A Pareto based Approach with Elitist Learning Strategy for MPLS/GMPS Networks

Mohsin Masood
Electronics and Electrical Engineering Department,
University of Strathclyde,
Glasgow, UK,
mohsin.masood@strath.ac.uk

Mohamed Mostafa Fouad
Arab Academy for Science, Technology and Maritime, Transport,
Cairo, Egypt
mohamed_mostafa@aast.edu

Ivan Glesk
Electronics and Electrical Engineering Department,
University of Strathclyde,
Glasgow, UK
ivan.glesk@strath.ac.uk

***Abstract*—Modern telecommunication networks are based on diverse applications that highlighted the status of efficient use of network resources and performance optimization. Various methodologies are developed to address multi-objectives optimization within the traffic engineering of MPLS/ GMPLS networks. However, Pareto based approach can be used to achieve the optimization of multiple conflicting objective functions concurrently. We considered two objective functions such as routing and load balancing costs functions. In the paper, we introduce a heuristics algorithm for solving multi-objective multiple constrained optimization (MCOP) in MPLS/ GMPLS networks. The paper proposes the application of a Pareto based particle swarm optimization (PPSO) for such network's type and through a comparative analysis tests its efficiency against another modified version; Pareto based particle swarm optimization with elitist learning strategy (PPSO_ELS). The simulation results showed that the former proposed approach not only solved the MCOP problem but also provide effective solution for exploration problem attached with PPSO algorithm.**

Keywords- MPLS, GMPLS networks optimization, particle swarm optimization, heuristic algorithm, traffic engineering, weight inertia

I. INTRODUCTION

In recent times, rapid growth in telecommunication applications increase the importance of quality of service, reliability and effective usability of resources, that made traffic engineering a critical technology for the operation of large backbone networks. Traffic engineering (TE) encompasses methodologies of controlling internet traffic, which are used to achieve reliable and coherent flow of traffic, effective utilization of network resources and a planning usage of network capacity. Using traffic engineering (TE), traffic movement within the network is accomplished by splitting it into multiple routes which are computed by routing algorithms. Traditional routing algorithms are based on shortest path computation from source to destination. Multiple path routing balances traffic over various routes by avoiding number of congested links, by efficient use of network resources utilization, minimizing packet losses and bound delays. This introduces multiple objectives for computation of multiple routes according to traffic engineering parameters. Multi-constrained based routing has gotten a rapid interest when considering the range of latest real-time multimedia applications. From network optimization prospective, the multiple constrained based network that has multiple objectives to be achieved is known as multi-objective multiple constrained (MCOP) optimization problem, which may increase the computational complexities. The trade-offs between multiple objectives can be addressed by an effective heuristic algorithm which may result in an approximate feasible solution [1], [2].

Traditionally, effective traffic engineering faces problems with conventional IP based network technologies. However, recent development of multiple protocol label switched (MPLS) networks and the extended version generalized MPLS networks, improved IP systems functional limitations regarding traffic engineering [3], [4]. This paper will focus on multipath routing traffic engineering in MPLS/GMPLS networks. The goal is to meet the expectations of requirements for emerging network applications by optimizing the network quality of services and effective utilization of network resources. We propose a heuristic algorithm which is based on the linear combination of multiple objectives as well as the pareto-based technique that consist of optimal sets of trade-offs.

