卒業論文

Solving SAT by Integral Transform
(積分変換によるSAT問題の解法)

平成20年2月12日提出

指導教員
坂井修一 教授
五島正裕 准教授

電子情報工学科
60373 順 関
Abstract

We propose an new approach to SAT by (1) Considering variables binary-valued periodic waves of different periods, to include all the possible assignments into time-domain of the waves. (2) Applying OR operation to disjunctions and AND operation to conjunctions of the waves, to acquire a result wave in which contains information of satisfiable assignments. (3) (if satisfiable) Acquiring a satisfiable assignment by measuring the result wave.

For mathematical implementation, (1) We represent the waves in frequency-domain by integral transform. The wave could be represented as simple expression in frequency-domain. (2) In frequency-domain Logical operations of the waves is defined by polynomial arithmetics. (3) By inversed integral transform, to get a satisfiable assignment in time-domain.

We show that if we can establish AND operation for the entire time-domain of waves in polynomial complexity, an polynomial order algorithm for SAT will be realized.

This research is under developing. We report the current achievements in this thesis.
Contents

1 Introduction 2
  1.1 Background 2
  1.2 Thesis Organization 2

2 SAT 3
  2.1 Notation and Definitions 3
  2.2 Existing Algorithms 4

3 Integral Transform 5
  3.1 Basic Definitions 5
  3.2 Z-Transform 5
    3.2.1 Definition 5
    3.2.2 Properties 6
  3.3 Inversed Z-Transform 7

4 Solving SAT by Integral Transform 8
  4.1 Basic Conception 8
  4.2 Notation and Definitions 10
  4.3 Representing Variable by Wave in Frequency Domain 10
  4.4 Logical Operations in Frequency Domain 11
    4.4.1 NOT operation 11
    4.4.2 OR operation 12
    4.4.3 AND operation and it’s Problem 12
  4.5 Acquiring a Satisfiable Solution 15
  4.6 Analysis of Complexities 17
  4.7 Summary 17

5 Conclusions 19
  5.1 Conclusions 19
  5.2 Future Works 19
1 Introduction

1.1 Background

A Boolean Satisfiability Problem (SAT) is: Giving a boolean formula \( \phi \) in conjunctive normal form (CNF), decide if \( \phi \) could be “TRUE”. SAT is a representative problem of NP-Complete. There are varieties of applications of SAT. For example, AI, Planning\(^5\), Microprocessor verification\(^7\), Bioinformatics\(^3\) and so on.

Each literal could be assigned 1 or 0, a formula including \( N \) literals has \( 2^N \) possible solutions. Due to the relation between literals, an heuristic method is effective for SAT. DPLL is a formalism of solutions based on heuristics. Currently most of the effective algorithms for SAT are based on DPLL form.

In this research, We challenged developing a new approach to SAT which is Not based on the heuristics. We see variables as waves. and by operation for waves to acquire the satisfiable assignments.

1.2 Thesis Organization

Firstly we will give definitions of SAT in Chapter 2. We’ll give a introduction of integral transform in Chapter 3. Then we’ll show the basic conception of our algorithm in the beginning of Chapter 4, then the details of it’s mathematical implementation of each step. We’ll make conclusions and future plans for the research in Chapter 5.
2 SAT

2.1 Notation and Definitions

Here we give the definitions related to SAT.

**Propositional Variable (Variable for short)**

A Variable can take either TRUE or FALSE as it’s value.

**Literal**

A Literal is \( x \) or \( \overline{x} \), where \( x \) is a variable and \( \overline{x} \) is it’s negation.

**Clause**

A Clause is a disjunction of \( l_1 \lor l_2 \lor \ldots \lor l_n \), where \( l_1, l_2, \ldots, l_n \) are literals.

**Boolean Formula**

A Boolean Formula \( \mathcal{F} \) is defined over a set of literals with standard propositional connectives \( \neg, \land, \lor, \Rightarrow, \iff \), and parenthesis.

**Conjunctive Normal Form (CNF)**

A formula in CNF is a conjunction of clauses. Any formula can be transformed into CNF in polynomial time.

**k-CNF**

A \( k \)-CNF is a CNF contains \( k(k \geq 2) \) literals in each clause. Any CNF can be converted into \( k \)-CNF in polynomial time[12].

