# MODIFIED T SEARCH ALGORITHM FOR WAVELENGTH ASSIGNMENT AND ROUTING

K. RaghavendraRao1 ,Vadipina Amarnadh2 1, 2Assistant Professor, Department of CSE, Anurag group of Institutions, Hyderabad, Telangana, India

***Abstrat*—*To minimize the wavelength assignment usage optimally we can go for the Staticd Routing and wavelength assignment. The solution for this Minimizing Wavelength Usage problem is based on two method approach, considering routing and wavelentgth assignment individually which can reduce computational cost. The existing Tabu Search algorithm considers routing and wavelength assignment jointly. The performance of this existing tabu search algorithm reduces the computational complexity. To solve the problem of Minimizing wavelength usage ,Tabu Search algorithm is compared with the Integer Linear Programming method. We propose a novel where routing and wavelength are done concurrently which gives optimal results for minimizing of wavelength.***

***Index Terms*—Usage of minimum wavelength, modified tabu search, static routing and wavelength assignment, optical network.**

## I.INTRODUCTION

The optimal networks, lightpath is an all optical channel trail which ends at a pair of access nodes. We can go for light path uses the same wavelength when no converter of wavelength. Light path can finds the route for assigning wavelength to the route. The problem of assignment and routing is use minimum wavelength.

The condition of this problem not to use two lightpaths for single wavelength in the same link. The objective is to use minimum wavelength .

The optimization problem is there in usage of minimum wavelength. The ILP provides a global solution for minimum wavelength usage [2,3].the ILP used different integer variables for formulations different techniques to find optimal solution for networks.
These techniques include randomized rounding[1]and column generation[2]. Due to the computational intensity of the ILP approach, heuristic algorithms, which are widely applied in solving various combinatorial optimization problems, have begun to surface as alternatives.

In[3],an evolutionary algorithm is proposed. In[4],a *tabu search*(TS) algorithm is designed to solve the RWA problem of scheduled light path requests, which is analogous to the MWU problem described here. Both algorithms are effective when applied to a small network. We notice that both[3]and[4]essentially follow the two-step approach of[1]in solving the RWA problem. In the two- step approach the MWU problem is decoupled into two sub- problems, which are solved separately in sequence. First, the routes of the requests are found by means of an ILP based algorithm[1], evolutionary algorithm[3] or [4]. Secondly, wavelength assignment is carried out by a graph coloring algorithm. In the ILP approach, this method significantly reduces the number of integer variables in the formulation, and can significantly improve the computation efficiency at the expense of optimality. However, we notice that when TS is applied, the complexity of the algorithm is mainly governed by the neighborhood selection of the search process instead of the whole search space, which is proportionate to the number of variables. In view of this ,we propose a tabu search algorithm, which considers route and wavelength assignment jointly. Our simulation study for a relatively large network of 50nodes shows that the results are almost as good as that yielded by the ILP solution. The proposed TS algorithm can solve the MWU problem with in half an hour while the ILP solution takes more than a day to find the solution.

## II.EXISTING TABU SEARCHALGORITHM

TS is an iterative procedure that starts with an initial solu- tion and repeatedly constructs new solutions by searching the neighborhoods of the current solution. Each solution has an associated neighborhood $N(\Gamma)$,a sub set of the whole solution Space $\Phi$. The step by which solution $\Gamma \in N(\Gamma)$ is reached is called a move. A cost function $C(\Gamma)$ is formulated to evaluate the fitness of the current solution in relation to the objective of the TS. For each iteration, the $\Gamma$ with the best $C(\Gamma)$is selected as the new current solution. The process continues to generate neighboring solutions until the *stopping criterion* is satisfied. In order to avoid revisiting solutions already encountered, a tabu list is maintained. The list contains a number of the latest selected solutions, which should not be chosen for a certain number of subsequent iterations. This mechanism allows or escape from the local minima and sometimes a jump to a new search region. For detailed description of a generic TS algorithm, interested readers may refer to[5].