II. RELATED WORK

Effective network utilization and performance optimization are challenging tasks for service providers [5]. This section provides a brief review of research related to multiple path routing traffic engineering in MPLS/GMPLS networks. In recent years, traffic engineering optimization has gained a significant importance and various approaches have been proposed to address this problem [5]. Routing protocols play a critical role from the prospective of multiple path routing traffic engineering optimization. MPLS/ GMPLS networks are dependent on interior gateway routing protocols such as the open shortest path first (OSPF) and intermediate system-to-intermediate system protocols [6]. Furthermore, constrained based routing is the computation of routes that have administrated constraints and limited bandwidths. At the early stages, the constrained shortest path first (CSPF) routing was to remove the edges from the network that violates certain constraints [7]. But this approach is not effective for the network where subsequent traffic requests are receiving. Therefore, computation of shortest path for one traffic request may block the other traffic requests. To resolve this problem, equal cost multipath (ECMP) routing is used, but then the issue of conflicting objectives arises. Multiple conflicted objectives along with multiple constraints is an optimization problem, considered as NP hard [8], [9], [10]. The multi-objective multiple constrained optimization problem is usually addressed by exact and approximate/ heuristic methodologies. Various approaches of exact and heuristic algorithms have been developed for the multi-objective multiple constrained (MCOP) optimization problem. However, due to computation complexity exact method takes long time to solve it. Therefore, heuristic based approaches become an appealing for solving this problem [10], [11].

III. PROBLEM FORMULATION

This section describes the multi-objective multiple constrained optimization (MCOP) problem in traffic engineering of MPLS/ GMPLS networks. Multiprotocol labelled switched (MPLS) network is dependent on the set of protocol suite used for data carrying in high performance telecommunication networks. In MPLS networks, virtual links are established between source and destination nodes by shortest path known as label switched path (LSP). The path is computed by a routing algorithm (embedded in routing protocols), relaying on labels rather than complex lookups of routing table in each router. MPLS based networks can encapsulates various switching technologies. Generalized MPLS is the extended version of MPLS networks that supports multiple switching technologies such as packet, time, wavelength and fiber. Edge routers are connected to local area networks (LANs) and work as gateway routers between LAN and MPLS/ GMPLS backbone networks. The edge router connected at the sender site is known as the ingress router while at the opposite side, which receives packets, is called the egress router. Both ingress and egress routers are label edge routers (LER). When a traffic request is received at the ingress route(LER), LER computes the shortest path using the routing protocol`s algorithm and establishes LSP towards the egress router. LSP is a virtual path or link developed for the moving user data or traffic from the ingress to egress router within the MPLS/ GMPLS domain. Intermediate routers are known as label switched routers (LSRs), and are only used for routing of traffic based on the label forwarding. This approach avoids the complex look-up into the routing table of IP addresses in each router. In modern telecommunication networks, many network service providers offer MPLS/ GMPLS based routers that can be used for packets, fibre, time and wavelength switching technologies

In this paper, an MPLS/ GMPLS core network is represented as graphs, where links are edges and routers are vertices. A number of traffic requests is received at ingress router/vertices which will use the routing algorithm to compute LSP from a source to destination. LSP is considered as an optimal path for each objective function. The algorithm can be used to compute LSP based on multiple objectives along with their constraints. However, multiple paths will be computed based on multiple objectives, therefore, the proposed technique follows Pareto approach to have feasible number of optimal solutions between the trade-off region of objectives function. Furthermore, the methodology must have improved results with multiple number of iteration in the given application scenarios. The following notations for the graph are presented: $G = (V_{set}, E_{set})$, where V_{set} is for vertices set [v= 1,2, 3,…, V] and E_{set} is set of edges [e = 1, 2, 3,…, E].

IV. PROPOSED METHODOLOGY

Many optimization techniques have been proposed for networks performance optimization to ensure network reliability and effective utilization of network resources. These optimization techniques belong to various classes of optimization such as swarm, artificial intelligence or evolutionary based methodologies [11]. We proposed heuristic algorithm known as Pareto Particle Swarm Optimization with Elitist Learning Strategy (PPSO_ELS) for optimizing the multipath selection in MPLS/ GMPLS networks. This section describes the basic structure of PSO and then proposes the PPSO_ELS version along with its pseudo code.

