Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

# Congestion Management in MPLS Network Based on Hybrid Particle Swarm Algorithm

***Samar Taha Yousif***

*Baghdad College of Economic Sciences University, Baghdad, Iraq. Email: samaral_ani@baghdadcollege.edu.iq*

ARTICLE INFO

*Article history:*
Received: 20 /03/2021
Rrevised form: 26 /03/2021
Accepted : 07 /04/2021
Available online: 08 /04/2021

*Keywords:*

Network congestion; Multiprotocol Label Switching (MPLS); Particle Swarm Algorithm; Tabu search; Mathematical model.

ABSTRACT

To minimize the network congestion problem, a multi-constraint optimization mathematical model is established and a hybrid particle swarm algorithm based on the Tabu search algorithm is proposed. The algorithm integrates Tabu search into particle swarm algorithm, which can prevent the algorithm from falling into the local optimal solution, and gives a better initial individual for Tabu search, so it can speed up the algorithm convergence and improve the quality of the solution. The simulation was done using Matlab and the results show that the optimization algorithm reduces the end-to-end delay and packet loss rate, thereby avoiding network congestion and optimizing the utilization of network resources.

**MSC. 41A25; 41A35; 41A36**.

**DOI: : https://doi.org/10.29304/jqcm.2021.13.1.786**

## 1. Introduction

When too many messages exist in the network, it will cause network performance degradation, that is, congestion occurs [1]. Congestion mainly occurs when network resources are insufficient to meet the needs of service flows, or when the distribution of network service flows is unbalanced, causing an increase in the packet loss rate of the link, increase in the delay between nodes, increase network jitter, and reduce network throughput [2], Therefore, the use of effective methods to achieve congestion control has become a hot issue in current network research. At present, many algorithms have been proposed, such as RED, REM [3], Drop Tail Queue Management, DRED [4], PI controller [5], PID controller [6], and so on. But they all have some shortcomings. For example, RED and Drop Tail algorithms are greatly affected by subjective factors, have poor stability, and are not robust. They are prone to error data or even lost during the control process; PI controller, REM, DRED, and PID, the dynamic performance of these algorithms such as the controller is not good, and the queue fluctuates greatly. Because the intelligent bionic optimization algorithm has the advantages of good parallelism, good adaptive ability, and strong robustness, it is especially suitable for network congestion control [7]. On the other hand, Multi-Protocol Label Switching (MPLS) is a transmission protocol for data routing using tags [8], [9]. The explicit routing technology of MPLS allows the source to fully control the transmission path of the data in the network, and the label switching path (LSP) meeting certain constraints can be pre-established before the packet is transmitted, so it can be integrated. Considering the global

*Corresponding author: Samar Taha Yousif

Email addresses: samaral_ani@baghdadcollege.edu.iq

Communicated by : Dr. Rana Jumaa Surayh aljanabi.

service flow requirements, network topology, and link characteristics, establishing an optimized path to achieve the purpose of balanced network flow distribution is very conducive to achieving the goal of traffic engineering which is considered to be one of the strengths of the MPLS technology, where it is used to optimize the use of network resources to avoid congestion [10]. A hybrid SDN approach was presented in [11] for the management of MPLS traffic engineering. quality of service capabilities has been applied to MPLS as specified in [12]. Different tools were developed in [13] to help operators in increasing the visibility of the MPLS network, the proposed tools are concerned with the observation of capacity and bandwidth usage, to avoid MPLS network congestion. The mathematical description of four types of MPLS network traffic engineering problems is given in [14], [15], and it is pointed out that these problems are all NP-hard problems, but no specific solving algorithm is given. [16] gives a heuristic method for establishing an explicit restricted LSP that conforms to the load balancing principle in the MPLS network, [17] proposes an algorithm to monitor delay, jitter, packet loss, and Maximum Transmission Unit (MTU) values for the MPLS network. The algorithms in literature [18] require the use of mathematical tools to solve mathematical programming problems, which are difficult to integrate with the actual application environment and are limited by the support capabilities of mathematical tools. In order to avoid congestion in the MPLS network and improve network performance, this paper combines particle swarm algorithm and Tabu search algorithm on the basis of existing optimization algorithms and proposes a hybrid particle swarm algorithm. Simulation results show that this algorithm can be implemented reliably and effectively.