**Definition of SAT Problem**

Giving a formula \( \mathcal{F} \) in CNF. Find if there exists an assignment to the variables of \( \mathcal{F} \) such that \( \mathcal{F} = 1 \). If exists such an assignment, output SATISFIABLE, else output UNSATISFIABLE.

For example, giving \((x + y)(\overline{x} + y)(x + \overline{y})\), we have \( x = \text{TRUE}, y = \text{TRUE} \) as a satisfiable assignment, thus the giving formula is SATISFIABLE. Giving \((x + y)(\overline{x} + y)(x + \overline{y})(\overline{x} + \overline{y})\), no assignment can make it be \( \text{TRUE} \), thus the giving formula is UNSATISFIABLE.

SAT is a representative problem of NP-Complete. There is not a existing polynomial time algorithm for solving SAT. Any algorithm solving SAT is in the worst case exponential to the number of variables.
2.2 Existing Algorithms

There are mainly two classes of algorithms for solving SAT: Complete Methods and Incomplete Methods (Stochastic Methods[6]). Complete Methods such as DPLL can provide a satisfiable solution. Incomplete Methods such as Local Search[2] can only prove satisfiability. Since our proposal algorithm is neither similar with DPLL nor Local Search, here we only give a brief introduction to DPLL as a reference of existing algorithm.

**DPLL**

A DPLL[4] method was developed by Davis, Putnam, Logemann and Loveland in 1960. DPLL is a non-probabilistic searching method. DPLL is mainly constructed by four phases.

**Phase 1: Preprocessing**

In Preprocessing, DPLL assigns obvious variables. If all the variables couldn’t be assigned, then go to the next phase: Decision.

**Phase 2: Decision**

In Decision, DPLL assigns Free Variables, which have not been assigned in the first phase.

**Phase 3: Deduction**

In Deduction, DPLL look for the result of current assignment. As the result, SATISFIABLE or CONFLICT or UNKNOWN is returned. If UNKNOWN, then return to the Decision phase.

**Phase 4: Conflict-Analysis**

If CONFLICT is returned in Deduction phase, DPLL backtracks the assignment of variables. After backtracking, return to Decision phase.

So far varieties of optimization methods for DPLL have been developed[1]. The optimization methods are mainly focused on reducing the searching amount of possible assignments. SAT solvers based on DPLL has improved significantly. However, as mentioned above, in the worst case the computation cost of DPLL based algorithm will grow exponentially.
3 Integral Transform

3.1 Basic Definitions

A Integral Transform converts function \( f(t) \) to \( g(\alpha) \). The Integral Transform is defined by

\[
g(\alpha) = \int_{a}^{b} f(t) K(\alpha, t) dt,
\]

(1)

where function \( K(\alpha, t) \) is called the Integral Kernel of the transform. Generally the interval of the integration is decided by the Integral Kernel.

Varieties of transforms could be defined by the kind of the Integral Kernel. For example, Fourier Transform[11][9] is defined by

\[
g(\alpha) = \mathcal{F}\{f(t)\} = \int_{-\infty}^{\infty} f(t) e^{i\alpha t} dt,
\]

(2)

where \( e^{i\alpha t} \) is the Kernel. Another example is Laplace Transform[8], which is defined by

\[
g(\alpha) = \mathcal{L}\{f(t)\} = \int_{0}^{\infty} f(t) e^{-\alpha t} dt,
\]

(3)

where \( e^{-\alpha t} \) is the Kernel.

Integral Transform converts a function defined in time domain to another domain, in where problems could be solved easier. For example, Differentiations and Integrations could be converted into arithmetic calculations by Laplace Transform. As we show later, we use integral transform to convert periodic rectangular waves of time domain into frequency domain. The expression and operation of periodic rectangular wave is troublesome in time domain. In other hand, they could easily be defined as rational expressions in frequency domain. To achieve this merit, varieties of integral transform could be applicated. Nevertheless, our need is only focused on periodic rectangular wave of discrete time-domain. Thereby in this thesis we only use Z-Transform, which is applied to discrete time sequence.

