Two Hybrid Tabu Scatter Search Meta-heuristics for Solving MAX-SAT Problems

Dalila Boughaci[1], Habiba Drias[2] and Belaid Benhamou[3]

[1] University of Sciences and Technology – ITS- USTHB
BP 32, El-Alia Beb-Ezzouar, 16111, Algiers, Algeria
Dalila_info@yahoo.fr
[2] Institute of Computer Science –INI-
BP 68M OUED SMAR EL HARRACH , ALGER
drias@wissal.dz
[3] LSIS - UMR CNRS 6168 MARSEILLE CEDEX 20
belaid.benhamou@cmi.univ-mrs.fr

Abstract. Tabu search is a meta-heuristic that has been successfully applied to hard optimization problems. In this paper, two new hybrid meta-heuristics are studied for the NP-Complete satisfiability problems, in particular for its optimization version namely MAX-SAT. At first, we present a tabu scatter search approach, TS+SS, which is a tabu search procedure extended by a commonly shared collection of scatter elite solutions. Then, we introduce a scatter tabu search approach, SS+TS, which is a scatter search procedure enhanced with a tabu search improvement strategy. Experiments comparing the two approaches for MAX-SAT are presented. The empirical tests are performed on DIMACS benchmark.

1 Introduction

Tabu search is one of the meta-heuristic methods. It has been applied to various optimization problems with a great success. In this work, we propose two hybrid approaches based on tabu search meta-heuristic to solve the satisfiability problems. Given a collection *C* of *m* clauses involving *n* Boolean variables, the satisfiability problem is to determine whether or not there exists a truth assignment for *C* that satisfies the *m* clauses. A clause is a disjunction of literals. A literal is a variable or its negation. A formula in conjunctive normal form (CNF) is a conjunction of clauses. The formula is said to be satisfiable if there exists an assignment that satisfies all the clauses and unsatisfiable otherwise. In the latter situation, we are interested in other variants of SAT. We mention among them the maximum satisfiability problem (MAX-SAT). The latter consists in finding an assignment that satisfies the maximum number of clauses. MAX-SAT is an optimization variant of SAT. They are an important and widely studied combinatorial optimization problem with applications in artificial intelligence and other areas of computing science. The decision variants of both SAT and MAX-SAT problems are NP-Complete [4, 8].

Many algorithms have been proposed and important progress has been achieved. These algorithms can be divided into two main classes:

© A. Gelbukh, R. Monroy. (Eds.)
Advances in Artificial Intelligence Theory
Research on Computing Science 16, 2005, pp. 45-54

- *Complete algorithms*: dedicated to solve the decision version of SAT problem. The well- known algorithms are based on the Davis-Putnam-Loveland procedure [5]. Satz [13] is a famous example of a complete algorithm.
- *Incomplete algorithms*: they are mainly based on local search and evolutionary algorithms. Local search[16], tabu search [14, 1, 2], simulated annealing [10], genetic algorithms [7], GRASP[15], scatter search [6] and recently memetic algorithms [3] are examples of incomplete algorithms for SAT. These meta-heuristics are a good approach for finding a near solution of very large instances, in particular for unsatisfiable or unknown instances.

In this paper, we propose, at first, a tabu scatter hybrid procedure for MAX-SAT problems. Its algorithmic backbone is a tabu search (TS) which is extended by a commonly shared collection of elite solutions. This collection is maintained by the tabu search, which inputs quality solutions and is used by the scatter search to construct combined solutions. Then, a scatter search variant is proposed for the same problem. Its algorithmic backbone is a scatter search (SS) combined with a tabu search (TS) improvement strategy. The latter performs an intensified search of solutions around the scatter search regions. Experiments comparing the two approaches for MAX-SAT are presented. The empirical tests are performed on some well-known DIMACS benchmark instances. The paper starts with a brief review of the tabu search. Section 3 introduces the scatter search approach. Section 4 presents our new tabu scatter search approach. Section 5 presents the scatter tabu search approach. Our comparative study and experiments results are summarized in section 6. Finally, conclusion and future work are explained in section 7.

