October 10, 2007
RT0754
Computer Science; Mathematics 10 pages

Research Report

A Formal Study of Algebraic Constraint

Issei Yoshida

IBM Research, Tokyo Research Laboratory
IBM Japan, Ltd.
1623-14 Shimotsuruma, Yamato
Kanagawa 242-8502, Japan

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

A Formal Study of Algebraic Constraint

Issei Yoshida*

Abstract

We present a model for computation of algebraic constraint. An algebraic constraint is defined to be a boolean formula of equations in which every equation is expressed by a polynomial over a field, and hence such constraint may contain negation of an equation, that is, a form $f \neq 0$ where f is some polynomial. Algebraic constraint appears in many application fields of data analysis such as study of geometries, computer aided design, robotics and mechanics. It is well-known that we can describe negations in a form of equations by using slack variables, but traditional approaches assume the same number of slack variables as that of negations. This means the dimension of the ambient space, in which the targeted manifold is embedded, depends on the number of negations used in the constraint, hence the dimension is not intrinsic in the constraint.

We construct an algebraic model that enables to describe an arbitrary boolean formula including multiple negations by using only one slack variable. Also the model provides boolean operations that commute with algebraic operations of polynomials in a natural way, in which we introduce a kind of semiring and its operations in order to make do with one slack variable. To use one slack variable means that we can always consider constraints on the ambient space of the same dimension $(n + 1)$ where n is the number of original variables. We present our approach to construct this model and show some important properties of it.

Keywords: Algebraic constraint, algebraic geometry, transformation of boolean formula, DNF

1 Introduction

In algebraic geometry, an algebraic set V in k^n is defined as a set of common zeroes of a (finite) set of polynomials $f_1, ..., f_m \in R = k[x_1, \cdots, x_n]$, where k is a field and R is a polynomial ring of dimension n over k. For example, $y - x^2 \in R = \mathbb{R}[x, y]$ corresponds to a parabola in the real plane $\mathbb{R}^2$. Given a set of polynomials, it is in general very difficult to compute the corresponding algebraic set and calculate some properties of it. For the purpose of this, various methods including Gröbner basis[2] and their applications are actively studied.

Although the main target of algebraic geometry is such "closed" algebraic sets, sometimes open constraint, that is, $f \neq 0$ for some $f \in R$, is also important in application. Such application fields include study of geometries[1][5], computer aided design, robotics and mechanics[7][8]. Taking study of geometry for example, on the real plane $\mathbb{R}^2$, the condition that two points (x_1, y_1) and (x_2, y_2) are distinct can be described as "$x_1 \neq x_2$ or $y_1 \neq y_2$ where negation naturally appears.

We cannot apply methods for computation of closed sets to these open constraints directly. Instead, it is well-known that we can use slack variable to convert negation to equation as follows[3]: By using a new variable z that is algebraically independent of $x_1, \cdots, x_n$, the condition on a point $x = (x_1, \cdots, x_n)$ that $f(x_1, \cdots, x_n) \neq 0$ is transformed into the condition on a point x that there exists z such that $1 - zf(x_1, \cdots, x_n) = 0$.

As far as we know, the previous works assume that the given constraint is a boolean expression of conjunction of multiple equations or negations, and they use as many slack variables as equations appeared in the given constraint. However, the number of slack variables depends on the number of equations used in the given constraint, and hence it is not intrinsic in the boolean function defined by the constraint. For example, consider two constraints $(f = 0) \wedge (g \neq 0)$ and $((f = 0) \vee (g \neq 0)) \wedge (g \neq 0)$. These are apparently equivalent as a boolean function but the number of negations is different. The number of slack variables may also cause a problem in applications when we compute a constraint having many negations because the number of variables affects heavily the computational time in general.

In this paper we construct an algebraic model to solve these problems. We start with a simple idea for converting a constraint in DNF into a set of equations by using only one slack variable, and then develop a thoery to handle sufficiently general constraints.

In our theory, for every algebraic constraint with n variables, we can express it as an intersection of closed algebraic sets in k^{n+1}. We call k^{n+1} the ambient space of dimension $(n + 1)$. The dimension of the ambient space depends only on the number of variables and does not depend on the expression of the given constraint. Also, we give a coherent correspondence between a set

*IBM Tokyo Research Laboratory, issei@jp.ibm.com

of constraints and a kind of semiring[4] where every element of the semiring represents a set of equations. For example, we can construct the expression of $C_1 \wedge C_2$ in the ambient space from the expressions of C_1 and C_2 in the ambient space in a natural way.

This paper is organized as follows. In section 2 we define algebraic constraint and basic terminologies that are necessary for our discussion. Section 3 describes a way to represent a constraint including negation as a closed set in a certain geometric space, by using only one slack variable, although at this stage we assume that the given constraint is in DNF. Section 4 introduce a constraint semiring in order to extend our technique for constraints that are not necessarily in DNF, and we conclude in section 5 and give some remarks for future research.

2 Representation of Algebraic Constraints

Here we prepare some notations and give the formal definition of algebraic constraint.

2.1 Notation and Preliminaries Let k be a field and $R = k[x_1, \cdots, x_n]$ be a polynomial ring over k. k is arbitrary such as the complex number field $\mathbb{C}$, the real number field $\mathbb{R}$, and so on, but we do not assume any further property on k in most of this paper.

We often write simply $f(x)$ or f for a polynomial $f(x_1, \cdots, x_n) \in R$. For a polynomial $f \in R$, we define $V(f) \stackrel{\text{def}}{=} \{a = (a_1, \cdots, a_n) \in k^n \mid f(a) = 0\}$ and we call $V(f)$ an elementary closed set defined by f. We can identify $V(f)$ with a boolean-valued function on k^n so that $V(f)(a)$ is true for $a \in k^n$ if and only if $f(a) = 0$, that is, $a \in V(f)$.

Similarly, for $f \in R$ we define $D(f) \stackrel{\text{def}}{=} \{a \in k^n \mid f(a) \neq 0\} = k^n \setminus V(f)$ and we call $D(f)$ an elementary open set defined by f. We can identify $D(f)$ with a boolean-valued function on k^n so that $D(f)(a)$ is true for $a \in k^n$ if and only if $f(a) \neq 0$. As a boolean-valued function on k^n, $D(f)$ is equivalent to $\neg V(f)$, which is clear from the definition. We use the symbol $\neg$ for both negation as a boolean operation and operation of taking complementary set of the given set, which will be clear from the context.

Example. Let us consider a simple example. On $\mathbb{R}[x, y]$, $V(x^2 - y^2) = V((x + y)(x - y))$ is a union of two lines on the plane $\mathbb{R}^2$. It works as a boolean-valued function on $\mathbb{R}^2$ such that it returns true if and only if the given point in $\mathbb{R}^2$ is on (at least) one of the two lines.

Note that as a function $V(f)$ and $D(f)$ depend on the underlying field k. For example, $V(x^2 + y^2 + 1)$ always returns false as a boolean-valued function on $\mathbb{R}^2$, while it may return true on $\mathbb{C}^2$ because $(x, y) = (i, 0)$ satisfies $x^2 + y^2 + 1 = 0$ where i is the unit imaginary number in $\mathbb{C}$. ■

2.2 Algebraic Constraints Now we define algebraic constraint.

DEFINITION 2.1. *A literal over R is a symbol f or $\neg f$ for $f \in R$. $\neg f$ is called a negative literal of f, or negation of f simply. We define a set of algebraic constraints $AC(R)$ over R to be a Boolean Algebra generated by literals over R.*

For $C \in AC(R)$, we define $V_c(C)$ to be a subset of k^n obtained by replacing $\wedge$, $\vee$, $\neg$ with $\cap$, $\cup$, complement of the specified subset, and replacing f_{ij}, $\neg h_{ij}$ with $V(f_{ij})$, $D(h_{ij})$ respectively. We call $V_c(C)$ the algebraic constraint set corresponding to C, or simply an algebraic constraint set when C is clear from the context.

