials, X and Y.

(i) $(X + Y)\downarrow \in I$ (or $X \equiv Y \pmod I$)

(ii) There exists a polynomial, Z, such that $X \overset{*}{\Rightarrow}_R Z$ and $Y \overset{*}{\Rightarrow}_R Z$.

Theorem 2.8

Let E be an arbitrary finite set of Boolean normal polynomials, then a Gröbner base for the ideal generated by E exists and, furthermore, we have an algorithm to construct it from E.

Intuitively, an element of the generated ideal is a polynomial of value 0 under the constraint that all elements in E have value 0. A Gröbner base can be viewed as a mechanism to determine whether a certain polynomial is in the ideal. First, we give an algorithm, then show its correctness. We need to define several notions.

Definition 2.9

Let R be a finite set of Boolean normal polynomials. For each Boolean normal polynomial X, $X\downarrow_R$ denotes a Boolean normal polynomial, Y, such that $X \overset{*}{\Rightarrow}_R Y$ and Y is irreducible by $\Rightarrow_R$, i.e., there exists no Boolean normal polynomial, Z, such that $Y \Rightarrow_R Z$. (Note that Corollary 2.5 assures the existence of such Y. However, it may not be unique. $X\downarrow_R$ denotes one Y.)

Definition 2.10

Let $A \oplus X$ be a Boolean normal polynomial, and a a variable in A. Then $(aX + X)\downarrow$ is called a *self-critical pair* of $A \oplus X$.

If $A \oplus X$ is in an ideal, I, then so are all the self-critical pairs of $A \oplus X$. In fact, let $a \in A$, i.e., $A = aB$ for some (possibly empty) Boolean monomial, B. Then, $aB \oplus X \in I$ implies $((a+1)(aB \oplus X))\downarrow = (aX + X)\downarrow \in I$.

Example 2.11

Let $A \oplus X$ be $ab \oplus b + c$. Then, $(a(b+c)+(b+c))\downarrow = ab+ac+b+c$ and $(b(b+c)+(b+c))\downarrow = bc + c$. Therefore, self-critical pairs of $A \oplus X$ are $ab + ac + b + c$ and $bc + c$.

Definition 2.12

Let $A \oplus X$ and $B \oplus Y$ be Boolean normal polynomials, and C the intersection of A and B as multisets. According to tradition, let us call C the GCD (greatest common divisor) of A and B. Suppose that $A = CA'$ and $B = CB'$. Then, $(B'X + A'Y)\downarrow$ is called the critical pair between $A \oplus X$ and $B \oplus Y$.

If $A \oplus X$ and $B \oplus Y$ are in an ideal, I, then so is the critical pair between $A \oplus X$ and $B \oplus Y$. In fact, $(B'(A \oplus X) + A'(B \oplus Y))\downarrow = (B'X + A'Y)\downarrow \in I$.

Example 2.13

Let $A \oplus X = abc \oplus a + b$ and $B \oplus Y = abd \oplus a + b$, then $(d(a + b) + c(a + b))\downarrow = ac + ad + bc + bd$. Therefore, $ac + ad + bc + bd$ is the critical pair between $abc \oplus a + b$ and $abd \oplus a + b$.

Definition 2.14

Let X be a Boolean normal polynomial and R be a finite set of Boolean normal polynomials, then $CP(X, R)$ denotes the set consisting of all the non-zero critical pairs between X and each element of R and all the self-critical pairs of X.

Now the algorithm can be presented.

input E
$R \longleftarrow \emptyset$
while $E \neq 0$
  choose $X \in E$
  $E \longleftarrow E - \{X\}$ and $X' \longleftarrow X\downarrow_R$
  if $X' \neq 0$ **then**
    for every $A \oplus Y \in R$
      if $A \Rightarrow_{X'} Z$
      then $E \longleftarrow E \cup \{(Z + Y)\downarrow\}$ and $R \longleftarrow R - \{A \oplus Y\}$
      else $R \longleftarrow (R - \{A \oplus Y\}) \cup \{A \oplus Y\downarrow_{R \cup \{X'\}}\}$
      end-if
    end-for
    $E \longleftarrow E \cup CP(X', R)$ and $R \longleftarrow R \cup \{X'\}$
  end-if
end-while
output R (R is a Gröbner base)

(In this algorithm, the choice of an element in E should be fair. That is, any element of E should be chosen at some stage in the outermost **while** loop.)