3.2 Z-Transform

3.2.1 Definition

A Z-Transform[10][14] is defined by

\[
F(z) = \mathcal{Z}\{f(n)\} = \sum_{n=0}^{\infty} f(n)z^{-n},
\]

(4)

where \( z \) is a complex number. Z-Transform is used to convert a discrete date sequence to a complex frequency expression. For example, we have a discrete sequence define below:

\[
f_1(n) = \{1, 2, 0, 1, 0, 0, 7, 0\}.
\]

(5)
According to (4), (5) is Z-Transformed by
\[
F_1(z) = \mathcal{Z}\{f_1(n)\} = 1 \cdot z^0 + 2 \cdot z^{-1} + 0 \cdot z^{-2} + 1 \cdot z^{-3} + 0 \cdot z^{-4} + 0 \cdot z^{-5} + 7 \cdot z^{-6} + 0 \cdot z^{-7}
\]
\[
= 1 + 2z^{-1} + z^{-3} + 7z^{-6},
\]
where the \( z \) is a complex number. The example above there is no restriction for the \( z \). Giving another example of a infinite sequence:
\[
f_2(n) = \{1, 0, 1, 0, 1, 0, 1, 0, \ldots\}.
\]
(7)
According to (4), (7) is Z-Transformed by
\[
F_2(z) = \mathcal{Z}\{f_2(n)\} = 1 + z^{-2} + z^{-4} + z^{-6} + \ldots = \frac{1}{1 - z^{-2}}, \text{ where } |z| > 1.
\]
(8)
In this case there exists a Range Of Convergence (ROC) that \(|z| > 1\).

3.2.2 Properties

We give some major properties of Z-Transform related to our work here. Note that \( f(n) \) and \( g(n) \) are two arbitrary sequences and \( F(z) = \mathcal{Z}\{f(n)\}, G(z) = \mathcal{Z}\{g(n)\} \).

**Linearity**
\[
\mathcal{Z}\{af(n) + bg(n)\} = aF(z) + bG(z).
\]
(9)

**Time Shifting**
\[
\mathcal{Z}\{f(n-k)\} = z^{-k}F(z).
\]
(10)

**Multiplication in Time-Domain**
\[
\mathcal{Z}\{f(n)g(n)\} = \frac{1}{2\pi j} \int_{\Gamma} w^{-1}F(w)G(wz^{-1})dw,
\]
(11)
where the closed integration path \( \Gamma \) separates the poles of \( w^{-1}F(w) \) from the poles of \( G(wz^{-1}) \).

**Convolution in Time-Domain**
\[
\mathcal{Z}\{f(n) * g(n)\} = F(z)G(z).
\]
(12)
Difference in Time-Domain

\[ \mathcal{Z} \{f(n) - f(n-1)\} = (z-1)F(z). \]  \hspace{1cm} (13)

Parseval's relation

\[ \sum_{n=-\infty}^{\infty} f(n)g^*(n) = \frac{1}{2\pi j} \int_{\Gamma} w^{-1}F(w)G^*(w^*-1)dw. \]  \hspace{1cm} (14)

3.3 Inversed Z-Transform

Here we give the Inversed Z-Transform by Power Series method[13]. According to the definition of Z-Transform, we have

\[ F(z) = f(0) + f(1)z^{-1} + f(2)z^{-2} + ... + f(n)z^{-n} + ... . \]  \hspace{1cm} (15)

Consider a \( F(z) \) represented by a rational formula below,

\[ F(z) = \frac{a_0 + a_1z^{-1} + ... + a_mz^{-m}}{1 + b_1z^{-1} + ... + b_nz^{-n}}, \quad n \geq m. \]  \hspace{1cm} (16)

By dividing the numerator by the denominator we expand (16) to the form of (15):

\[ F(z) = c_0 + c_1z^{-1} + c_2z^{-2} + ... + c_nz^{-n} + ... . \]  \hspace{1cm} (17)

The division is done by continued fraction. Let \( f(n) = c_n \) then we have the inversed Z-Transform of (16). For example, giving a \( F(z) \) below:

\[ F(z) = \frac{z^{-1}}{1-z^{-2}}. \]  \hspace{1cm} (18)

The (18) is expand by

\[ \frac{z^{-1} + z^{-3} + ...} {1-z^{-2}}z^{-1} = \frac{z^{-1} - z^{-3}}{z^{-3} - z^{-5} + ...}. \]  \hspace{1cm} (19)

Where the coefficients of \( z^{-1} + z^{-3} + ... \) is the inversed sequence \( \{0, 1, 0, 1, ...\} \).

