

# Static Routing and Wavelength Assignment Inspired by Particle Swarm Optimization

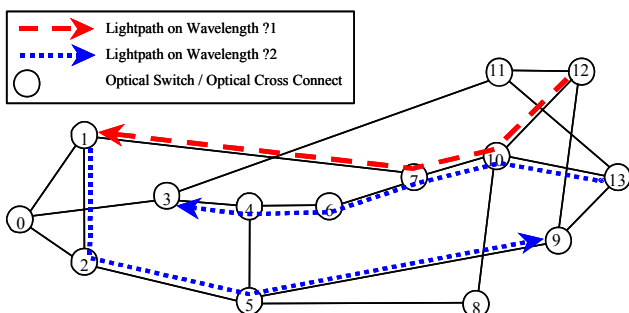
A. Hassan, C. Phillips

**Abstract**— Static Routing and Wavelength Assignment (RWA) problem for wavelength routing optical networks has traditionally been solved using Integer Linear Programming (ILP) and graph colouring. These problem domain search schemes become inefficient for solving large NP-hard (Non Polynomial Time - hard) optimization problems and in dynamic environments because of their computational complexity and execution time. In this paper, a heuristic approach inspired by Particle Swarm Optimization (PSO) is proposed for solving static RWA. A novel encoding scheme for members of the swarm population is proposed. To help the particles to converge towards optimal solution quickly, a novel scheme is devised for route selection, during particle's search. In order to diversify the search in problem space, a special operation is proposed for global best particle. The proposed scheme achieves optimal solution in significantly lower number of iterations.

**Index Terms**— All Optical networking, Particle Swarm Optimization, ASON, Wavelength Continuity Constraint, Static RWA, ILP.

## I. INTRODUCTION

In WDM networks, communication is achieved by setting up all-optical channels called lightpaths between end-points [1]. A lightpath can traverse multiple optical link fibres; however, the information sent by the lightpath does not need Optical-Electrical-Optical conversion at intermediate nodes.



**Figure 1:- 14 Node, Wavelength Routed WDM NSFNET with lightpath connections**

Establishment of lightpaths creates a logical topology on the top of physical topology of WDM optical network. If the

intermediate nodes in the route, chosen for lightpath setup, are not equipped with wavelength-conversion capability, then same wavelength needs to be assigned to the lightpath over all fibre links. This property is called the *wavelength continuity constraint*. Two lightpaths sharing a common edge of the network need to be assigned unique wavelengths. This is called the *wavelength clash constraint*. To set up a lightpath, an appropriate route from the source to destination node needs to be selected and unique wavelength assigned. Typically, connection requests to setup lightpaths can be categorized into following types: (1) Static (2) Scheduled (3) Incremental (4) Dynamic.

In the static demands, all the connection requests are known in advance. In scheduled lightpath demands, all the connection requests plus their set-up and tear-down times are also known in advance. In incremental lightpath demands, connection requests arrive one-by-one, without any prior knowledge, and will stay in the network for indefinite time [1]. In dynamic lightpath demands, connection requests arrive unexpectedly without any prior knowledge, and after lightpath setup, they will remain in the network for limited amount of time. The problem of finding an appropriate route and wavelength for setting up lightpath is known as the Routing and Wavelength Assignment problem (RWA).

## II. STATIC ROUTING & WAVELENGTH ASSIGNMENT

In this paper the static routing and wavelength assignment problem, also known as Static Lightpath Establishment (SLE) problem, is addressed for wavelength routing optical networks with wavelength continuity constraint. Static RWA is NP (Nondeterministic - polynomial time) hard optimization problem and is performed offline [1, 2]. The objective is to minimize the number of wavelengths required to setup lightpaths for a given set of connection requests. Alternatively, the objective of static RWA can be defined as maximizing the number of lightpaths that can be setup from a given number of wavelengths.

