# Network Slice Resource Adaptive Dynamic Adjustment Method Based on Bat Algorithm

**Hua Qu[a], Zhelin Duan[b], and Bin Zhang[c]**

School of Software Engineering, Xi'an Jiaotong University Xi'an, Shaanxi, China.

[a]qh@mail.xjtu.edu.cn, [b]duanzhelin@163.com, [c]lbjzhangbin@163.com

**Keywords:** Network Slice; Resource Allocation; Bat Algorithm; Dynamic.

**Abstract:** With the continuous improvement of material life and spiritual life, people continue to innovate in business and put forward various demands for network. Network slice can be used to construct virtual logical network according to the needs of different application scenarios, such as network rate, delay and reliability, etc., according to the needs of business for network function and security, etc., which is one of the key technologies of 5G. Aiming at the QoS demand of network slice diversification, a resource allocation method of network slice based on discrete binary particle swarm optimization algorithm is proposed. This paper proposes an adaptive dynamic resource adjustment strategy based on bat algorithm. Faced with huge resource allocation and its characteristics of dynamic, it is no longer enough to balance the advantages and disadvantages of resource allocation strategy from a single aspect. Aiming at these problems, from two aspects of the user and resource provider, introduces the bat algorithm in resource allocation strategy, set up the minimum resource utilization and resources adjustment cost as the double constraints of resources adjustment quantity decision model, the simulation results show that the bat algorithm can effectively solve the network section of dynamic allocation of resources to improve the utilization ratio of resources.

## 1. Introduction

As an emerging industry, there are still many problems in SDN based network slicing, among which resource allocation is one of them. Dynamic resource allocation is a NP complete problem and heuristic algorithm is one of the important directions in this field. In literature [1], the traditional genetic algorithm is introduced into the resource allocation model, and the quality of service is introduced to improve the fitness function, but it is easy to fall into the local optimal problem. In literature [2], the improved particle swarm optimization is applied to the resource allocation model, and the idea of multi-dynamic multi-group writing and variable particle flying is introduced to coordinate the global search and local search, which improves the execution efficiency of resources. Literature [3] proposed a resource allocation model based on multi-objective integrated ant colony optimization, introduced the concept of entropy into the model, and measured the uncertainty of resources.

## 2. Ease of Use

### 2.1 Theoretical Significance

The dynamic adjustment strategy of slice resources based on SDN has the significance of optimizing and improving the allocation of virtual network resources, which can help to improve the utilization rate of network resources and avoid overloading of traffic.

### 2.2 Application Value

The dynamic adjustment strategy of SDN-based sharding resources can have a positive impact on the operator's network revenue.

## 3. System Model

In fact, the resource mapping problem of network slice can be reduced to a np-hard constrained optimization problem. This paper proposes a adaptive dynamic resource adjustment strategy based on the bat algorithm research , starting from he requirements of the allocation of resources, through in-depth study the bat algorithm, first established in resource utilization and resources adjustment minimum cost as the double constraints of resources quantity decision model, and then the bat algorithm used in this model, the simulation results show that the bat algorithm can effectively solve the network section of dynamic allocation of resources to improve the utilization ratio of resources.

We assume that the total bandwidth allocated to the i th tenant on the network slice is $A_i$ , the rental price of the unit bandwidth from the network infrastructure provider is $p_i$ and the revenue from the unit bandwidth is $g_i$ .