Notice that to get the first term of inversed series, we only have to know the first terms of numerator and denominator respectively. This property plays an important role in acquiring satisfiable assignment in our algorithm, we’ll see the detail in next chapter.
4 Solving SAT by Integral Transform

4.1 Basic Conception

We consider each variable a periodic binary-valued wave (we simply call it wave below). The wave could be considered as periodic clock signal with amplitudes of only 0 and 1. By assigning different variables 2 times of periods, all the assignment combinations will be included in the Time-Domain of the waves. For example, giving a SAT formula of 3 variables: \((x + y)z\), assume we assign the \(x\) a wave of period \(T\), the \(y\) period of \(2T\) which is 2 times the period of \(x\), and like wise the \(z\) \(4T\). In this way all the \(2^3\) possible assignments for the formula are included into the time-domain of the 4 waves (Figure 1).

![Figure 1: Waves and assignments of x, y and z](image)

Firstly, the disjunctions inside clause is done by applying \(OR\) operation for waves. In the example, we apply \(OR\) operation to the waves of \(x\) and \(y\) inside the clause, thus we get the result wave of \(x OR y\) (Figure 2).

![Figure 2: x OR y](image)

Secondly, the conjunction clauses is done by applying \(AND\) operation for waves. After we applied \(AND\) operation for all the result waves of each clause, we acquire a final result wave (Figure 3).

![Figure 3](image)
Thirdly, if the giving formula is satisfiable, there must be more than 1 none-0 amplitude somewhere in the time-domain of the final wave. By measuring the average amplitude of the wave, we can know whether the formula is satisfiable. Moreover, by finding the the none-0 time of the wave, we can acquire the satisfiable assignments for the formula. In the example above, compare the result wave of \((x + y)z\) to the assignments in Figure 1, then we acquire the first three satisfiable assignments(Figure 4, in which the satisfiable assignments are reddened).

The conception requires methods of applying logical-operations for the entire time-domain of two waves "at once". Because If we have to "Sampling" each time of wave, the conception would be useless. Thereby the critical questions are: (1)How to represent each variable by wave mathematically? (2)How to do logical-operations of waves for the entire time-domain at once? (3)How to acquire a satisfiable assignment from the final wave mathematically?

We challenged some ways to reduce the complexity for each step. Our ap-
The approach to the questions is: (1) Representing variables by periodic waves in frequency-domain in which the wave will simply be a rational formula. (2) In frequency-domain, represent logical-operations for waves as operations of the rational formulas. (3) By Inversed Integral-Transform the final wave, acquire times of none-0 amplitudes in time-domain and those are satisfiable assignments.

The most critical problem in this approach is, Integral Transform is normally designed for linear system while logical-operation is not. Thereby in (2) and (3) we have to develop means to acquire the same "effect" of logical-operations in Frequency-Domain. Which is to "simulate" none-linear system by linear system.

Our approach is not complete yet and there’re still plenty of problems to be solved, nevertheless, we give the initial achievements of the details for each step in the following sections of this chapter.

4.2 Notation and Definitions

Here we give the notation and definitions used in the following issue.

Wave: A periodic wave in discrete time-domain, which can be considered as a periodic sequence. As mentioned above, initially each variable is assigned a unique wave. By wave it also represents a variable. We represent a wave by \( f(n) \) in time-domain, or \( F(z) \) in frequency-domain. We call the frequency-domain form \( F(z) \) Z-Expression.

Result Wave: A wave acquired by applying logical-operations to two waves.

Final Wave: A result wave acquired after all the logical-operations of a SAT formula. As mentioned above, The times(see below) of none-0 amplitude in final wave are satisfiable assignments.

Time: An exact time in time-domain of a wave. As in (Figure 1), a time in waves of variables represents an assignment.

Assignment: The assignment for variables of a SAT formula.