## 1. MPLS routing optimization mathematical model

In order to be able to analyze various network congestion control strategies and meet the user's network service quality requirements, an MPLS explicit routing global optimization mathematical model needs to be established to minimizes network congestion. The optimization principle is: ensuring that parameters such as bandwidth and delay meet the requirements under the premise of reducing the network resources used as much as possible and evenly distribute the network load everywhere. This is an NP-hard problem. The optimal solution cannot be obtained, and only the sub-optimal solution can be found. Therefore, the expression of the mathematical model constructed in this paper as follow [19]:

$$M = K_1 \times A\,(P) + K_2 \times \sigma^2 \quad (1)$$

In Equation (1), $A\,(p)$ is called the resource consumption function, which reflects the number of network resources occupied by a certain service volume in the selected path $p$, which can be expressed as:

$$A\,(P) = h\,(p) \times B \times d\,(p) \quad (2)$$

In Equation (2), $h(p)$ is the number of hops of the selected path $p$; $B$ is the requested bandwidth of the traffic, $d(p)$ is the delay of path $p$. Obviously, if the path has fewer $h(p)$ and $d(p)$, it will occupy fewer network resources.

$\sigma^2$ is the variance of the link utilization rate on a certain path $p$, which reflects the degree of balanced load distribution on the path, and can be expressed as:

$$\sigma^2 = \sum_{m=1}^{n} \sum_{n=1}^{n} ( E_{m,n} - \bar{E} )^2 \rho_{m,n} \quad (3)$$

In Equation (3), $E_{m,n}$ is the link utilization rate from node $m$ to node $n$; $\bar{E}$ is the mean value of $E_{m,n}$; if the link from $m \rightarrow n$ belongs to the selected path, then $\rho_{m,n}$ is 1, otherwise, $\rho_{m,n}$ is 0.

$k_1$ and $k_2$ are two positive real numbers. If the value is appropriate, the two parts of the objective function $M$ can be basically equal.

The particle swarm algorithm (PS) has strong global search ability, but it has poor mountain climbing ability and is prone to prematureness. Tabu search (TS) has strong mountain climbing ability, but it requires experience to select initial parameters, and its robustness is poor. In order to make up for the shortcomings of these two algorithms and improve the optimization efficiency, this paper incorporates the idea of TS into PS and proposes a hybrid particle swarm algorithm.

## 2. Implementation of hybrid particle swarm algorithm

The particle swarm algorithm was first proposed by Dr. Kennedy and Dr. Eberhart in the United States in 1995 [19]. The algorithm regards all individuals in a population as an object and performs global optimization search through, initialization, fitness function determination, speed and location updating, and termination condition. This algorithm is simple, easy to implement, robust, and is widely used in various fields.

The flow chart of the particle swarm algorithm is shown in Fig. 1, and the programming flowchart for the hybrid particle swarm algorithm is shown in Fig. 2.