It is well-known that any boolean formula can be converted into DNF, so any algebraic constraint is equivalent to a constraint C in the following form:

$$C = \bigvee_{i=1}^{r} \Big((\bigwedge_{j=1}^{p_i} f_{ij}) \bigwedge (\bigwedge_{j=1}^{q_i} \neg h_{ij}) \Big) \tag{2.1}$$

where f_{ij} and h_{ij} are elements of R, and either one of p_i and q_i can be zero. When $r = 0$ we define $C = 0$, and when $r = 1$ we call C a minimal algebraic constraint.

In this case, $V_c(C)$ is given by

$$V_c(C) = \bigcup_{i=1}^{r} \Big(\big(\bigcap_{j=1}^{p_i} V(f_{ij}) \big) \bigcap \big(\bigcap_{j=1}^{q_i} D(h_{ij}) \big) \Big) \tag{2.2}$$

DEFINITION 2.2. *We say that $C \in AC(R)$ is true on k^n when $V_c(C)$ is nonempty, and for a point $a = (a_1, \cdots, a_n) \in k^n$, we say that C is true at a when $a \in V_c(C)$.*

Also, for constraints C and C', we say that C is equivalent to C' when $V_c(C) = V_c(C')$ and denote $C \sim C'$. This means that C is equal to C' as a boolean function.

We omit "over R" or "on k^n" when it is clear from the context.

We consider only DNF constraints for the time being, and call C in (2.1) a DNF constraint. We shall discuss on constraints in the form of arbitrary boolean formulas later. When we consider DNF constraints, $\neg$ occurs only immediately above literals, that is, occurs in the form of $\neg h$ for some $h \in R$.

Before we state a relation between our definition and algebraic geometry, we would give an example of algebraic constraints.

Example. Let $C = ((y - x^2) \wedge \neg(y - x) \wedge \neg(y + x)) \vee ((y - 1) \wedge \neg(x - 1))$ be an algebraic constraint over $\mathbb{R}[x, y]$. The

corresponding algebraic constraint set $V_c(C)$ ($\subset \mathbb{R}^2$) is

$$\begin{aligned} V_c(C) &= \big(V(y-x^2) \cap D(y-x) \cap D(y+x)\big) \\ &\quad \cup \big(V(y-1) \cap D(x-1)\big) \\ &= \{\ (x,y) \in \mathbb{R}^2 \ \big| \\ &\quad (y = x^2 \wedge y \neq x \wedge y \neq -x) \\ &\quad \vee\ (y = 1 \wedge x \neq 1)\ \} \end{aligned}$$

This is union of a parabola and a line except two points $(0,0)$ and $(1,1)$. ■

In algebraic geometry, an *algebraic set* V in k^n is defined as a set of common zeroes of a (finite) set of polynomials $f_1, ..., f_m \in R$, so

$$V = \textstyle\bigcap_{i=1}^{m} V(f_i).$$

We can get this representation by letting $r = 1$ and $q_i = 0$ in (2.2), so our definition of algebraic constraint set is a natural extension of algebraic set.

However, if we consider irreducible decomposition of every f_i, conjunctive normal form (CNF) also seems natural. For example, if $f_i = g_{i1} \times \cdots \times g_{ip_i}$ for some $g_{i1}, \cdots, g_{ip_i} \in R \setminus k$, we easily see that

$$V(f_i) = \textstyle\bigcup_{j=1}^{p_i} V(g_{ij})$$

and hence V can be represented as

$$V = \textstyle\bigcap_{i=1}^{m} \big(\bigcup_{j=1}^{p_i} V(g_{ij})\big)$$

Thus as far as we consider algebraic sets that do not contain negation of an equation, CNF is also available. However, irreducible decomposition of a polynomial itself is non-trivial operation, although some numerical approaches have been studied[9]. We shall see in the following sections that DNF is more appropriate when we consider a model including negation. The following lemma is a key for this.

LEMMA 2.1. *Every algebraic constraint C is equivalent to a constraint C' such that for every i, $f_{ij} = 0$ for some $j \in \{1, \cdots, p_i\}$ and $q_i = 1$ in the right hand of the equation* (2.1).

Proof. For every $i \in \{1, \cdots, r\}$ we transform

$$(\bigwedge_{j=1}^{p_i} f_{ij}) \bigwedge (\bigwedge_{j=1}^{q_i} \neg h_{ij}) \tag{2.3}$$

in the right hand of (2.1) in order to make C'.

If $p_i = 0$ or $f_{ij} \neq 0$ for all j, we add (0) to (2.3) to get

$$\Big\{(0) \bigwedge (\bigwedge_{j=1}^{p_i} f_{ij})\Big\} \bigwedge (\bigwedge_{j=1}^{q_i} \neg h_{ij}) \tag{2.4}$$

and this is equivalent to (2.3) because (0) is transformed to $V(0) = k^n$ and it gives no constraint on the final subset of k^n.

Similarly, if $q_i = 0$, we add $(\neg 1)$ to (2.3) to get

$$(\bigwedge_{j=1}^{p_i} f_{ij}) \bigwedge (\neg 1) \tag{2.5}$$

and this is equivalent to (2.3) because $(\neg 1)$ is transformed to $D(1) = k^n$ and it gives no constraint on the final subset of k^n.

If $q_i > 1$, let $h \stackrel{\text{def}}{=} \prod_{j=1}^{q_i} h_{ij}$ be a product of all of h_{ij} for fixed i. For any point $a = (a_1, \cdots, a_n) \in k^n$,

$$\begin{aligned} & a \in \bigcap_{j=1}^{q_i} D(h_{ij}) \\ \Leftrightarrow\ & h_{ij}(a) \neq 0, \quad 1 \leq \forall j \leq q_i \\ \Leftrightarrow\ & h \neq 0 \end{aligned}$$

Hence $\bigcap_{j=1}^{q_i} D(h_{ij}) = D(h)$ in the expression of $V_c(C)$, so we can replace $\bigwedge_{j=1}^{q_i} \neg h_{ij}$ with $(\neg h)$ wherever $q_i > 1$. ■

By Lemma 2.1, when we consider algebraic constraint sets, we assume that every DNF constraint C is in the form of

$$C = \bigvee_{i=1}^{r} \Big((0) \bigwedge (\bigwedge_{j=1}^{p_i} f_{ij}) \bigwedge (\neg h_i)\Big) \tag{2.6}$$

where $f_{ij}, h_i \in R$.

DEFINITION 2.3. *We say a DNF constraint C is canonical when C is in the form of* (2.6). *For an arbitrary constraint C, the C' in Lemma 2.1 is called the canonicalization of C.*

Example. Let C be the same constraint as in the Example in section 2.2. The canonicalization of C is $((0) \wedge (y - x^2) \wedge \neg(y^2 - x^2)) \vee ((0) \wedge (y-1) \wedge \neg(x-1))$ ■

In the following sections we see how we can relate boolean operations on algebraic constraints with algebraic operations over a (non-boolean) polynomial ring.