### A. Integer Linear Programming (ILP) Formulation

Static RWA problem can be formulated as multi-commodity problem with integer flows in each link. The objective function in this case is to minimize the flow in each link, which, in turn, corresponds to minimizing the number of lightpaths passing through a particular link [2]. Let,

$\lambda_{sd}$  = Traffic (i.e. lightpath) from source 's' to destination 'd';

$\lambda_{sd} = 1$  if there is a lightpath from source 's' to destination 'd', considering at most one lightpath at a time, otherwise  $\lambda_{sd} = 0$ ;

$F_{ij}^{sd}$  = Traffic (in terms of number of lightpaths) that is flowing from source 's' to destination 'd' on physical link 'ij'.

The ILP formulation can be written as:

$$\text{Minimize: } F_{\max}$$

Such that:

$$F_{\max} \geq \sum_{s,d} F_{ij}^{sd} \forall ij$$

$$\sum_i F_{ij}^{sd} - \sum_k F_{jk}^{sd} = \begin{cases} \lambda_{sd} & \text{if } s = j \\ -\lambda_{sd} & \text{if } d = j \\ 0 & \text{otherwise} \end{cases}$$

$$\lambda_{sd} = 0, 1$$

$$F_{ij}^{sd} = 0, 1$$

### B. Static RWA Problem

When static RWA is solved by integer linear programming using a multi-commodity formulation, it results in rapid increase in the number of equations and variables. Let's consider an example of a NSFNET having 14 nodes, and 21 edges, and an average of 5 unidirectional lightpaths per node. Then...

The number of  $\lambda_{sd}$  variables =  $14 * 13 = 182$  'sd' pairs.

The number of  $F_{ij}^{sd}$  variables =  $182$  'sd' pairs \*  $21$  'ij' pairs =  $3822$ .

The number of Equations =  $[1 + 21 + 2548 + 182 + 3822]^2 = (6574)^2$

In order to decrease the computational complexity, different problem space pruning techniques are used like randomized rounding, path stripping etc. These search space reduction schemes can be found in [2, 3]. Even after applying problem-space reduction techniques, solving static RWA using ILP is computationally extensive and typically requires a huge execution time.

Therefore, to decrease the computational power and time required to solve NP-hard optimization problems, different heuristic based algorithms like greedy search, Tabu search, local search, simulated annealing, stochastic diffusion search, swarm intelligence algorithms (like ant colony optimization, genetic algorithms, particle swarm optimization) and many more have been used successfully. Unlike mathematical problem-domain search algorithms like ILP, heuristic algorithms don't guarantee an optimal solution to the problem domain. Each heuristic technique has its own advantages and disadvantages and the efficiency of applying a particular technique depends on many factors like computational complexity, execution time, quality of solution, robustness

and so forth.

### C. Related Work

Heuristic algorithms usually divide static RWA problem into two sub-problems; routing sub-problem and wavelength assignment sub-problem. An overview of well-known static RWA algorithms, their functional classification, advantages and disadvantages can be found in [4]. In [2], routing and wavelength assignment sub-problems are solved independently. For solving routing sub-problem, multi-commodity flow formulation is used. For reducing the size of problem search space, randomized rounding is employed. The wavelength assignment sub-problem is solved using a graph-colouring technique. In [5], the authors used an iterative algorithm based on local random search for finding a reasonably good route. To assign an appropriate wavelength, a greedy graph colouring algorithm is employed. A generalization of graph colouring problem, called partition colouring problem, and its application for solving static RWA in WDM all-optical network is proposed in [6]. In [7], a conflict graph is constructed by computing alternative routes for each connection request and then a Tabu search heuristic is applied for solving partition colouring problem.