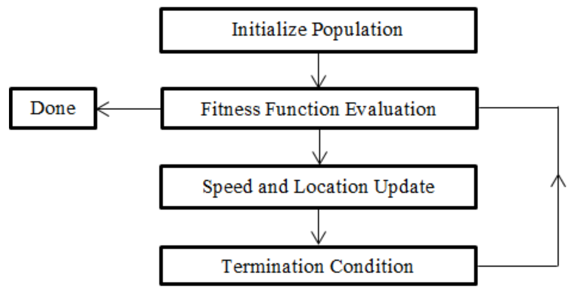


**Figure 1 - particle swarm algorithm flowchart**

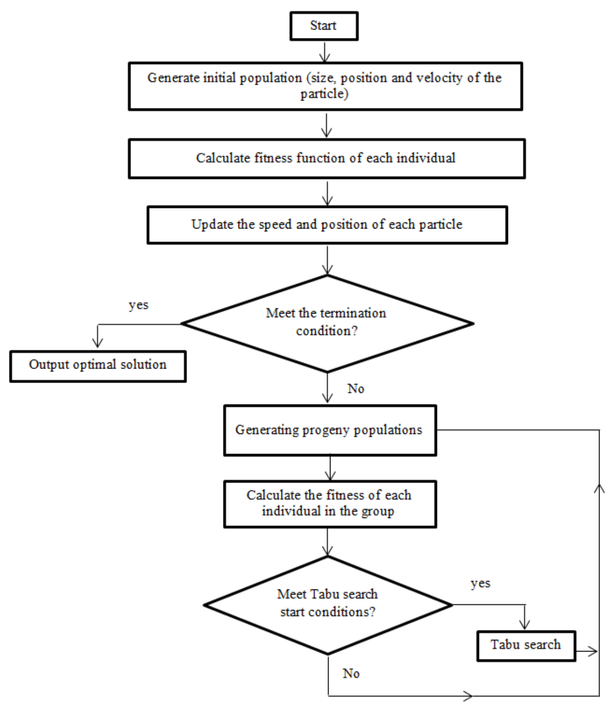


**Figure 2 - flow chart for hybrid particle swarm algorithm**

### *3.1. Initial population*

The initialization of position and speed was generated randomly as a 50 x1 data matrix from 0 to 1. The position constraint here can also be understood as the position limit, and the speed limit is to ensure that the particle step length does not exceed the limit. Generally, the speed limit is set to [-1, 1].

### *3.2. Fitness Function*

The fitness function can reflect the pros and cons of different individuals in the same group and is the basis for selection operations, usually in the form of the maximum value. Since Equation (1) of the optimization mathematical model is a minimum problem, this paper uses the method of reciprocal of it to obtain the fitness function, as follow:

$$Fitness = \frac{1}{M} = \frac{1}{K1 \times A(P) + K2 \times \sigma 2} \tag{4}$$

### *3.3. Speed and location update*

Speed and position update is the core of particle swarm algorithm, its principle expression, and update method as follow:

$$V_{nt} = W.\, V_{nt} + L_1 .\, R_1 .\, (P_{nt} - X_{nt}) + L_2 .\, r_2 .\, (P_{gd} - X_{nt}) \tag{5}$$

$$X_{nt} = X_{nt} + V_{nt} \tag{6}$$

Where: $n$ represents the particle number; $t$ represents the time, which is reflected in the number of iterations; $w$ is the inertial weight, which is generally set at about 0.4; $L_1$ and $L_2$ are learning factors, generally 2; $P_{nt}$ represents the most experienced particle $n$ best position; $P_{gd}$ represents the optimal position of the global particle; $r_1$, $r_2$ are random values between 0 and 1.
Each search needs to compare the current fitness and optimal solution with the historical record value. If it exceeds the historical optimal value, update the historical optimal position and optimal solution of the individual and the population.

### *3.4. Termination condition*
Two terminating conditions may be selected; the first is the maximum algebraic; the second is the adjacent deviation between generations within a specified range. In this paper; the maximum algebraic termination condition were used.

### *3.5. Tabu Search*
After speed and position updating and termination condition operations, the particle swarm optimization algorithm produces a better initial individual for the Tabu search.
Tabu search starts from an initial feasible solution, tests a series of specific search directions (movements), and selects the move that increases the value of the specific objective function. In order to avoid falling into the local optimal solution, Tabu search records the search process information that has been experienced to guide the next search direction.

### *3.6. Tabu List*
The Tabu list records and selects the optimization process that has been performed, which can prevent the Tabu search from falling into the local optimum. The dynamic setting method is adopted here, that is, each time the loop ends, the “move” in the opposite direction that enters the Tabu list first is replaced with the current “move” in the opposite direction, and the Tabu length of the “move” is defined, finally the other opposite directions are “move”. The Tabu length is reduced by 1.

### *3.7. Amnesty Criterion*
In order not to miss the "movement" that produces the optimal solution as much as possible, Tabu search also adopts the "amnesty criterion" strategy and keeps the record of the best individual to replace the worst individual in the next generation.

### *3.8. Termination Conditions*
There are two loops in the hybrid optimization algorithm: inner loop and outer loop. The termination condition of the inner loop is the maximum allowable number of iterations of the Tabu search; while there are two termination conditions of the outer loop: the maximum allowable number of iterations of the particle swarm algorithm and the optimization criterion, where the optimization criterion is the optimal individual fitness value before and after the Tabu search, whether the change is not greater than a given constant.