3 Representing Algebraic Constraints by Equations

An algebraic constraint is a boolean-valued function and it does not fit in so easily with algebraic operations over R. For example, we cannot even "multiply" two literals f and $\neg g$ because $\neg g$ is not a polynomial any more. First we consider an elementary open set $D(f)$. This represents a condition $f \neq 0$ and this is not an equation. It is well-known that we can adopt a new variable (slack variable) z to come down to:

$$f(x) \neq 0 \Leftrightarrow 1 - zf(x) = 0 \text{ for some } z \in k$$

As far as we know, the previous works use slack variables only for the purpose of turning inequalities into equations, and do not have unifying formulation that takes into consideration both base variables (x_i in $R = k[x_1, \cdots, x_n]$) and slack variables. Also, this simple application of slack variables requires the same number of slack variables as of the elementary open sets that occur in the given constraint.

In the following discussion we show that only one slack variable is sufficient to model general algebraic constraint defined in Section 2. In this section we give a formulation of our idea to do with only one slack variable for DNF, and in the next section we construct an algebraic model to expand our idea to arbitrary constraints.

3.1 Ambient Space for Algebraic Constraints

In order to treat constraints including negations in our algebraic model, we introduce a variable z that is algebraically independent of $x_1, \cdots, x_n$, and consider a little larger space than the original space k^n, that is, k^{n+1}.

Definition 3.1. *Let π be a natural projection map from k^{n+1} to k^n as follows:*

$$\begin{array}{rcl} \pi : k^{n+1} & \longrightarrow & k^n \\ (a_1, \cdots, a_n, c) & \longmapsto & (a_1, \cdots, a_n) \end{array}$$

when we consider $AC(R)$, this k^{n+1} is called the ambient space of $AC(R)$ or simply the ambient space.

We would like to give a subset $W \subset k^{n+1}$ for every $C \in AC(R)$ such that $\pi(W) = V_c(C)$. If such W is found, then

$$\begin{array}{rcl} C \text{ is true on } k^n & \Leftrightarrow & V_c(C) \text{ is nonempty} \\ & \Leftrightarrow & W \text{ is nonempty} \\ C \text{ is true at } a \in k^n & \Leftrightarrow & a \in V_c(C) \\ & \Leftrightarrow & (a, c) \in W \text{ for some } c \end{array}$$

hold, and we can discuss on the ambient space to study the original constraint.

It is very important that such W must be an algebraic set, that is, a subset of k^{n+1} in the form $\{f_1(x) = 0\} \wedge \cdots \wedge \{f_m(x) = 0\}$, in order to apply various methods of algebraic geometry to W. Note that when W is a union of more than one algebraic sets, we can still say that W is an algebraic set because when $W = \bigcup_{i=1}^{r} W_i$ and $W_i = \{f_{i1} = 0\} \wedge \cdots \wedge \{f_{ip_i} = 0\}$, then W is expressed as an intersection of $f_{1j_1} f_{2j_2} \cdots f_{rj_r} = 0$ for all possible $(j_1, \cdots, j_r)$. Hence we call a union of algebraic sets also as an algebraic set in the following discussion. Refer to [2] or [3] for details in terms of algebraic geometry.

The next example shows that canonicalization of a DNF constraint is essential.

Example. For one elementary open set $D(f)$, we can easily find such $W \subset k^{n+1}$, that is,

$$\begin{array}{rcl} W & = & V(1 - zf(x)) \\ & = & \{(a_1, \cdots, a_n, c) \in k^{n+1} \mid 1 - cf(a) = 0\} \end{array}$$

However, for the intersection of two open sets $D(f)$ and $D(g)$, the natural interpretation by the formula $(1 - z_1 f(x) = 0) \cap (1 - z_2 g(x) = 0)$ requires two slack variables z_1 and z_2 and we cannot use a common z here because the value of z completely determines the value of f and g. Instead, by using the fact that $D(f) \cap D(g) = D(fg) \subset k^n$, we can find an appropriate algebraic set $W = V(1 - zf(x)g(x)) \subset k^{n+1}$. ■

The above example motivates us to define an algebraic set in the ambient space as follows.

3.2 Correspondence between Algebraic Constraint Sets and Algebraic Sets in Ambient Space

Definition 3.2. *For a canonical DNF constraint C defined by (2.6), we define an algebraic set $\widetilde{V}_c(C)$ on the ambience space k^{n+1} as*

$$\widetilde{V}_c(C) \stackrel{\text{def}}{=} \bigcup_{i=1}^{r} \Big(\big(\bigcap_{j=1}^{p_i} V(f_{ij}) \big) \bigcap V\big(1 - zh_i\big) \Big) \tag{3.7}$$

When $C = 0$ we define $\widetilde{V}_c(C) = k^{n+1}$, the whole ambient space.

Note that f_{ij} and h_i can be seen as polynomials of $R[z]$ although they do not contain the variable z. The following Proposition 3.1 is the first result of our formulation.

Proposition 3.1. $\pi(\widetilde{V}_c(C)) = V_c(C)$

Proof. Let $a = (a_1, \cdots, a_n) \in V_c(C)$. By the definition of $V_c(C)$, there exists $i_0 \in \{1, \cdots, r\}$ such that

$$a \in \big(\bigcap_{j=1}^{p_{i_0}} V(f_{i_0 j}) \big) \bigcap D(h_{i_0}) \tag{3.8}$$

So the following equations hold.

$$f_{i_0 1}(a) = 0, \cdots, f_{i_0 p_{i_0}}(a) = 0,\ h_{i_0}(a) \neq 0,$$

Let $c \stackrel{\text{def}}{=} 1/h_{i_0}(a)$. c is well-defined because $h_{i_0}(a)$ is nonzero. Take $\widetilde{a} \stackrel{\text{def}}{=} (a_1, \cdots, a_n, c) \in k^{n+1}$. By the construction of c,

$$1 - c \times h_{i_0}(a) = 1 - 1 = 0$$

Hence we have

$$\widetilde{a} \in \Big(\bigcap_{j=1}^{p_{i_0}} V(f_{i_0 j})\Big) \bigcap V\big(1 - z h_{i_0}\big) \tag{3.9}$$

(Note that $f_{ij}(\widetilde{a}) = f_{ij}(a), h_i(\widetilde{a}) = h_i(a)$ because f_{ij} and h_i are polynomials independent of the variable z) So $\widetilde{a} \in \widetilde{V}_c(C)$ and $\pi(\widetilde{a}) = a$. This means $V_c(C) \subset \pi(\widetilde{V}_c(C))$.

Conversely, take $a \in \pi(\widetilde{V}_c(C))$. We can take $c \in k$ such that $\widetilde{a} = (a, c) \in k^{n+1}$ and $\widetilde{a} \in \widetilde{V}_c(C)$. Then there exists $i_0 \in \{1, \cdots, r\}$ such that (3.9) holds. In particular, $1 - c \times h_{i_0}(a) = 0$, which implies $h_{i_0}(a) \neq 0$. Hence we have (3.8) and $\pi(\widetilde{V}_c(C)) \subset V_c(C)$ follows. ■

By Proposition 3.1, we can calculate algebraic constraint sets completely by using algebraic equations on the ambient space, that is, $\widetilde{V}_c(C)$ is a union of sets of common zeros of polynomials, while $V_c(C)$ is not.