2 A Tabu Search Meta-heuristic

Tabu search is a meta-heuristic that has been proposed by Fred Glover [9]. It has been applied to various optimization problems including the satisfiability problem [14, 1, 2] and job shop scheduling [17]. Tabu search starts with an initial configuration generated randomly, then, the best neighbor solutions are selected. Tabu search uses also a list called "tabu list" to keep information about solutions recently selected in order to escape the solutions already visited. In the case where a tabu move applied to a current solution gives a better solution; we accept this move in spite of its tabu status by aspiration criterion. The search stops when the quality of the solution is not improved during a maximum number of iterations or when we reach a global optimal.

2.1 Tabu Search Items

In order to use tabu search for solving MAX-SAT problem, we define the following items:

- A Solution is represented by a binary chain X (*n* Vector); each of whose components xi receives the value 0 or 1. It is defined as a possible configuration verifying the problem constraints and satisfying the goal that consists in finding an assignment of truth values to the *n* variables that maximizes the sum of satisfied clauses.

- A move is an operator used to generate neighbor solutions. An elementary move consists in flipping one of the variables of the solution. The neighborhood of a solution is constituted by all the solutions obtained by applying an elementary move on this solution. A variable is in tabu state if it has been modified during the current move and it keeps it during a certain number of iterations called **tabu tenure**.
- A Tabu List is used to keep information about the solutions already visited in order to escape local optima by searching in new regions not already explored.

3 A Scatter Search Meta-heuristic

Scatter search [12] is a population-based meta-heuristic. It is an evolutionary method that constructs solutions by combining others**.** The approach starts with an initial population (collection of solutions) generated using both diversification and improvement strategies, then, a set of best solutions (reference set that incorporates both diverse and high quality solutions) are selected from the population. These collections of solutions are a basis for creating new solutions consisting of structured combinations of subsets of the current reference solutions.

3.1 A Scatter Search Template

Four methods are used to achieve the scatter search template:

- A Diversification Generator. The generator creates, from a seed solution, a collection of diverse solutions, applies a heuristic process for improving these solutions and designates a subset of the best solutions to be reference solutions. Solutions gain membership to the reference set according to their quality or their diversity.

- An Improvement Method. An Improvement method transforms a trial solution into one or more enhanced trial solutions. To improve the quality of solutions we often apply a heuristic process.

- A Subset Generation Method. A subset generation method operates on the reference set (collection of elite solutions), to produce a subset of its solutions as a basis for creating a combined solution.

- A Combination Operator. A solution combination method transforms a given subset of solutions created by the subset generation method into one or more combined solutions. In this step, we create new points consisting of structured combinations of subsets of the current reference solutions.

4 A Tabu Scatter Search

In order to take advantage of the individual benefits of a single-solution oriented approach and a population oriented approach, we propose a hybrid tabu scatter search approach. Its backbone is a basic tabu search that works on a single solution by building neighborhoods from which a best admissible candidate is passed to the scatter search process. The hybrid tabu search makes use of a population based strategy and maintains a collection of elite solutions. More precisely, the tabu scatter hybrid (TS+SS) procedure starts with an initial solution generated randomly; then, a basic tabu search is started. The duration of this second phase (TS) is given by an input parameter $iter_{TS}$ corresponding to the number of iterations of the basic TS process (see code below). During the TS phase a new best solution is always deposited into the collection. Every "$iter_{SS}$" iterations the algorithm calls the subroutine of the scatter search phase, operating on the solutions in the collection. Those solutions represent the reference set in the basic scatter search. They are a basis for creating new combined solutions using a combination operator. The combination method, that we have used, randomly selects a position *K* to be the crossing point from the range [1,…n]. The first *K* elements are copied from the one reference point while the second part is copied from the second reference point to create the new trial solution. After having built new combined solutions via the combination method cited above, the best solution is returned to TS to serve as an initial starting point which may be enhanced after resetting the tabu list. The algorithm terminates after a certain number of iterations.

