Hybrid Scatter Search Algorithm for 4-Color Mapping Problem

Mohammed Salah Ibrahim
College of Computer, University of Anbar, Iraq

Ahmed T. Sadiq
Computer Science Department, University of Technology, Iraq

Ali M. Sagheer
College of Computer, University of Anbar, Iraq

Abstract: In this paper scatter search algorithm has been hybridized with simulated annealing algorithm. Scatter search algorithm has some disadvantages which will be solved by using concepts of simulated annealing algorithm. Simulated annealing algorithm will provide scatter search algorithm with more of exploration for problem search space, this will increase ratio of getting optimal solutions. The hybrid and original scatter search algorithm have been tested by 4-Color Mapping problem. 6 local instances for Middle East maps have been tested and reported in tables. The computational results illustrate that the hybrid Scatter Search algorithm is better than the original Scatter Search algorithm.

Keywords: Metaheuristic, Scatter Search, Simulated Annealing, Combinatorial Problems, 4-Color Mapping Problem.

Received September 4, 2002; accepted May 13, 2003

1. Introduction

Four Color Mapping is one of the combinatorial optimization problems that was first conjectured in 1852 by Francis Guthrie, and after over a century of work by many famous mathematicians [1][2]. The 4-Color Mapping can be defined as: The regions of any simple planar map can be colored with only four colors, in such a way that any two adjacent regions have different colors [3] see Fig. 1 which illustrate vector of 20 regions have been colored with 4 color and no region beside region has the same color.

Figure 1. Graph Colored with 4-Color

4-Color Mapping problem will be test ground for Scatter Search algorithm and its hybridization. Scatter Search algorithm which is one of Population-based metaheuristic will be hybridized with Simulated Annealing algorithm which is one of algorithm that was inspired from physical nature.

The rest of the paper is organized as follows. Scatter Search algorithm is described in Section 2. Section 3 presents brief description for Simulated Annealing algorithm. Section 4 presents the Hybrid Scatter Search. Section 5 describes Hybrid Scatter Search for 4-Color Mapping Problem. In section 6, experimental results are presented. Finally, some concluding remarks are presented in Section 7.

2. Scatter Search Algorithm

Scatter search (SS) was first introduced in Glover (1977) as a heuristic for integer programming. In the original proposal, solutions are purposely generated to take account of characteristics in various parts of the solution space [4]. The method uses strategies for search diversification and intensification that have proved effective in a variety of optimization problems [5].

Fig.2 illustrates the basic SS algorithm [6]. SS works on a population of solutions of the problem to be solved, which are stored in a set of solutions called the Reference Set. The solutions in this set are combined in order to obtain new ones, trying to
generate each time better solutions, according to high quality and diversity criteria [5,7].

**Algorithm 1: Scatter Search**

**Input:** P = Population, Maxitr.

**Output:** Best solution found.

- Initialize the population (P) using a Diversification Generation Method;
- Apply the Improvement Method to the P;
- Reference Set Update Method (Good solutions for RefSet1 and Diversity solutions for RefSet2);
- While (itr < Maxitr) do
  - Subset Generation Method;
  - While (subset-counter < > 0) do
    - Solution Combination Method;
    - Improvement Method;
    - Reference Set Update Method;
  - End while
- End while

Figure 2. Basic Scatter Search algorithm

The design of a basic SS algorithm is generally based on the following five steps [5][6][7]:

- A Diversification Generation Method to generate a P of diverse trial solutions within the search space.
- An Improvement Method to transform a trial solution into one or more enhanced trial solutions.
- A Reference Set Update Method to build and maintain a Reference Set (RefSet). The objective is to ensure diversity while keeping high-quality solutions. For instance, one can select RefSet1 solutions with the best objective function and then adding RefSet2 solutions with the best diversity solutions (RefSet = RefSet1 + RefSet2).
- A Subset Generation Method to operate on the reference set, to produce several subsets of its solutions as a basis for creating combined solutions.
- A Solution Combination Method to transform a given subset of solutions produced by the Subset Generation Method into one or more combined solution vectors. Fig. 3 illustrates the five SS steps.

The combined solutions will produce new solutions and the fitness of these new solutions will be evaluated. The new solutions will subject to the Reference Set Update Method which are [4]:

1) The new solution has a better objective function value than the solution with the worst objective value in RefSet1.
2) The new solution has a better diversity value than the solution with the worst diversity value in RefSet2.