Consider case 1: when the number of users of tenant I on the network slice suddenly increases, and the actual traffic generated gradually approaches the amount of resources allocated, the actual benefit of tenant I on the network slice is shown in formula (1).

$$k_i = u_i g_i - A_i p_i \tag{1}$$

Where $U_i$ is the actual traffic of tenant i on the network slice, is the bandwidth value allocated by tenant i on the network slice $A_i$ and $\sigma$ is the resource utilization of tenant i on the network slice.

$$U_i = \sigma A_i \tag{2}$$

In order not to cause network congestion and affect user experience, the bandwidth value allocated by tenant i on the network slice needs to be increased. Benefits after increasing the allocated bandwidth value:

$$K_i^x = U_i g_i - A_i^x p_i \tag{3}$$

Where $A_i^x$ is the bandwidth value allocated after the adjustment by tenant i on the network slice, and is the resource utilization ratio of tenant i on the network slice.

$$U_i = \rho A_i^x \tag{4}$$

The difference between adjusted and adjusted earnings is:

$$\Delta K = A_i p_i (1 - \frac{\sigma}{\rho}) \tag{5}$$

Formula (5) can be used to analyze: $\rho$ is less than $\sigma$ , so ΔK is less than 0. It can be concluded that the higher the utilization, the higher the benefit.

Consider case 2: when the number of users of tenant i on the network slice suddenly decreases and the actual traffic generated is much less than the amount of resources allocated, it is necessary to reduce the allocated resources of tenant i on the network slice, and the formula (5) is also obtained, but $\rho$ is greater than $\sigma$ , so ΔK is greater than 0.

Assuming that there are m tenants in the network slice, the total number of the whole network slice is $A_{all}$ , and $\mu_i$ is the resource utilization ratio of tenant i in the network slice, it can be obtained:

$$\mu_i = \frac{\sum_{j=1}^{n} a_j}{A_i^x} \tag{6}$$

Where $a_j$ is the actual traffic of the user of tenant i on the network slice, and there are n users in total. Adjusted allocation of resources for tenant i on the network slice. Objective function 1 is to maximize resource utilization adjusted by tenants on all network slices.

$$\overline{\mu} = \frac{1}{m}\sum_{i=1}^{m}\mu_i \tag{7}$$

However, there is an overhead associated with adjusting network resources, such as the latency associated with updating tenants on a network slice. Assume that $c_i$ is the overhead of bandwidth adjustment for tenant i units on the network slice. The second objective function can be obtained, even if the bandwidth adjustment overhead is minimized:

$$C = \sum_{i=1}^{m} c_i \left| A_i^x - A_i \right| \tag{8}$$

There are two objective functions above, one is to maximize resource utilization, the other is to minimize resource adjustment cost, and the goal can be normalized. The final goal is to solve the redistribution value of each tenant on the network slice, so as to maximize the result value of the normalization function:

$$result = \frac{\frac{1}{m}\sum_{i=1}^{m}\frac{\sum_{j=1}^{n} a_j}{A_i^x}}{\sum_{i=1}^{m} c_i \left| A_i^x - A_i \right|} \tag{9}$$

The first line in the constraint condition indicates that the bandwidth value allocated by all tenants on the network slice is less than or equal to all bandwidth values of the network slice. The second line indicates that the bandwidth value allocated by each tenant on the network slice is less than all the bandwidth values in the network slice. The third line indicates that resource utilization per tenant on the network slice is greater than the set minimum resource utilization and less than the set maximum resource utilization.

## 4. Resource allocation based on bat algorithm

### 4.1 Basic The Bat Algorithm

Bat algorithm (BA) is a new heuristic optimization algorithm proposed by Yang in 2010, inspired by the relevance between the echolocation behavior pattern of bats and the function of the optimization target. Now, it has been applied to various optimization problems by many scholars. Similar to many existing optimization algorithms, bat algorithm is also a kind of random optimization algorithm based on species group. Bat individual is the basic unit of bat algorithm, which gives specific meaning to specific problems. The optimization process of BA is a dynamic evolution process, from the random population at the beginning of disorder to the search process of constantly updating the current optimal solution (local optimal solution) and gradually ordering to find the dynamic evolution of the global optimal solution. As the basic unit of search, each bat will make flight exploration for the specific search space and continuously evolve according to the change of fitness. Each bat individually can follow the current optimal bat through the solution space by adjusting the frequency, sound intensity, and pulse emittance.