The key point is that we use DNF to represent constraints in order to choose one $i_0 \in \{1, \cdots, r\}$ and use only one h_{i_0} in the proof of Proposition 3.1, which enables us to handle general combination of multiple negations with only one variable z as we have already seen.

Example. Consider an algebraic constraint $C = (\neg x) \wedge ((y) \vee (\neg(x^3 - 1)))$ over $k[x, y]$. Let $W = V(1 - zx) \cap (V(y) \cup V(1 - z(x^3 - 1)))$ be a corresponding algebraic subset of k^2. C is not a DNF constraint, and $\pi(W) \neq V_c(C)$ because $(-1, 1) \in V_c(C)$ and $(-1, 1) \notin \pi(W)$. ■

4 Constraint Semiring

In Section 3 we define an algebraic set $\widetilde{V}_c(C)$ in the ambient space that is mapped to the target algebraic constraint set $V_c(C)$ by the projection π. Also, by Proposition 3.1, we can get a set of equations on the ambient space explicitly from a DNF constraint.

In order to obtain the algebraic set for the arbitrarily given constraint C, it may be sufficient to transform C into a DNF constraint C' and apply Proposition 3.1 to C', because C is equivalent to C' and they give the same algebraic constraint set.

However, such C' cannot be uniquely determined and hence it is nontrivial to choose a "good" set of equations on the ambient space, which is necessary for applying tools of algebraic geometry or computational methods such as Gröbner basis. For example, when $C_1 \wedge C_2$ is a constraint and we know a set of algebraic equations for every C_i on the ambient space, how can we know a set of equations for $C_1 \wedge C_2$? Since we use only one slack variable, things are not so simple.

Here we study a model that enables us to calculate a set of equations for $C_1 \wedge C_2$ from *equations* for C_1 and C_2. We consider operations $\wedge, \vee, \neg$ for sets of equations on the ambient space, and we shall show that these operations commute with $\wedge, \vee, \neg$ for algebraic constraints that are not necessarily DNF.

First we define the constrant semiring, a kind of semiring[4] that enables us to compute an algebraic constraint set for an arbitrary constraint that is not necessarily in DNF. Then we give a relation among algebraic constraint, constraint semiring and set of equations on the ambient space.

4.1 Motivation and Definition

It is known that a set of all of ideals of R is a commutative semiring with its natural operations of addition and multiplication of ideals[4]. However, addition of ideals is not suitable for our model because the slack variable z cannot be shared with conjunction of constraints. For example, consider an algebraic set W corresponding to an algebraic constraint $(\neg x) \wedge (\neg(x-1))$. The corresponding ideal of W should be $I = (1 - zx, 1 - z(x - 1))$ because $(\neg x)$ and $(\neg(x-1))$ correspond to ideals $(1-zx)$ and $(1-z(x-1))$ respectively, but as an ideal $I = (1)$ and $V(I) \subset \mathbb{R}^2$ is an empty set, and hence $\pi(V(I)) \subset \mathbb{R}$ is also empty. This is different from our expected result $(x \neq 0 \wedge x \neq 1)$, a whole line except two points in $\mathbb{R}$.

We tackle this by defining addition and multiplication on a set of generators of an ideal that are different from the normal operations of a set of all of ideals of R.

Definition 4.1. *Let $R[z] = k[x_1, \cdots, x_n, z]$ be a polynomial ring over k with $(n+1)$-variables. We define $\mathcal{I}(R[z])$ a set of finite sets of elements of $R[z]$ such that every element $I \in \mathcal{I}(R[z])$ contains $0 \in R[z]$. We say that $I \in \mathcal{I}(R[z])$ is a minimal constraint generator, or $I \in \mathcal{MCG}(R)$, when I is in the form of*

$$I = \{0, f_1(x), \cdots, f_r(x), 1 - zh(x)\} \tag{4.10}$$

where $f_i, h \in R \setminus \{0\}$ and $r \geq 0$.

In particular, $\{0, f, 1 - z\}$ is denoted by $L(f)$, and $\{0, 1 - zh\}$ is denoted by $\neg L(h)$.

We construct the target semiring by extending $\mathcal{MCG}(R)$ with operations of addition and multiplication.

Definition 4.2. *For $I = \{0, f_1(x), \cdots, f_r(x), 1 - zh_1(x)\}$ and $J = \{0, g_1(x), \cdots, g_s(x), 1 - zh_2(x)\} \in \mathcal{MCG}(R)$, we define multiplication $I \times J$, or simply IJ, as*

$$\begin{aligned} IJ \stackrel{\text{def}}{=} & \{0\} \cup \{f_1, \cdots, f_r\} \\ & \cup \{g_1, \cdots, g_s\} \cup \{1 - zh_1h_2\} \end{aligned} \tag{4.11}$$

It is clear that $IJ \in \mathcal{MCG}(R)$ by this construction, and it is easy to see that $\mathcal{MCG}(R)$ is a commutative

monoid with the identity element $\mathbf{1} \stackrel{\text{def}}{=} \{0, 1-z\}$ with respect to multiplication. Note that we define minimal constraint generator so that it corresponds to minimal algebraic constraint(See subsection 2.1).

Next we give a definition of the constraint semiring.

DEFINITION 4.3. *Let $\mathcal{CS}(R)$ be a free monoid over $\mathcal{MCG}(R)$ with respect to addition. Every element of $\mathcal{CS}(R)$ is expressed as a formal sum $I_1 + \cdots + I_m$ for some $I_i \in \mathcal{MCG}(R)$.*

The zero element is intrinsically defined, and we give a structure of set to the zero element of $\mathcal{CS}(R)$ as $\mathbf{0} \stackrel{\text{def}}{=} L(1) = \{0, 1, 1-z\}$. that is, we define $I + \mathbf{0} = I$ for any $I \in \mathcal{CS}(R)$.

PROPOSITION 4.1. *For $I = I_1 + \cdots + I_r$, $J = J_1 + \cdots + J_s \in \mathcal{CS}(R)$, we define multiplication IJ to be $\sum_{1 \le i \le r, 1 \le j \le s}(I_i J_j)$. Then $\mathcal{CS}(R)$ is a semiring with respect to this multiplication and addition.*

Proof. Since $I_i J_j$ is again a minimal constraint generator, $\mathcal{CS}(R)$ is closed under multiplication defined here. We can easily check that $\mathcal{CS}(R)$ satisfies the following properties of semiring by straightforward calculation.

$$I + J = J + I, I + (J + K) = (I + J) + K,$$

$$IJ = JI, \quad I(JK) = (IJ)K,$$

$$I(J + K) = IJ + IK,$$

$$I + \mathbf{0} = I, \quad I \times \mathbf{1} = I$$

where $I, J, K \in \mathcal{CS}(R)$ are arbitrary elements and $\mathbf{0}, \mathbf{1}$ are defined above. ■

DEFINITION 4.4. *We call $\mathcal{CS}(R)$ the constraint semiring over R. An element of $\mathcal{CS}(R)$ is called a constraint generator over R.*

We defined the constraint semiring so that addition and multiplication of two generator I, J correspond to taking $\vee$ and $\wedge$ of the two constraints respectively. Note that for $\mathbf{0} = \{0, 1, 1-z\}$, $I \times \mathbf{0} = \mathbf{0}$ does not hold in general. However, in actual calculation of algebraic constraint sets, we can replace $I \times \mathbf{0}$ with $\mathbf{0}$ as we can see below.