A unified approach to solve static RWA is proposed in [8] where both routing and wavelength assignment sub-problems are solved simultaneously using an integer formulation and a column generation technique. A heuristic based algorithm using bin-packing approach is proposed in [9]. In [10] the static RWA is solved using a genetic algorithm where members of population called chromosomes, use genetic operators to generate new members and then members are selected for new generation based on fitness function. In [11], ant colony optimization is used to minimize the number of wavelengths required to provision all the given connection requests. Particle swarm optimization is used to solve the static RWA in [12]. In [12], like traditional techniques, the objective is to minimize the number of wavelengths. Routing and wavelength assignment sub-problems are solved separately. While solving routing sub-problem, a first-fit algorithm is applied to determine the number of wavelengths required, in every iteration of each particle. This increases the computational complexity and time required to solve the RWA problem. A desirable feature of any heuristic algorithm for static RWA is its ability to solve problem space optimally using fewer iterations and thus in lesser time.

## III. ROUTING & WAVELENGTH ASSIGNMENT USING PARTICLE SWARM OPTIMIZATION (STATIC TRAFFIC CASE)

In this paper, a PSO technique is used to solve Routing and Wavelength Assignment problem (RWA) for static traffic, where all the connection requests are known in advance. PSO is a population-based algorithm based on the social psychology metaphor. Members of the population observe other members of the population and try to improve themselves. The whole population of individuals is called the swarm, while each individual is called a particle. A particle is

an encoding of a candidate solution. Every particle has a fitness value that represents the quality of the particle as a solution. Unlike ACO, in PSO no pheromone table needs to be maintained for next hop decision-making. In GA, the fitness function is used to determine which member of the population will stay in the next generation and which will be excluded. This requires the need for careful design of fitness function. In PSO, there is no selection operation based on the fitness function. Rather, each member of the population evolves towards better a solution through the iterative process. So there is always a chance that a member of population having a bad initial fitness value, evolving over time to become the best member in the entire population of particles.

The neighbourhood for a particle is defined as a sub-collection of particles that are within a certain distance from it. Distance here is defined in terms of degree of similarity in the number of routes between two particles. The size of the neighbourhood is a user defined parameter. When using PSO for solving any problem, first of all particles are initialized randomly or using some suitable scheme. Then fitness function is applied to each particle in order to quantize its fitness. The particle with best fitness value in the neighbourhood is marked as local best particle and the particle having best fitness value in the swarm is marked global best. For each particle, its velocity is computed. Then that velocity is applied to that particle. As a result, the particle will move to another position, representing another candidate solution for the problem domain. The magnitude of this move is a function of current position of the particle and the distance between itself and global best (or local best) particle. Particles continue to move around synchronously in this manner trying to better themselves in comparison with their own performance and that of their neighbours.

#### A. PSEUDO CODE for PSO Algorithm

The generalized pseudo-code of particle swarm optimization is as follows:

- Initialize the particles
- Quantize each particle using fitness function.
- The particle having best fitness value will be marked as GLOBAL BEST, and the particle having best fitness value in the neighbourhood will be marked as LOCAL BEST particle.
- For each particle Do:
  - Find its velocity according to Global best and/or local best particle.
  - Apply the velocity to the particle.
  - After this, the particle will move to another (new) position.
  - Then re-apply fitness function to update the Global/Local best particles.
- Iterate until optimal solution or solution of desired quality is not found.

#### B. Modified PSO equations for solving RWA

In order to apply PSO for solving the RWA problem, the general PSO equations are modified so that PSO can be mapped for RWA. In the proposed static RWA algorithm, the *velocity* of movement for each particle is either influenced/governed according to global best or local best particle but not both at any one time as shown in equation 1. The velocity is then used in the determination of the next position to move to in the solution space where this movement is represented by equation 2. In any iteration, we will change the position of the particle either according to the global best particle or local best particle. So the equation to find out velocity for the particle has been modified as follows.

$$V_{i+1} = \alpha * C1 (P_{gb} - P_{xi}) + (1 - \alpha) * C2 (P_{lb} - P_{xi}) \quad (1)$$

where  $\alpha$  is either 0 or 1

$$X_{i+1} = X_i + V_{i+1} \quad (2)$$

C1 & C2 are social learning parameters

$P_{gb}$  = Position of global best particle

$P_{lb}$  = Position of local best particle

$P_{xi}$  = Position of current particle.