4.1 A Tabu Scatter Search Outline

```
Step 1. Initialization
  Set tabu scatter search(TS+SS)parameters
  //iter is the current iteration of TS+SS process,
  //maxiter is the maximum number of iterations of TS+SS
  //iter1 is the current iteration of TS process,
  // iterTS is the maximum number of iterations of TS, //iterss
  is the number of iteration in which the scatter search
  (SS )is called,                                    // TL
  is the tabu list,
  // S* is the best solution with the minimum F* corresponds
  to S*, F* objective function value that is F*=F(S*),
  - Generate an arbitrary solution S;                        -
  Evaluate F (S); S*= S; F* = F; iter=0; iter1 = 0;
Step 2. Iteration
 While (iter < maxiter) do
  begin
    While (iter1 < iterTS  and iter < maxiter)  do
      begin
      - iter  = iter + 1; iter1 = iter1 +1;
       - Apply a basic TS process;/* iteratively execute
        iterTS iterations using neighborhood operators*/
       - Add the good solution found to the collection
         of elite solutions to construct the reference
         set for the next phase;
   If  (iter = iterss )  then
     begin
      /*while performing the TS, execute an SS
                phase every iterss */
        - Apply a TS process using the second move (SWAP) to
           diversify the search, and add the diverse
           solution to the collection;
```

```
          - Generate subsets of the reference set as a basis
              for creating combined solutions;
          - For each subset produced, use the combination
              operator to produce new solutions;
          - Improve the combined solutions
       end;
  end;
  - iter1=0;
 end;
 end;
 Step 3. Termination. Print the best solution with the best
 objective value.
```

5 A Scatter Tabu Search

Our scatter tabu search-based evolutionary approach starts with an initial population of solutions created using a diversification generator. The latter creates, from a seed solution *V*, a collection of solutions associated with an integer h $(1<h<=n)$. A solution is represented by a binary chain *V* (*n* Vector). Two types of solutions *V'* and *V''* are created from the seed solution V and given as:

Type1 solutions $V'[1+k*h]= 1- V[1+k*h]$, $k=1,2,\ldots..n/h, k<n$.

Type2 solutions: *V''* are the complement of *V'*.

Then, each solution in the population makes tabu search to improve its fitness. After that, a set of solutions (reference set) are selected from the current population. The resulting reference set has *B1* high quality of individuals plus *B2* diverse solutions. The reference set is a basis for creating new solutions consisting of structured combination of subsets of the current reference set. The combination method (the same described in the preview section) is applied to all subsets of solutions of the current reference set. After having built new combined solutions, the combined solution is returned to TS procedure to serve as an initial starting point which may be enhanced. With all this components: diversification generator, reference set selection, combination method and intensified tabu search procedure, we hope to be able to achieve a good compromise between intensification and diversification in the search process. The search terminates after a certain number of generations or when we reach the optimum global.

5.2 A Scatter Tabu Search Outline

```
Step 1. Initialization
Set scatter tabu search parameters
//Psize is the size of the population P,
//B is the size of the reference set
//maxiter is the maximum number of generations,
// iter is the current generation,
- Call the Diversification generator to create an
  initial population P;
- Use TS to create enhanced trial solutions of P;
- Evaluate and order the solutions in P according
     to their objective function value;
- iter=0 ;
```

```
Step 2. Iteration
 While (iter < maxiter)do
  begin
      - Create the Reference Set of selections by
       choosing B1 high quality and B2 diverse
       solutions from P where B1+B2= B;
      - Subsets Generation Method
     While(continuing to maintain and update
         reference Set) do
      begin
        - For each subset produced, use the
         combination method to produce new solutions;
        - Use TS to create enhanced trial solutions;
         - Update the Reference Set: If the resulting
           solution improves the quality then add it to
           the B1 high quality solutions and remove the
           worst one else add it to the B2 solutions and
           removes the low diverse one in B2;
       end;
         - Build a new population P by initializing the
             generation process with the reference set;
         - Use TS to create enhanced new trial solutions;
        - iter= iter+1;
     end;
Step 3. Termination, print the best solution with the best
objective value.
```