The search is continues while RefSet is changed. If no change in RefSet, the algorithm will check if number of iteration (itr) is reach the max iteration (MaxItre) that detected by user then the algorithm will display the good solution(s) reached, else, the new P will be generated and RefSet1 will be added to the start of new P [6].

**3. Simulated Annealing Algorithm**

Simulated Annealing (SA) is single-solution based metaheuristic applied to optimization problems which emerges from the work of S. Kirkpatrick et al. [8] and V. Cerny [9]. In these pioneering works, SA has been applied to graph partitioning and VLSI design. In the 1980s, SA had a major impact on the field of heuristic search for its simplicity and efficiency in solving combinatorial optimization problems. Then, it has
been extended to deal with continuous optimization problems [10,11,12]. SA is based on the principles of heating and then slowly cooling a substance to obtain a strong crystalline structure. The strength of the structure depends on the rate of cooling metals. Strong crystals are grown from careful and slow cooling [6].

The basic steps of the SA algorithm are explained in Algorithm 2 [13]. The algorithm requires a number of parameters to be set:
- s: current solution.
- s*: new solution.
- T: control temperature.
- ΔE: the change in objective function between s and s*.
- f(s): the objective function (fitness).

### Algorithm 2: SA algorithm

**Input:** T=temperature, s= initial solution.

**Output:** Best solution found.

1. s = s₀; (Generation of the current solution)
2. Evaluate the objective function of s: f(s).
3. T = Tₘₐₓ; (Starting temperature)
4. Repeat (temperature loop)
   - Repeat (At a fixed temperature)
     - Generate a random neighbor s*;
     - Evaluate the objective function of s*: f (s*).
     - If ΔE ≤ 0 Then s = s* (Accept the neighbor solution)
     - Else Accept s=s* with a probability $e^{\frac{-\Delta E}{T}} > random[0,1]$);
   - Until stopping criteria (number of neighbor search)
   - T = g(T); (Temperature update)
5. Until Stopping criteria satisfied (e.g. T < Tₘᵢₙ)

Figure 4. Simulated Annealing Algorithm

We'll explain the SA algorithm as follow:

As initial, the SA algorithm start with single solution then the fitness of the current solution will be evaluated. High degree is assigned for Tₘₐₓ. The high degree is required to increase probability of acceptance for the new solution.

After assigning degree to the Tₘₐₓ, the Temperature loop will begin. At fixed temperature loop of number of neighborhood search on solution s will begin. The neighborhood search will be made with fixed temperature.

Random neighborhood on current solution s will be used to generate new solution s* and the objective function f(s*) is evaluated.

E for the new and current solution is computed. The symbol E refers to the change in energy E in the natural annealing process.

If the changes in the objection function E was less than zero then the s* will be accepted and the current solution will be equal to the new solution. Else, the acceptance of the new solution will depend on the probability.

The probability is computed by using the exponential function where the parameters of this function are E and T:

$$P(\Delta E,T) = e^{\frac{-\Delta E}{T}},$$ where $P$ is The Probability.

If the probability P result was larger than value between 0 and 1 then the new solution will be accepted and the s which is the current solution will be equal to the new solution s*.

Loop of neighbor search will be continues until reaching the limited number of neighborhood search.

The effective part in SA algorithm is Temperature updating. It contains the update (decrement) of Temperature T where it represents the cooling process. The cooling must be carefully achieved. Linear update method was used to update the T [12]:

$$T = T - \beta,$$

where $\beta$ is a specified constant value [0,1].

Finally, the search will stop and the best solution will be displayed.

### 4. Hybrid Scatter Search Algorithm

Reference Set Update Method and Improvement Method are the most effective steps in SS algorithm. When we try to use SA with Improvement Method appeared big problem represented by time, where the improvement method applied to all population and apply to every new solution generated and this will take a lot of time, this time will affect the algorithm as one of Metaheuristic algorithms which the main objective of it is to find the nearest optimal solution in reasonable time.

### Algorithm 3: Hybrid SS algorithm

Initialize the population Pop using a Diversification Generation Method;
Apply the Improvement Method to the population;
Reference Set Update Method (Good solutions for RefSet₁ and Diversity solutions for RefSet₂);
While (itr < Maxitr) do
  While (Reference set is changed) do
    Each Solution in Reference Set Enter to SA Algorithm
    SA retrieve the updated Reference Set
    Subset Generation Method;
    While (subset-counter < > 0) do
      Solution Combination Method;
      Improvement Method;
      Reference Set Update Method;
    End while
  End while
End while

Figure 5. Hybrid Scatter Search Algorithm
But when we try to use Reference Set Update Method with SA algorithm we get good solutions and ratio of getting nearest optimal solution increased, so we use the solutions of $RefSet_1$ with SA algorithm where each solution in $RefSet_1$ will pass through steps of SA to generate new one and the new solution will subject to rules of Reference set update Method. Fig. 5 illustrates the place of SA algorithm in the body of SS algorithm to produce the Hybrid SS algorithm.

