

# A Clonal Selection Algorithm Based Tabu Search for Satisfiability Problems

Abdesslem Layeb

Computer Science Department - Mentouri University Constantine, Algeria.

Layeb.univ@gmail.com

**Abstract**— We present in this paper a new memetic algorithm to deal with the Max Sat problem. The objective is to find the best assignment for a set of Boolean variables, which gives the maximum of verified clauses in a Boolean formula. Unfortunately, this problem has been shown to be NP-hard if the number of variables per clause is greater than 3. The proposed approach is based on clonal selection principles and local search method. In order to increase the performance of clonal selection to deal with Max Sat problem, an adaptive fitness function based on weighted clauses has been used. The underlying idea is to harness the optimization capabilities of clonal selection algorithm to achieve good quality solutions for Max SAT problem. A local search based on Tabu Search has been embodied in the clonal selection algorithm leading to an efficient hybrid framework which achieves better balance between exploration and exploitation capabilities of the search process. The obtained results are very encouraging and show the feasibility and effectiveness of the proposed hybrid approach.

**Index Terms**—Maximum Satisfiability problem, Artificial Immune System, Clonal Selection, Tabu Search.

## I. INTRODUCTION

The combinatorial optimization plays a very important role in operational research, discrete mathematics and computer science. The aim of this field is to solve several combinatorial optimization problems that are difficult to solve. The propositional satisfiability problem (SAT) is among the well-known combinatorial optimization problems in the literature. It is the task of determining the satisfiability of a given Boolean formula by looking for the variable assignment that makes this formula evaluating to true [1]. The SAT problem is very important in many fields. In fact, many problems are easily representable in propositional logic areas such as model checking [2], graph coloring [3] and task planning[4] to cite just few.

In some cases, we want to know how many clauses are true. This is another instance of the Sat problems called MAX-SAT (MAXimum SATisfiability). In recent years, there is growing interest in this problem, given its use in a wide range of domains. For a propositional formula in CNF, the problem is to determine the maximum number of clauses of this formula that can be fulfilled. An instance of this problem is the problem Max k-Sat [5]. This problem relates to instances of clauses with at most k literals. Max 3-SAT is special case of Max k-Sat

problems where the Boolean expressions are written in CNF (Conjunctive Normal Form) with 3 variables per clause. The problem Max k-Sat has been shown to be NP-complete for any  $k > 2$  [5]. There are other variants of the Max-SAT problem such as Weighted Max-SAT [6] and Partial Max-SAT [7].

To solve the Max Sat problem, many algorithms were proposed. In fact, there are two classes of algorithms for solving instances of Max-Sat in practice: exact and heuristics methods. The exact algorithms are able to find the exact solution of the Max Sat problem, although they have an exponential complexity [8]. The most popular algorithms of this class are based on the Davis-Putnam-Loveland algorithm (DPLL) [9, 10]. On the other hand, the metaheuristic methods find an optimal solution in good time, but they don't guarantee to give the exact solution of the problem. This class of methods contains Evolutionary Algorithms (EA) [11, 12], Stochastic Local Search (SLS) methods [13, 14, 15] and hybrid methods of EA and SLS or Exact methods [16, 17, 18, 19].

Evolutionary computation has been proven to be an effective way to solve complex engineering problems. It presents many interesting features such as adaptation, emergence and learning. Artificial neural networks, genetic algorithms and artificial immune systems are examples of bio-inspired systems used to this end. An Artificial Immune System (AIS) [20, 21, 22] is a type of optimization algorithm inspired from the principles and processes of the vertebrate immune system. One of the most used variant of AIS is the Clonal Selection Algorithms (CSA) [23, 24]. They are most usually applied to optimization and pattern recognition problems. However, the use of a pure clonal selection algorithm for satisfiability problems is not good enough. To overcome this weakness, the integration of the Stochastic Local Search (SLS) is required in order to increase the performances of the evolutionary algorithms to deal with the SAT problems.

The stochastic local search methods were demonstrated to be useful in solving many complex problems. Tabu Search (TS) [25] is among the most popular and robust local search methods. TS is found to be practical in many hard combinatorial optimization problems. The main idea underlying is to diversify the search and avoid becoming trapped in local by forbidding or penalizing moves which take the solution, in the next iteration, to point in the solution space previously visited.

For this, the algorithm creates a memory list of undesirable moves called "tabu list". However, Tabu restrictions may be overruled under certain conditions, in which, a Tabu move leads to a better solution, this principle is called aspiration criterion. Tabu search algorithms are amongst the most effective local search based methods for the Max Sat problems [14, 26]. The main difficult of the tabu search is the choice of the best parameter setting.

The present study was designed to investigate the use of clonal selection algorithm hybridized with tabu search to deal with the maximum satisfiability problem. The proposed approach is a population based method where every individual represents a potential solution to the problem at hand. The features of the proposed method consist in applying different immune system principles like clonal selection and mutation to govern the dynamics of the population in a way to optimize an adaptive objective function based on the clause weights. To foster the convergence to optimality, a local search based on tabu search has been embodied within the optimization process.

The remainder of the paper is organized as follows. In section 2, a formulation of the tackled problem is given. Section 3 presents a state of art on the Max sat solvers. Some basic concepts of Artificial Immune Computing are presented in section 4. In section 5, the proposed method is described. Experimental results are discussed in section 6. Finally, conclusions and future work are drawn.

## II. PROBLEM FORMULATION

Throughout this paper,  $n$  will represent the number of Boolean variables and  $m$  will denote the number of clauses in formula  $F$ . Given a Boolean formula  $F$  expressed in CNF and having  $n$  Boolean variables  $x_1, x_2, \dots, x_n$ , and  $m$  clauses. The k-SAT problem can be formulated as follows:

- An assignment to those variables is a *vector*  
 $v = (v_1, v_2, \dots, v_n) \in \{0, 1\}^n$
- A clause  $C_i$  of length  $k$  is a disjunction of  $k$  literals,  $C_i = (x_1 \vee x_2 \vee \dots \vee x_k)$
- Each literal is a variable or a negation of a variable
- Each variable can appear multiple times in the expression.