A. Particle Swarm Optimization technique (PSO)

Particle Swarm Optimization (PSO) is swarm intelligence based optimization technique that was proposed by Kennedy and Eberhart in 1995. The algorithm was inspired by the social behaviour of fish school and bird's flocks. We proposed algorithm based on PSO technique while enhancing its performance with elitist learning strategy. The basic structure of PSO is based on multi-dimensional problem hyperspace, where each particle in the swarm represents a potential solution dependent on two vectors such as i^{th} particle current position x_i and its velocity v_i. Which can be represented as x_i=x_{i1}, x_{i2}, x_{i3}, …., x_{iD} and v_i=v_{i1}, v_{i2}, v_{i3}, …., v_{iD}. where D for all dimensions in the searching space. During searching process, each particle stochastically adjusted by its local searching component ($P_{local.best}$) and social searching component ($G_{social.best}$). The mathematical expression for each particle`s velocity and its position is updated is given as follow [11].

$$v_{i\,(ite+1)} = v_{i\,(ite)} + C_1 r_1 (P_{best\,(ite)} - x_{i\,(ite)}) + C_2 r_2\,(G_{best\,(ite)} - x_{i\,(ite)}) \quad (1)$$

$$x_{i\,(ite+1)} = x_{i\,(ite)} + v_{i\,(ite)} \quad (2)$$

Where ite = iteration/generation during algorithm, i = [1,2,…, N] represents particle`s index and $N$ is the total population size. $P_{local.best}$ = [$P_{best1}$, $P_{best2}$,…, $P_{bestD}$] is the particle`s best position computed according to the fitness function evaluation while $G_{social.best}$ is the global best particle found in the group of $P_{local.best}$. C_1 and C_2 are coefficient constants that represents the cognitive and social behavior of each particle, respectively. r_1 and r_2 are random numbers within the range of [0,1]. $W_{inertia}$ factor is used for local and global searching but has got a significant importance for balancing exploration and exploitation searching of the algorithm [12].

B. Pareto Strengthen Particle Swarm Optimization with Elitist Learning Strategy (PPSO_ELS)

First, it is important to consider a brief about the two objectives functions. Then the preceding algorithms illustrated the sequence of the proposed PPSO_ELS along with explanation.

1) Routing Costs Objective Function

In MPLS / GMPLS core networks, service providers usually assign specific link cost per unit of traffic flow. While total cost incurred for t^{th} traffic on computed path (LSP) is the summation of connected links computed in the network [11].

$$R_t^e = \sum e_r\, i_{t,e} \quad (3)$$

Where, R_t^e is the routing cost over a path for a traffic request or sum of e_r of all connected links in a path. e_r is the routing cost for t^{th} traffic over a link and $i_{t,e}$ = 1 if the link belongs to computed path for the traffic. In the paper, the objective function is to minimize the sum of routing costs of the network, as described by the following expression [11].

$$\sum_{t \epsilon Tset} \sum_{e \epsilon\ Eset} R_t^e\, x_t^P \quad (4)$$

Where, t^{th} traffic is member of all traffics set (T_{set}) and x_t^P represents the traffic flow over computed path.

2) Load Balancing Costs Objective Function

In our topology, the second objective function is related to even distribution of the traffic over various links, which is load balancing. To achieve better load balancing, two parameters are considered such as a link capacity e_c and link utilization e_u, used for load balancing costs. Where e_u defines a total link load and e_c is its capacity [11]

$$L_{bal.} = link\ utilization\ e_u \,/\, link\ capacity\ e_c \quad (5)$$

$L_{bal.}$ explains that each link utilization is associated with its capacity. In our experiments, the objective function is to minimize the sum of the load balancing costs in the network. Mathematically it can be described as follow.

$$\sum_{e \epsilon\ Eset} L_{bal.} \quad (6)$$

Pareto based Particle Swarm Optimization with Elitist Learning Strategy (PPSO_ELS) Algorithm

---

Initialization stage for basic structure of the algorithm:

1. Initialize random positions of particles in searching space (x_i) for $init_1$ and $init_2$.
2. Initialize particles velocities (v_i).
3. Define weight ($W_{inertia}$) for exploration process of local and global particle minima in searching space. Initial value of weight inertia is 0.9.
4. Initialize two constant coefficients C_1 (for cognitive behaviour) and C_2 (for social behaviour). C_1 and C_2 both have constant values of 2.
5. Take initial values of r_1 and r_2. Both r_1 and r_2 have random values within the limit of [0,1].