To describe the MWU problem, we use variable $L_i$to represent a light path assigned to a request $R_i \in \Delta$,where $\Delta=\{R_1,R_2...R_m\}$is the given set of requests. The value Of $L_i$ is represented by a pair($p_i$,$w_i$), where $p_i$ is a route From $R_i$'s source node$s_i$ to destination node $d_i$ on the given Directed network, and $w_i$ is the wavelength assigned to that route. If each variable is assigned a value, we call this set of assignment $\{(p_1,w_1),(p_2,w_2)...(p_m,w_m)\}$a solution Γ to the MWU problem. If there are no two assigned light paths in Γ sharing the same link and using the same wavelength; i.e., no light paths are in conflict with each other ,solution Γ is dubbed a *feasible solution*.

To find a *feasible solution* with minimum wavelength usage, our proposed algorithm starts with a relatively large set of available wavelengths $W$. If the algorithm can find a *feasible Solution* with $/W/$ wavelengths using the TS sub routine, the cardinality of $W$ is reduced by one, and the TS sub routine is run again. The iteration ends when the algorithm is no longer able to find a feasible solution with certain $W$, and returns $/W/$+1astheresult.Theinitialavailablewavelengthset$W$is Obtained by the *first fit* algorithm proposed in[3].

The TS sub routine is designed to find a feasible solution with a given $W$. The problem-specific details of the TS approach are described below:

1)Let $P_i$ denote the candidate paths set associated to $R_i$. $P_i$is pre-calculated using the $k$-shortest paths algorithm in [6].The initial solution $\Gamma_{ini}$ is constructed by randomly Assigning a $p_i \in P_i$ and a$w_i \in W$ to each variable $L_i$.

2)The cost function $C(\Gamma)$ Is computed as:

$|\Delta|$

$C(\Gamma)=C_i(1)$

$i$

A conflict cost $C_i$ is associated with each variable $L_i$ and is calculated as the number of conflicts the assigned light path of $L_i$ experienced in the current solution. For example, in a solution $\Gamma=\{(p_1,w_1),(p_2,w_2),(p_3,w_3)\}$,if ($p_1$,$w_1$)is in conflict with($p_2$,$w_2$)and($p_3$,$w_3$),$C_1$=2. If($p_1$,$w_1$) is not in conflict with any other light path in Γ,$C_1$=0.Clearly,Γ is a *feasible solution* if and only if The total number of conflicts $C(\Gamma)$=0.

3)The neighborhood $N(\Gamma)$ of the current solution is con- structed in the following way. A *conflict request list* (CRL) is maintained in the whole TS subroutine. The CRL is derived from $\Gamma_{ini}$; specifically, CRL is derived from $\Gamma_{ini}$ by noting the $L_i$ whose $C_i$>0.We denote the first variable(*head*) in the CRL as $L_f$. A neighbor Γ of Γ is constructed by assigning a new value to $L_f$ while keeping all other variables of Γ unchanged.

4)CRL is updated at the end of each iteration. For each iteration, a neighbor Γ with the best $C(\Gamma)$ is found to replace the current solution Γ. We use $\Omega_f$ to represent the set of variables that are in conflict with $L_f$ in Γ. The Cost so f $L_f$ and $L_x \in \Omega_f$ are recalculated for Γ. If the New cost of $L_x$ is zero, it means the light path assigned To $L_x$ is no longer in conflict with the other light paths in Γ; thus, $L_x$ is deleted from the CRL. If the cost of $L_f$ Is zero, $L_f$ is deleted and the next variable in the CRL Becomes the new *head*; otherwise, it means $L_f$ must be In conflict with one or more variables in Γ. Let $\Omega_f$ represents that set of variables. If a $L_y \in \Omega_f$ is already in the CRL,$L_y$ is moved to the head of the CRL. Other variables in $\Omega_f$ are inserted into the head of the CRL one by one in a random sequence. With this method, the *head* $L_f$ is renewed for each iteration, and the search process is guided to a new neighborhood based on the new *head*.

5)In order to prevent the TS from searching a recently searched neighborhood, a *tabu list* is used to store the recently searched CRL head $L_f$.We apply the *tabu rule* (TR) where by, a Γ, which may lead to a new CRL *head* that is forbidden by the *tabu list*, is marked with a tabu tag. A tagged solution is then dropped from the neighborhood $N(\Gamma)$ and no longer eligible to be selected as the next current solution.