If each clause  $C_i$  in the CNF formula is a disjunction of  $k$  literals  $C_i = (x_1 \vee x_2 \vee \dots \vee x_k)$ , the problem is called k-SAT. Another important variant of satisfiability problem is the Max k-Sat, this is a combinatorial optimization problem. In this problem, the objective is to find the best assignment  $v \in \{0, 1\}^n$  that maximizes the number of satisfied clauses in the Boolean formula. Therefore, it is categorized as an optimization problem. The Max-Sat problem can be defined by specifying implicitly a pair  $(\Omega, SC)$ , where  $\Omega$  is the set of all potential solutions  $(\{0, 1\}^n)$  and  $SC$  (SCore) is a function  $\Omega \rightarrow \mathbb{N}$ , called score of the assignment, equal to the number of true clauses. Consequently, the problem consists of defining the best binary assignment that maximizes the number of true clauses in the Boolean formula.

Clearly, there are  $2^n$  potential satisfying assignments for this problem, and it has been proven that the k-SAT problem is NP-complete for any  $k > 2$ . In this paper we deal with the Max 3-SAT problem. It is obviously that this problem is a combinatorial optimization problem. It appears to be impossible to obtain exact solutions in polynomial time. The main reason is that the required computation grows exponentially with the size of the problem. Therefore, it is often desirable to find near optimal solutions to these problems.

## III. MAX 3-SAT SOLVERS STATE-OF-THE-ART

Modern Max-Sat solvers have deeply improved the techniques and algorithms to find optimal solutions. In practice there are two broad classes of algorithms for solving instances of SAT: Complete and Incomplete methods. The complete methods examine the entire search space, they fully answer for the problem of satisfiability of an instance. In the worst case, the computation time required for the execution of such methods, increases exponentially with the size of the instance to solve. The method of Davis and Putnam (DP) [27] is one of the first methods dedicated to solving SAT problem. The DP method is based on the application of the resolution rule in order to eliminate variables. The procedure of Davis, Putnam, Logemann and Loveland (commonly known DPLL) is an enhancement of the DP algorithm. DPLL escapes the exponential clause generation and the high memory usage of DP. This method replaces the resolution mechanism for the separation of the problem into two problems. The idea is to construct a binary tree in which each node represents a DPLL recursive procedure. Each leave represents the search stopping unless an empty clause if the formula is satisfiable. The process search uses the backtracking technique to go back in the search tree and thus to test new branches. The DPLL takes advantage of two techniques that improve the selection of assignments to variables: the pure literal elimination and the unit propagation [16]. Currently, the most of exact methods are based mainly on the DPLL procedure [28, 29].

The fact that complete algorithms require huge amount of computing time, even for relatively small problem instances, incomplete methods have been developed to resolve SAT or Max-Sat problems of large size in reasonable time. Incomplete methods do not guarantee finding the best solutions, because they do not explore the entire space of possible solutions. However, the incomplete methods are particularly suitable large-scale practical Max-Sat problem instances. They are able to find quickly a good solution towards the cost and constraints of the problem. The incomplete methods are principally based on metaheuristics such as: local search methods, evolutionary algorithms or hybrid methods. The local search methods are widely used to solve Max 3-Sat problems [5]. Among the popular methods of this class, we can cite the popular method GSAT [13] and its improved variant Walksat [30]. GSAT starts with a random assignment and iteratively apply a set of flips by using a specific heuristics in order to enhance the number

of satisfied clauses. Walksat is an evolution of GSAT in which the heuristics used to select the variable to flip are improved. The main difference lies in the estimated size of the neighborhood: Walksat greatly reduces the number of neighbors as it selects a false clause, and only one variable of this clause may be flipped. One biggest improvement to GSAT method is adopting the strategy of "random walk" while solving the problem. This strategy allows us to accelerate the convergence and allow avoiding some local minima. The idea of this strategy is to choose a variable to flip among the variables involved in randomly selected false clause. Unfortunately, Walksat suffers from local minima problem. To overcome this problem, Walksat insert some randomness in the search process. Walksat swaps between greedy moves and noisy moves. Those moves are randomly chosen from the variables that appear in unsatisfied clauses. New heuristics were integrated in Walksat to choose a variable of the clause to flip such as Random, Best Novelty, Rnovelty and Tabu [31].

Evolutionary algorithms have been applied to SAT problems. However, the use of a pure evolutionary algorithm has been unsuccessful in Max Sat problems. In fact, Rana and Whitley [32] showed that a classical genetic algorithm is unsuitable for Max 3-Sat problem, because this later requires more intensification search than diversification search. Therefore, hybridization between evolutionary algorithms and local search methods are needed to find success. This kind of hybridization is called memetic algorithms. Recently, several memetic algorithms were proposed to deal with the Max Sat problems for example: FlipGA [11] based on hybrid genetic algorithm and a special flip heuristics; GASAT [12] based on hybrid genetic algorithm and tabu search, QGASAT [15] based on quantum evolutionary algorithm and local search procedure.

Recently, a new kind of hybridization based on incomplete and complete algorithms has emerged. Hybridization consists to use heuristics inspired DPLL for local search heuristics, and reciprocally the local search methods for DPLL algorithm. In such combinations, the incomplete methods are used to quickly identify promising areas of search space in which the exact methods practiced an exhaustive exploration. There are several methods in the literature based on this kind of hybridization [16, 17, 18].

#### IV. ARTIFICIAL IMMUNE SYSTEMS

In order to solve complex problems, ideas gleaned from natural mechanisms have been exploited to develop heuristics. Nature inspired optimization algorithms has been extensively investigated during the last decade paving the way for new computing paradigms such as neural networks, evolutionary computing, artificial immune systems, etc.