Pre-Processing stage for defining constraints and multiple objectives:

6. Apply constraints related to routing cost objective function.
7. Initialize load balancing costs objective functions constraints.
8. Discard disconnected links from the matrix. Remove links and routers/ nodes that do not come within the feasible region of routing cost and load balancing functions. Feasible region is the constraints based searching space for possible optimal solutions.
9. After constraints applied, initialize the network as output matrix for both objective functions as a searching space for algorithm.
10. Evaluate fitness function of routing costs (fun_{xir}), as a part of $init_1$.
11. Evaluate the load balancing costs function (fun_{xil}), for $init_1$.
12. Based on estimated routing function costs (fun_{xir}), find each particle`s local best position ($P_{best.routing}$) as a solution in searching space of routing cost objective function.
13. Evaluate particle`s local best position ($P_{best.load}$) in the feasible region of load balancing objective function.
14. Choose the global best particle position ($G_{best.routing}$) according to minimum (fun_{xir}) among group of $P_{best.routing}$ particles.
15. Find the global best particle position for minimum load balancing function (fun_{xil}) among particles best positions $P_{best.load}$
16. Initialize the linear combination objective function for $init_2$.
17. Find the global best position for $init_2$ as $G_{best.current}$

Processing stage as the main body of Proposed Algorithm:

18. Initialize While loop until a condition for termination in number of iterations have met.
19. For every i^{th} particle will be searching for optimal solution within the feasible regions of routing cost, load balancing cost for $init_1$ and Linear combination objective function for $init_2$.
    i) update $W_{inertia}$ from W_{max}= 0.9 to W_{min}= 0.4 in each iteration
    ii) update each x_i^{th} particle`s velocities for both objective functions separately of $init_1$ and linear combination function of $init_2$.
    iii) update each x_i^{th} particle`s positions for $init_1$ and $init_2$.
    iv) evaluate the fitness function as routing costs and load balancing costs function for $init_1$ and linear combination function of $init_2$ according to updated particles positions and velocities.
    v) update particles best positions in searching space for each objective function as P_{best} according to following condition
        (a) if $fun_{routing.previous} > fun_{routing.current}$, if updated fitness function is less than (better than) previous fitness function, update $P_{best.routing}$
        (b) Similarly, if $fun_{load.previous} > fun_{load.current}$, update $P_{best.load}$
    vi) Update $G_{best,routing}$, $G_{best,load}$ of $init_1$ and $G_{best,current}$ for $init_2$.
        (a) $G_{best.routing.previous} > P_i^{th}{}_{best.routing}$ update $G_{best.current.routing}$ for $init_1$
        (b) $G_{best.load.previous} > P_i^{th}{}_{best.load}$ update $G_{best.current.load}$ for $init_1$
        (c) $G_{best.previous} > P_i^{th}{}_{best.current}$ update $G_{best.current.}$ for $init_2$
    vii) Evaluate the exploration process of global minima for each objective function (for $init_1$ and $init_2$)
        (a) if global best position or global minima of routing costs function is repetitive as $G_{best.routing.previous} = G_{best.routing.current}$
        (b) Similarly, if $G_{best.load.previous} = G_{best.load.current}$
        (c) And, if $G_{best.previous} = G_{best.current}$ for $init_2$ objective function
        (d) if any one of these objective functions have true condition met, apply Elitist Learning Strategy (ELS) for that specific objective function take random one dimension (g^{dim}) of current best global position

($G_{best.routing.current}$, $G_{best.load.current}$, $G_{best.current}$)

