Marek BALCER

Department of Mathematics Applications and Methods for Artificial Intelligence, Silesian University of Technology, Gliwice, Poland

AN EXAMPLE OF THE SATISFIABILITY PROBLEM IN THE CONTINUOUS STRUCTURE

Abstract. The paper presents and demonstrates the theorem showing the equivalence of the problem of the verifiability test of a logical expression in the discrete model $\mathfrak{N}$ of the logic with the search for the minimum value of a continuous function generated by this expression in the structure $\mathfrak{M}$, which is a simple extension of $\mathfrak{N}$. Theoretical considerations are illustrated by the example of a certain semi-heuristic algorithm seeking the minimum value of function φ with a short statistics of its.

1. Introduction

Many efficient algorithms have been developed for satisfiability testing. Many of them are presented and classified by Jun Gu in [1]. One branch of this classification includes algorithms namely continuous and constrained algorithms. Its' idea depends on creating multidimensional function which interpolates or approximates some Boolean formulas or logic expressions. The satisfiability problem of these formulas is changed to searching the minimum or the maximum of constructed functions. For example, in 1998 Back et al. proposed to transform SAT into continuous optimisation [3]. Similarly, this method is also used for other satisfiability

problems like Max-Sat, Uni-SAT etc. [2]. Replacing the problem of satisfiability
with the problem of finding the minimum value of the objective function allows us
to use many well-known standard or heuristic optimization algorithms. However,
in order to make the algorithms working more effectively it is necessary to make
some changes in their design.

In this article particle swarm approach is used to logic systems development,
where proposed application is used to model logic system with the search for the
minimum value of a continuous function generated for expressions formulated in
a predefined structure.

2. Mathematical Background

2.1. Language

Let $\mathfrak{L}$ be the set of all logical expressions formed from the symbols of atomic
formulas (atoms) and the logical symbols $\sim, \wedge, \vee$ defined as follows [4]:

If we take the non-empty finite set of $At(\mathfrak{L}) = \{p, q, r, \ldots\}$ as a set of atoms,
then:

$\mathfrak{L}$ is the smallest set such that:

1. $At(\mathfrak{L}) \subset \mathfrak{L}$

2. For any $A, B \in \mathfrak{L}$

$$\sim A \in \mathfrak{L}$$
$$A \wedge B \in \mathfrak{L}$$
$$A \vee B \in \mathfrak{L}$$

i.e. $(p \wedge q) \vee (\sim p \vee q) \in \mathfrak{L}$.

For constructing language we can also use the parenthesis free notation which
has better properties for decomposition of expressions [6].

2.2. Structure

A structure $\mathfrak{N}$ is $\mathfrak{N} = \langle X, f_\sim, f_\wedge, f_\vee \rangle$ where X is a non-empty set called a range
of assignment and

$$f_\sim : X \to X, \qquad f_\wedge : X \times X \to X, \qquad f_\vee : X \times X \to X,$$

will be certain functions defined in the domain X and $X \times X$.

2.3. Assignment

Let Z be a subset of the set of expressions $\mathfrak{L}$. Let us denote the set of all atoms occurring in expressions from the set Z by $At(Z)$.

Definition 1. *The assignment in the set X of atomic sets of Z is every function* $\mathbf{w}$, *which assigns the atomic proposition of the set Z to the set of X.*

Definition 2. *The value of an expression A for assignment $\mathbf{w}$ is the element of the set X, obtained by the function $f_\sim, f_\wedge, f_\vee$ from the values assigned by the function $\mathbf{w}$ for atomic proposition evaluated from the expression A.*

To illustrate this definition, let's take the following shortcut:
$$W(\mathbf{w}, A) = \text{expression value } A \text{ for assignment } \mathbf{w} \text{ [5]}$$

1. (I CONDITION). IF A is an atom, then

$$W(\mathbf{w}, A) = \mathbf{w}(A).$$

2. (INDUCTION CONDITIONS).

$$W(\mathbf{w}, \sim A) = f_\sim(W(\mathbf{w}, A)),$$
$$W(\mathbf{w}, A \wedge B) = f_\wedge(W(\mathbf{w}, A), W(\mathbf{w}, B)),$$
$$W(\mathbf{w}, A \vee B) = f_\vee(W(\mathbf{w}, A, W(\mathbf{w}, B))),$$

If the symbols of atoms are in order $At(\mathfrak{L}) = \{p_1, p_2, \ldots, p_m\}$ then assignment $\mathbf{w}$ can be a vector $\mathbf{w} = (\mathbf{w}(p_1), \mathbf{w}(p_2), \ldots, \mathbf{w}(p_m))$.