DEFINITION 4.5. *Let $I = I_1 + \cdots + I_r$ be a constraint generator where every I_i is minimal. We define $S(I)$, a subset of $\mathcal{I}(R[z])$, to be*

$$S(I) \stackrel{\text{def}}{=} \{\ f_1 \cdots f_r \mid f_i \in I_i,\ 1 \le i \le r\ \} \tag{4.12}$$

where $f_1 \cdots f_r$ is a normal product of f_i in $R[z]$.

Note that we identify $S(mI)$ with $S(\overbrace{I + \cdots + I}^{m})$ for a constraint generator I and a positive integer m. Also note that the definition of $S(I)$ depends on representation of $I = I_1 + \cdots + I_r$, but $S(I)$ is a free monoid generated by elements of $\mathcal{MCG}(R)$, so the representation is unique and $S(I)$ is well-defined.

Intuitively, for $I = I_1 + \cdots + I_r$, $S(I)$ gives a set of equations on the ambient space for a DNF that corresponds to (2.1). In the following discussion we will give the correspondence between an algebraic constraint and a set of equations on the ambient space in a natural way.

4.2 Algebraic Constraint Set for General Constraints We analyze arbitrary algebraic constraints that are not necessarily in DNF by using the constraint semiring. First we study constraints in negation normal form(NNF), that is, a form where negation occurs only immediately above literals.

DEFINITION 4.6. *Let $C \in AC(R)$ be an algebraic constraint in NNF. We define $I(C) \in \mathcal{CS}(R)$ by converting C as follows.*

- *Replace a literal $f \in R$ with $L(f) = \{0, f, 1-z\} \in \mathcal{CS}(R)$ in C. Note that when $f = 0$ then $L(f) = \mathbf{1}$,*
- *Replace a non-zero negative literal $\neg h$ ($h \in R \setminus \{0\}$) with $\neg L(h) = \{0, 1-zh\} \in \mathcal{CS}(R)$ in C ,and $\neg\mathbf{0}$ with $\mathbf{0} = \{0, 1, 1-z\}$, and*
- *Replace $\wedge$, $\vee$ with $\times$, $+$ respectively in C.*

$I(C)$ is an element of $\mathcal{CS}(R)$, which is clearly from its construction, and $S(I(C))$ is a finite subset of $R[z]$. It is easy to see that every element of $S(I(C))$ is expressed as $f = f_1 \times \cdots \times f_r \times (1 - zh_1) \times \cdots \times (1 - zh_s)$ for some $f_i, h_j \in R$ that occur in C.

We define $\widetilde{V}_c(C)$ to be a subset of k^{n+1} as

$$\widetilde{V}_c(C) \stackrel{\text{def}}{=} V(S(I(C))) = \bigcap_{f \in S(I(C))} V(f) \tag{4.13}$$

that is, $\widetilde{V}_c(C)$ is a set of common zeros of polynomials in $S(I(C))$.

First we check that this definition is consistent with the previous definition of $\widetilde{V}_c(C)$.

LEMMA 4.1. *For a canonical DNF constraint $C \in AC(R)$, $\widetilde{V}_c(C)$ in Definition 4.6 is equal to the one* (3.7) *in Definition 3.2.*

Proof. Let C be a constraint in the form of (2.6). By the definition of $I(C)$, we get

$$
\begin{aligned}
I(C) &= \sum_{i=1}^{r}\Big(\mathbf{1}\times\prod_{j=1}^{p_i}\{0,f_{ij},1-z\}\times\{0,1-zh_i\}\Big)\\
&= \sum_{i=1}^{r}\{0,f_{i1},\cdots,f_{ip_i},1-zh_i\} \qquad \cdots(*)
\end{aligned}
$$

By Definition 3.2, for $(a,c)=(a_1,\cdots,a_n,c)\in k^{n+1}$,

$$
\begin{aligned}
&(a,c) \in (3.7)\\
\Longleftrightarrow\ &\exists i_0\in\{1,\cdots,r\}\\
&s.t. \quad (a,c)\in\big(\bigcap_{j=1}^{p_{i_0}}V(f_{i_0j})\big)\bigcap V\big(1-zh_{i_0}\big)\\
\Longleftrightarrow\ &\exists i_0\in\{1,\cdots,r\}\\
&s.t. \quad f_{i_0j}(a)=0\ (1\le j\le p_{i_0}),\ 1-ch_{i_0}(a)=0
\end{aligned}
$$

This implies that for every element $f\in S(I(C))$, $f(a,c)=0$ because $f=F_1\cdots F_r$ where F_i is either one of f_{ij} or $1-zh_i$, and the product always contains one of f_{i_0j} or $1-zh_{i_0}$. Hence (3.7) $\subset$ (2.6) holds. Conversely, if such i_0 does not exist, for every i there exists some $\tilde{F}_i\in\{f_{i1},\cdots,f_{ip_i},1-zh_i\}$ such that $\tilde{F}_i(a,c)\neq 0$, so $f\stackrel{\text{def}}{=}\tilde{F}_1\cdots\tilde{F}_r$ also satisfies $f(a,c)\neq 0$, which implies that $(a,c)\notin$ (2.6). ■

What we want to do is to generalize Proposition 3.1. The following Lemma 4.2 is proved by similar techniques used in the proof of Lemma 4.1, but it is a little more complicated.

LEMMA 4.2. *For any constraints $C_1,C_2\in AC(R)$, the following holds.*

$$
\begin{aligned}
&(4.14)\quad \pi(\widetilde{V}_c(C_1\wedge C_2)) = \pi(\widetilde{V}_c(C_1))\cap\pi(\widetilde{V}_c(C_2))\\
&(4.15)\quad \pi(\widetilde{V}_c(C_1\vee C_2)) = \pi(\widetilde{V}_c(C_1))\cup\pi(\widetilde{V}_c(C_2))
\end{aligned}
$$

Proof. We prove only (4.14) here. (4.15) can be proved in a similar manner but it is easier. Let $I(C_1),I(C_2)\in\mathcal{CS}(R)$ and $I(C_1)=I_1+\cdots+I_r$ and $I(C_2)=J_1+\cdots+J_s$ where I_i,J_j are minimal constraint generators. It is clear that $I(C_1\wedge C_2)=I(C_1)\times I(C_2)$ by Definition 4.6, and we have

$$
\begin{aligned}
\widetilde{V}_c(C_1\wedge C_2) &= \bigcap_{f\in S(I(C_1\wedge C_2))}V(f)\\
&= \bigcap_{f\in S(I(C_1)\times I(C_2))}V(f)\\
&= \bigcap_{f\in S(\sum_{1\le i\le r,1\le j\le s}(I_iJ_j))}V(f)
\end{aligned}
$$

Hence

$$
\begin{aligned}
&a = (a_1,\cdots,a_n)\in\pi(\widetilde{V}_c(C_1\wedge C_2))\\
\Leftrightarrow\ &\exists c\in k\ \ s.t.\ (a,c)\in V(f)\\
\text{for}\ \ &\forall f\in S(\sum_{1\le i\le r,1\le j\le s}(I_iJ_j))
\end{aligned}
$$

Now take any $a\in\pi(\widetilde{V}_c(C_1\wedge C_2))$. By definition of $S()$, $f=\prod_{1\le i\le r,1\le j\le s}F_{ij}$ for some $F_{ij}\in I_iJ_j$, so the above condition says that

$$
\begin{aligned}
&(4.16)\quad \exists c\in k\\
&\text{s.t.}\qquad \prod_{1\le i\le r,1\le j\le s}F_{ij}(a,c)=0,\ \forall F_{ij}\in I_iJ_j
\end{aligned}
$$

If there exists $E_{ij}\in I_iJ_j$ for every (i,j) such that $E_{ij}(a,c)\neq 0$, then (4.16) does not hold for $\prod_{1\le i\le r,1\le j\le s}E_{ij}$, which is a contradiction. Hence there exists (i_0,j_0) such that $E(a,c)=0$ for any $E\in I_{i_0}J_{j_0}$.