## 4. Simulation research
### *4.1 Network parameters setting*
In order to verify the effectiveness of the hybrid particle swarm algorithm, the MPLS network shown in Fig. 3 is simulated.

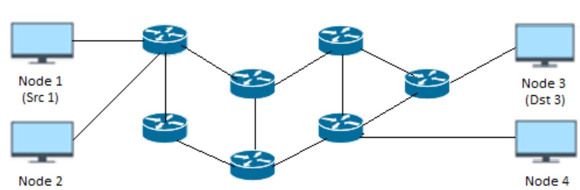


**Figure 3 - MPLS network**

Each link in the network sets as [bandwidth delay] = [1Mb 10ms]. Assuming that there are two identical traffic generators T0 and T1 on source nodes 1 and 2, and 500 Byte data packets are generated every 5 ms and flow to destination node 3. The T0 packet starts sending at 0.5s and stops at 4.5s; the T1 packet starts sending at 1.0s and stops at 4.0s. Since the requested bandwidth of node 1 and node 2 is 0.8Mb, while the bandwidth of the link between nodes (2-3) is only 1Mb, when two sets of the business flow through the link (2-3) together, there will be competition for network resources causing network congestion. The hybrid particle swarm algorithm proposed in this paper was used to optimize the bandwidth and delay parameters of the link (2-3).

### 4.2 Result analysis

#### 4.2.1 Comparison of network performance

Fig. 4 and Fig. 5 show the end-to-end delay time of T0 and T1 packets before and after optimization. It can be seen from the simulation results that the use of a hybrid particle swarm algorithm to optimize the network parameters reduces the end-to-end delay time of the packet and avoids network congestion.

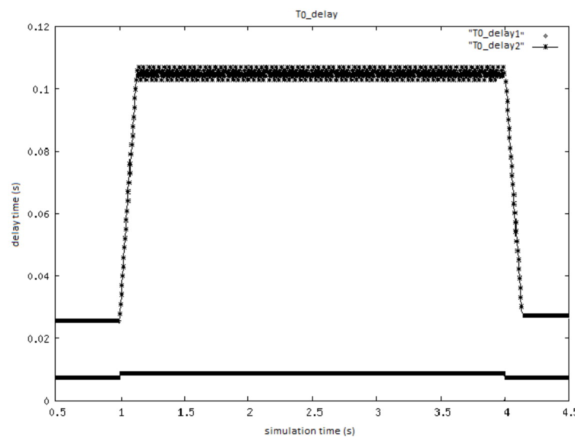


**Figure 4 - end-to-end delay of T0 packets**

Where T0_delay1 represents end-to-end delay time before optimization and T0_delay2 represents end-to-end delay time after optimization.

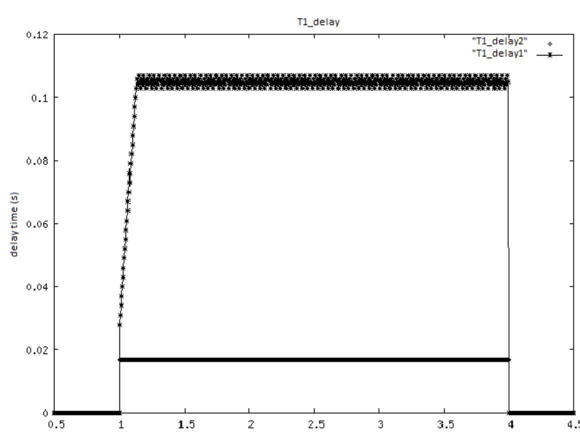


**Figure 5 - end-to-end delay of T1 packets**

Where T1_delay1 represents end-to-end delay time before optimization and T1_delay2 represents end-to-end delay time after optimization.