2.4. Satisfiability

Let $\mathfrak{N} = \langle X, f_\sim, f_\wedge, f_\vee \rangle$ be a structure. In a set X we can distinguish a non-empty subset T.

Definition 3. *(Generalized concept of truth in the structure). The expression A is called T-expression in the structure $\mathfrak{N}$, if for any assignment in this structure the value of the expression A belongs to the set T.*

2.5. Function generated by the expression

Definition 4. *The function generated by the expression A in the $\mathfrak{N}$ structure is called the function*

$$\varphi_{\mathfrak{N}}^A : X^m \to X$$

that, for any assignment $\mathbf{w}$

$$\varphi_{\mathfrak{N}}^A(\mathbf{w}(p_1), \mathbf{w}(p_2), \ldots, \mathbf{w}(p_m)) = W(\mathbf{w}, A),$$

where $p_1, p_2, \ldots, p_m$ are all atoms in A.

Example 5. Let be given a non empty set of expressions $Z \subset \mathfrak{L}$ comprised by the use of operators $\sim, \wedge, \vee$. Let $\mathfrak{N} = \langle X, f_\sim, f_\wedge, f_\vee \rangle$ be a structure, where $X = \{0, 1\}$ and

$$f_\sim(x) = 1 - x, \qquad f_\wedge(x, y) = xy, \qquad f_\vee(x, y) = \max(x, y),$$

$\mathfrak{N} = \langle \{0, 1\}, 1 - x, xy, \max(x, y) \rangle$.

Then, for $A = \ulcorner (p \wedge q) \vee (\sim p \wedge q) \urcorner$

$$\varphi_{\mathfrak{N}}^A(x, y) = \max(xy, (1 - x)y).$$

2.6. Normal form

Definition 6. *The expression A is in a disjunctive normal form if and only if A is an alternative of a finite number of conjunctions $A_1, A_2, \ldots, A_n$, so as*

$$A = A_1 \vee A_2 \vee \ldots \vee A_n.$$

Members of the conjunction A_i are atoms or negation of atoms, so as

$$A_i = B_i^1 \wedge B_i^2 \wedge \ldots \wedge B_i^{k_i},$$

where $B_i^j = \ulcorner p \urcorner$ or $\ulcorner \sim p \urcorner$, for p as an element from the set $At(A)$.

If expression $\mathbf{A}$ is composed as presented above, then we say that it is of a class $[n, k]$, where

$$k = \max_{i=1...n} k_i.$$

Example 7. Let $A = \ulcorner (p \wedge q \wedge r) \vee (p \wedge (\sim q)) \vee ((\sim p) \wedge q \wedge r) \vee (r \wedge s) \vee (p \wedge (\sim r) \wedge s) \urcorner$. Expression A is in the normal disjunctive form class $[5, 3]$. $At(A) = \{p, q, r, s\}$.

Let $\mathfrak{N} = \langle \{0,1\}, 1-x, xy, \max(x,y) \rangle$. Having the following assignment $\mathbf{w}$

$$\mathbf{w}(p) = x_1, \mathbf{w}(q) = x_2, \mathbf{w}(r) = x_3, \mathbf{w}(s) = x_4$$

and using assumption $\max(x, \max(y,z)) = \max(x,y,z)$ function $\varphi_{\mathfrak{N}}^A$ takes the form

$$\varphi_{\mathfrak{N}}^A(x_1, x_2, x_3, x_4) = \max\Big(x_1 x_2 x_3, \; x_1(1-x_2), \; (1-x_1)x_2 x_3, \; x_3 x_4, x_1(1-x_3 x_4) \Big).$$

In classical logic it is true that for any expression A there exists an A' expression in the normal form such that

$$A \equiv A'.$$

Consequently, further consideration of this article will only be restricted to the examination of expressions in the normal form.

3. From discrete to continuous universe

Let the structure be given

$$\mathfrak{N} = \langle \{0,1\}, 1-x, xy, \max(x,y) \rangle.$$

Let us create a new structure by changing only the set X from $\{0,1\}$ to $< 0,1 > \subset \mathbf{R}$

$$\mathfrak{M} = \langle \langle 0,1 \rangle, 1-x, xy, \max(x,y) \rangle.$$

Let A be an expression in the normal form of class $[n,k]$ and $|At(A)| = m$.

Theorem 8. *The expression A is a tautology if and only if*

$$\varphi_{\mathfrak{M}}^A(x) \geqslant \frac{1}{2^k}$$

for all $x \in \langle 0,1 \rangle^m$.