This algorithm terminates and returns a Gröbner base. To prove the correctness of the algorithm, we study a more general form of the algorithm.

Definition 2.15

We define inference rules on pairs (E, R) of finite sets of Boolean normal polynomials.

Rule 1
$$\frac{E \cup \{X\}, R}{E \cup \{Y\}, R} \quad \text{where } X \Rightarrow_R Y$$

Rule 2
$$\frac{E \cup \{0\}, R}{E, R}$$

Rule 3
$$\frac{E, R \cup \{A \oplus X\}}{E, R \cup \{A \oplus Y\}} \quad \text{where } X \Rightarrow_R Y$$

Rule 4
$$\frac{E, R \cup \{AB \oplus X\}}{E \cup \{(AZ + X)\downarrow\}, R} \quad \text{where } B \oplus Z \in R \text{ and } A \neq \emptyset$$

Rule 5
$$\frac{E \cup \{A \oplus X\}, R}{E, R \cup \{A \oplus X\}}$$

Rule 6
$$\frac{E, R}{E \cup \{(CX + BY)\downarrow\}, R} \quad \text{where } AB \oplus X, AC \oplus Y \in R \text{ and } B \cap C = \emptyset \quad \text{(critical pair)}$$

Rule 7
$$\frac{E, R}{E \cup \{(aX + X)\downarrow\}, R} \quad \text{where } aA \oplus X \in R \quad \text{(self-critical pair)}$$

Definition 2.16 (General form of the algorithm)

Let $E_0 = E, R_0 = \emptyset$. For each i, let E_{i+1} and R_{i+1} be obtained from E_i and R_i by one of the above rules. In the following, $\cup_{n=1}^{\infty} \cap_{i=n}^{\infty} E_i$ is denoted by E^{∞} and $\cup_{n=1}^{\infty} \cap_{i=n}^{\infty} R_i$ by R^{∞}. We give priority to Rules 1 and 2. We need two restrictions to make the algorithm correct.

(i) The algorithm must be fair, i.e., $E^{\infty} = \emptyset$.

(ii) Any possible critical pair or self-critical pair must be taken, i.e., for each $X \in R^{\infty}$, any self-critical pair of X must be put in some E_i by Rule 7 and for each $X, Y \in R^{\infty}$, any critical pair of X and Y must be put in some E_i by Rule 6.

Then for some i, E_i is empty and R_i is a Gröbner base. (Note that the previous algorithm takes the form defined here.)

To prove the last statement, we need some more definitions.

Definition 2.17

Let R be a finite set of Boolean normal polynomials. A rewriting rule on Boolean polynomial $\rightarrow_R$ is defined as follows. $X \rightarrow_R Y$ iff $X \rightarrow_Z Y$ for some $Z \in R$. $\stackrel{*}{\rightarrow}_R$ is defined as a reflexive and transitive closure of $\{\rightarrow_R, \rightarrow_\times, \rightarrow_+\}$ and $\stackrel{*}{\leftrightarrow}_R$ as a symmetric, reflexive and transitive closure of $\rightarrow_R$.