A Natural Immune System (NIS) is responsible for defending the body against viruses and other infections. The natural immune system is a complex system that can

be analyzed at different levels: molecules, cells and organs. The immune system is divided into two major defense systems: the innate immune system, which represents the immune defenses present from the birth and the adaptive immune system which represents all defenses learned over time. Complex interactions between entities within each level enable the resulting immune system to protect the body from any harmful entity, or exogenous agent, called antigen. A specific kind of cells, known as B-cells, is responsible for the destruction of the antigen (Fig. 1). The B-cell produces antibodies that binds with the antigens and marks them for destruction. The strength of the antibody/antigen binding is termed antigenic affinity [20, 21]. Interesting features of the immune system have encouraged their adaptation to information technology for solving real world problems.

Artificial immune system (AIS) is a recent paradigm that attempts to capture interesting features of the systems natural immunity, such as memorization, pattern recognition, learning and skills adaptation. An artificial immune system has been defined as an adaptive system inspired by the immune system biological problem solving [16, 17, 18]. Artificial immune systems can be viewed as a composition of intelligent methodologies inspired from natural immune systems for solving real world problems. Many models have been inspired from natural immune systems to solve various problems.

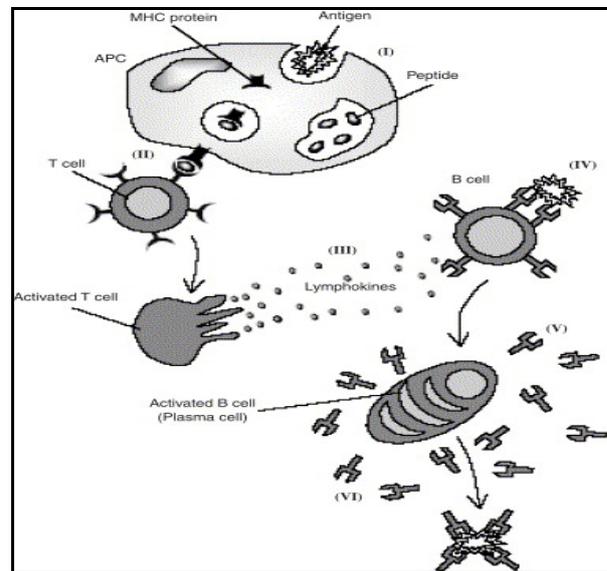


Figure. 1 Immune response

One particular feature of the natural immune system is its ability to construct specific antibodies against new antigens. The clonal selection principle [23, 24] handles this as follows: when a new antigen is detected, the available B-cells start producing antibodies. Those B-cells that best recognize the antigen proliferate by cloning. The clones then undergo hyper-mutation mechanisms to promote their genetic variation. B-cells

with high affinity differentiate into plasma cells and memory cells, and B-cells with low affinity are either destroyed or mutated. Plasma cells produce a high number of antibodies against the invading antigen (Fig. 2). Memory cells, on the other hand, are long living cells that confers the system a memory of the encountered antigen.

Artificial Clonal selection is an artificial immune system particularly based on clonal selection theory. CLONALG (Clonal selection Algorithm) is a popular algorithm that implements the clonal selection algorithm [23, 24, 33]. It has been developed not only for realizing machine learning and pattern recognition tasks but also for solving optimization problems. A basic clonal selection algorithm is given by figure 3.

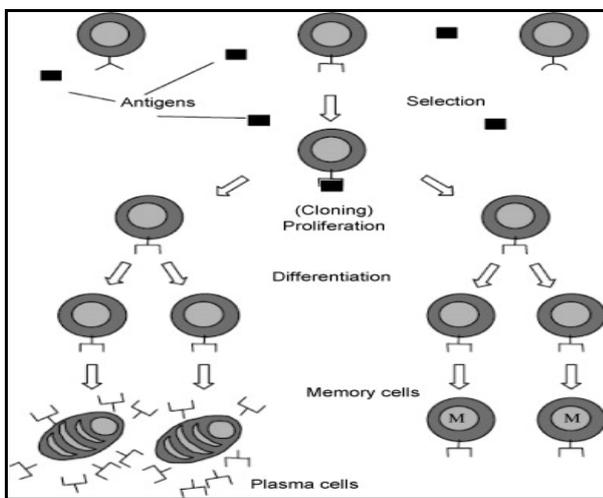


Figure. 2 The process of clonal selection.

```

Step 1  $k = 0$ ; Initialize randomly antibody population  $\mathbf{A}(0)$  then calculate the affinity of initialized population.
Step 2 Produce a new antibody population  $\mathbf{A}(k)$  by Clonal Selection Operator taking into account the antibody affinity and the clonal size.  $k = k + 1$ ;
Step 3 if the stopped condition is satisfied return the best antibody; else return to Step2.
    
```

Figure. 3 A simple schema of clonal selection algorithm.

The big problem encountered when using AIS to solve optimization problems is how to map the problem domain into an immune system. De Castro and Timmis [21] propose a framework to simplify the process of the correspondence between the biological immune systems and artificial immune systems. The core of this Framework is composed of the following three elements:

- *Representation of system components:* The representation of space search and the solutions.
- *Affinity function:* Similar to the fitness function in evolutionary algorithms. It's used to evaluate the interaction of components with the environment and between them.
- *Immune dynamics:* A set of adaptation procedures that govern the system dynamics. The adaptation procedures are determined according to the problem.

V. THE PROPOSED APPROACH

To solve the Max Sat problem, a new approach is proposed based on the clonal selection principles and enhanced by a local search procedure. In this approach, a new objective function is adapted based on the weight of clauses. In order to show how clonal selection concepts have been tailored to the problem at hand, we need first to derive a representation scheme which includes the definition of an appropriate clonal representation of potential Boolean assignments and the definition of clonal selection operators. Then, we describe how these defined concepts have been combined with local search algorithm to deal with the Max 3-Sat problem. For correspondence with the immune system, an antigen represents Max 3-Sat problem environments and the antibodies correspond to different solutions.