#### 4.2.2 Average optimization results

In order to verify the optimization performance of the hybrid particle swarm algorithm in this paper, the particle swarm algorithm is used to optimize the same network, and the convergence characteristics of the two algorithms are compared. The results are shown in Fig. 6. It can be seen from Figure 5 that the hybrid particle swarm algorithm generated the optimal objective function value around the 70th generation, while the particle swarm algorithm did not generate the optimal objective function value until the 80th generation; moreover, the objective function value optimized by the hybrid particle swarm algorithm was less than the objective function value after particle swarm optimization. This shows that the hybrid particle swarm algorithm has a faster convergence rate and good global optimization performance.

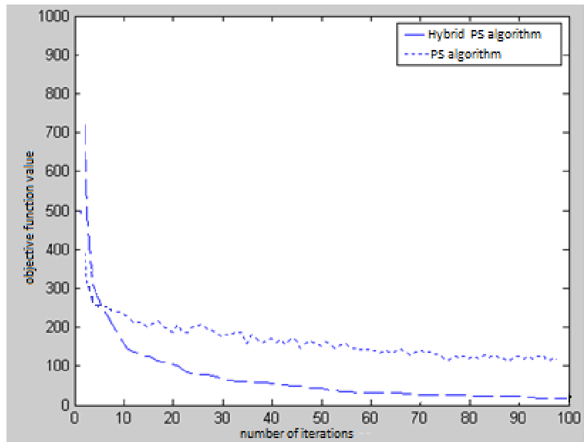


**Figure 6 - Comparison of average convergence properties**

## 5. Conclusions

This paper proposes a hybrid particle swarm algorithm to solve the congestion problem of the mpls network where a mathematical model is established to determine the optimization objective function on the basis of meeting performance indicators such as delay and packet loss rate. this algorithm combines the characteristics of particle swarm and tabu search to provide a better initial individual for the tabu search, reduces the number of calls, and speeds up the convergence of the algorithm. the simulation results reflect that a hybrid particle swarm algorithm can reduce the utilization of network resources and balance the distributed load while reducing the packet loss rate and reducing the end-to-end delay time to avoid network congestion and provide an effective way for further implementation of congestion control.

## References


[1] S. Murtaza, S. Anjum, and E. Sciences, "An Advanced Practice for Congestion Control, Routing and Security Issues in Multi-Protocol Label Switching ( Mpls )," vol. 01, no. 01, 2020.

[2] S. T. Yousif and Z. A. A. Al-Haboobi, "An Improved Algorithm for Congestion Management in Network Based on Jitter and Time to Live Mechanisms," *Al-Nahrain J. Eng. Sci.*, vol. 23, no. 4, pp. 352–356, 2020, DOI: 10.29194/njes.23040352.

[3] A. K. Sharma, "A Survey on Active Queue Management Techniques," *Int. J. Eng. Comput. Sci.*, no. December, pp. 6–11, 2016, DOI: 10.18535/ijecs/v5i11.47.

[4] S. Rastogi and H. Zaheer, "Comparative analysis of queuing mechanisms: Droptail, RED and NLRED," *Soc. Netw. Anal. Min.*, vol. 6, no. 1, 2016, DOI: 10.1007/s13278-016-0382-5.

[5] S. Chebli and A. El Akkary, "PI Stabilization for Congestion Control of AQM Routers with Tuning Parameter Optimization," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 4, no. 1, p. 52, 2016, doi: 10.9781/ijimai.2016.4110.

[6] Y. Shatnawi and M. Quwaider, "Congestion Control in ATM networks using PID Controller with Immune Algorithm," *2019 10th Int. Conf. Inf. Commun. Syst. ICICS 2019*, pp. 19–24, 2019, DOI: 10.1109/IACS.2019.8809150.

[7] Z. Zhang, J. Ye, and S. Cheng, "Research on Traffic Congestion Resolution Mechanism based on Genetic Algorithm and Multi-Agent," vol. 87, no. Icmeit, pp. 473–483, 2019, DOI: 10.2991/icmeit-19.2019.77.

[8] F. Bensalah, N. El kamoun, M. El Hamzaoui, and A. Bahnasse, "Behavior study of SIP on IP multimedia subsystem architecture MPLS as transport layer," *Int. J. Inf. Technol.*, vol. 10, no. 2, pp. 113–121, 2018, DOI: 10.1007/s41870-018-0106-3.