#### C. Novel Encoding Scheme for Particles

For each connection request, an appropriate route is selected randomly from pre-computed k-shortest paths, where each route is identified by a unique route-id. Particle is represented as a vector of route ids. E.g. In the case NSFNET shown in figure 1, let's say we have eight connection requests. The chosen routes, routes id and the particle having these chosen routes may look like, as shown in the figure 2.5 (a) & 2.5 (b) respectively.

(a)

Connection#	Connection Request	Chosen Routes	Route Id
1	1 → 3	1 → 0 → 3	R39
2	4 → 8	4 → 6 → 7 → 10 → 8	R73
3	9 → 11	9 → 13 → 11	R312
4	12 → 5	12 → 10 → 8 → 5	R1096
5	9 → 2	9 → 5 → 2	R747
6	6 → 13	6 → 4 → 5 → 9 → 13	R201
7	8 → 7	8 → 10 → 7	R627
8	6 → 0	6 → 4 → 5 → 2 → 0	R559

(b)

R39	R73	R312	R1096	R747	R201	R627	R559
-----	-----	------	-------	------	------	------	------

**Figure 2.5: (a) Connection requests to be provisioned, chosen routes and their corresponding IDs. (b) Particle's representation corresponding to the chosen routes.**

With each particle, a *common edge usage table* is attached, which will show the edge usage in the network, after assigning the routes represented in that particle. This table will help us to determine which edges of the network will be overloaded, if we choose the routes of the current particle.

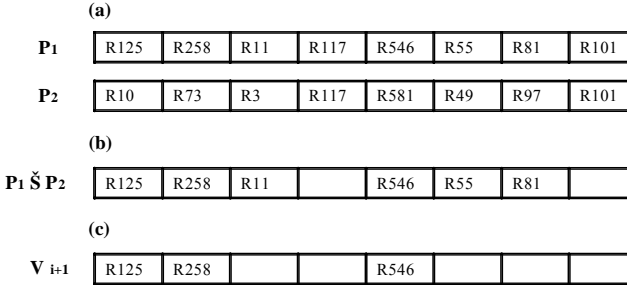
#### D. Calculating the Velocity of a Particle

Equation 1 is used to calculate the velocity for any particle.

Velocity here is a vector of routes that will be replaced in the current particle according to routes in global or local best particle.

- $(P_{gb} - P_{xi})$  = Routes that are different in the global best particle and current particle.
- $(P_{lb} - P_{xi})$  = Routes that are different in the local best particle and current particle.
- C1 and C2 represent social learning parameters and will determine the number of routes to be replaced.
- ‘ $\alpha$ ’ is used to select whether we will be changing routes in the current particle according to either global best particle or local best particle, but not both in a single iteration for any particle.

Unlike [12], while calculating velocity, instead of randomly picking up routes from  $(P_{gb} - P_{xi})$  or  $(P_{lb} - P_{xi})$  vector, those routes are chosen which traverse the most congested edges in the network. For this purpose, edge usage table will give information about degree of congestion (number of lightpaths, traversing) on each edge of the network. Velocity here will be a vector of route ids, which indicate the routes to be replaced in the current particle by the corresponding routes in the global best or local best particle. For example: As shown in the figure 2.6(a), let’s say particle  $P_2$  needs to update its position according to the position of particle  $P_1$ . The particles  $P_1$  and  $P_2$  have six different routes (at position 1, 2, 3, 5, 6 and 7) as shown in figure 2.6(b).