Definition 2.18

Let X and Y be arbitrary Boolean polynomials such that $X \rightarrow_{A \oplus Z} Y$ for $A \oplus Z \in E_i$. We associate the rewriting $X \rightarrow_{A \oplus Z} Y$ with a triple $(\{X, X\}, A, Z)$, where $\{X, X\}$ is a multiset. Similarly, we associate the rewriting $X \rightarrow_{A \oplus Z} Y$ for $A \oplus Z \in R_i$ with a triple $(\{X\}, A, Z)$. We also associate the rewriting $X \rightarrow_\times Y$ or $X \rightarrow_+ Y$ with a triple $(\{X\}, \bullet, \bullet)$, where $\bullet$ is a special constant. We introduce an ordering on the above triples defined as follows. The first component is compared by the multiset ordering induced by the ordering on Boolean polynomials and the second or third component is compared as a Boolean monomial or Boolean normal polynomial, respectively. We

define $\bullet$ as bigger than any Boolean monomial and any Boolean normal polynomial. Finally, we define an ordering on triple lexicographically by the components. We denote this ordering $\geq$. Note that this is well-founded. We also consider this ordering as an ordering on rewritings.

Definition 2.19

Let X and Y be arbitrary Boolean polynomials. *A proof* of $X \overset{*}{\leftrightarrow}_{E_i \cup R_i} Y$ is a sequence $\Xi_1, \Xi_2, \ldots, \Xi_m$ of rewritings such that each Ξ_j is a rewriting from X_j to X_{j+1} or from X_{j+1} to X_j where $X_0 = X$ and $X_m = Y$. Note that there might be many proofs of $X \overset{*}{\leftrightarrow}_{E_i \cup R_i} Y$ in general. We define an ordering on proofs as the multiset ordering induced by the ordering on rewritings defined above. Note that this is also well-founded.

Note that the definition of the ordering does not depend on i. Therefore, we can compare a proof of $X \overset{*}{\leftrightarrow}_{E_i \cup R_i} Y$ and a proof of $X \overset{*}{\leftrightarrow}_{E_j \cup R_j} Y$, even if i and j are different.

Lemma 2.20

The equivalence relation, $\overset{*}{\leftrightarrow}_{E_i \cup R_i}$, is the same for every i. Moreover, if a proof, Ψ, of $X \overset{*}{\leftrightarrow}_{E_i \cup R_i} Y$ is given, we can construct a proof, Φ, of $X \overset{*}{\leftrightarrow}_{E_j \cup R_j} Y$ for each $j > i$ such that $\Psi \geq \Phi$.

Proof: It is enough to show the following.

(i) If a proof, Φ, of $X \overset{*}{\leftrightarrow}_{E_{i+1} \cup R_{i+1}} Y$ is given, we can construct a proof, Ψ, of $X \overset{*}{\leftrightarrow}_{E_i \cup R_i} Y$.

(ii) If a proof, Ψ, of $X \overset{*}{\leftrightarrow}_{E_i \cup R_i} Y$ is given, we can construct a proof, Φ, of $X \overset{*}{\leftrightarrow}_{E_{i+1} \cup R_{i+1}} Y$ such that $\Psi \geq \Phi$.

The first claim is shown by checking that the rule, $\rightarrow_X$, for X, which is a new Boolean normal polynomial added to $E_{i+1} \cup R_{i+1}$ by Rule 1–8, can be simulated by a combination of $\rightarrow_\times, \rightarrow_+$ and the rules in $E_i \cup R_i$. The second claim is shown similarly, but in this case it must be verified that the simulation generates the same proof or less than it. Later, we will show such simulation in several interesting cases. However, since such simulation is a routine in general, we omit details. ∎

(We have this property since we discussed a rewriting rule, $\rightarrow$, on Boolean polynomials. Note that this property does not hold for $\Rightarrow$ on normal polynomials.)

We use the simple notation, $\overset{*}{\leftrightarrow}$, for the above equivalence relation, $\overset{*}{\leftrightarrow}_{E_i \cup R_i}$, since it does not depend on i.

Before going to the next step, we will show several lemmata.