4.3 Representing Variable by Wave in Frequency Domain

Here we give the method to represent variables by waves in frequency-domain. Assume a variable \( x \) is assigned a wave of period \( T \), it’s mathematical representing in time-domain would be

\[
x(n) = \sum_{i=0}^{\infty} (-1)^i u(n - \frac{iT}{2}), n > 0,
\]  

(20)
where the \( u(n) \) is Heaviside function. The \( \sum \) in expression(20) made it difficult in calculations. For this reason, we express the wave in frequency-domain in which the wave will simply be a rational expression. The wave is discrete in amplitude and time domain, thus it could be seen as a sequence of period \( T \)

\[
x(t) = \{1, 1, \ldots, 0, 0, 1, 1, \ldots, 0, 0, \ldots\}.
\]  

According to the definition of Z-Transform, \( x(t) \) is transformed to frequency-domain by

\[
X(z) = \mathcal{Z}\{x(t)\} = \frac{z^{-1} + z^{-2} + \ldots + z^{-T} + z^{-2T} + z^{-2T-1} + \ldots + z^{-3T} + \ldots}{(1 + z^{-T/2}) (1 - z^{-1})}, \quad |z| > 1.
\]

We call such an formula Z-Expression below. Any periodic binary-valued sequences can be transformed into the form of the Z-Expressions like wise.

### 4.4 Logical Operations in Frequency Domain

Here we show Logical Operations for Z-Expressions.

#### 4.4.1 \textit{NOT} operation

\textit{NOT} operation could be seen as ”Time Shifting” for wave in Time-Domain. Giving \((x+y)\) and \((\bar{x}+y)\) for comparison, assume the \(x\) is represented by a wave of period \(T\). Then \(\bar{x}\) could be represented by shifting the wave of \(x\) \(T\) times forward(or backward)(Figure 5).

![Figure 5: NOT(x) by time shifting](image)

According to the Time Shifting property of Z-Transform, Shifting a sequence \(T\) times forward in Time-Domain is done by multiplying \(z^{-T}\) to it’s corresponding Z-Expression. Therefore the negate of \(x\) could be defined as following

\[
\overline{X}(z) = z^{-T} X(z).
\]  

11
where T is the period assigned to variable x.

4.4.2 OR operation

In CNF, OR operation is required in the disjunctions inside each clause. Giving a clause \((x + y)\) for example, assume the variable \(x\) is assigned a wave of period \(T\), and \(y\) of period \(2T\), thus their Z-Expressions are

\[
X(z) = \frac{1}{(1 + z^{-T/2})(1 - z^{-1})}, \quad Y(z) = \frac{1}{(1 + z^{-T})(1 - z^{-1})}.
\]  (24)

In Frequency-Domain, there’s not any elementary operation could make the same result corresponds to \(x \text{ OR } y\). For this reason, we permit multiple amplitudes. We define amplitude 0 as FALSE, otherwise TRUE. Thereby we can define OR operation as

\[
x(z) \text{ OR } y(z) = x(z) + y(z) = \frac{1}{(1 + z^{-T/2})(1 - z^{-1})} + \frac{1}{(1 + z^{-T})(1 - z^{-1})}.
\]  (25)

4.4.3 AND operation and it’s Problem

In CNF, AND operation is required in conjunctions of clauses. Since we permitted multiple none-0 amplitudes for representing TRUE, AND operation can not be done by the De Morgan’s laws with negation. Here firstly we give the definition of a very special case of AND operation, which can be done easily but not practical. Secondly we discuss the general case, which is the most critical problem in the entire research.
A very special case. Giving \( (x \cdot y) \) for example of the special case, where the \( x \) and \( y \) are two sequences defined in time-domain:

\[
x(n) = \{1, 1, 0, 0, 1, 1, 0, 0\ldots\}, \quad y(n) = \{1, 0, 1, 0, 1, 0, 1, 0\ldots\}. \tag{26}
\]

To find the result of \( x \) AND \( y \) in the case, we first calculate their convolution sequence in time-domain:

\[
x(n) * y(n) = \{1, 1, 1, 1, 2, 2, 2\ldots\}. \tag{27}
\]

Secondly we difference the convoluted sequence, thus we have

\[
\Delta\{x(n) * y(n)\} = x(n) * y(n) - x(n-1) * y(n-1)
\]

\[
= \{1, 0, 0, 0, 1, 0, 0, 0\ldots\}. \tag{28}
\]