This means that in the continuous structure $\mathfrak{M}$ T-expressions are the $< \frac{1}{2^k}, 1 >$ expressions or $R+$ expressions.

In order to prove the above theorem, we first show the truth of several properties of the function $\varphi_{\mathfrak{M}}^A$.

Let's presume the following assumptions.

Let A be an expression from language $\mathfrak{L}$ in the normal form class $[n, k]$, where $At(A) = \{p_1, p_2, \ldots, p_m\}$, so that

$$A = A_1 \vee A_2 \vee \ldots \vee A_n, \tag{1}$$

where

$$A_i = B_i^1 \wedge B_i^1 \wedge \ldots \wedge B_i^{k_i} \tag{2}$$

and for $1 \leqslant i \leqslant n$, $1 \leqslant j \leqslant k_i$ $B_i^j = \ulcorner p \urcorner$ or $\ulcorner \sim p \urcorner$, $p \in \{p_1, p_2, \ldots, p_m\}$

$$k = \max_{i=1\ldots n} k_i.$$

Let us accept at the same time for any $i(1 \leqslant i \leqslant m)$, $x_i = w(p_i)$.

Corollary 9. *For any* $(x_1, x_2, \ldots, x_m) \in \{0, 1\}^m$

$$\varphi_{\mathfrak{M}}^A(x_1, x_2, \ldots, x_m) \in \{0, 1\}.$$

Corollary 10. *If for any* $(x_1, x_2, \ldots, x_m) \in \{0, 1\}^m$

$$\varphi_{\mathfrak{M}}^A(x_1, x_2, \ldots, x_m) = 1$$

and only if A is a tautology.

The above properties result directly from the structure extension from $\mathfrak{N}$ to $\mathfrak{M}$.

Corollary 11.
$$\varphi_{\mathfrak{M}}^A\left(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\right) = \frac{1}{2^l},$$

where $l = \min_{i=1\ldots n} k_i$.

Lemma 12. *If for any* $(x_1, x_2, \ldots, x_m) \in \langle 0, 1 \rangle^m$

$$\varphi_{\mathfrak{M}}^A(x_1, x_2, \ldots, x_m) > 0$$

then A is a tautology.

Proof. Let $x = (x_1, x_2, \ldots, x_m) \in \{0,1\}^m$. From the assumption results $\varphi_{\mathfrak{M}}^A(x_1, x_2, \ldots, x_m) > 0$. Then from Corollary 9 $\varphi_{\mathfrak{M}}^A(x_1, x_2, \ldots, x_m) = 1$, so using Corollary 10 A is a tautology. $\square$

Let us proof Theorem 8.

Proof. ($\Leftarrow$) If for any $x = (x_1, x_2, \ldots, x_m) \in \langle 0,1 \rangle^m$ $\varphi_{\mathfrak{M}}^A(x) \geqslant \frac{1}{2^k}$, that $\varphi_{\mathfrak{M}}^A(x) > 0$. Using Lemma 12, A is a tautology.

($\Rightarrow$) Let A be a tautology and let $x = (x_1, x_2, \ldots, x_m) \in \langle 0,1 \rangle^m$. Let us create a new vector of assignment $\epsilon = (\epsilon_1, \epsilon_2, \ldots, \epsilon_m) \in \langle 0,1 \rangle^m$ that

$$\epsilon_i = \begin{cases} 0 & \text{if } x_i \leqslant \dfrac{1}{2} \text{ for } i = 1 \ldots m, \\ 1 & \text{if } x_i > \dfrac{1}{2}. \end{cases} \tag{3}$$

Because $\epsilon = (\epsilon_1, \epsilon_2, \ldots, \epsilon_m) \in \{0,1\}^m$ therefore using Corollary 10:

$$\varphi_{\mathfrak{M}}^A(\epsilon_1, \epsilon_2, \ldots, \epsilon_m) = 1.$$

Using (1) A is in a form

$$A = A_1 \vee A_2 \vee \ldots \vee A_n.$$

Therefore

$$\varphi_{\mathfrak{M}}^A = \max(\varphi_{\mathfrak{M}}^{A_1}, \varphi_{\mathfrak{M}}^{A_2}, \ldots, \varphi_{\mathfrak{M}}^{A_n}).$$

Because $\varphi_{\mathfrak{M}}^A(\epsilon) = 1$, then must exist $i(1 \leqslant i \leqslant n)$ that $\varphi_{\mathfrak{M}}^{A_i}(\epsilon) = 1$. Using (2),

$$A_i = B_i^1 \wedge B_i^2 \wedge \ldots \wedge B_i^{k_i},$$

and therefore

$$\varphi_{\mathfrak{M}}^{A_i} = \varphi_{\mathfrak{M}}^{B_i^1} \varphi_{\mathfrak{M}}^{B_i^2} \cdots \varphi_{\mathfrak{M}}^{B_i^{k_i}}.$$