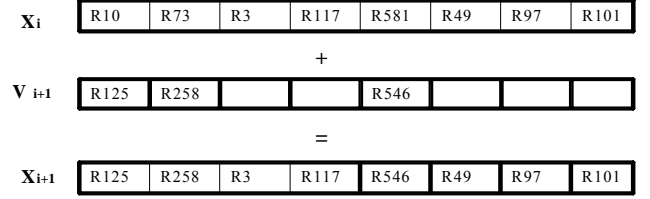
**Figure 2.6: (a, b) Two particles  $P_1$  and  $P_2$  (b) Routes that are different  $P_1$  and  $P_2$ . (c) New velocity for particle  $P_2$**

If the social learning parameter is 0.5, then three routes in  $P_2$  need to be replaced by the routes in the  $P_1$  (corresponding to the respective connection request). Instead of picking up routes randomly from  $P_1 - P_2$  vector, those routes are picked that traverse the most overloaded edges of the networks. According to the edge usage table of  $P_2$ , let’s say route R11, R55 and R81 are traversing most overloaded edges of the network, then the resulting new velocity for particle  $P_2$  is shown in the figure 2.6 (c).

#### E. Applying Velocity to a Particle (Redefining the ‘+’ operator):

Equation 2 is used to give the velocity calculated, to the current particle so that it can move to a new position. For the RWA problem, we need to redefine the meaning of ‘+’ operator. The routes in the velocity vector will replace the

corresponding routes in the current particle as shown in the example in figure 2.7.



**Figure 2.7: Applying velocity to a particle to move it to a new position.**

#### F. The Fitness Function:

Equation 3 is used to quantize the particles in terms of their fitness.

$$F(x) = 1 / \text{Cost}(x) \quad (3)$$

$$\text{Cost}(x) = P_1 * \text{APL} + P_2 * \Theta$$

APL = Average Path Length,  $\Theta$  = Number of ‘directed edge disjoint route’ sets.  $P_1$  &  $P_2$  are user-defined constants.

‘ $\Theta$ ’, here is equivalent to the number of wavelengths required to set up given lightpath requests, using routes in the current particle. All the routes in each of these ‘directed edge disjoint route’ set can be assigned same wavelength, as no two routes in a single set will share a common directed edge of the network. In other words, all the routes in the same set will be assigned same wavelength, however each set will be assigned a distinct wavelength. This also removes the need to have a separate ‘wavelength assignment’ algorithm, for calculating appropriate wavelength for each route calculated.

To find number of edge disjoint route sets from the routes chosen for any particle, create an empty set and sort all the chosen routes of the current particle in non-increasing order. Pick up a route, and starting from first set, attempt to insert it in the set such that no two routes with a common directed edge should lie in the same set. If route cannot be placed in all the previous sets, create a new set and place the route there.

#### G. Proposed Novel Strategies to Improve Problem Space Search:

In order to help the particles, find a combination of routes that can move them to a position with better fitness value, a strategy called *St. (1)* here, has been proposed. This scheme helps to prioritize the replacement of routes within in a particle. To help the particles to move towards a better position quickly, while avoiding local minima as much as possible, a strategy called *St. (2)* has been proposed. At the same time, to diversify the problem space search, a novel operation has been proposed for global best particle, in *St. (3)*. This operation attempts to improve the fitness value of global best particle. Unlike traditional particle swarm optimization, *St. (3)* causes the global best particle to explore the problem search space. However, the advantage of this operation is, that the global best particle will always move to a better position, i.e. towards the best position within the locality.

- *St. (1)*: For the current particle, select those routes which traverse the most congested edges of the

network. Edge usage table associated with the particle can help to determine this.