A. Clonal representation of Max Sat solutions

To successfully apply clonal selection principles on Max 3-Sat problem, we have needed to map potential solutions into a clonal representation that could be easily manipulated by clonal selection operators. We used a binary representation to encode B-cells which represent potential solutions. There are several possible representation of the Max 3-Sat space search like the floating point representation and the clausal representation. However the most successful evolutionary algorithms for the Sat problems use the bit string representation. Thus, one potential solution is represented by a binary vector of size  $n$  where  $n$  is the number of Boolean variables in the CNF formula (Figure. 4). The digit 0 digit represents a false value and the digit 1 represents true value. The initial solutions are randomly generated. However, we can start with initial solutions obtained by using greedy heuristics in order to accelerate the optimization process.

0	1	0	1	1	0	1	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---

Figure. 4 Binary representation of B-cell

B. The hypermutation process

Its goal is to maintain the maximum diversity among antibodies that compose the memory set. The hypermutation process consists in changing the structure of an antibody  $Ab$  by applying  $p$  random mutations in order to improve its affinity. The result is a mutated antibody  $Ab_m$  that's generally better than the original antibody  $Ab$ . Several hypermutation techniques are

reported in the literature [21, 22, 23]. In the proposed approach, we have used the inversely proportional hypermutation. In this kind of hypermutation, antibodies that have high affinity are mutated slightly, while antibodies that have poor affinity undergo hypermutation process most important.

### C. Selection strategy

The selection is to select antibodies that have the best affinity to construct the intermediate population in order to apply the cloning process and the hypermutation. Each antibody is selected with a probability related to its affinity. Thus, antibodies with high affinity value have more chance of being selected for the next generation. Selection strategies that we can use in clonal selection algorithms are the same as those applied in the evolutionary algorithms. In our approach, we have used the roulette wheel selection. Classical roulette wheel selection provides higher chances of being selected to fitter solutions. It is commonly regarded as an intensification component, contrary to diversification operators.

### D. Replacement strategy

After the hypermutation process, new antibodies are introduced into the population. In order to keep the population size constant from generation to generation, a replacement operator is used. There are different strategies taking into account the memory component. After replacement, some antibodies are removed and replaced by randomly generated new one to diversify the population. Although, the affinity of the antibody is low, its chance to be replaced is great.

### E. Adaptive Affinity

The affinity between the antibody and antigen measures the compatibility degree between them. This measure is very important because it helps to determine if a solution is enough good and generally has a great impact on the performance of immune algorithms. From a computational point of view, the affinity is usually related to the distance. However, in the case of Max 3-Sat problem and optimization problems in general, we use appropriate evaluation functions to solve the problem. The choice of the affinity measure is generally reliant on the represented problem. Thus, for the Max 3-sat problem, several fitness functions [11] were proposed to be used by the evolutionary algorithms. The standard evaluation function used in the most incomplete methods is the number of satisfied clauses. The objective is to maximize the number of the true clauses. Unfortunately, this kind of objective function cannot lead an evolutionary algorithm to find the best solution. In fact, there are many different solutions having the same quality concerning the Max 3-sat evaluation function, which makes it hard for an evolutionary algorithm to select the appropriate solution leading to the best solution [11]. This observation is obviously clear when dealing with the satisfiable SAT instances. For this reason, we

have used an adaptive affinity function based on the mechanism of *stepwise adaptation of weights* (SAW) introduced by Eiben et al. [34]. The affinity function is given by the following formula:

$$F(x) = \sum_{i=1}^m W_i C_i(x) \quad (1)$$

Each clause  $C_i$  has a weight  $W_i$ . The affinity function is adapted by modifying the weights  $W_i$  appropriately in order to identify the hard clauses in the preceding search process. A high weight  $W_i$  indicates that the clause  $C_i$  is difficult and leads to a higher fitness of those solutions that satisfy  $C_i$ . Consequently, the weights lead the search process towards solutions satisfying the most difficult clauses. In the beginning all weights are initialized by  $W_j = 1$ , and later the weights are adjusted according to following formula ( $x^*$  is the current best antibody):

$$W_{i+1} = W_i + 1 - C_i(x^*) \quad (2)$$

### F. Local Search

The study of the behavior of the clonal selection algorithm with simple hypermutation for Max-Sat problem [19] showed that the convergence speed is slow. In order to increase intensification capabilities of search, we apply a local search algorithm when the memory cell has not been improved for more than a given number of generations. In [19], the authors have integrated the walksat method in the core of the clonal selection algorithm in order to improve its performances. However, the experiments showed that the performance to find the exact solutions for satisfiable instances is low and the convergence speed is slowed down. In this section, we show how a clonal selection algorithm can be greatly improved by hybridizing with a new local search based on tabu search. The optimization process is governed by the affinity function SAW (eq.1) in order to focus on difficult clauses and to escape from local optima.

The local search used in our approach is hybridization between the Flip heuristic used in flipGA [11] and the tabu search. Flip heuristic consists in flipping the variables of the Boolean formula (fig. 6). The flip is accepted if there is improvement in the number of satisfied clauses. The process is repeated until there is no improvement. Unfortunately, the big problem of this heuristic is the large neighborhood size. To overcome this problem, we have introduced the tabu search principles in order to reduce the number of neighbors. We put in the tabu list each variable that does not improve the affinity function. In more detail, the tabu search method is given by the figure 5.

The tabu variables are stocked in ordered FIFO list and they are prevented to being flipped again during a given number of iterations. The tabu list length is set empirically according to the number of variables in the Boolean formula. Based on the experiments of Mazure [26], the tabu list length is set to 10% of the number of

variables. Concerning the aspiration criterion, we have not used in our search local any aspiration condition because it can slow down the approach without improving the results. However, we suggest using a good adaptive aspiration criterion in order to escape from the local minima.

---

```

Input: B-cell (Boolean assignment),
formula  $\phi$  in CNF, maxflip.