6 A Comparative Study

The purpose of this comparative experiment is to evaluate the performance of each one of the proposed techniques to solve MAX-SAT instances. First of all, we compare on the table 1 the approaches regarding their principles and the operators used by each approach. Further, we give some numerical results obtained by applying each algorithm on MAX-SAT instances. The objective is to explore the influence of population and combination strategies by comparing SS+TS and TS+SS. To compare the hybrids and to explore the influence of the hybridization, we have compared SS+TS and TS+SS with SS and TS alone. The results are given on the tables below.

6.1 Computational Results

All experiments were run on a 350 MHZ Pentium II with 128 MB RAM. All instances have been taken from the SATLIB [11]. They are hard Benchmark Problems. On each instance the different algorithms have been executed in order to compute the average of the maximum number of the sum of the satisfied clauses.

The DIMACS Benchmarks

Two kinds of experimental tests have been undertaken. The goal of the first ones is the setting of the different parameters of the TS+SS, and SS+TS algorithms like the Tabu tenure, the number of iterations, the population size and the interaction between the two algorithms parameters. These parameters are fixed as:

Table 1. Comparison of TS, SS, TS+SS, and SS+TS approaches

	TS	**SS**	**TS+SS**	**SS+TS**
Principles	-neighbor search	- Evolutionary meta-heuristic	- neighbor search meta-heuristic	-Evolutionary meta-heuristic
	-Single current solution	- Population-based	-Population-based	- Population-based
	-Interdiction mechanism	Biological evolution	Interdiction mechanism	-Biological evolution and interdiction
Operators	-Move -Tabu list - Aspiration criterion	-Reference set selection - Structured combination - Improvement local technique	- Move - Tabu list -Aspiration criterion	- Reference set selection - Structured combination - Improvement local technique
Solution or Population generation	At Random	Using diversification generator	At Random or using a heuristic	Using a diversification generator

- TS+SS. The maximum total number of iterations was set to maxiter=1000. The basic TS phase parameter, $iter_{TS,}$ was set to 100 iterations, the population size was set to 40, the SS procedure was called every $iter_{SS}$ = 10 iterations. The move operator for TS intensification phase was the variable flipping. A second move operator is used in order to diversify the search consisting in permuting between two variables chosen at random. This phase is executed before calling the SS subroutine, that, in order to create a collection of best solutions including diverse and high quality solutions.
- TS. The maximum total number of iterations was set to maxiter=10000. The move operator was the variable flipping. and tabu tenure was set to 7.
- SS+TS. The basic SS phase parameter, the maximum total number of iterations was set to maxiter=3, the reference set was set to 10, the population size was 100, and the TS parameter was set to 30 iterations and tabu tenure was set to 7.
- SS. is a scatter search with a simple local search as an improvement technique. The SS parameters are: the maximum total number of iterations was set to maxiter=3, the reference set was set to 10 and the population size was 100.
The second kind of experiments concerns MAX-SAT instances. All these instances are encoded in DIMACS CNF format [11]. The tables below show the results obtained by our algorithms. These columns contain the name of instance, the number of variables, the number of clauses, the solution found by each algorithm, and the algorithm running time in second. The results found are classed by class:

AIM class: Artificially generated random 3-SAT, defined by Kazuo Iwama, Eiji Miyano and Yuichi Asahiro [18]. We have chosen six instances.

Table 2. Solutions quality and running time results obtained by TS, TS+SS, SS and SS+TS on AIM instances.