The (28) perfectly matches the result of sequence \( (x \cdot y) \). Now consider it’s corresponding operations by Z-Expressions. The \( x \) and \( y \) are transformed to Z-Expression as

\[
X(z) = \frac{1}{(1+z^{-2})(1-z^{-1})}, \quad Y(z) = \frac{1}{(1-z^{-2})}. \tag{29}
\]

Firstly, according to the property of convolution in time-domain of Z-Transform, the convolution result of \( X(z) \) and \( Y(z) \) is done by their multiplication:

\[
\mathcal{Z}\{x(n) * y(n)\} = X(z)Y(z) = \frac{1}{(1+z^{-2})(1-z^{-1})} \cdot \frac{1}{(1-z^{-2})}. \tag{30}
\]

Secondly, according to the difference property of Z-Transform, the difference of the convolution result is acquired by

\[
\mathcal{Z}\{\Delta\{x(n) * y(n)\}\} = (z-1)X(z)Y(z)
\]

\[
= (z-1) \frac{1}{(1+z^{-2})(1-z^{-1})} \cdot \frac{1}{(1-z^{-2})}. \tag{31}
\]

As showed above, in this special case we can define AND as

\[
X(z) \text{ AND } Y(z) = (z-1)X(z) \cdot Y(z). \tag{32}
\]

General case. In general case we consider \( x \) and \( y \) two arbitrary waves, AND of the two waves equals to their multiplication in time-domain, which can be done by the convolution of their corresponding Z-Expressions,

\[
X(z) \text{ AND } Y(z) = \mathcal{Z}\{x(n) * y(n)\} = \frac{1}{2\pi j} \oint_{\Gamma} w^{-1}X(w)Y(zw^{-1})dw. \tag{33}
\]

The integration can be done by Residue Theorem[8]. The integration path \( \Gamma \) separates the poles of \( w^{-1}X(w) \) from those of \( Y(zw^{-1}) \). Thereby the integration equals to the sum of all the residues at either the poles of \( w^{-1}X(w) \) or \( Y(zw^{-1}) \):

\[
X(z) \text{ AND } Y(z) = \sum X_i = - \sum Y_i, \tag{34}
\]
where the \( X_i \) are the residues at the poles of \( w^{-1}X(w) \) and \( Y_i \) are those of \( Y(zw^{-1}) \).

In the case of (24), from (33) and (34) we have

\[
X(z) \text{ AND } Y(z) = \sum \text{Res}\left\{ \frac{1}{(1+w^{-T/2})(1-w^{-1})} \right\} = -\sum \text{Res}\left\{ \frac{1}{(1+z^{-T}w)(1-w^{-1})} \right\}. \tag{35}
\]

From (35), the first and the second sum contain \( T+1 \) and \( 2T+1 \) residues respectively. The \( T \) is the period of assigned waves. As the assignment rule of waves, \( T \) grows exponentially to the number of variables. Thereby the raw calculation of residues costs exponential order. In order to avoid the exponential complexity, we have to develop some means to calculate the convolution in less order. Due to the time limitation we haven’t establish a complete method, nevertheless we give the outlines of two alternative expectations bellow.

**Alternative 1.** Consider two arbitrary waves \( x(n) \) and \( y(n) \),

\[
x(n) = \{x_0, x_1, x_2, ..., x_n\}, \quad y(n) = \{y_0, y_1, y_2, ..., y_n\}. \tag{36}
\]

AND operation requires the two terms of the same time of \( x(n) \) and \( y(n) \) to be multiplied. In general case the problem is, the multiplication \( X(z) \cdot Y(z) = x(n) \ast y(n) \) can not make the two terms of the same time interacted.

In order to make the two terms interacted, firstly flip \( y(n) \) in time-domain by \( Y(z^{-1}) \), notate the flipped \( y(n) \) as \( y'(n) \). Secondly slide \( y'(n) \) one by one time forward in time-domain(Figure 7).

\[
\begin{array}{c|cccc|c}
\text{Slide} & X_0 & X_1 & X_2 & \ldots & X_n & \Rightarrow X(z) \\
\text{Yn} & \cdots & Y_2 & Y_1 & Y_0 & \Rightarrow Y(z^{-1}) \\
\text{Yn} & \cdots & Y_2 & Y_1 & Y_0 & \Rightarrow z^{-1}Y(z^{-1}) \\
\text{Yn} & \cdots & Y_2 & Y_1 & Y_0 & \Rightarrow z^{-2}Y(z^{-1}) \\
\end{array}
\]

Figure 7: Flipping and sliding \( Y(z) \)

In each time sliding, convolute the \( X(z) \) with the \( Y(z^{-i}) \) to acquire a new convoluted series. Put all the convoluted series line by line, the diagonal terms are wanted part(reden part in Figure 8).