[9] F. Bensalah, N. El Kamoun, and A. Bahnasse, "Evaluation of tunnel layer impact on VOIP performances ( IP – MPLS – MPLS VPN – MPLS VPN IPsec )," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 17, no. 3, pp. 87–92, 2017.

[10] F. Bensalah and N. El Kamoun, "Novel software-defined network approach of flexible network adaptive for VPN MPLS traffic engineering," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 4, pp. 280–284, 2019, DOI: 10.14569/ijacsa.2019.0100433.

[11] A. Bahnasse, F. E. Louhab, H. Ait Oulahyane, M. Talea, and A. Bakali, "Novel SDN architecture for smart MPLS Traffic Engineering-DiffServ Aware management," *Futur. Gener. Comput. Syst.*, vol. 87, pp. 115–126, 2018, DOI: 10.1016/j.future.2018.04.066.

[12] P. Biradar and P. T L, "Design of Traffic Engineered Multi-Protocol Label Switching-Transport Profile (Mpls-Tp) for the Enhancement of Quality of Service," *Int. J. Res. -GRANTHAALAYAH*, vol. 5, no. 4RACEEE, pp. 58–67, 2017, doi: 10.29121/granthaalayah.v5.i4raceee.2017.3323.

[13] M. Arafah, H. Alhindi, H. Mathkour, M. Faisal, and M. D. Lytras, "Smart systems approach for development of explicit congestion marking and traffic engineering model for Diffserv/MPLS networks," *J. Ambient Intell. Humaniz. Comput.*, vol. 0, no. 0, pp. 1–11, 2018, doi: 10.1007/s12652-018-0739-x.

[14] Z. Hu, *Advances in Computer Science for Engineering and Education II*. 2020.

[15] H. Huhphqnr, L. V Wkdw, W. K. H. Rswlpl, D. Sureohp, R. I. Ordg, and E. Dw, "( Qjlqhhulqj ) Dvw 5H5Rxwh Iru / Lqn 1Rgh Dqg," vol. 6,

pp. 5–9.

[16] S. A. Magsi, R. U. Amin, A. S. Malik, B. K. Kasi, and M. K. Kasi, "Mitigating Network Congestion through MPLS-TE Based Load Balancing Using Alternate Optimal Path," vol. 9, no. 2, pp. 39–45, 2016.

[17] I. Ramadaa, J. Ozegovic, and V. Pekic, "Network performance monitoring within MPLS traffic engineering enabled networks," *2015 23rd Int. Conf. Software, Telecommun. Comput. Networks, SoftCOM 2015*, pp. 315–319, 2015, doi: 10.1109/SOFTCOM.2015.7314107.

[18] C. M. Restrepo, O. J. Salcedo-Parra, and J. M. Sánchez-Céspedes, "Traffic model for the interconnection of networks and operators using MPLS-TE," *Rev. Fac. Ing.*, vol. 26, no. 44, p. 85, 2017, doi: 10.19053/01211129.v26.n44.2017.5774.

[19] S. Talukder, "2nd International Conference on Advanced Machine Learning Technologies and Applications, AMLTA 2014," *Commun. Comput. Inf. Sci.*, vol. 488, no. February, pp. 1–541, 2014.

[20] Kh. Alzubaidi, W., & H. Shaker, S. (2018). Methods of Secure Routing Protocol in Wireless Sensor Networks. Journal of Al-Qadisiyah for Computer Science and Mathematics, 10(3), Comp Page 38 - 55. https://doi.org/10.29304/jqcm.2018.10.3.437

[21] Al-Shammary, D. (2019). Jaccard Coefficients based Clustering of XML Web Messages for Network Traffic Aggregation. Journal of Al-Qadisiyah for Computer Science and Mathematics, 11(2), comp 82-91. https://doi.org/10.29304/jqcm.2019.11.2.592

[22] Abdul-Hussain, R., & Awad, O. (2019). Studying The Effect of Sampling Time and Network Load on Wireless Networked Control Systems. Journal of Al-Qadisiyah for Computer Science and Mathematics, 11(3), Comp Page 31 -42 . https://doi.org/10.29304/jqcm.2019.11.3.604