Instance/ satisfiable	# Var	# claus es	TS	TS Time	TS+ SS	TS+SS Time	SS	SS Time	SS+ TS	SS+ TS Time
Aim50-1-1	50	80	78	21,6	79	44,8	79	9,6	79	54,8
Aim50-2-1	50	100	99	26,5	99	43,3	99	11,5	99	15,7
Aim50-3-1	50	170	169	44,3	170	12,0	167	18,9	165	25,5
Aim100--1	100	160	154	87,5	159	95,1	158	34,5	157	43.7
Aim100-2	100	200	195	105,5	199	156,1	196	46,8	195	63.6
Aim100-3	100	340	334	185,6	335	243.9	330	70,9	327	183.9

JNH class: Randomly generated instances- constant density model. The instances have originally been contributed by John Hooker [18].

Table 3. Solutions quality and running time results obtained by TS, TS+SS, SS and SS+TS on JNH instances.

Instance satisfiable	# Var	# claus es	TS	TS Time	TS+ SS	TS+SS Time	SS	SS Time	SS+ TS	SS+TS Time
Jnh201- yes	100	800	797	885,3	800	155,7	795	31.4	799	670.2
Jnh202- no	100	800	792	1084,2	796	896,0	795	30,6	797	777.3
Jnh203-no	100	800	798	1159,0	796	1135,1	790	31,3	794	753.7
Jnh204-yes	100	800	796	707,7	798	728,7	796	28,7	797	821.1
Jnh205-yes	100	800	800	697,3	799	894,1	794	34,5	797	802,2
Jnh206-no	100	800	799	701,1	797	750,2	794	30,4	796	821.1
Jnh207-yes	100	800	797	701,4	797	841,8	793	28,3	796	831,4
Jnh208-no	100	800	797	700,7	794	761,1	795	34,2	796	789.1
Jnh209-yes	100	800	797	697,8	797	797,8	794	27,8	796	748.0
Jnh210-yes	100	800	800	699,0	800	83,2	795	28,6	797	767.6

Parity8 class: Instance arises from the problem of learning the parity function. Defined by James Crawford (jc@research.att.com). All the instances are satisfiable by construction [18].

The results obtained by the different approaches are acceptable (we have reached the optimum for some instances in reasonable time). In many cases (Jnh 201, Jnh 210, par8-1-c for example), the tabu scatter search (TS+SS) performs the other approaches in solving such instances. In some case (Jnh 205, for example) a TS alone performs the others. Also, for some benchmarks, the SS alone performs the SS+TS which means that the choice of adequate parameters for a meta-heuristic in solving a given benchmark is a difficult operation and the hybridization in some situation is not very interesting. However, for the most benchmarks the hybridization improve the quality of solutions and gives a good result. So, according to our results, we can see that, in general, when SS is incorporated in TS (TS+SS), the solutions space is better searched. When intensified improvement tabu search and diversified components are incorporated in SS (SS+TS), the solutions space is better searched but the process search takes more time to find a solution. We precise, that the role of a tabu search technique in scatter search is to locate the solution more efficiently.

Table 4. Solutions quality and running time results obtained by TS, TS+SS, SS and SS+TS on parity8 instances.

Instance	# Var	# claus es	TS	TS Time	TS+ SS	TS+SS Time	SS	SS Time	SS+ TS	SS+TS Time
Par8-1	350	1149	1115	1768,1	1126	986,7	1141	76,7	1141	303,94
Par8-1-c	64	254	250	68,9	254	10,9	248	4,1	245	121.12
Par8-2	350	1157	1114	1813,6	1137	937,9	1146	76,0	1146	401,99
Par8-2-c	68	270	267	78,9	267	137,7	263	5,13	260	99,17
Par8-3	350	1171	1127	1850,6	1154	1131,0	1162	72,8	1162	536.33
Par8-3-c	75	298	296	96,2	294	971,4	291	11,8	289	117.23
Par8-4	350	1155	1105	1816,9	1154	1351	1149	70,9	1149	278.26
Par8-4-c	67	266	261	74,5	264	141,2	260	4,8	260	91,06
Par8-5	350	1171	1110	1842,7	1164	1581.1	1163	74,8	1163	584.16
Par8-5-c	75	298	293	95,9	294	157,0	290	7,2	290	119,91