6)An *aspiration rule*(AR) is applied to overrule the *tabu rule* when a good solution is found. For each iteration, if the cost of a tagged solution is better than the best cost achieved among all the iterations so far, the tabu tag is removed by this rule.

7)The TS sub routine stops based on two conditions. The first stop page condition is when $C(\Gamma)$=0,in this case,a *feasible solution* is found. The second condition is when

The best cost achieved does not improve for $|\Delta|$ number
Of iterations. In this case, the TS sub routine is deemed to have failed in finding a *feasible solution*.

The pseudo codes of the proposed TS algorithm

are as follow:

```
Begin of Main//Main TS algorithm
Construct the initial $W$ with first fit algorithm; While(Tb Sub($W$)=feasible)do{$|W|:=|W|-1$;} Return $|W|+1$;
    End Of Main
    Tb Sub($W$)Subroutine //Begin of Subroutine
      Construct initial solution $\Gamma_{ini}$; Create the CRL from $\Gamma_{ini}$;
      $\Gamma:=\Gamma_{ini}$;Cycle:=0;
      While($C(\Gamma)=0$&Cycle=$|\Delta|$)do
        Begin
          $L_f$:=head of the CRL; Construct $N(\Gamma)$ based on $L_f$;
          Apply TR &AR to find the best $C(\Gamma)$ for all $\Gamma \in N(\Gamma)$;
          $\Gamma:=\Gamma$; Update the CRL ,the tabu list and Cycle;
        End of while
      If($C(\Gamma)=0$)return feasible;
        Else return in feasible;
    End of Subroutine
```

III. PROPOSED MODIFIED TABU SEARCH ALGORITHM

The modified tabu search is concurrent procedure with concurrently construct the new solutions by searching the neighbourhoods. The process will continue until its satisfies.

In the modified search algorithm maintains a visiting information for avoiding revisiting in the list maintained latest selection which is not chosen for another iteration.
To describe the MWU problem, we use variable $L_i$to

represent a light path assigned to a request $R_i \in \Delta$,where
$\Delta=\{R_1,R_2...R_m\}$is the given set of requests. The value
Of $L_i$ is represented by a pair$(p_i,w_i)$,where $p_i$ is a route
From $R_i$'source node $s_i$ to destination node $d_i$ on the given
Directed network, and $w_i$ is the wavelength assigned to that
route. If each variable is assigned a value, we call this set of assignment$\{(p_1,w_1),(p_2,w_2)...(p_m,w_m)\}$ a solution $\Gamma$ to the MWU problem. If there are no two assigned light paths in $\Gamma$ sharing the same link and using the same wavelength; i.e., no light paths are in conflict with each other, solution $\Gamma$ is dubbed a *feasible solution*.

To find a *feasible solution* with minimum wavelength usage, our proposed algorithm starts with a relatively large set of available wavelengths $W$. If the algorithm can find a *feasible*
*Solution* with $|W|$ wavelengths using the TS sub routine, the
cardinality of $W$ is reduced by one, and the TS sub routine is run again. The iteration ends when the algorithm is no longer able to find a feasible solution with certain $W$, and returns
$|W|+1$as the result. The initial available wavelength set $W$ is
Obtained y the *first fit* algorithm proposed in[3].

The TS sub routine is designed to find a feasible solution with a given $W$. The problem-specific details of the TS approach are described below:

1) Let $P_i$ denote the candidate paths set associated to $R_i$. $P_i$ is pre-calculated using the *k*-shortest paths algorithm in [6].The initial solution $\Gamma_{ini}$ is constructed by randomly
Assigning a $p_i \in P_i$ and a $w_i \in W$ to each variable $L_i$.