II. Find the maximum limit of that global best dimension as (X_{max}^{dim})

III. Find the minimum limit of that global best dimension as (X_{max}^{dim})

IV. Compute the new global best minima G^{Dim} within the maximum and minimum dimension of current global minima as

$$(G^{Dim}) = g^{dim} + (X_{max}^{dim} - X_{min}^{dim}) \times Gaussian\ (\mu_{mean}, \sigma^2) \quad (7)$$

unless

G^{Dim} < && ≠ $G_{best.current}$

(G^{Dim} is not minimum and not equal to $G_{best.current}$)

end (ELS loop)

end ($G_{best.current}$ loop)

archiving Pareto points

end (processing stage loop)

---

where $\sigma = \sigma_{max}, (\sigma_{max} - \sigma_{min})$

$\sigma_{max} = 1.0$ and $\sigma_{min} = 0.1$

Gaussian distribution is used with mean value $\mu_{mean} = 0$. σ is time varying standard deviation value. σ_{max} and σ_{min} values are computed on empirical study [13].

We used two versions of initializations for the algorithm. Both versions are based on linear programming (LP) methodology. In the first initialization method ($init_1$), two objective functions (routing costs and load balancing costs) are initialized separately While the next approach is initialization method 2 ($init_2$), where both objective functions are sum together as a fitness function and being assessed during all stages of algorithm. Mathematically linear combination (LP) function can be described as follow.

$$\text{Fitness function} = \alpha \sum_{t \epsilon Tset} \sum_{e \epsilon\ Eset} R_t^e\, x_t^P + (1-\alpha) \sum_{e \epsilon\ Eset} L_{bal} \quad (8)$$

During process stage, whenever the repetitive solution or global minima comes then ELS policy implements within algorithm. In ELS process, one dimension is selected randomly from its present global best solution, denoted as g^{dim}. From the same dimension, find the maximum and minimum limits such as X_{max}^{dim}, X_{min}^{dim} . Compute the G^{Dim} new global best position from the feasible region.

Number of constraints are considered for both objective functions. For load balancing costs function, each link load must be less than its capacity. Furthermore, the difference between link load and its capacity must not be very high, to ensure maximum use of link capacity. Similarly, load balancing must not be greater than or even equal to link loads in the network. With routing costs functions, the constraints are to discard nodes/ routers from the network which do not connected at least two more nodes within the network. Further, the routing cost assigned for each link connected between routers must be higher than the specific value which can shows connectivity of router to the network.

V. Experiments and Evaluation

Experiment settings

The experimental scenarios and the proposed algorithm has been implemented on MATLAB tool. The simulation scenarios consider 20 nodes (GMPLS routers) per each network. The proposed algorithm (PPSO_ELS) were ran for 10 iterations with $W_{inertia}$ starting by 0.9 and decreasing per followed iterations till reach its minimum value 0.4. While, as per the constant coefficient the value were fixed to 2, the random numbers r_1 and r_2 were selected within the range [0,1].

Experiment results

Experiments were based on two scenarios; one case is dependent on Pareto based Particle Swarm Optimization while the latter is based on proposed methodology PPSO_ELS. The purpose was to overcome a problem that were discovered when applying the first algorithm as it will be described next paragraphs.

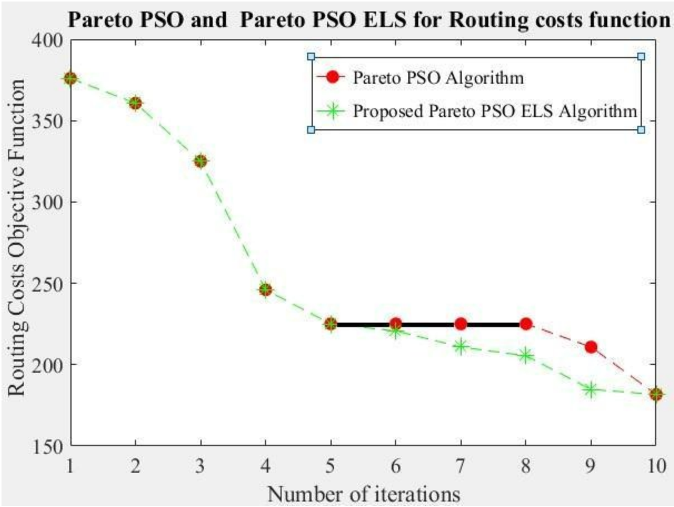

Figure 1: Comparative results between PPSO and proposed PPSO_ELS for Routing costs function with $init_1$