7 Conclusion and Perspectives

In this paper, we have presented, at first, the single–oriented meta-heuristic called tabu search. We have proposed to hybridize it with a scatter search evolutionary algorithm.. Then, we have presented a scatter tabu search approach. The proposed approaches have been implemented for solving MAX-SAT hard instances. Our objective is to explore the influence of both population and combination strategies on the ability of generating high quality solutions in single solution-oriented approaches and vice versa. We have shown that the impact of a population strategy on a single-oriented approach is like the import of a local search in a population- based approach. After an intensified experimentation, we conclude that the tabu search (TS) can be considered as a powerful procedure capable to organize and to direct operations of subordinate methods. We plan to improve our framework by implementing a parallel environment including the two approaches.

References

1. D.Boughaci H.Drias "Solving Weighted Max-Sat Optimization Problems Using a Taboo Scatter Search Meta-heuristic". *In Proceedings of ACM* Sac 2004, pp35-36, 2004.
2. D.Boughaci, H.Drias. "PTS: A Population-based Tabu Search for the Maximum Satisfiability Problems". *In Proceedings of the 2 IEEE GCC conferences,* pp 622-625, 2004.
3. D.Boughaci, Drias H and Benhamou B. "Solving Max-Sat Problems Using a memetic evolutionary Meta-heuristic". *In Proceedings of 2004 IEEE CIS,* pp 480-484, 2004.
4. S.Cook . "The Complexity of Theorem Proving Procedures". I*n Proceedings of the 3rd Annual ACM Symposium on the Theory of Computing, 1971*.
5. M.Davis, G. Putnam and D.Loveland. "A machine program for theorem proving". C*ommunications of the ACM*, 394-397, Jul 1962.
6. H.Drias. "Scatter search with walk strategy for solving hard Max-W-Sat problems". I*n Proceedings of IEA- AIE2001. Lecture Notes in Artificial Intelligence, LNAI-2070,* Springer-Verlag, Boudapest, pp 35-44, 2001.
7. J.Frank, "A study of genetic algorithms to find approximate solutions to hard 3CNF problems". *In Proceedings of Golden West International Conference on Artificial Intelligence,1994.*
8. MR.Garey, S.Johnson. "Computer and intractability, a guide to the theory of NP-Completeness, Freeman company. Sanfranscsco, 1978.
9. F. Glover, "Future paths for integer programming and links to Artificial intelligence", *Operational Research*, Vol 31, 1986.
10. P.Hansen P; B.Jaumard. "Algorithms for the Maximum Satisfiability". Journal of Computingm 44- pp 279-303, 1990.
11. H. H. Hoos and T. Stutzle. SATLIB: An online resource for research on SAT. In I. Gent, H. van Maaren, and T. Walsh, editors, *SAT2000: Highlights of Satisfiability Research in the year 2000*, Frontiers in Artificial Intelligence and Applications, pp 283–292. Kluwer Academic, 2000.
12. M.Laguna, F.Glover *"Scatter Search"*, Graduate school of business, University of Colorado. Boulder, 1999.
13. CMLi and. Anbulagan, "Heuristic based on unit propagation for satisfiability". In Proceedings of CP 97, springer-Verlag, LNCS 1330, pp 342-356, Austria, 1997.
14. B.Mazure, L.Sais and E.Greroire,." A Tabu search for Sat". In Proceedings of AAAI 1997.
15. PM.Pardalos, L.Pitsoulis and MGC. Resende. "A Parallel GRASP for MAX-SAT Problems. PARA96 Workshop on Applied Parallel Computing in Industrial Problems and Optimization", Lynghy. Denmark August 18-21, 1996.
16. B.Selman , H. .Kautz and B.Cohen . "Local Search Strategies for Satisfiability Testing". P*resented at the second DIMACS Challenge on Cliques, Coloring, and Satisfiability, October 1993.*
17. Taillard , "Parallel Tabu Search Techniques for Job Shop Scheduling Problem". *ORSA Journal on Computing,* 6:108-117, 1994.
18. http://www.satlib.org**.**