5. Hybrid Scatter Search For 4-Color Mapping

SS and the Hybrid SS algorithm will be test by 4-Color Mapping problem which is one of NP-hard problems. We will illustrate siting of 4-Color Mapping problem to SS and hybrid SS algorithms as follow:

In Diversification Method, $P$ of solutions of colored region will be generated by using randomized local search procedure that depends on seed which represents the initial solution that will enter to local search to generate $P$ from it. This procedure generates random solutions each solution is different from another and each solution is represented as array containing the colors of each region. The colors in each solution will be represented by the numbers (1,2,3,4) where each number represents a color in solution such as 1=red, 2=green, 3=blue and 4=yellow.

After generating $P$ by Diversification Generation Method, the fitness of each solution in $P$ will be evaluated. The fitness in 4-Color Mapping problem is the summation of neighbored regions whose have the same color.

After evaluating the fitness of each solution, all the solutions in $P$ will be improved or remain without change by Improvement Method. The improvement achieved by using local search algorithm which is heuristic replaces the current solution by a neighbor that improves the objective function. The neighborhood is done by mutating the solution by inserting new color instead of another Fig. 6 illustrates the mutation in solution.

After local search generates a new solution, the fitness of this new solution will be evaluated. If the new solution is better than current solution then current solution = new solution else the current solution will remain without change. These operations will be applied to all solutions in $P$ to obtain more of feasible solutions.

Solutions of $P$ will be ordered depending on their fitness and Reference Set Update Method will chose $RefSet_1$ which will take the first $b_1$ solutions in $P$ and delete them from $P$, while $RefSet_2$ will be generated by Euclidean distance to select diversity solutions. Where the Euclidean distance computes the dissimilarity of the solutions in $RefSet_1$ with solutions of $RefSet_1$-$P$ and the solutions of $P$ which are dissimilar with solutions of $RefSet_1$ will be chosen as solutions of $RefSet_2$.

Solutions of $RefSet_1$ will enter to SA Algorithm, SA will take each solution in $RefSet_1$ to and generate new one. The new solution $s^*$ will be evaluated, if the fitness of the $s^*$ is better than the solution that was generated from it then replace with it else the $s^*$ will subject to probability of acceptance. In this case the SA will provide SS with more of exploration for problem search space and this will increase ratio of getting optimal solution. After SA generate the new solutions from $RefSet_1$, all solutions in $RefSet_1$ will be updated using reference set update methods.

After updating the solutions in $RefSet$ each solution in $RefSet$ will make subset with all other solutions in $RefSet$ by Subset Generation Method.

Solution Combination Method uses the subsets generated with the Subset Generation Method to combine the solutions in each subset with the purpose of creating new trial solutions. Depending on the form of the solution in 4-Color Mapping problem the new solution can be generated by two techniques:

1) Crossover the colors of solutions in subset (see Fig 7).

2) Mutation the solutions in subset (see Fig. 6).

The combination method can use either one of these two techniques or it can hyper the combination for solution by using these two techniques. During the experiments, the second technique is more appropriate for generating new feasible solutions in SS, so in our experiments we will use the second technique with the following conditions [4]:

1) If both $x$ and $x^*$ in subset are elements of $RefSet_1$, then generate 3 solutions by applying
mutation(x(i)a,x(i+1)b,x(i+2)c) where x = first solution in subset, x’ = second solution in subset i=number of solution.

2) If only one of x and x’ in subset is a member of RefSet₁, then generate 3 solutions for member of RefSet₁ and generate 2 solutions for RefSet₂ solutions by applying mutation(x(i)a,x(i+1)b) where i=number of solution.

The combination method will search for regions in solution that have neighbors and neighbors have same color, if they found the combination will mutate these regions to create new solution.

After generating the new solution by applying Solution Combination Method to each subset, it will be subjected to reference set update rules where the new generated solution may become a member of the reference set if either one of the following conditions is satisfied:

- The new solution has a better objective function value than the solution with the worst objective value in RefSet₁.
- The new solution has a better dissimilarity value than the solution with the worst dissimilarity value in RefSet₂.