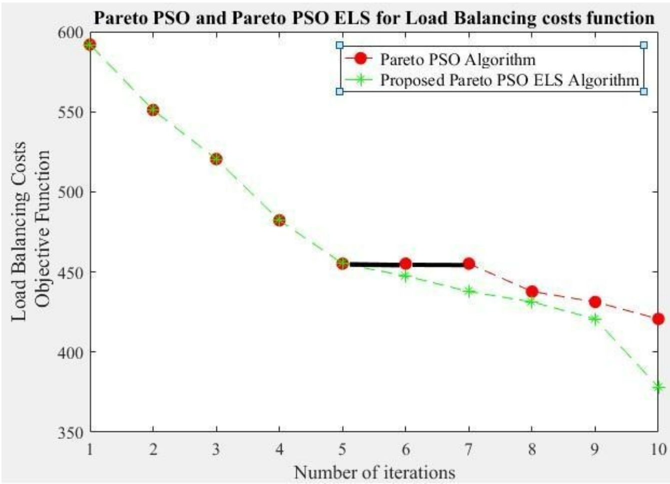

Figure 2: Comparative results between PPSO and proposed PPSO_ELS for Load Balancing costs function with $init_1$

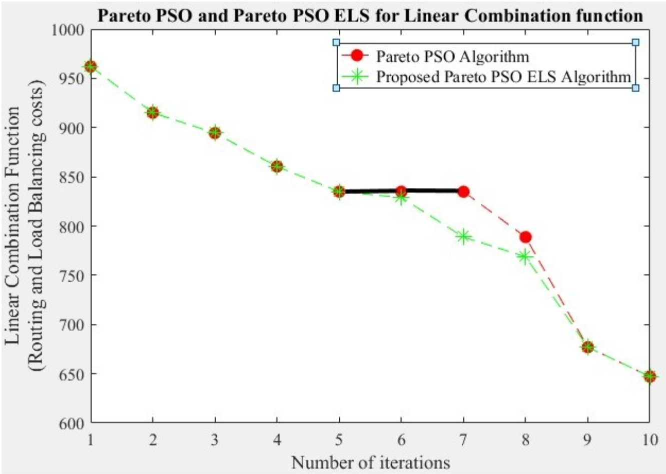

Figure 3: Comparative results between PPSO and proposed PPSO_ELS for Linear combination function with $init_2$

The experiments considered firstly, a comparison between the performance of PPSO against the PPSO_ELS when the objective functions for the routing costs and for the load balancing costs ($init_1$ and $init_2$) is to minimize both.

Fig. 1, 2 and 3 are the obtained comparative results between both algorithms. The figures depict that there are redundant results per some consecutive iterations for the PPSO. The redundant solutions during iterations are highlighted within the figures with thick lines. This stuck with the results attached with the PPSO is related to the behaviour of the PSO algorithm, where the objective algorithm trapped within local minima within the search space (lack of the exploration). On contrary, the PPSO_ELS had overcame the exploration problem and for each iteration provide distinct value for both routing cost and load balance cost objects. As for more clarifications consider figure 1, the routing costs functions gives the same G_{best} values during 6^{th}, 7^{th} and 8^{th} iteration. For more analysis, we had combined both figures using a linear combination functions (Fig. 3). It was discovered that during 6^{th} and 7^{th} iterations, the problem of exploration of the PPSO is clearly exist.

One of the findings of that showing the advancement of the PPSO_ELS is the achievement of more minimization over the PPSO with the iteration's scaling up. The reasons may be the searching for a new global best position from the maximum and minimum current feasible region.