```

---

```

improvement=1;
nbrflip=0 ;
repeat
improvement=0;
For i=1 to nVar
if ( var i  $\notin$  tabuList){
    flip the i-th variable of the B-
cell;
    nbrflip++;
    compute the gain of flip;
    if (gain>=0){
        accept the flip;
        improvement= improvement +gain; }
    Else {
        tabuList.push(var i);
        undo the flip of i-th variable
        if(tabuList.size()==tabuListMax)
        tabueList.pop (); }
    End For
Until (improvement > 0 && nbrflip <
maxflip)

```

---

```

output: Boolean assignment VB

```

---

Figure. 5 Local search procedure.

### G. Outline of the proposed framework

Now, we describe how the representation scheme including clonal representation and clonal selection operators has been hybridized with local search algorithm. The whole process is repeated until the stopping criterion is met; the number of generations is used as stopping criterion. In more details, the proposed approach can be described as in Figure 6.

Firstly, a population of antibodies is initialized using binary strings, each encoding a given solution. The population size is important factor in evolutionary algorithms. A large population represents a large space search of solutions, but increases the computational cost. On the other hand, a small population size can lead to local solutions. Seen, we use a local search method in the core of clonal selection algorithm, it is preferably to reduce the population size. We have found that a population between 10 and 30 can give good results.

---

```

Input: Boolean formula in CNF form A

```

---

```

Initialization: Generate a population
of B-cells called pop.

```

---

```

Repeat for G generations

```

1. Evaluate the affinity of the elements of pop.
2. Select N best B-cells from pop with respect to their affinities. Then clone this selected B-cells to produce a clone population.
3. Mutate each clone to produce a mature population.
4. Apply the local search method
5. Evaluate the affinity of the mature population and select the best B-cell as a candidate cell. If it is better than the memory cell, it becomes the new memory cell.
6. Replace all the B-cells of pop with the best ones from the mature population. Make sure that pop contains the memory cell.
7. Replace the worst elements of pop with new randomly generated ones.

---

```

Output: best variable assignment,
number of satisfied clauses

```

---

Figure. 6 The proposed approach for max 3-sat problem

At each iteration, our method begins by evaluating the affinity of each B-cell. The affinity value is the sum of weighted satisfied clauses in the Boolean formula (eq. 1). Next, Antibodies in the population will be duplicated proportional to their affinity, the N selected B-cells, using roulette wheel mechanism, are allowed to clone themselves. The better the B-cell, the more clones it will produce. The number of clones for each B-cell is calculated by using the following formula:

$$nbClones_i = \frac{affinity_i}{\sum_j affinity_j} \times \beta \tag{3}$$

Where  $affinity_i$  is the affinity of the  $i^{th}$  selected B-cell and  $\beta$  is a parameter that indicates the desired clone population's size.

After the population cloning, the mutation process starts which transforms the clones to a degree inversely proportional to their affinity. The mutation process is as follows:

- a. For each clone  $clone_i$ ; compute its normalized affinity:

$$affN_i = \frac{aff_i - \min Aff}{\max Aff - \min Aff} \quad (4)$$

$\min Aff$  and  $\max Aff$  are the minimal and maximal affinities found in the *clone* population.

- b. Calculate the number of mutations to apply to *clone*<sub>*i*</sub>:

$$nbMutations_i = affN_i \times \min + (1 - affN_i) \times \max \quad (5)$$

The *min* and *max* parameters indicate how many mutations to apply to the worst and best individual respectively.

- c. For each mutation of *nbMutations*<sub>*i*</sub> a random bit is flipped.

However, we have noticed that the use of simple flip mutation is inefficient for Max 3-Sat. In order to enhance the capacity of mutation, we apply the local search procedure. The local search is applied after a specified number of generations passes without any improvement in the population's fitness.

After the mutation step, we evaluate the affinity of the mature population and select the best B-cell as a candidate cell. If it is better than the memory cell, the candidate cell becomes the new memory cell. Finally, we replace all the B-cells of the current population with the best ones from both the current population and the mature population and we replace the worst elements of the current population with new randomly generated ones.

## VI. EXPERIMENTAL RESULTS

The developed method called WClonTS (Weighted Clonal selection with Tabu for Satisfiability problems) is implemented in Java and tested on a micro-computer 3 GHz and 1 GB of memory. To assess the experimental performance of WClonTS, several tests are conducted with instances taken from the AIM benchmark instances. The AIM instances are all generated with a particular Random-3-SAT instance generator [35]. The particularity is that the generator produces yes-instances and no-instances independently for wide ranges. The used AIM instances include yes-instances with low and high clause/variable ratios that have exactly one solution. We have used only the satisfiable benchmarks because we want to evaluate the capabilities of our approach to find the exact solutions for yes-instances.

To analyze the performances of the approach, we performed a comparison with other popular methods in the literature. We have compared WClonTS against two programs based on clonal selection algorithm: ClonsatW based on Walksat method [19] and ClonTS based on flip heuristic and the standard affinity function (number of true clauses). The purpose of this comparison is to show the impact of local search and the affinity functions on clonal selection optimization. Furthermore, we compared

our results with those of another evolutionary algorithm (QGASAT) which is based on a hybrid quantum evolutionary algorithm and flipping procedure [12]. We have made a comparison with three state-of-art stochastic local search programs: Walksat, Novelty [36] and an Iterated Robust Tabu Search for Max-Sat IROTS [37], for these programs, we have used the default parameters used in the program UBCSAT [31]. Finally, statistical tests of Freidman were carried out to test the significance of the difference in the accuracy of each method in this experiment. In all experiments, we have run the WClonTS, ClonsatW and ClonTS programs using the parameters' settings shown in Table 1.