2)The cost function $C(\Gamma)$ Is computed as:

$|\Delta|$

$$C(\Gamma)=C_i \qquad (1)$$

$i$

A conflict cost $C_i$ is associated with each variable $L_i$ and is calculated as the number of conflicts the assigned light path of $L_i$ experienced in the current solution. For
example, in a solution $\Gamma=\{(p_1,w_1),(p_2,w_2),(p_3,w_3)\}$,if
$(p_1,w_1)$ is in conflict with$(p_2,w_2)$and$(p_3,w_3)$,$C_1=2$.
If$(p_1,w_1)$ is not in conflict with any other light path in
$\Gamma$,$C_1=0$.Clearly,$\Gamma$isa*feasiblesolution*ifandonlyif
The total number of conflicts $C(\Gamma)=0$.

3)The neighborhood $N(\Gamma)$ of the current solution is con- structed in the following way. A *conflict request list* (CRL)is maintained in the whole TS sub routine. The CRL is derived from $\Gamma_{ini}$; specifically, CRL is derived from $\Gamma_{ini}$ by noting the $L_i$ whose $C_i$>0.Wedenotethe first variable(*head*)in the CRL as $L_f$. A neighbor $\Gamma$ of $\Gamma$ is constructed by assigning a new value to $L_f$ while keeping all other variables of $\Gamma$ unchanged.

4)CRL is updated at the end of each iteration. For each iteration, a neighbor $\Gamma$ with the best $C(\Gamma)$ is found to replace the current solution $\Gamma$. We use $\Omega_f$ to represent these to f variables that are in conflict with $L_f$ in $\Gamma$. The

Costs of $L_f$ and $L_x \in \Omega_f$ are recalculated for $\Gamma$.If the New cost of $L_x$ is zero, it means the light path assigned To $L_x$ is no longer in conflict with the other light paths in $\Gamma$; thus, $L_x$ is deleted from the CRL. If the cost of $L_f$ Is zero, $L_f$ is deleted and the next variable in the CRL Becomes the new *head*; otherwise ,it means $L_f$ must be In conflict with one or more variables in $\Gamma$. Let $\Omega_f$

represents that set of variables. If a$L_y \in \Omega_f$is already in the CRL,$L_y$is moved to the head of the CRL. Other variables in $\Omega_f$ are inserted into the head of the CRL one by one in a random sequence. With this method, the *head* $L_f$ is renewed for each iteration, and the search process is guided to a new neighborhood based on the new *head*.

5)In order to prevent the TS from searching a recently searched neighborhood, a *tabu list* is used to store the recently searched CRL head $L_f$.We apply the *tabu rule* (TR) where by ,a $\Gamma$, which may lead to a new CRL *head* that is forbidden by the *tabu list*, is marked with a tabu tag. A tagged solution is then dropped from the neighborhood $N(\Gamma)$ and no longer eligible to be selected as the next current solution.

6)An *aspiration rule*(AR)is applied to overrule the *tabu rule* when a good solution is found. For each iteration, if the cost of a tagged solution is better than the best cost achieved among all the iterations so far, the tabu tag is removed by this rule.

7)The TSsub routine stops based on two conditions. The first stop page condition is when $C(\Gamma)$=0,in this case,a *feasible solution* is found. These condition is when The best cost achieved does not improve for$/\Delta/$number Of iterations. In this case ,the TS sub routine is deemed to have failed in finding a *feasible solution*

The pseudo codes of the proposed TS algorithm are as follow:

```
Begin of Main/ /Main TS algorithm
  Construct the initial $W$ with first fit algorithm; While( Tb Sub($W$)=feasible)do{$/W/$:=$/W/$-1;} Return $/W/$+1;
End Of Main
Tb Sub($W$)Subroutine //Begin of Subroutine
  Construct initial solution $\Gamma_{ini}$; Create the CRL from $\Gamma_{ini}$;
  $\Gamma$:=$\Gamma_{ini}$;Cycle:=0;
  If($C(\Gamma)$=0) return feasible;
    Else

  While($C(\Gamma)$=0&Cycle=$/\Delta/$)do
    Begin
      $L_f$:= head of the CRL; Construct $N(\Gamma)$basedon$L_f$;
                                  Apply TR &AR to find the best $C(\Gamma)$ for all $\Gamma \in N(\Gamma)$;
                                 $\Gamma$:= $\Gamma$; Update the CRL ,the tabulist and Cycle;
                                       Else if not feasible solution
End of while End of Subroutine
```

## III.SIMULATIONS

We are running two sets of simulation for getting results, one for small networks and another for large networks. The network consist of 50 nodes and 164 unidirectional links. The path of light is randomly generated with atmost one request for each pair node to the stochastic nature of the TS algorithm, the construction of $\Gamma_{ini}$ may lead to significantly different results. To evaluate the stability of our proposed algorithm, the following simulation isrun.Thetestnetworkisalargenetworkwith50 nodesand164unidirectional links[7].