Because $\epsilon = (\epsilon_1, \epsilon_2, \ldots, \epsilon_m) \in \{0,1\}^m$ for any $j(1 \leqslant j \leqslant k_i)\varphi_{\mathfrak{M}}^{B_i^j}(\epsilon) = 1$ using corollary 1 and assumption $\varphi_{\mathfrak{M}}^{A_i}(\epsilon) = 1$. Because $B_i^j = \ulcorner p \urcorner$ or $\ulcorner \sim p \urcorner$, where $p \in \{p_1, p_2, \ldots, p_m\}$ for $1 \leqslant i \leqslant n, 1 \leqslant j \leqslant k_i$, therefore exist $t(1 \leqslant t \leqslant m)$ that

$$\varphi_{\mathfrak{M}}^{B_i^j} = \begin{cases} w(p_t) & \text{if } B_i^j = \ulcorner p_t \urcorner, \\ 1 - w(p_t) & \text{if } B_i^j = \ulcorner \sim p_t \urcorner. \end{cases}$$

Because $\varphi_{\mathfrak{M}}^{B_i^j}(\epsilon) = 1$, therefore $\varphi_{\mathfrak{M}}^{B_i^j}(\epsilon_t) = 1$

$$\varphi_{\mathfrak{M}}^{B_i^j}(\epsilon_t) = \begin{cases} \epsilon_t & \text{if } B_i^j = \ulcorner p_t \urcorner \text{ or when } \epsilon_t = 1, \\ 1 - \epsilon_t & \text{if } B_i^j = \ulcorner \sim p_t \urcorner \text{ or when } \epsilon_t = 0. \end{cases}$$

Therefore

$$\varphi_{\mathfrak{M}}^{B_i^j}(x_t) = \begin{cases} x_t & \text{if } \epsilon_t = 1, \\ 1 - x_t & \text{if } \epsilon_t = 0. \end{cases}$$

Because $x_t > \frac{1}{2}$ when $\epsilon_t = 1$ and $x_t \leqslant \frac{1}{2}$ when $\epsilon_t = 0$ using construction (3) so

$$\varphi_{\mathfrak{M}}^{B_i^j}(x_t) \geqslant \frac{1}{2}$$

and further for any i $(1 \leqslant i \leqslant n)$:

$$\varphi_{\mathfrak{M}}^{A_i}(x) = \varphi_{\mathfrak{M}}^{B_i^1} \varphi_{\mathfrak{M}}^{B_i^2} \dots \varphi_{\mathfrak{M}}^{B_i^{k_1}} \geqslant \frac{1}{2^{k_i}}.$$

Because

$$\varphi_{\mathfrak{M}}^{A} = \max(\varphi_{\mathfrak{M}}^{A_1} \varphi_{\mathfrak{M}}^{A_2} \dots \varphi_{\mathfrak{M}}^{A_n}),$$

that

$$\varphi_{\mathfrak{M}}^{A}(x) \geqslant \varphi_{\mathfrak{M}}^{A_i}(x) \geqslant \frac{1}{2^{k_i}} \geqslant \frac{1}{2^k},$$

where $k = \max_{i=1\dots n} k_i$. What ends the proof of the theorem. $\square$

This theorem demonstrates that the satisfiability test of a logical expression in the discrete model of logic $\mathfrak{N}$ is equivalent to finding extremum of the continuous function φ in the area $\langle 0, 1 \rangle^m$ generated by that expression in the structure $\mathfrak{M}$. Summing up, we are looking for the smallest value of a function knowing that it is 0 in the case that its generating statement is false or greater than 0 when the expression is true, with additional information about its lower limit resulting from the above statement.

Using the mathematical analysis of continuous property we know that the smallest value of a function can be realized at the points of its extremes inside the examined area or on its border.

Since every expression in the normal class $[n, k]$ $(k > 3)$ can be reduced to a task in the form of a normal class $[n_1, 3]$ for $n_1 > n$, where numerical tests are reduced to testing only this class.

4. Some numerical experiments

In order to test the effectiveness of the developed method of investigating the fulfillment of logical expressions with the function generated by the expression, the following algorithm was used. Algorithm in pseudo-code

data:

m – the number of atomic formulas which can built A,

n – the number of conjunctives in A,

k – the number of atomic formulas in each conjunctive,

s – the number of particles,

T_{max} – maximal number of iterations.