The table 2 summarizes the obtained results. For each program, the best of ten consecutive runs is taken. The results found seem to be very promising and demonstrate the feasibility and efficiency of the combining an immune artificial system with local search based tabu search and the adaptive affinity function (Table 2). Indeed, the found results are similar to the exact solutions as it's confirmed by the Friedman test (Fig. 7). As we can see, except the program WClonSAT, all the other programs used in this experiment fail to find the exact solutions for the benchmarks *aim-100-1\_6-yes1-1:4* and *aim-100-2\_0-yes1-1:4*. The results show that the performance of the clonal selection algorithm with the local search based on Flip heuristics and tabu search is better than one based on walksat ClonSatW or flip heuristic with standard affinity function ClonTS. Compared to the second evolutionary algorithm used in this experiment QGASAT, WClonTS is obviously better. QGASAT is not competitive to deal with yes-instances; it ranks the last in the Freidman test (Fig. 7). By another hand, the experiment shows that the programs Walksat, Novelty+ and ClonTS have similar performances, but they rank third in the Freidman test. However, the results of the programs IROTS and ClonsatW are very close to the exact solutions.

TABLE 1 PARAMETERS OF PROGRAMS BASED CLONAL SELECTION

Parameter name	Parameter
population size	20
mutation min	1 / number-of-variables
mutation max	0.01
clone size ( $\beta$ )	30
replacement size	5
total number of generations before stopping	100
number generations before applying local search	10
Maxflip in Flip heuristic	40000

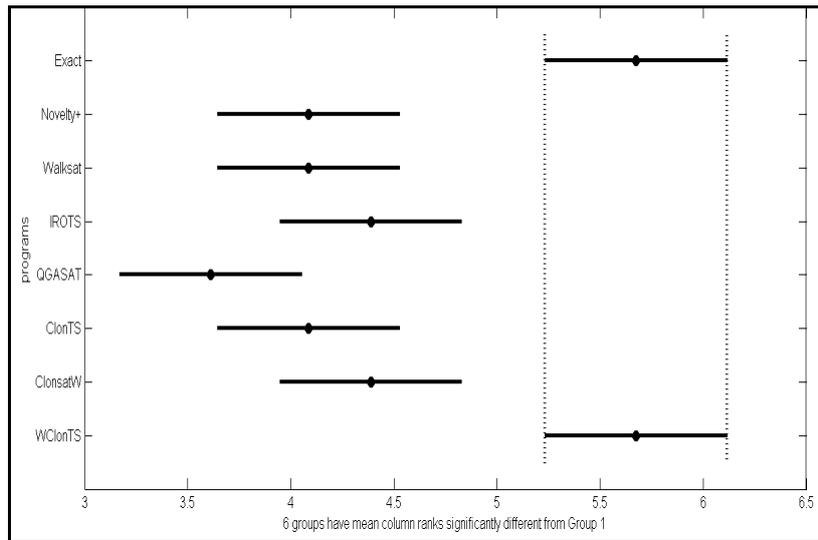


Figure. 7 Friedman test.

TABLE 2 COMPARATIVE RESULTS.

Tests	Novelty+	Walksat	IROTS	QGASAT	ClonTS	ClonsatW	WClonTS
aim-50-1_6-yes	79	79	79,75	79	79	79,75	80
aim-50-2_0-yes	100	100	100	99	100	100	100
aim-50-3_4-yes	170	170	170	170	170	170	170
aim-50-6_0-yes	300	300	300	300	300	300	300
aim-100-1_6-yes	159	159	159	159	159	159	160
aim-100-2_0-yes	199	199	199	199	199	199	200
aim-100-3_4-yes	340	340	340	339	340	340	340
aim-100-6_0-yes	600	600	600	600	600	600	600
aim-200-2_0-yes	399	399	399	399	399	399	400
aim-200-6_0-yes	1200	1200	1200	1200	1200	1200	1200

VII. CONCLUSION

In this paper, a new approach to solve the Max-3-Sat problem is proposed. The developed approach is based on hybridizing clonal selection algorithm with a new local search algorithm based on flip heuristic and tabu search. Our approach yields high quality solutions. It reaps advantage from characteristics of clonal selection algorithm and tabu search, better balance between intensification and diversification of the search is achieved. The use of the adaptive affinity function based on the clauses weights has increased the performance of our approach. According to the experimental results, the choice of the local search procedure is crucial for the effectiveness of the resulting algorithm. In most cases, our program gives comparable or better solutions than other programs of literature. In addition, the proposed framework provides an extensible platform for evaluating different variant of satisfiability problems. We envisage also improving our framework to implement a parallel hybrid clonal selection algorithm to reduce the CPU running time.

REFERENCES

- [1]. S.A. Cook, "The Complexity of Theorem Proving Procedures". In: *Proc. 3rd Ann. ACM Symp. On Theory of Computing, Association for Computing Machinery*, pp 151–158, 1971.
- [2]. A. Biere, A. Cimatti, E., Clarke, and Y. Zhu, "Symbolic model checking without BDDs", In *Proc. of the Workshop on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'99), LNCS, 1579: pp. 193–207,1999.*
- [3]. N. Bouhmala, and O. Granmo, "Solving Graph Coloring Problems Using Learning Automata". In *Proceedings of the Eighth European Conference on Evolutionary Computation in Combinatorial Optimisation, LNCS 4972, pp. 277–288, 2008.*
- [4]. H. Kautz, and B. Selman, "Pushing the envelope: planning, propositional logic and stochastic search," *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI 96), AAAI Press*, pp 1194 -1201, 1996.
- [5]. H. Kautz, and B. Selman, "The state of SAT,," *Discrete Applied Mathematics*, 155 (12):1514–1524, 2007.
- [6]. B. Borchers and J. Furman, "Two-Phase Exact Algorithm for MAX-SAT and Weighted MAX-SAT Problems," *Journal of Combinatorial Optimization*, Vol. 2, No. 4, pp. 299–306, 1999.