Since I_{i_0} and J_{j_0} are minimal, we assume that $I_{i_0}=\{0,f_{i_01},\cdots,f_{i_0p_{i_0}},1-zh_{i_0}\}$ and $J_{j_0}=\{0,g_{j_01},\cdots,g_{j_0q_{j_0}},1-zv_{j_0}\}$ where $f_{i_0l},g_{j_0m},h_{i_0},v_{j_0}\in R$. Then $I_{i_0}J_j=\{0,f_{i_01},\cdots,f_{i_0p_{i_0}},g_{j_01},\cdots,g_{j_0q_{j_0}},1-zh_{i_0}v_{j_0}\}$ (Even if $f_{i_0l}=g_{j_0m}$ for some l and m, it does not affect the following discussion). By combining this with the above fact, we get

$$
(*)\begin{cases}
f_{i_0l}(a)=0 & (1\le l\le p_{i_0}),\\
g_{j_0l}(a)=0 & (1\le l\le q_{j_0}),\\
1-ch_{i_0}(a)v_{j_0}(a)=0
\end{cases}
$$

In particular, $h_{i_0}(a)\neq 0$ and $v_{j_0}(a)\neq 0$ hold by the last equality.

Let $c_1\stackrel{\text{def}}{=}1/(h_{i_0}(a))$. Every element f in $S(I(C_1))=S(\sum_{i=1}^{r}I_i)$ is expressed as $\prod_{1\le i\le r}F_i$ for some $F_i\in I_i$. In particular, F_{i_0} is either one of f_{i_0l} for some l or $1-zh_{i_0}$, and in any way $F_{i_0}(a,c_1)=0$ holds by the construction of a and c_1. This means that $f(a,c_1)=0$ for every $f\in S(I(C_1))$, which implies that $(a,c_1)\in\widetilde{V}_c(C_1)$, and hence $a\in\pi(\widetilde{V}_c(C_1))$. Also, we obtain $a\in\pi(\widetilde{V}_c(C_2))$ by replacing $I(C_1)=\sum_{i=1}^{r}I_i$ with $I(C_2)=\sum_{j=1}^{s}J_j$ in the above discussion, so $a\in\pi(\widetilde{V}_c(C_1))\cap\pi(\widetilde{V}_c(C_2))$ follows.

In order to see the converse inclusion, take any $a\in\pi(\widetilde{V}_c(C_1))\cap\pi(\widetilde{V}_c(C_2))$. $a\in\pi(\widetilde{V}_c(C_1))$ means that there exists $c_1\in k$ such that $\prod_{1\le i\le r}F_i(a,c_1)=0$ for every choice of $F_i\in I_i$. This implies that there exists i_0 such that $F_{i_0}(a,c_1)=0$ for all $F_{i_0}\in I_{i_0}$, and hence

$$
\begin{aligned}
f_{i_0l}(a) &= 0\ \ (1\le l\le p_{i_0}),\\
1-c_1h_{i_0}(a) &= 0
\end{aligned}
$$

follows. Similarly, we can choose $c_2\in k$ and j_0 such that

$$
g_{j_0m}(a) = 0\ \ (1\le m\le q_{j_0}),
$$

$$1 - c_2 v_{j_0}(a) \quad = \quad 0$$

hold. Now let $c \stackrel{\text{def}}{=} c_1 c_2$. By the construction of c, $(*)$ holds for (i_0, j_0) and this implies $a \in \pi(\widetilde{V}_c(C_1 \wedge C_2))$. ■

Note that Lemma 4.2 holds for any constraints that are not in NNF. Now we state generalized version of Proposition 3.1 for NNF.

PROPOSITION 4.2. *Let $C \in AC(R)$ be an algebraic constraint in NNF. Then*

$$\pi(\widetilde{V}_c(C)) = V_c(C) \tag{4.17}$$

holds.

Proof. Lemma 4.2 shows that $\wedge$ and $\vee$ commute with operation $\pi(\widetilde{V}_c(\ ))$. Since C is in NNF, the problem is reduced to prove the case when $I(C) \in \mathcal{MCG}(R)$, which follows from Proposition 3.1 and Lemma 4.1. ■

Example. Consider $C = (\neg x) \wedge (y \vee (\neg(x^3 - 1))) \in AC(R)$. C is not in DNF but in NNF.

$$\begin{aligned} I(C) &= (\neg L(x)) \times (L(y) + \neg L(x^3 - 1)) \\ &= ((\neg L(x)) \times L(y)) + ((\neg L(x)) \times (\neg L(x^3 - 1))) \\ &= \{0, y, 1 - zx\} + \{0, 1 - zx(x^3 - 1)\} \\ &= \{0, y(1 - zx(x^3 - 1)), (1 - zx)(1 - zx(x^3 - 1))\} \end{aligned}$$

the right hand of the equation gives a set of equations for $\widetilde{V}_c(C)$, and Proposition 4.2 ensures that, by projecting $\widetilde{V}_c(C)$ on k^n via π, we get the desired $V_c(C) = \{x \neq 0\} \cap (\{y = 0\} \cup \{x^3 \neq 1\})$.

Note that the order of operations is essential in computation. For example, if we compute $(L(y) + \neg L(x^3 - 1))$ first, then the result is not a sum of minimal constraint generators and cannot perform multiplication with $\neg L(x)$ because it is not defined. This corresponds to taking $S(I)$ for a constraint generator I *after* calculating multiplication in $\mathcal{CS}(R)$. ■

4.3 Negation and Main Result We define the operation of general negation on $\mathcal{CS}(R)$, and extend Proposition 4.2 to general constraints. We give our main result in Proposition 4.3 and introduce a few properties of the correspondence between constraints and algebraic sets on the ambient space.

DEFINITION 4.7. *For a minimal constraint generator $I = \{0, f_1, \cdots, f_r, 1 - zh\}$,we define $\neg I$ as*

$$\begin{aligned} \neg I &\stackrel{\text{def}}{=} \{0, 1 - zf_1\} + \cdots + \{0, 1 - zf_r\} + \{0, h, 1 - z\} \\ &= (\neg L(f_1)) + \cdots + (\neg L(f_r)) + L(h) \end{aligned}$$

$\neg I$ is again a finite sum of minimal constraint generators, so $\neg I \in \mathcal{CS}(R)$.

For a general constraint generator $I = I_1 + \cdots + I_m \in \mathcal{CS}(R)$, $I_i \in \mathcal{MCG}(R)$, we define $\neg I$ as

$$\neg I \quad \stackrel{\text{def}}{=} \quad (\neg I_1) \times \cdots \times (\neg I_m) \tag{4.18}$$

where every I_i is defined above. For an algebraic constraint C, we define $I(C)$ by converting C following the rules in Definition 4.6 and calculateing negation by the above rules. Note that $I(C)$ is also an element of $\mathcal{CS}(R)$ after resolving negation.