VI. CONCLUSION

The paper describes the traffic engineering in MPLS/ GMPLS networks and the proposed multi-objective multiple constrained optimization (MCOP). Then the paper proposes the application of Pareto based particle swarm optimization versions to find the optimal paths between the networks routers. The Pareto based Particle Swarm Optimization with ELS strategy (PPSO_ELS) had proven its superiority over the original Pareto based Particle Swarm Optimization version in terms of both expanding the exploration with the search space and finding a much minimized G_{best} solutions out of both multi-objective functions (route cost, and load balance cost). As a future work, further enhancement to be evaluated along with other adaptive approaches of PSO algorithm.

VII. RFERENCES

[1] Jabbari, B., Oki, E., Vasseur, J. and Chong, S. (2007). Routing, path computation and traffic engineering in future Internet. *Journal of Communications and Networks*, 9(4), pp.345-349.

[2] Awduche, D. (1999). MPLS and traffic engineering in IP networks. *IEEE Communications Magazine*, 37(12), pp.42-47.

[3] Myoung Lee, G. and Kyun Choi, J. (2006). Multipath Routing for Traffic Engineering with Flow Classification in GMPLS Network. In: *Communications, 2006. APCC '06. Asia-Pacific Conference on*. Busan: IEEE.

[4] Hussein, H. (2015). Multipath Bandwidth Capacity Allocation and MPLS Internet Traffic Engineering. *Journal of Advances in Computer Networks*, 3(3), pp.239-242.

[5] Coello, C., Pulido, G. and Montes, E. (2005). Current and Future Research Trends in Evolutionary Multi-Objective Optimization. In: M. Grana, R. J. Duro, A. d'Anjou and P. P. Wang, ed., *Information Processing with Evolutionary Algorithms*, 1st ed. London: Springer-Verlag London, pp.213-231.

[6] Masood, M., Abuhelala, M. and Glesk, I. (2016). Detailed Analysis of Routing Protocols with Different Network Limitations. In: *Proc. SPIE 10142, 20th Slovak-Czech-Polish Optical Conference on Wave and Quantum Aspects of Contemporary Optics, 101421T*. SPIE.

[7] M. El-Alfy, E., Z. Selim, S. and N. Mujahid, S. (2007). Solving the minimum-cost constrained multipath routing with load balancing in MPLS networks using an evolutionary method. In: *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*. Singapore: IEEE.

[8] Young Cho, H., Yong Lee, J. and Chul Kim, B. (2003). Multi-path constraint-based routing algorithms for MPLS traffic engineering. In: *Communications, 2003. ICC '03. IEEE International Conference on*. Anchorage: IEEE.

[9] SC., E. (2004). *Traffic engineering in MPLS networks with multi-objective: modelling and optimization*. PhD. RWTH Aachen, Germany.

[10] Garroppo, R., Giordano, S. and Tavanti, L. (2010). A survey on multi-constrained optimal path computation: Exact and approximate algorithms. *Computer Networks*, 54(17), pp.3081-3107.

[11] El-Alfy, E., Mujahid, S. and Selim, S. (2013). A Pareto-based hybrid multi-objective evolutionary approach for constrained multipath traffic engineering optimization in MPLS/GMPLS networks. *Journal of Network and Computer Applications*, 36(4), pp.1196-1207.

[12] Lim, W. and Mat Isa, N. (2014). An adaptive two-layer particle swarm optimization with elitist learning strategy. *Information Sciences*, 273, pp.49-72.

[13] Zhan Z., Zhang J. (2008) Adaptive Particle Swarm Optimization. In: Dorigo M., Birattari M., Blum C., Clerc M., Stützle T., Winfield A.F.T. (eds) Ant Colony Optimization

and Swarm Intelligence. ANTS 2008. Lecture Notes in
Computer Science, vol 5217. Springer, Berlin, Heidelberg.