- *St. (2)*: Instead of randomly selecting the routes over the most congested edges to be replaced by routes in global/local best particle, replace a route in the current particle with an alternate route (from global/local best) only when the number of channels (congestion) of the most loaded link in the alternative route is lower than the congestion of the most loaded link in the previously assigned route.
- *St. (3)*: For global best and local best particle in the swarm, attempt ‘t’ times to find an alternate route from pre-computed k-shortest paths, and replace it, such that congestion on the most loaded link in the alternative route is lower than the congestion of the most loaded link in the previously assigned route. (For all simulation results presented in this paper, the value of ‘t’ is assumed to be 3)

#### H. Simulation Results and Analysis:

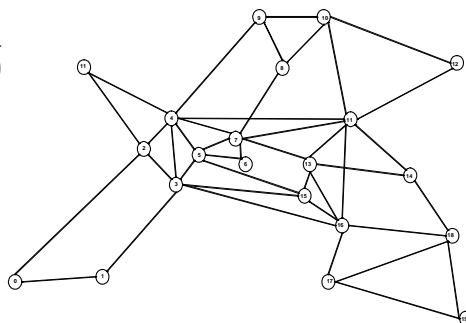
The proposed algorithm has been implemented using Visual C++.net, without any code optimization on Dell optiplex GX520 (CPU 3.00 GHz, 1.99 GB of RAM) in windows environment. In order to determine the performance measures of the proposed algorithm, lower bound for the number of wavelength required presented in [9] has been used. The bound on the number of wavelengths needed to establish a given set ‘T’ of ‘n’ lightpath requests in a network ‘G’ with |V| nodes and |E<sub>p</sub>| edges can be calculated by (4), as follows:

$$LB_w = \max \left\{ \max_{i \in V} \left[ \frac{\Delta_l(i)}{\Delta_p(i)} \right], \left[ \frac{\sum_{j=1}^n l(SP_j)}{2 * |E_p|} \right] \right\} \quad (4)$$

$\Delta_l(i)$  represents the logical degree of node  $i$ , i.e. the number of lightpaths for which node  $i$  is the source node.  $\Delta_p(i)$  represents the physical degree of node  $i$ .  $l(SP_j)$  is the length of the shortest path in ‘G’ of lightpath request  $(s_j, d_j)$ , where ‘ $s_j$ ’ is the source node and ‘ $d_j$ ’ is the destination node.

For the experiments, uniform traffic demand is assumed, where all the node pairs are assigned a lightpath consisting of a physical path and a unique wavelength. Then, if  $l(SP_j)$  is the length of the shortest path in ‘G’ of lightpath request  $(s_j, d_j)$ , and |V| is the number of nodes in the network, then the lower bound for the average path length can be calculated by (5), as follows:

$$LB_{APL} = \frac{\sum_{j=1}^{|n|} l(SP_j)}{|V| \times (|V| - 1)} \quad (5)$$



These lower bounds are theoretical bounds and may not be achievable practically. Simulations have been done using proposed algorithm for the networks shown in the figure 1, 2.8 and 2.9. For each network, table 1 illustrates the  $LB_w$  (Lower bound on the number of wavelength required),  $LB_{APL}$  (Lower bound on the average path length), user defined algorithmic parameters like number of particles used, size of neighbourhood (NS), social learning parameter ( $C_1$ ,  $C_2$ ) and maximum iterations allowed in each set of simulations. It also shows the iteration number when the swarm achieved best fitness value; average path length (APL) and the number of edge disjoint route sets (9).

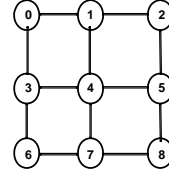


Figure 2.8

Figure 2.9:  
20 Node, EON

The results in table 1, shows clearly that the proposed algorithm achieves lower bound on the number of wavelengths required while minimizing the average path length (APL) as compared to [12], in significantly lesser number of iterations. For example, in the case on NSFNET, [12] achieves  $LB_w = 13$ , in 9299 iterations with average path length of 2.5400. On the other hand, the proposed algorithm achieves  $LB_w = 13$ , in just 3640 iteration with an average path length of 2.39011. So there’s significant improvement both in terms of number of iterations required and average path length, while reaching  $LB_w = 13$ , for NSFNET. Similarly, as compared to a genetic algorithm proposed in [10], which requires 23 wavelengths to set up 100 SD pairs (source-destination pair) in NSFNET, the proposed algorithm can set up 182 SD pairs in just 13 wavelengths (First-Fit algorithm requires 41 wavelength to set up 100 SD pairs in NSFNET [10]).