- [7]. M. Menai, M. Batouche, "A Backbone-Based Co-evolutionary Heuristic for Partial MAX-SAT," *Artificial Evolution*, pp 155–166, 2005.
- [8]. H. Zhang, H. Shen, "Exact Algorithms for MAX-SAT," *Electronic Notes in Theoretical Computer Science*, Vol. 86, No.1, 2003.
- [9]. M. Davis, G. Logemann, D. Loveland, "A machine program for theorem proving," *communication of the ACM*, pp 394–397, 1962.
- [10]. CM. Li, F. Manyà, J. Planes, "Exploiting unit propagation to compute lower bounds in branch and bound Max-SAT solvers," In: *Proc. of the 11th Int. Conf. on Principles and Practice of Constraint Programming, CP'05, Sitges, Spain*, pp. 403–414, 2005.
- [11]. J. Gottlieb, E. Marchiori, and C. Rossi, "Evolutionary algorithms for the satisfiability problem," *Evolutionary Computation*, Vol.10, No. 2, pp. 35–50, 2002
- [12]. A. Layeb, D. Saidouni, "A New Quantum Evolutionary Local Search Algorithm for MAX 3-SAT Problem," In *Proc., of the 3rd international Workshop on Hybrid Artificial intelligence Systems*. LNAI, Vol. 5271, pp. 172–179, 2008.
- [13]. E.S. Skvortsov, "A Theoretical Analysis of Search in GSAT," *Proceedings of the SAT*, pp. 265–275, 2009.
- [14]. M. Mastrolilli and L. M. Gambardella, "Maximum satisfiability: how good are tabu search and plateau moves in the worst-case?," *European Journal of Operational Research*, Vol. 166, No. 1, pp. 63–76, 2005.
- [15]. F. Lardeux, F. Saubion, J.K. Hao, "GASAT: A genetic local search algorithm for the satisfiability problem," *Evolutionary Computation*, Vol. 14, No. 2, pp. 223–253, 2006.
- [16]. Y. Kilani, "Comparing the performance of the genetic and local search algorithms for solving the satisfiability problems," *Applied Soft Computing*, Vol. 10, No. 1, pp. 198–207, 2010.
- [17]. A. Bhalla, I. Lynce, J. T. de Sousa and J. Marques-Silva, "Heuristic based backtracking for Propositional Satisfiability," In *Proceedings 11th Portuguese Conference on Artificial Intelligence (EPIA'03), Lecture Notes in Artificial Intelligence, vol. 2902*, pp. 116–130, 2003.
- [18]. L. Kroc, A. Sabharwal, C. P. Gomes and B. Selman, "Integrating systematic and local search paradigms: A new strategy for MaxSAT," *IJCAI*, pp 544–551, 2009.
- [19]. A. Layeb, A.H. Deneche, M. Meshoul, "A New Artificial Immune System for Solving the Maximum Satisfiability Problem," In: *N. Garcia-Pedrajas et al. (Eds.): IEA/AIE 2010, part II, LNAI*, Vol. 6097, pp. 136–142, 2010.
- [20]. D. Dasgupta, F. Nino, "Immunological Computation: Theory and Applications," *Auerbach Publications*, USA, 2008.
- [21]. L.N. De Castro and J. Timmis, "Artificial Immune Systems: A New Computational Intelligence Approach," *Springer*, UK, 2002.
- [22]. I. Aydin, M. Karakose, E. Akin, "Artificial immune classifier with swarm learning," *Engineering Applications of Artificial Intelligence*, Vol.23, No. 8, pp. 1291–1302 2010.
- [23]. L.N.D. Castro and F.J.V. Zuben, "Learning and optimization using the clonal selection principle," *IEEE Trans. on Evolutionary Computation*, Vol. 6, No. 3, pp. 239–251, 2002.
- [24]. V. Cutello, G. Nicosia, "The Clonal Selection Principle for in silico and in vitro Computing, in Recent Developments in Biologically Inspired Computing," In *De Castro LN, Von Zuben FJ (Eds)*, 2004.
- [25]. F. Glover, "Tabu Search for Nonlinear and Parametric Optimization (with links to genetic algorithms)," *Discrete Applied Mathematics*, Vol. 49, pp. 231–255, 1994.
- [26]. B. Mazure, L. Sais, E. Geroire, "Tabu search for SAT," In: *Proceeding of the AAAI-97/IAAI-97*, pp 281–285, 1997.
- [27]. M. Davis, H. Putnam, "A computing procedure for quantification theory," *Journal of the ACM*, Vol. 7, No. 3, pp. 201–215, 1960.
- [28]. M. W. Moskewicz, C. F. Madigan, Y. Zhao, L. Zhang, and S. Malik, "Chaff: Engineering an Efficient SAT Solver," In *Proceedings of 12th International Conference on Computer Aided Verification*, LNCS, Vol. 1855, 2001.
- [29]. L. Xu, F. Hutter, H.H. Hoos, and K. Leyton-Brown, "SATzilla: portfolio-based algorithm Selection for SAT," *Journal of Artificial Intelligence Research*, Vol. 32, pp. 565–606, 2008.
- [30]. B. Selman, H. Kautz, B. Cohen, "Local Search Strategies for Satisfiability Testing," *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, Vol. 26, pp. 521–532, 1993.
- [31]. D.A.D. Tompkins, H.H. Hoos, "UBCSAT: An Implementation and Experimentation Environment for SLS Algorithms for SAT and MAX-SAT," In: *Hoos HH and Mitchell DG (Eds.) SAT 2004*, LNCS, Vol. 3542, pp. 306–320, 2005.
- [32]. S. Rana, D. Whitley, R.B. Heckendorn, "A Tractable Walsh Analysis of SAT and its Implications for Genetic Algorithms," In *proceedings of the 15th National Conference on Artificial Intelligence*, pp 392–397, 1998.
- [33]. A. Layeb, A.H. Deneche, "Multiple Sequence Alignment by Immune Artificial System," *IEEE/ACS International Conference on Computer Systems and Applications*, pp. 336–342, 2007.
- [34]. J. Gottlieb, N. Voss, "Adaptive fitness functions for the satisfiability problem," In: *Parallel Problem Solving from Nature - PPSN VI 6th International Conference*, LNCS, Vol. 1917, 2000.
- [35]. Y. Asahiro, K. Iwama, E. Miyano, "Random Generation of Test Instanzas with Controlled Attributes," *DIMACS Series on Discr. Math. and Theor. Comp. Sci*, Vol. 26, pp. 377–394, 1996.
- [36]. D. McAllester, B. Selman, H. Kautz, "Evidence for invariants in local search," In: *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)*, pp. 321–326, 1997.
- [37]. K. Smyth, H.H. Hoos, T. Stützle, "Iterated Robust Tabu Search for MAX-SAT," In: *Canadian Conference on AI*, LNCS, Vol. 2671, pp.129–144, 2003.