Lemma 2.21

Let A be a monomial and X_1, X_2, and Z polynomials. Then there exists a polynomial U such that $Z + AX_1 \stackrel{*}{\rightarrow}_{(X_1+X_2)\downarrow} U$ and $Z + AX_2 \stackrel{*}{\rightarrow}_{(X_1+X_2)\downarrow} U$. We denote this situation simply

$$Z - AX_1 \stackrel{*}{\rightarrow}_{(X_1+X_2)\downarrow} \stackrel{*}{\leftarrow} Z + AX_2.$$

Proof: If $(X_1 + X_2)\downarrow = 0$, clearly $X_1\downarrow = X_2\downarrow$. Then,

$$Z + AX_1 \stackrel{*}{\rightarrow} Z + A(X_1\downarrow) = Z + A(X_2\downarrow) \stackrel{*}{\leftarrow} Z + AX_2,$$

where $\stackrel{*}{\rightarrow}$ denotes application of $\rightarrow_\times$ and $\rightarrow_+$, performed a finite number of times. If $(X_1 + X_2)\downarrow \neq 0$, then let $(X_1 + X_2)\downarrow = C \oplus W$, then, either $C \in X_1\downarrow$ or $C \in X_2\downarrow$. Without loss of generality, we can assume that $C \in X_1$. Let $X_1\downarrow = C \oplus X$. Then,

$$Z + AX_1 \stackrel{*}{\rightarrow} Z + AX_1\downarrow \rightarrow_{C\oplus W} Z + A(W + X) \stackrel{*}{\rightarrow} Z + A((W + X)\downarrow)$$

$$Z + AX_2 \stackrel{*}{\rightarrow} Z + A(X_2\downarrow).$$

It is clear that $(W + X)\downarrow = X_2\downarrow$. ∎

Lemma 2.22

Suppose that Rule 1, 3, 4, or 5 is applied in the i-th step and a polynomial in E_i or R_i, say X, is eliminated. Any rewriting using $\rightarrow_X$ can be replaced by a smaller proof in $E_{i+1} \cup R_{i+1}$.

Proof: The lemma is clear for Rule 5 by the definition of the ordering. We show the lemma only for Rule 4. The proof is almost the same for Rules 1 and 3, and much easier. Let $X = AB \oplus X'$ for some B, A, and X' such that $B \oplus Z \in R_i$ and $A \neq \emptyset$. Then $(AZ + X')\downarrow \in E_{i+1}$. If $P \rightarrow_{AB\oplus X'} Q$, P is of form $W + CAB$ and Q is of form $W + CX'$. Therefore, by the above lemma,

$$P = W + CAB \rightarrow_{B\oplus Z} W + CAZ \stackrel{*}{\rightarrow}_{(AZ+X')\downarrow} \stackrel{*}{\leftarrow} W + CX' = Q.$$

Finally, let us verify that this proof is smaller than the original proof, $P \rightarrow_{AB\oplus X'} Q$. The first rewriting, $P \rightarrow_{B\oplus Z} W + CAZ$, is smaller, since $B < AB$. Therefore, the whole proof is smaller, since the other rewritings are clearly smaller. ∎

Lemma 2.23

Let X and Y be arbitrary Boolean polynomials, and $\Xi_1, \Xi_2, \ldots, \Xi_m$ a minimal proof of $X \stackrel{*}{\leftrightarrow} Y$.

(i) There is no rewriting in it which uses a rule in some E_i.

(ii) There is no j such that Ξ_{j-1} is a rewriting from X_j to X_{j-1} and Ξ_j is a rewriting from X_j to X_{j+1} (we denote this situation $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$).

Proof of (i): Suppose some Ξ_i is a rewriting, $X \rightarrow_Z Y$, for Z in some E_j. By condition (i) of the definition of the algorithm, Z is eliminated at some stage, $k > j$. By the above lemma, there is a proof of $X \stackrel{*}{\leftrightarrow} Y$ in $E_k \cup R_k$ which is less than Ξ_i. This contradicts the minimality.

Proof of (ii): Also by the above lemma, if Ξ_i is a rewriting, $X \rightarrow_Z Y$, for Z in some R_j, Z should be in R^∞. Suppose we have $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$. There are several possibilities.