#### 4.1.1 Bat Speed Updates And Location Updates

If the search space is d dimension, the position of the i th bat at time t is $x_i^t$, and the velocity is $v_i^t$ then the position at time t+1 is $x_i^{t+1}$ and the velocity $t_i^{t+1}$ are more modern:

$$f_i = f_{min} + (f_{max} - f_{min})\beta \tag{10}$$

$$v_i^{t+1} = v_i^t + (x_i^t - x_*) f_i \tag{11}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{12}$$

$f_i$, $f_{max}$ and $f_{min}$ espectively represent the frequency of sound wave emitted by the ith bat at the current moment, the minimum value and the maximum value of sound wave. $\beta \in [0,1]$ is a randomly generated number; x * represents the current global optimal solution.

Once a solution (bat) is selected from the existing optimal solution set, the new location of the solution can be generated by formula (4) :

$$x_{new}(*\mathrm{i}) = x_{old} + \varepsilon A^{t} \tag{13}$$

This can be understood as a local search in which a new solution is generated in the region adjacent to the selected solution. $x_{old}$ represents a solution selected randomly from the current optimal solution set, $A^{t}$ represents the mean loudness of the current generation bat population, and ε is the d-dimensional random vector belonging to [-1, 1].

### 4.1.2 Loudness and Pulse Rate

The loudness A(i) and emission rate R(i) of the pulse are constantly updated as the iteration progresses. In general, when approaching the optimal solution, the loudness will gradually decrease and the pulse impulse emission rate will gradually increase. When A(i) = 0, it indicates that bat I just found an optimal solution and no longer sends out detection signals. Equations (5) and (6) are the update equations of pulse loudness and emission rate.

$$A_i^{t+1} = \alpha A_i^{t} \tag{14}$$

$$R_i^{t+1} = R_i^{0} \times \left[1\text{-}\exp(\text{-}\gamma \mathrm{t})\right] \tag{15}$$

$0 < \alpha < 1$ ,$\gamma > 0$, are often.

In bat algorithm, the increase coefficient of pulse frequency γ and attenuation coefficient of pulse α sound intensity have important influence on the performance of algorithm. The change mode of bat individual's current spatial state is determined according to the true or false expression, β > R(i), where $\beta \in [0,1]$ is a random variable, and R(i) is the current search pulse frequency of bat individual i, and calculation method can be obtained from equation (15). If β> R(i) is true, then the current spatial state of the ith bat is generated from the vicinity of the optimal solution in the current space. If eq. β> R (i) is not true, then the current spatial state of the i th bat can be calculated from equation (14).

### 4.1.3 Discrete The Bat Algorithm

For basic algorithm bat population every bat in a small range of search ability is stronger, but the lack of effective mechanism of variation for the entire group the algorithm used to the current population optimal individual learning mechanism, once attracted by a local extremum, because there is no effective mutation mechanism, make the whole population is likely to be trapped in local optimal solution. In addition, bat algorithm also has some problems, such as the late convergence speed is slow, it is difficult to get rid of the local optimal, the population quickly gathers to the superbody, and the overall search ability is poor.

### 4.1.4 Harmonic Search Algorithm

Harmony Search (HS) algorithm is a new intelligent optimization algorithm proposed by z.w. geem et al., a Korean scholar in 2001. The algorithm simulates the process that musicians can achieve a beautiful Harmony state by repeatedly adjusting the tone of each instrument in the band based on their own memory in music composition. HS algorithm compares the harmony of Musical Instruments to the solution vector of optimization problem, and the evaluation is the objective function value of each pair. The algorithm introduces two main parameters, namely Pitch Adjusting Rate (PAR) and Harmony Memory Considering Rate (HMCR).