**Abdesslem Layeb** is currently an associate professor in the department of computer science at the University of Constantine. He received his PhD degree in computer science from the University Mentouri of Constantine, Algeria. He is researcher in MISC laboratory. The author's major field of study is combinatorial optimization methods and its application to solve several problems from different domains like Bioinformatics, imagery, formal methods. He has many publications in theoretical computer science, bioinformatics, and formal methods.

# Call for Papers and Special Issues

## Aims and Scope

JAIT is intended to reflect new directions of research and report latest advances. It is a platform for rapid dissemination of high quality research / application / work-in-progress articles on IT solutions for managing challenges and problems within the highlighted scope. JAIT encourages a multidisciplinary approach towards solving problems by harnessing the power of IT in the following areas:

- **Healthcare and Biomedicine** - advances in healthcare and biomedicine e.g. for fighting impending dangerous diseases - using IT to model transmission patterns and effective management of patients' records; expert systems to help diagnosis, etc.
- **Environmental Management** - climate change management, environmental impacts of events such as rapid urbanization and mass migration, air and water pollution (e.g. flow patterns of water or airborne pollutants), deforestation (e.g. processing and management of satellite imagery), depletion of natural resources, exploration of resources (e.g. using geographic information system analysis).
- **Popularization of Ubiquitous Computing** - foraging for computing / communication resources on the move (e.g. vehicular technology), smart / 'aware' environments, security and privacy in these contexts; human-centric computing; possible legal and social implications.
- **Commercial, Industrial and Governmental Applications** - how to use knowledge discovery to help improve productivity, resource management, day-to-day operations, decision support, deployment of human expertise, etc. Best practices in e-commerce, e-commerce, e-government, IT in construction/large project management, IT in agriculture (to improve crop yields and supply chain management), IT in business administration and enterprise computing, etc. with potential for cross-fertilization.
- **Social and Demographic Changes** - provide IT solutions that can help policy makers plan and manage issues such as rapid urbanization, mass internal migration (from rural to urban environments), graying populations, etc.
- **IT in Education and Entertainment** - complete end-to-end IT solutions for students of different abilities to learn better; best practices in e-learning; personalized tutoring systems. IT solutions for storage, indexing, retrieval and distribution of multimedia data for the film and music industry; virtual / augmented reality for entertainment purposes; restoration and management of old film/music archives.
- **Law and Order** - using IT to coordinate different law enforcement agencies' efforts so as to give them an edge over criminals and terrorists; effective and secure sharing of intelligence across national and international agencies; using IT to combat corrupt practices and commercial crimes such as frauds, rogue/unauthorized trading activities and accounting irregularities; traffic flow management and crowd control.

The main focus of the journal is on technical aspects (e.g. data mining, parallel computing, artificial intelligence, image processing (e.g. satellite imagery), video sequence analysis (e.g. surveillance video), predictive models, etc.), although a small element of social implications/issues could be allowed to put the technical aspects into perspective. In particular, we encourage a multidisciplinary / convergent approach based on the following broadly based branches of computer science for the application areas highlighted above:

## Special Issue Guidelines

Special issues feature specifically aimed and targeted topics of interest contributed by authors responding to a particular Call for Papers or by invitation, edited by guest editor(s). We encourage you to submit proposals for creating special issues in areas that are of interest to the Journal. Preference will be given to proposals that cover some unique aspect of the technology and ones that include subjects that are timely and useful to the readers of the Journal. A Special Issue is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

The following information should be included as part of the proposal:

- Proposed title for the Special Issue
- Description of the topic area to be focused upon and justification
- Review process for the selection and rejection of papers.
- Name, contact, position, affiliation, and biography of the Guest Editor(s)
- List of potential reviewers
- Potential authors to the issue
- Tentative time-table for the call for papers and reviews

If a proposal is accepted, the guest editor will be responsible for:

- Preparing the "Call for Papers" to be included on the Journal's Web site.
- Distribution of the Call for Papers broadly to various mailing lists and sites.
- Getting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Instructions for Authors.
- Providing us the completed and approved final versions of the papers formatted in the Journal's style, together with all authors' contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

## Special Issue for a Conference/Workshop

A special issue for a Conference/Workshop is usually released in association with the committee members of the Conference/Workshop like general chairs and/or program chairs who are appointed as the Guest Editors of the Special Issue. Special Issue for a Conference/Workshop is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

Guest Editors are involved in the following steps in guest-editing a Special Issue based on a Conference/Workshop:

- Selecting a Title for the Special Issue, e.g. "Special Issue: Selected Best Papers of XYZ Conference".
- Sending us a formal "Letter of Intent" for the Special Issue.
- Creating a "Call for Papers" for the Special Issue, posting it on the conference web site, and publicizing it to the conference attendees. Information about the Journal and Academy Publisher can be included in the Call for Papers.
- Establishing criteria for paper selection/rejections. The papers can be nominated based on multiple criteria, e.g. rank in review process plus the evaluation from the Session Chairs and the feedback from the Conference attendees.
- Selecting and inviting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Author Instructions. Usually, the Proceedings manuscripts should be expanded and enhanced.
- Providing us the completed and approved final versions of the papers formatted in the Journal's style, together with all authors' contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

More information is available on the web site at <http://www.academypublisher.com/jait/>.