Case 1: Both reductions are $\rightarrow_\times$ or $\rightarrow_+$. In this case, $X_{j-1}{\downarrow} = X_{j+1}{\downarrow}$. Therefore, $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$ can be replaced by

$$X_{j-1} \stackrel{*}{\rightarrow} X_{j-1}{\downarrow} = X_{j+1}{\downarrow} \stackrel{*}{\leftarrow} X_{j+1},$$

which is easily verified to be less than $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$. This contradicts the minimality.

Case 2: One reduction is $\rightarrow_\times$ and the other $\rightarrow_{R^\infty}$. We can assume $X_{j-1} \leftarrow_\times X_j \rightarrow_{R^\infty} X_{j+1}$. There are three subcases.

Subcase 1: $X_j = P + aaA$, $X_{j-1} = P + aA$, $P \rightarrow_{R^\infty} P'$, and $X_{j+1} = P' + aaA$

Subcase 2: $X_j = P + aaA$, $X_{j-1} = P + aA$, $A \rightarrow_{R^\infty} S$, and $X_{j+1} = P + aaS$

SubcaU0se 3: $X_j = P + BaaA$, $X_{j-1} = P + BaA$, $aA \oplus S \in R^\infty$, and $X_{j+1} = P + BaS$

We consider only Subcase 3. The others are much simpler. Since $aA \oplus S \in R^\infty$, its self-critical pair $(aS + S){\downarrow} \in E_k$ for some k by condition (ii) of the definition of the algorithm. Then $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$ can be replaced by

$$X_{j-1} = P + BaA \rightarrow_{aA \oplus S} P + BS \stackrel{*}{\rightarrow}_{(aS+S){\downarrow}} \stackrel{*}{\leftarrow} P + BaS = X_{j+1},$$

which is easily verified to be less than $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$. This contradicts the minimality.

Case 3: One reduction is $\rightarrow_+$ and the other $\rightarrow_{R^\infty}$. This case is handled much more easily.

Case 4: Both reductions are $\rightarrow_{R^\infty}$. There are two subcases.

Subcase 1: $X_j = P + A + B$, $A \rightarrow_{R^\infty} S$, $X_{j-1} = P + S + B$, $B \rightarrow_{R^\infty} T$, and $X_{j+1} = P + A + T$

Subcase 2: $X_j = P + DABC$, $AB \oplus S, AC \oplus T \in R^\infty$, $X_{j-1} = P + DCS$, and $X_{j+1} = P + DBT$ where $A \cap B = \emptyset$

Subcases 1 and 2 are easy. For Subcase 3, since $AB \oplus S, AC \oplus T \in R^\infty$, their critical pair $(CS + BT)\downarrow \in E_k$ for some k by condition (ii) of the definition of the algorithm. Then $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$ can be replaced by

$$X_{j-1} = P + DCS \overset{*}{\underset{(CS+BT)\downarrow}{\leftrightarrow}} P + DBT = X_{j+1},$$

which is easily verified to be less than $X_{j-1} \leftarrow X_j \rightarrow X_{j+1}$. This contradicts the minimality. ∎

Lemma 2.24

$\{\rightarrow_X \mid X \in R^\infty\} \cup \{\rightarrow_*, \rightarrow_+\}$ is a confluent and terminating rewriting system on Boolean polynomials for equivalence relation $\overset{*}{\leftrightarrow}$.

Proof: Confluence is clear from the above lemma and its proof. Termination is Corollary 2.5. ∎

Lemma 2.25

Let S and T be arbitrary Boolean normal polynomials such that $S \overset{*}{\leftrightarrow} T$, then there is a Boolean normal polynomial, Z, such that $S \overset{*}{\Rightarrow}_{R^\infty} Z$ and $T \overset{*}{\Rightarrow}_{R^\infty} Z$.