### 4.1.5 Improvements to the Bat Algorithm

Bat algorithm has a good search ability. But it takes a lot of iterations to produce the desired results. Here, we define the dynamic scaling factor parameter $\theta$, which is a factor in local search that limits the random walk of step size, and improve the proposed algorithm. With appropriate adjustment, this parameter reduces the number of iterations from the calculation time. In addition, this improvement can provide a discrete optimization problem of bat algorithm that is easy to tune. The dynamic scale factor parameter $\theta$ can be defined as:

$$\theta(iter) = \theta_{max} \exp\left(\left(\frac{Ln(\frac{\theta_{max}}{\theta_{min}})}{\theta_{max}}\right).iter\right) \tag{16}$$

This formula defines the relationship between broadband parameters in the harmony search algorithm. Instead of formula (13), the further evolved formula can be expressed as follows:

$$x_{new}(*i) = x_{old} + \varepsilon\theta(iter) \tag{17}$$

In the above formula, iter refers to the current iteration process. In order to solve the convergence example, the above enhanced version of the equation will be used. In a continuous optimization problem, $\theta_{max}$ and $\theta_{min}$ should be 1 and 0.001, respectively. At the same time, in discrete optimization problems, $\theta_{max}$ and $\theta_{min}$ can be taken as 10 and 0.01, respectively. However, these values are usually recommended for parameter integration in optimization problems. At the same time, these values are also recommended in cases where some of the parameters are highly correlated with the problem, but the problem needs to be adjusted well.

### 4.2 Network Slice Resource Allocation Algorithm Steps

Resource allocation is to search all possible scheduling sequences and finally find the best scheduling scheme. When bat algorithm is applied to resource scheduling, it is necessary to partition resources and encode bat individuals. In order to facilitate the calculation and improve the search effect of bat algorithm, this paper adopts binary encoding as the bat mode of bat individuals, that is, bat individuals correspond to the resources of each task, and each encoded bat individual actually corresponds to each resource.The steps of network slice resource allocation based on bat algorithm are as follows:

Step 1. The location of bat population x(i), velocity v(i), pulse emission rate r(i), pulse loudness A(i) and pulse frequency f(i) were initialized.ing and related conditions such as constraint functions.

Step 2. Initialize the number of bats in the current area (I $x_i \in \{1... N\}$).

Step 3. Different frequency generation strategies are adopted according to different adaptive values of individuals.

Step 4. Formula (2)~(4) were used to calculate the flying speed and vacancy position of bats.

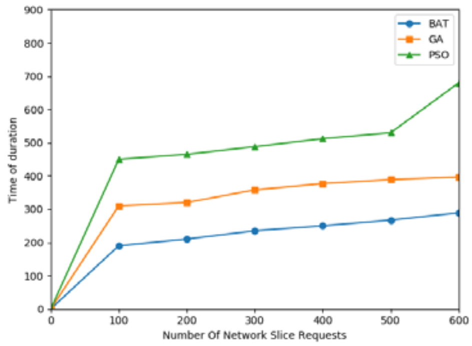
Step 5. Evaluate the new solution, update the position and speed if the conditions are met, and update the pulse loudness A(i) and emission rate r(i) by using formulas (9) and (6).

Step 6. Judge the termination condition of the algorithm. If the search precision w is satisfied or the search times are reached, exit and output the final result, otherwise return 2).

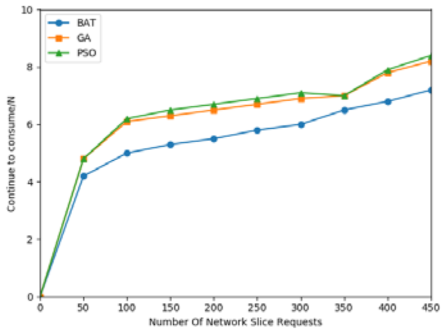
Step 7. Output the optimal extremum and individual.