In both cases, the new solution replaces the worst and the ranking which is updated to identify the new worst solution in terms of either quality or diversity. This loop terminates when the reference set does not change and all the subsets have already been subjected to the Solution Combination Method. At this point, the regeneration by Diversification Generation Method is used to construct new diversity solutions and the search continues. The regeneration consists of keeping RefSet₁ in first of new population and using the Diversification Generation Method to construct new diverse solutions.

6. Computational Experiments

SS and hybrid SS algorithms were implemented in Microsoft Visual C# 2005 Express Edition and run on a computer whose processor is AMD Turion™ 64 2 Mobile Technology TL60 (2 CPU) 2.0 GHz, with 3 GB main memory, 200 GB hard disk. The algorithms were applied to local instances for meddle east maps with small sizes ranging from 5 to 30 regions (such as map contain the following countries: Iraq, Kuwait, Jordan, Lebanon and Syria until 30 countries). The stop criteria are chosen as follow:

1. If no change in Reference Set.
2. To reach a maximum number of iterations = 10.

The following parameters are chosen:

- Initial population $P = 100$.
- The size of $|RefSet₁| = b₁ = 10$, the size of $|RefSet₂| = b₂ = 10$ and the size of reference set $|RefSet| = |RefSet₁|+|RefSet₂| = 20$.
- Temperature $T = 10$.

As the 4-Color Mapping can color small instance and find the optimal solution (which represented by the vector that contain regions adjacent and have different colors) rapidly, the comparison of the original and hybrid SS algorithms will depend on the time required to find the optimal solutions for each vector of regions. Fifteen run for each instances have been achieved and the average of time required to reach the optimal solutions for each vector of regions have been reported in Table 1.

Table 1. Average Of Elapsed Time For SS And Hybrid SS For 4-Color Mapping Problem

<table>
<thead>
<tr>
<th>instances</th>
<th>Average of elapsed time for SS (Sec)</th>
<th>Average elapsed time for Hybrid SS (Sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 Regions</td>
<td>0.015173</td>
<td>0.01776</td>
</tr>
<tr>
<td>10 Regions</td>
<td>0.05152</td>
<td>0.0356</td>
</tr>
<tr>
<td>15 Regions</td>
<td>0.07536</td>
<td>0.0634</td>
</tr>
<tr>
<td>20 Regions</td>
<td>0.083613</td>
<td>0.067307</td>
</tr>
<tr>
<td>25 Regions</td>
<td>0.131133</td>
<td>0.080813</td>
</tr>
<tr>
<td>30 Regions</td>
<td>0.402173</td>
<td>0.126107</td>
</tr>
</tbody>
</table>

Table 1 show how the original SS is faster than the hybrid SS with small instance such as with 5 regions but when the instance extended the hybrid SS is faster than the original SS algorithm. In Fig 8 chart illustrate the difference in reaching to the optimal solution in faster time for original and hybrid SS algorithms.

![Figure 8. Difference between SS and Hybrid SS algorithms](image)

7. Conclusions

This paper presents hybridization for Scatter Search algorithm. The Hybridization achieved by adding SA Algorithm to the Scatter Search algorithm. The original and hybrid Scatter Search algorithms have been tested on 4-Color Mapping problem which is one of the combinatorial optimization problems. The computational results illustrate that the hybrid Scatter Search algorithm present better performance than original Scatter Search algorithm where the hybrid Scatter Search reached to the optimal solution faster than the original Scatter Search.

For Future works we hope to do the following: 1) The presented algorithms are still being developed; the next step would be testing them on a greater variety of
problems with different parameters, stopping criterion, and problem-specific combination operators. 2) The SS and hybrid SS contain several of parameters that can use it to improve the performance such as using Population-based Metaheuristic in step 2 which is improvement method. 3) Using ant colony or partial swarm concepts on reference set which represents the most effective step in SS.

References

Mohammed Salah Ibrahim was born in Anbar, Iraq (1985), now live in Fallujah, Anbar. Complete study of secondary school in Fallujah (2005). He obtained B.Sc. (2008), M.Sc. (2011) in Computer Science from the College of Computer, Anbar University. Mohamed worked as an administrator in the Department of calculating in Al-Safa Co. to oversee the reconstruction work in Fallujah (2007), and worked with Afaq Co. as accountant (2010). He was a lecturer in Administration and Economic Anbar University. Now, He is Faculty staff member in Computer Science Department, College of Computer, Anbar University.