	No. of particles	NS	$C_1, C_2$	Max. Iterations	Iteration no. having best fitness	APL	$\mathcal{Q}$			
9 Nodes, 12 Edges (figure 2.8) $LB_W = 6$ $LB_{APL} = 2.0$	10	3	0.0009	100	98	4.16667	18			
				500	485	3.75	16			
				1000	982	3.19444	13			
				1500	1462	2.88889	11			
				2000	1932	2.52778	9			
				2500	2494	2.03055	8			
				3000	2868	2.1111	7			
				<b>3000</b>	<b>2993</b>	<b>2.05556</b>	<b>7</b>			
NSFNET (figure 1) $LB_W = 13$ $LB_{APL} = 2.14$	14	3	0.05	100	100	4.97253	34			
				500	481	4.1978	26			
				1000	987	3.79121	22			
				1500	1475	3.57692	19			
				2000	1981	3.3956	18			
				2500	2471	3.08791	16			
				3000	2995	2.68681	14			
				3500	3487	2.52747	13			
				<b>4000</b>	<b>3640</b>	<b>2.39011</b>	<b>13</b>			
				EON (figure 2.9) $LB_W = 18$ $LB_{APL} = 2.36$	20	3	0.05	1000	997	3.63684
2000	1927	3.38158	23							
4000	3789	3.14474	21							
9000	8856	2.90526	19							
<b>12000</b>	<b>11238</b>	<b>2.77895</b>	<b>18</b>							
25	3	0.05	<b>12000</b>					<b>13233</b>	<b>2.77368</b>	<b>18</b>

**Table 1: Static RWA PSO Performance for Networks Shown in the Figure 1, 2.8 and 2.9.**

Values in bold indicates the best values of ‘APL’ and ‘ $\mathcal{Q}$ ’ achieved by the swarm. Using all three strategies, St. (1), St. (2) & St.(3)

Table 2 summarize the comparative results of using different combinations of proposed strategies for NSFNET under the same network conditions. The results show that St. (1) and St. (3) significantly improve the results both in terms of wavelengths and APL. St. (2) shows improvement, but not as significant as St. (1) and St. (3) does alone. St. (1) indirectly decreases the number of wavelengths required and average path length, by replacing routes traversing the most congested edges of the network. St. (3) on the other hand, diversifies the problem search by giving the global best particle in the swarm, an opportunity to fine tune and improve its position towards better fitness value.

	Strategies Used:	Max. Iterations	Iteration no. having best fitness	APL	$\mathcal{Q}$
NSFNET (figure 2.9) $LB_W = 13$ $LB_{APL} = 2.14$ Particles= 14 NS = 3 $C_1 = 0.05$ $C_2 = 0.05$	None	1500	1487	3.93956	26
		B.F.V	1504	3.93407	26
	St. (1)	1500	1498	3.6044	21
		B.F.V	1534	3.5989	21
	St. (1) and St. (2)	1500	1455	3.41758	21
		B.F.V	1602	3.4036	21
	St. (1) and St. (3)	1500	1497	3.58791	20
		B.F.V	3820	2.62088	14
	St. (2) and St. (3)	1500	1486	2.68681	15
		B.F.V	1962	2.52747	14
St. (1), St. (2) and St. (3)	1500	1475	3.57692	19	
	B.F.V	3640	2.39011	13	

**Table 2: Static RWA PSO Performance for Different Combinations of the Proposed Strategies**

(B.F.V = Iterate for Best fitness Value, until all the particles converge)

#### IV. CONCLUSIONS

In this paper swarm intelligence based, particle swarm

optimization is proposed to solve static routing and wavelength assignment problem, with wavelength continuity constraint applied. Unlike traditional schemes where the objective of static RWA problem is to minimize the number of wavelengths required, the proposed algorithm takes into account both average path length of the chosen routes and the number of wavelengths required. A novel encoding scheme has been proposed for particles of the swarm. In order to diversify the search, a novel operation, St. (3) has been proposed for global best particle. For replacement of routes with the respective routes in global/local best particle, a novel strategy, St. (1) has been adopted that helps to better converge towards optimal solution in significantly lesser iterations, as compared to [12]. Instead of random selection, St. (2) has been proposed that also shows some improvement in the results. The proposed scheme also shows significantly better results as compared to a genetic algorithm presented in [10].