TALEI

RESULTSFOR10TRIALS

| | Ave. | Max. | Min. | Std.Dev./Ave. |
|---|---|---|---|---|
| /∆/=500 | 44.6 | 45 | 43 | 1.6% |
| /∆/=300 | 28.8 | 29 | 28 | 1.5% |

The standard deviations of the results for the 300 and 500 sets are about1.5% and1.6% of the average value respectively. This shows that our proposed algorithm exhibits reasonable performance stability.

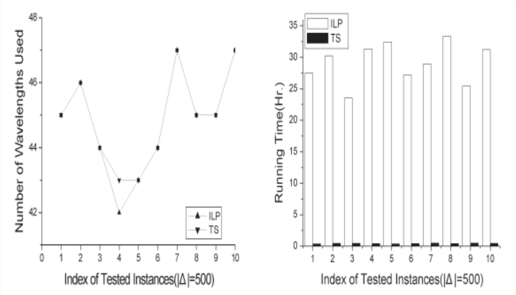

Fig.1. Result for 10 trials with /∆/=500: (a) number of wavelengths used; (b) running time.

## IV.CONCLUSION

We propose new concurrent tabu search algorithm to solve the wavelength problem. By our knowledge our approach will give the optimal solution to wavelength and routing. The simulation results will give the solution.

## REFERENCES

[1] D.BanerjeeandB. Mukherjee, “A practical approach for routing and wavelength assignment in large wavelength-routed optical networks,” *IEEEJ.Select.AreasCommun.*,vol.14,pp.903-908,June1996.

[2] K.LeeandK.Choe,“Routing and wavelength assignment in WDM all-opticalnetworks,”*IEEEElectron.Lett.*,vol.36,pp.964-966,May 2000.

[3] N.BanerjeeandS.Sharan,“A evolutionary algorithm for solving the single objective static routing and wavelength assignment problem in WDMnetworks,”in*Proc.IEEEISIP’04*,Jan.2004,pp.13-18.

[4] J.Kuri,N.Puech,M.Gagnaire,E.Dotaro,andR.Douville,“Routing and wavelength assignment of scheduled lightpath demands,” *IEEEJ. Select.AreasCommun.*,vol.21,pp.1231-1240,Oct.2003.

[5] F.GloverandM.Laguna,*Tabu Search*.Kluwer Academic Publishers, 1997.

[6] J. Y. Yen, “Finding the K shortest loopless paths in a network,” *ManagementScience*,vol.17,pp.712-716,July1971.

[7] S.OhtaandA.Greca,“Comparison of routing and wavelength assign- ment algorithms for optical networks, ”in *Proc .IEEE Workshop on High PerformanceSwitchingandRouting*,May2001,pp.146-149.

[8] Y. Wang, T. H. Cheng, andB. Mukherjee, “Dynamic routing and Wavelength assignment scheme for protection against node failure ,”in *Proc.IEEEGLOBECOM2003*,Dec.2003,pp.2585-2589.


Author Profile : K. Raghavendra Rao, working as assistant professor in Computer Science Engineering from Anurag group of Instituions (Formerly CVSR College of Engineering), Venkatapur(V), Ghatkesar(M), Ranga Reddy District, Hyderabad-500088, Telangana State, India.


Vadipina Amarnadh, Working as Assistant Professor in computer science and Engineering from Anurag group of Institutions (Formerly CVSR college of Engineering), Venkatapur(V), Ghatkesar(M), Ranga Reddy District, Hyderabad-500088, Telangana State,