From the above definition, $\neg(I_1 + I_2) = (\neg I_1) \times (\neg I_2)$ always holds for $I_1, I_2 \in \mathcal{CS}(R)$. Although $\neg(I_1 I_2) = (\neg I_1) + (\neg I_2)$ does not hold in general, the next lemma claims that the above definition is natural.

LEMMA 4.3. *For an algebraic constraint $C \in AC(R)$,*

$$\pi(\widetilde{V}_c(\neg C)) = \neg \pi(\widetilde{V}_c(C)) \tag{4.19}$$

holds.

Note that $\neg\pi(\widetilde{V}_c(C)) = k^n \setminus \pi(\widetilde{V}_c(C))$. We prove this lemma similarly to the proof of Lemma 4.2, but $S(I(\neg C))$ is more complicated than $S(I(C))$ so we have to look at its construction carefully.

Proof. For the given C, let $I(C) = I_1 + \cdots + I_r$, $I_i = \{0, f_{i1}, \cdots, f_{ip_i}, 1 - zh_i\}$, $f_{ij}, h_i \in R$. First we analyze $\widetilde{V}_c(C)$ and $\widetilde{V}_c(\neg C)$. By definition, $\widetilde{V}_c(C) = \bigcup_{f \in S(\sum_i I_i)} V(f)$ where every $f \in S(\sum_i I_i)$ is expressed as $f = F_{1j_1} F_{2j_2} \cdots F_{rj_r}$ for some $F_{ij_i} \in I_i$.

As for $\widetilde{V}_c(\neg C)$,

$$\begin{aligned} I(\neg C) &= (\neg I_1) \times \cdots \times (\neg I_r) \\ &= \prod_{i=1}^{r} \left(\neg L(f_{i1}) + \cdots + \neg L(f_{ip_i}) + L(h_i)\right) \\ &= \sum \neg L(f_{i_1 j_{i_1}}) \cdots \neg L(f_{i_m j_{i_m}}) L(h_{i_{m+1}}) \cdots L(h_{i_r}) \qquad (4.20) \\ &= \sum \{0, h_{i_{m+1}}, \cdots, h_{i_r}, 1 - z(f_{i_1 j_{i_1}} \cdots f_{i_m j_{i_m}})\} \qquad (4.21) \end{aligned}$$

where the sum of (4.20) is taken over minimal constraint generators with indices $\{i_1, \cdots, i_r, j_{i_1}, \cdots, j_{i_m}\}$ satisfying $\{i_1, \cdots, i_r\} = \{1, \cdots, r\}, 0 \leq m \leq r, i_1 < \cdots < i_m, i_{m+1} < \cdots < i_r$ and $1 \leq j_{i_k} \leq p_{i_k}$ for $k = 1, \cdots, r$. Note that if $m = 0$ then the corresponding generator does not contain $\neg L(f_{ij})$, so the expression of the generator in (4.21) is $\{0, h_{i_1}, \cdots, h_{i_r}, 1 - z\}$.

And we have $\widetilde{V}_c(\neg C) = \bigcup_{f \in S(I(\neg C))} V(f)$ where every $f \in S(I(\neg C))$ is zero or a product of elements of $\{h_{i_{m+1}}, \cdots, h_{i_r}, 1 - z(f_{i_1 j_{i_1}} \cdots f_{i_m j_{i_m}})\}$ such that one and only one element is chosen from one index $\{i_1, \cdots, i_r, j_{i_1}, \cdots, j_{i_m}\}$.

We shall prove the lemma by showing bidirectional inclusions. Let $a \in \pi(\widetilde{V}_c(\neg C))$. There exists $c \in$

k such that $f(a,c) = 0$ for every $f \in S(I(\neg C))$. Since f is expressed by a product of elements of R as above, as in the discussion of Lemma 4.2, there exists $\{i_1, \cdots, i_r, j_{i_1}, \cdots, j_{i_m}\}$ such that for every $f \in \{0, h_{i_{m+1}}, \cdots, h_{i_r}, 1 - z(f_{i_1 j_{i_1}} \cdots f_{i_m j_{i_m}})\}$, $f(a,c) = 0$ holds. We assume that $i_1 = 1, \cdots, i_r = r$ without loss of generality, so we have

$$(*) \begin{cases} h_i(a) = 0 \ \text{ for } \ m+1 \leq i \leq r, \text{and} \\ 1 - c f_{1j_1}(a) \cdots f_{mj_m}(a) = 0 \end{cases}$$

In particular $f_{ij_i}(a) \neq 0$ for $1 \leq i \leq m$. Let $\tilde{f} \stackrel{\text{def}}{=} f_{1j_1} \times \cdots \times f_{mj_m} \times (1 - zh_{m+1}) \times \cdots \times (1 - zh_r)$. $\tilde{f}$ is an element of $S(I(C))$ and, by $(*)$, $f(a,c') \neq 0$ for any choice of $c' \in k$. This shows that $(a,c') \notin \widetilde{V}_c(C)$, and $a \in \neg\pi(\widetilde{V}_c(C))$ follows.

Conversely, let $a \notin \pi(\widetilde{V}_c(\neg C))$, then for every $c \in k$, there exists $f \in S(I(\neg C))$ such that $f(a,c) \neq 0$. Assume that $a \notin \neg\pi(\widetilde{V}_c(C))$. Then for every i, at least one of $h_i(a) = 0$ or $f_{ij}(a) \neq 0$ $(\exists j)$ hold. By renumbering indices if necessary, without loss of generality we assume that

$$(**) \begin{cases} f_{i1}(a) \neq 0 \ \text{ for } \ 1 \leq i \leq t, \text{and} \\ h_i(a) = 0 \ \text{ for } \ t+1 \leq i \leq r \end{cases}$$

hold for some $0 \leq t \leq r$.

Now consider the minimal constraint generator $J \stackrel{\text{def}}{=} \{0, h_{t+1}, \cdots, h_r, 1 - z(f_{11} \cdots f_{t1})\}$ (if $t = 0$, then use $1 - z$ instead of $1 - z(f_{11} \cdots f_{t1})$), which must appear in (4.21). Take $c \stackrel{\text{def}}{=} 1/(f_{11}(a) \cdots f_{t1}(a))$ if $t \geq 1$ or $c = 1$ if $t = 0$. By definition of J and $(**)$, it follows that $f(a,c) = 0$ for every $f \in J$. That implies $f(a,c) = 0$ for every $f \in S(I(\neg C))$, which contradicts our hypothesis. ■

Now we are ready for our main result.

PROPOSITION 4.3. *Let C be an arbitrary algebraic constraint over R. Then*

$$\pi(\widetilde{V}_c(C)) = V_c(C) \tag{4.22}$$

holds.

Proof. Since Lemma 4.2 holds for arbitrary constraints including general negation, as in the proof of Proposition 4.2, it is sufficient to show that negation commutes with $\pi(\widetilde{V}_c())$, which we have just shown in Lemma 4.3. ■

By Proposition 4.3, we can construct a set of equations on the ambient space from an arbitrarily given constraint, that is, $S(I(C))$. Moreover, $\wedge, \vee$ and $\neg$ commute with operation of taking algebraic set on the ambient space as we have already seen.