In the future, the quality of solutions obtained from proposed scheme will be compared against techniques like ILP and other swarm intelligence algorithms like genetic algorithms and ant colony optimization, in terms of different network performance parameters. This work will be extended to consider optical networks with full or partial wavelength conversion capability and survivable optical networks.

#### REFERENCES

- [1] Hui Zang, Jason P. Jue and Biswanath Mukherjee. “A Review of Routing and Wavelength Assignment Approaches for Wavelength-Routed Optical WDM Networks”, Optical Networks Mag., vol. 1, no. 1, pp. 47-60, Jan. 2000.
- [2] Dhritiman Banerjee and Biswanath Mukherjee. “A practical approach for routing and wavelength assignment in large wavelength-routed optical networks”. Selected Areas in Communications, IEEE Journal on Volume 14, Issue 5, June 1996 Page(s):903 – 908.
- [3] Krishnaswamy, R.M.; Sivarajan, K.N. “Algorithms for Routing and Wavelength Assignment Based on Solutions of LP-Relaxations”. IEEE Communications Letters, vol. 5, No. 10, October 2001, Page(s):435 – 437.
- [4] J.S. Choi, N. Golmie, F. Lapeyere, F. Mouveaux, D. Su, “A functional classification of routing and wavelength assignment schemes in DWDM networks: Static case”, Proceedings of the VII International Conference on Optical Communication and Networks, January 2000.
- [5] E. Hyttiä and J. Virtamo, “Wavelength Assignment and Routing in WDM Networks”, Fourteenth Nordic Teletraffic Seminar, NTS-14, pp. 31-40, 1998, Copenhagen, Denmark.
- [6] Guangzhi Li and Rahul Simha, “The Partition Coloring Problem and its Application to Wavelength Routing and Assignment”, First Workshop on Optical Networks, Dallas, TX, 2000.
- [7] Altan K., Demeter G. and Semith B. “All-Optical Networking”. Turkish Journal of Electrical Engineering & Computer Sciences, Volume 9, Issue 2 (2001), page: 69-122.
- [8] Kyungsik Lee, Kug Chang Kang, Taehan Lee, and Sungsoo Park, “An Optimization Approach to Routing and Wavelength Assignment in WDM All-Optical Mesh Networks without Wavelength Conversion”, ETRI Journal, vol.24, no.2, Apr. 2002, pp.131-141.
- [9] Nina Skorin-Kapov, “Routing and wavelength assignment in optical networks using bin packing based algorithms”, European Journal of

Operational Research, Volume: 177, Issue 2, March 2007, Pages: 1167-1179.

- [10] Nilanjan Banerjee, Vaibhav Mehta and Sugam Pandey, "A Genetic Algorithm Approach for Solving the Routing and Wavelength Assignment Problem in WDM Networks", IEEE/IEE International Conference on Networking, pp. 70-78, March 2004,
- [11] Varela, G.N. and Sinclair, M.C., "Ant Colony Optimization for Virtual-Wavelength-Path Routing and Wavelength Allocation", *Proceedings of the Congress on Evolutionary Computation*, IEEE Press, volume 3, pages 1809-1816, Washington D.C., USA, 6-9 July 1999.
- [12] T.M. Rao and V. Anand, "Routing and Wavelength Assignment in Optical Networks using Particle Swarm Optimization", IEEE Sarnoff Symposium 2006, Publication date: 15 Dec. 2006, ISBN-13: 9781424400027.