Proof: Let Z be the normal form of S and T by $\{\rightarrow_X \mid X \in R^\infty\} \cup \{\rightarrow_\times, \rightarrow_+\}$. Since the rewriting system is confluent and terminating, whichever order we take for applying rewriting rules, we finally reach Z from S or T. Apply $\rightarrow_\times$ or $\rightarrow_+$ as far as possible in the reductions from S and T. Then we get reductions $S \overset{*}{\Rightarrow}_{R^\infty} Z$ and $T \overset{*}{\Rightarrow}_{R^\infty} Z$. ∎

Lemma 2.26

The same statement as the above lemma holds for some R_i instead of R^∞.

Proof: Since E_0 is finite, only a finite number of Boolean variables appear in the algorithm. Therefore, there are only a finite number of Boolean normal polynomials, hence R^∞ is finite. Therefore, there exists some R_i such that $R^\infty \subseteq R_i$ by definition of R^∞. Clearly the assertion holds for this R_i. ∎

Proof of the last statement of the definition of the algorithm:

Take i such that the above lemma holds. Since any X in E_i is reduced to 0 by $\Rightarrow_{R_i}$, by applying Rules 1 and 2 several times, say k-times, E_{i+k} will be empty. Note that the above lemma also holds for R_{i+k}. To complete the proof, it suffices to show the next lemma. ∎

Lemma 2.27

Let I be an ideal generated by a finite set, E, of Boolean normal polynomials. Then for each Boolean normal polynomial, X and Y,

$$X \equiv Y \pmod{I} \quad \text{iff} \quad X \overset{*}{\leftrightarrow}_E Y.$$

Proof:

(if) It suffices to check the following. For each Boolean polynomial, X and Y, if $X \rightarrow_Z Y$ for Z in E, then $X\downarrow \equiv Y\downarrow \pmod{I}$. Let $X = S + BA, Y = S + BW$ and $Z = A \oplus W$, then $(X + Y)\downarrow = (BA + S + BW + S)\downarrow = (B(A \oplus W))\downarrow = (BZ)\downarrow \in I$.

(only if) Suppose $X + Y = P_1X_1 + P_2X_2 + \ldots + P_nX_n$ for $P_1, P_2, \ldots, P_n \in E$. Let $Z_i = Y + P_1X_1 + P_2X_2 + \ldots + P_iX_i$ for each $0 \leq i \leq n$. Then $Z_{i-1} \overset{*}{\leftrightarrow}_{P_i} Z_{i-1} + P_iX_i = Z_i$. Combining these proofs obtains a proof of $X = Z_0 \overset{*}{\leftrightarrow}_E Z_n = Y$. ∎

This completes the proof of Theorem 2.8.

REFERENCES

[Bachmair 86] Bachmair, L., Dershowitz, N., and Hsiang, J.: *Ordering for equational proof,* Proc. Symp. Logic in Computer Science, Cambridge, Massachusetts (June 1986)

[Buchberger 83] Buchberger, B.: *Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory,* Technical Report, CAMP-LINTZ (Nov. 1983)

[Dershowitz 79] Dershowitz, N. and Manna, Z.: *Proving termination with multiset orderings,* Comm. ACM 22, pp. 465-467 (1979)

[Dickson 13] Dickson, L. E.: *Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors,* Am. J. of Math., vol. 35, pp. 413-426 (1913)

[Huet 80] Huet, G. and Oppen, D. C.: *Equations and Rewrite Rules: a survey,* Formal Language: Perspectives and Open Problems Academic Press, pp. 349-405 (1980)

[Knuth 70] Knuth, D. E. and Bendix, P. B.: *Simple word problems in universal algebras,* Computational problems in abstract algebra, Pergamon Press, Oxford (1970)

[Waerden 37] van der Waerden, B. L.: *Moderne Algebra I,* Berlin-Leipzig (1937)

[Waerden 40] van der Waerden, B. L.: *Moderne Algebra II,* Berlin-Leipzig (1940)