Example. It follows from Proposition 4.3 that

$$\begin{aligned} \pi(\widetilde{V}_c(\neg(C_1 \wedge C_2))) &\stackrel{Prop.\ 4.3}{=} V_c(\neg(C_1 \wedge C_2)) \\ &= \neg V_c(C_1 \wedge C_2)) \\ &= \neg(V_c(C_1) \cap V_c(C_2)) \\ &= V_c(\neg C_1) \cup V_c(\neg C_2) \\ &= V_c((\neg C_1) \vee (\neg C_2)) \\ &\stackrel{Prop.\ 4.3}{=} \pi(\widetilde{V}_c((\neg C_1) \vee (\neg C_2))) \end{aligned}$$

This shows that we can use both $I(\neg(C_1 \wedge C_2))$ and $I((\neg C_1) \vee (\neg C_2))$ in calculation of the algebraic set $V_c(\neg(C_1 \wedge C_2))$, although $I(\neg(C_1 \wedge C_2))$ is not equal to $I((\neg C_1) \vee (\neg C_2))$ in general. ■

Example. Consider the relation between $I \times \mathbf{0}$ and $\mathbf{0}$. In $\mathcal{CS}(R)$, $I \times \mathbf{0} \neq \mathbf{0}$ in general. But $I \times \mathbf{0} = (I_1 + \cdots + I_r) \times \mathbf{0} = \sum_{i=1}^r I_i \times \mathbf{0}$, and for each i,

$$\begin{aligned} I_i \times \mathbf{0} &= \{0, f_1, \cdots, f_r, 1 - zh\} \times \{0, 1, 1 - z\} \\ &= \{0, f_1, \cdots, f_r, 1, 1 - zh\} \end{aligned}$$

In particular, every $I_i \times \mathbf{0}$ contains $1 \in R$. Hence $S(I \times \mathbf{0})$ also contains $1 \in R$, and $\bigcap_{f \in S(I \times \mathbf{0})} V(f) = \emptyset$ follows. This shows that we can replace $I \times \mathbf{0}$ with $\mathbf{0}$ in actual calculation, which corresponds to the fact that $I + (1) = (1) \subset R[z]$ for any ideal $I \subset R[z]$. ■

The following proposition shows the relation between $V_c(C)$ and $\widetilde{V}_c(C)$.

PROPOSITION 4.4. *Let $C \in AC(R)$ be a constraint such that $V_c(C)$ is nonempty. Then projection $\pi|_{\widetilde{V}_c(C)} : \widetilde{V}_c(C) \to V_c(C)$ is a finite morphism, that is, for every point $x \in V_c(C)$ the fiber $\pi^{-1}(x)$ is a finite set. In particular every fiber is a single point when C does not contain negations.*

Proof. Since $S(I(C))$ always contains an equation in the form of $(1 - zh_1(x)) \times \cdots \times (1 - zh_r(x))$, for any $x \in V_c(C)$ z is one of $1/h_i(x)$ $(1 \leq i \leq r)$ where $h_i(x) \neq 0$, and hence $\pi^{-1}(x)$ is a finite set. The second statement of the proposition is clear from the fact that all of h_i is 1 when C does not contain negations. ■

This shows that when $V_c(C)$ is an algebraic (closed) set, $\widetilde{V}_c(C)$ is on the hyperplane $z = 1$ and there is one-to-one correspondence between them.

Example. When $C = \neg(x) \vee \neg(x - 1)$ on $\mathbb{R}$, $I(C) = \{0, 1 - zx\} + \{0, 1 - z(x-1)\}$ and $S(I(C)) = \{0, (1 - zx)(1 - z(x-1))\}$ respectively. If $x \neq 0$ and $x \neq 1$, the fiber $\pi^{-1}(x)$ consists of two points, that is, $(x, \frac{1}{x})$ and $(x, \frac{1}{x-1})$. If $x = 0$ or 1, the fiber is a point. ■

5 Conclusion

In this paper we studied algebraic constraint that corresponds to a subset of k^n. We developed a theory to represent a constraint in a set of equations by using only one slack variable, and constructed a semiring that have a natural correspondence to algebraic constraint. Since arbitrary boolean formula can be transformed into an algebraic set of k^{n+1}, we can now use tools of algebraic geometry in order to develop techniques to handle general boolean formulas in a unified way. Here we mention about related issues and the possible directions of future research.

As we can easily see from the definition, we can define the semiring $\mathcal{CS}(R)$ over an arbitrary ring R that is not necessarily a polynomial ring over a field k, when $1 \in R$ (and hence $1 - zf$ is defined for $z, f \in R$). In this case we can regard an equation $f = 0$ as a condition that a single boolean variable f is true, and all of the propositions in this paper still hold. However, note that here we give not only true/false but an algebraic set for a given constraint. It is important that $S(I(C))$ gives a set of equations for a constraint C and we can apply methods of algebraic geometry to calculate manifolds on the ambient space.

In actual calculation of a set of equations $S(I)$ for a constraint generator I, we can think of $S(I)$ as an ideal of $R[z]$ and replace $S(I)$ with the radical $\sqrt{S(I)}$ so we can use reduction techniques that have been analyzed in [6]. (Although in [6] such techniques are shown as heuristics, they are very natural from the viewpoint of algebraic geometry, and in the actual calculation they will be hidden in computation by Gröbner basis)

We are interested in deeper analysis of Proposition 4.3. Since we define $\{0, f, 1 - z\}$ as a constraint generator corresponding to the constraint $C = (f)$, $\widetilde{V}_c(C) = \{(x, z) \in k^{n+1} \mid f(x) = 0 \ \wedge \ z = 1\}$ and we can easily see that the mapping $\widetilde{V}_c(C) \longmapsto \pi(\widetilde{V}_c(C))$ gives an isomorphism between (closed) algebraic sets of k^n and (closed) algebraic sets of k^{n+1} that are on the hyperplane $z = 1$. However, this does not holds for general (non-closed) sets, and it is an interesting problem how to measure "discrepancy" between $V_c(C)$ and $\widetilde{V}_c(C)$. Toward application to engineering areas, it is another interesting issue to extend these ideas to inequalities such as $f > 0$, although this is more difficult to do with one variable.

References

[1] S. C. Chou, *Automated Reasoning in Geometries Using the Characteristic Set Method and Gröbner Basis Method*, Proceeding of ISSAC, 255-260, 1990.

[2] D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties and Algorithms*, Undergraduate Texts in Mathematics, Springer, 1992.

[3] R. Hartshorne, *Algebraic Geometry*, Graduate Texts in Mathematics, Springer, 1977.

[4] J. S. Golan, *The theory of semirings with applications in mathematics and theoretical computer science*, Addison-Wesley Longman Ltd., 1992.

[5] D. Kapur, *Automated Geometric Reasoning: Dixon Resultants, Gröbner Bases, and Characteristic Sets*, Lecture Notes In Computer Science, Vol. 1360, Selected Papers from the International Workshop on Automated Deduction in Geometry, 1-36, 1996.

[6] D. Kapur and H. K. Wan, *Refutational Proofs of Geometry Theorems vis Characteristic et Computation*, Proceeding of ISSAC, 277-284, 1990.

[7] O. E. Ruiz S. and P. M. Ferreira, *Algebraic Geometry and Group Theory in Geometric Constraint Satisfaction*, Proceedings of ISSAC, 224-233, 1994.

[8] H. Sawada and X. T. Yan, *Applying a Generic Constraint Solving Technique to Engineering Design*, ECAI Workshop notes on Knowledge-Based Systems for Model-Based Engineering, 52-58, 2000.

[9] A. J. Sommese, J. Verschelde and C. W. Wampler, *Numerical Irreducible Decomposition using PHCpack*, Algebra, Geometry, and Software Systems, Springer, 109-129, 2003.