Generate random (testing) formula A in a normal form of the class $[n, k]$.

Generate random particles $x_1, x_2, \ldots, x_s$:

---

    while $t < T_{\max}$ **do**

2:    Evaluate $\varphi^A(x_1), \varphi^A(x_2), \ldots, \varphi^A(x_s)$

    Find minimum of $\{\varphi^A(x_1), \varphi^A(x_2), \ldots, \varphi^A(x_s)\}$

4:    **if** minimum $< \frac{1}{2^k}$ **then**

      A is "false"; stop

6:    **end if**

    Create new particles $x_1, x_2, \ldots, x_s$ by PSO method

8: **end while**

Classic PSO method [7] build new particles (for $i = 1 \ldots s$):

$$x_i^{new} = x_i^{old} + \alpha(x_{\max}^{old} - x_i^{old}) + \beta(t) random([0, 1]^m),$$

where α and $\beta(t)$ are moderating coefficients.

Results of testing this algorithm in original classic version of PSO were unsatisfied. Only some random cases were positive. More of tests were divergent.

After many experiments one of the best has the following form.

If $x_i^t = (x_{i,1}^t, x_{i,2}^t, \ldots, x_{i,m}^t)$ is an old position of particle that construction of new particle is (for $l = 1 \ldots m$):

$$x_{i,l}^{new} = \begin{cases} 1 - \beta(t) rnd[0, 1] & \text{if } x_{i,l}^{old} < \frac{1}{2}, \\ 0 + \beta(t) rnd[0, 1] & \text{if } x_{i,l}^{old} \geqslant \frac{1}{2}, \end{cases}$$

when $\varphi(x_i^{old}) \geqslant \varphi(x_{best}^{old})$.

We can also accelerate the process if one or more of the started particles are $(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2})$.

Since each expression in the normal form $[n, k]$ $(k > 3)$ can be reduced to an expression in the form of a normal class $[n_1, 3]$, where $n_1 > n$ numerical tests are reduced to testing only class $(k = 3)$.

Hint: It is also good if the number of particles is equal of the number of atomic formulas $(m = s)$ After these changes we have the next statistic results of constructed algorithm.

Table 1

Benchmark testing – number of iterations
depending M and N

M \ N	30	35	40	45
8	10.66	16.35	20.15	25.35
10	10.13	18.46	34.02	46.79
12	6.52	15.52	32.58	59.21
14	5.19	12.05	25.00	47.67

The array represents the average amount of iterations after which the algorithm identifies the expression as non-satisfiability. Parameters of this statistics are: M – the number of atoms in the tested expression, N – the number of alternative members i.e. the number of different triple conjunctions (clouses) in the expression. Tests were performed for randomly generated 1000 false statements for each pair of parameters.

5. Conclusion

Equivalence of SAT examinations – a discrete problem in the $\mathfrak{N}$ model with the search for the minimum value of the continuous function φ in the $\mathfrak{M}$ structure opens the way to using heuristic algorithms to solve both problems. The problem, however, is to find an effective algorithm, which briefly signals the example discussed above. This, however, opens up a wide field for further research both in constructing new algorithms and for finding other discrete structures and their continuous extensions similar to those presented in the article.

Looking from another perspective, the functions generated by logical statements can become very demanding test functions for evaluating heuristic algorithms.

We can also risk the assertion that, similarly to the discrete-matter classification, it is also possible to classify the functions of the NP class as exemplified by the functions of the class NP, which are examples of functions generated by expressions and others that are difficult to judge at the current level of research.

References

1. Gu J.: *The multi-SAT algorithm.* Discrete Appl. Math. **96–97** (1999), 111–126.

2. Battiti R., Protasi M.: *Approximate algorithms and heuristics for MAX-SAT.* In: Handbook of Combinatorial Optimization, vol.1., Du D.-Z., Pardalos P.M. (eds.), 77–148. Kluwer Acad. Publ., Boston 1998.

3. Gottlieb J., Marchiori E., Rossi C.: *Evolutionary algorithms for the satisfiability problem.* Evolutionary Comput. **10** (2002), 35–50.

4. Malitz J.: *Introduction to Mathematical Logic.* Springer Verlag, New York 1979.

5. Grzegorczyk K.: *An Outline of Mathematical Logic.* PWN, Warszawa 1985 (in Polish).

6. Balcer M.: *Characteristics and decomposition of expressions in the PF-notation.* Silesian J. Pure Appl. Math. **6** (2016), 5–22.

7. Kennedy J., Eberhart R.: *Particle swarm optimization.* Proc. IEEE Int. Conf. Neural Networks vol.IV, Perth 1995, 1942–1948.