To get the sum of the diagonal terms, we have to cut the \( n \)th term out of each series of \( X(z)z^{-n}Y(z) \). Assume that we could cut it out by some means(by window
function of signal processing techniques, or by the difference of correlation), we have the integrated series of the result of $\text{AND}$, notate this series as $r'_{xy}(n)$, we have

$$r'_{xy}(n) = \sum_{i=0}^{n} x_iy_i,$$  \hspace{1cm} (37)

immediately the result of $\text{AND}$ is acquired by multiplying (37) $(z - 1)$ to acquire the difference,

$$x(n) \text{AND} y(n) = \Delta \sum_{i=0}^{n} x_iy_i = \Delta r'_{xy}(n),$$  \hspace{1cm} (38)

Of which the operation by $Z$-Expression is

$$X(z) \text{AND} Y(z) = (z - 1)R'_{xy}(z).$$  \hspace{1cm} (39)

**Alternative 2.** The denominators of (35) only contains poles of first degree. All the $T/2 + 1$ poles of denominator $(1 + w^{-T/2})(1 - w^{-1})$ of the first sum are positioned in the unit circle of $Z$-plane(Figure 9). Consider the residues a series, if we can find it’s convergence we find the result of $\text{AND}$ operation.

### 4.5 Acquiring a Satisfiable Solution

Here we give the method to acquire a satisfiable assignment. Assume the $\text{AND}$ operation could be done by the operations of $Z$-Expressions. As the calculating result of conjunctions a final wave $R(z)$ is acquired. Consider the final wave $R(z)$ is represented as a rational formula like below,

$$R(z) = \frac{(a_0z^{-c_0} + a_1z^{-c_1} + ... + a_nz^{-c_n})(b_0z^{-d_0} + ... + b_nz^{-d_n})...}{(u_0z^{-j_0} + u_1z^{-j_1} + ... + u_nz^{-j_n})(v_0z^{-k_0} + ... + v_nz^{-k_n})...},$$  \hspace{1cm} (40)

where the $a_i, b_i, c_i, d_i, u_i, v_i, j_i, k_i$ are constants.
If there exists a satisfiable assignment in any time, it’s amplitude in the Time-Domain of the final wave must be none-0(TRUE). According to the Power Series method of inversed Z-Transform (16), the degree of a term equals to time in Time-Domain, and the coefficient equals to it’s amplitude. Thereby the times of 0 amplitude have 0 coefficients, thus their therms will not show in the expanded series. Thereby the degrees of each terms in the series are satisfiable assignments. We only have to find the degree of any terms to acquire a satisfiable assignment. Finding the degree of the first term is the easiest. From (40), we only have to divide the first term of numerator by the first term of denominator, the quotient is the wanted term and it’s degree is a satisfiable assignment.

In the numerator of (40), inside each parenthesis we choose the term of the highest degree for multiplying, to acquire the first term of numerator. In the same way we acquire the first term of denominator of (40).

Giving a \( R(z) \) below for example:

\[
R(z) = \frac{(3z^{-1} + 7z^{-5})(1 - z^{-2})(4z^{-4} - z^{-7})}{(1 - 2z^{-1})(2z^{-2} - z^{-9})(z^{-2} + z^{-3})}. \tag{41}
\]

Multiplying the terms of highest degree(underlined) in each parenthesis of the nominator of (41), we have the first term \( 12z^{-5} \). In the same way we have the first term of denominator: \( 2z^{-4} \). According to (19), divide \( 12z^{-5} \) by \( 2z^{-4} \) results \( 6z^{-1} \), thus the degree \(-1\)(=the time of 1) is a satisfiable assignment.

(If there exists more than one satisfiable assignment)To get another satisfiable assignment we have to find the second terms of the numerator and denominator of (40). These cost more than the case of the first one. We’ll give the analysis of complexity in the next section.
4.6 Analysis of Complexities

Here we give the analysis of complexity of each part of our algorithm. Giving a SAT problem, we define \( N \) as the number of the variables, and \( M \) as the number of the clauses.

The complexity of Representing waves to Z-Expressions

Initially we assign each variable an elementary periodic wave. As mentioned above in (22), the wave can be represented in Z-Expression by it’s period \( T \) instantly. Thereby the construction of each Z-Expression costs constant order, and all the \( N \) variable cost \( O(N) \). The construction part is in polynomial order.

The complexity of Logical Operations by Z-Expressions

From (23), \( \text{NOT} \) operation is done by multiplying a \( z^{-T} \) to Z-Expression. Thereby the operation is in polynomial order.

From (25), \( \text{OR} \) operation is done by adding two Z-Expressions. Thereby the operation is in polynomial order.

The complexity of \( \text{AND} \) operation is the most critical. As mentioned above, in the special case \( \text{AND} \) operation is done by multiplying the two Z-Expressions, thus it’s complexity is in polynomial. Nevertheless the definition of the special case is not practical. In general case, finding the operation of \( \text{AND} \) with less than exponential complexity order is paramount to the advantage of the entire algorithm.

The complexity of acquiring satisfiable assignment by fraction

From the form of final wave (40), Selecting the term of the smallest degree in each parenthesis is in polynomial order. Then finding and adding the first term is also in polynomial order. Thereby finding the two first terms of numerator and denominator is polynomial order. Thereby to find one satisfiable assignment will be done in less than exponential order.

4.7 Summary

This chapter firstly presents the conception of our proposal algorithm, which can be summarized by three steps. Giving a SAT problem, (1) Considering variables periodic waves. Assign the variables waves of different period so that all the possible assignments are included in the time-domain of the waves. (2) By doing logical-operations for the entire time-domain of waves at once, acquire a final wave in which exists information about satisfiable assignments. (3) By inversed Z-Transform, acquire the time of none-0 amplitude in time-domain of the final wave, and this time is a satisfiable assignment.

Secondly we show the mathematical implementation of the conception. (1) Representing the waves in frequency-domain by Z-Transform. In frequency-domain
the waves will be expressed as $Z$-Expressions, which is a rational formula of complex variable $z$. (2) \textit{NOT} and \textit{OR} operation for the waves can be done by simple calculations of $Z$-Expressions. \textit{AND} operation can be done by multiplication of $Z$-Expressions in a very special case, however in general case, how to develop a \textit{AND} method of less than exponential complexity is the most critical. (3) Based on the assumed form of the final wave, by dividing the first terms of numerator and denominator, a satisfiable assignment can be acquired.

We show that the complexity of the operations in the three steps are less than exponential order except \textit{AND} operation of general case. How to establish a method of \textit{AND} in less than exponential order complexity is the paramount of the entire algorithm.
5 Conclusions

5.1 Conclusions

We presented a new approach to SAT Problems. Variables of SAT could be seen as waves, and after applying logical operations for waves, we have the final wave in which contains the information of satisfiable assignments. We presented that the most critical problem is how to establish a calculation method for AND operation to two waves. We also presented two alternative expectations for the solution. If we can find it we have the possibility to solve SAT in less than exponential complexity.

5.2 Future Works

Developing AND operation for Z-Expressions in less than exponential order complexity is the most critical problem. The algorithm take advantages If and only If AND operation could be established. Following the two expectations of solutions above, our work shall continues the developing of AND method. In other hand, the important point is that, we do not have to develop a method which exactly results AND operation in Time-Domain. The amplitudes of TRUE could be changed as far as it’s not 0. There by varieties of possibilities should be considered.
References


Acknowledgement

I thank to Professor Shuichi Sakai, who had advised me plenty of comments in meetings and daily researches.

I thank to Associate Professor Masahiro Goshima, who advised me a lot and had worried about my research theme all the time, notwithstanding some times i’ve deviated from his guidance.

I thank to Mr.Ryota Shioya, who had taught me a lot of knowledges of architectures.

I also thank to Mr.Kenichi Watanabe, Mr.Hironori Ichibayashi, who gave me a great deal of valuable suggestions on my middle report.