## 5. Simulation results and analysis

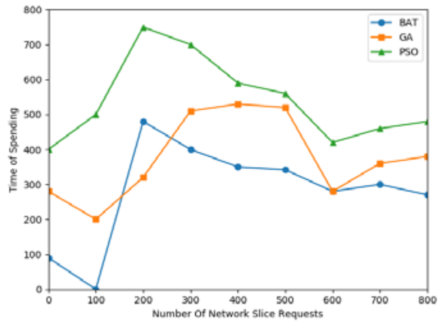
Experimental verification uses Python programming language for simulation. In this experiment, the number of task scheduling in the network slice is simulated as the number of bats, the optimal resource scheduling is regarded as the process of individual bat search, and the resource scheduling formula is simulated as the advantages and disadvantages of the location of bats. The algorithm in this paper adopts CloudSim platform test. Set the parameter virtual simulation tasks as 500, the virtual nodes as 8, and the iteration times as 500, and compare the algorithm in this paper with PSO and GA in the cloud computing model.




(a) Task completion time comparison of three resource load algorithms




(b) Comparison of energy consumption time of three resource load algorithms




(c) Comparison of overhead and time of three resource loading algorithms

Fig.1

As can be seen from the above two figures, the completion time of this algorithm in this paper is not significantly different at the beginning. However, with the increasing number of tasks, the difference is obvious. In terms of the resource cost of the three algorithms, the time of the proposed algorithm is better than that of the other two algorithms, and the time of resource allocation is saved to some extent.

## 6. Conclusion

Resource allocation is the key to network section, section of this paper is based on the network resource allocation, in view of the current cloud computing resource scheduling algorithm has slow convergence speed, low utilization rate of resources, such as defects, less use the bat algorithm control parameters, easy to implement, the advantages of simple calculation, this paper proposes a bat algorithm is based on harmony search algorithm of resource scheduling model. The simulation results show that compared with genetic algorithm and particle swarm optimization (pso) algorithm, the algorithm not only found the ideal cloud computing resource scheduling scheme, task completion time decreased dramatically, and overcome the deficiency existing in the current cloud computing resource scheduling algorithm, It improves the utilization rate of cloud computing resources and has a broad application prospect in modern large-scale cloud computing.

**Acknowledgments**

This work was financially supported by the National Natural Science Foundation of China (no.61531013) and National Key Research and Development Plan (2018YFB1800300).

## References

[1] Gandomi A H, Yang X S, Alavi A H, et al. Bat algorithm for constrained optimization tasks [J]. Neural Computing & Applications, 2013, 22(6): 1239-1255.

[2] Zong W G, Kim J H, Loganathan G V. A new heuristic optimization algorithm: harmony search. [J]. Simulation Transactions of the Society for Modeling & Simulation International, 2001, 76 (2):60-68.

[3] Calheiros R N, Ranjan R, De Rose C A F, et al. Cloudsim: A novel framework fo modeling and simulation of cloud computing infrastructures and services [R]. Technical report,Grids-TR-2009-1,Grid computing and distributed system laboratory,The University of Melbourne, Australia, 2009.

[4] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[5] ZHU Kai,SONG Hua-guang,LIU Li-jing. Hybrid genetic algorithm for cloud computing applications [C]/ /Proc of IEEE Asia-Pacific Services Computing Conference. 2011: 182-187.

[6] Zhu Kai,SONG Hua-guang,LIU Li-jing. Hybrid genetic algorithm for cloud computing applications [C]/ /Proc of IEEE Asia-Pacific Services Computing Conference. 2011: 182-187.

[7] Kong L R, Wang H, Yin H T, et al. Re-search on spectrum allocation algorithm based on discrete binary particle swarm optimization [J]. Automated Instrumentation, 2018, v.39; No.442 (06): 65-69.

[8] Abdelquoddouss Laghrissi, Tarik Taleb, Miloud Bagaa. Conformal Mapping for Optimal Network Slice Planning Based on Canonical Domains[J]. IEEE Journal on Selected Areas in Communications, 2018, PP(99):1-1.

[9] INTHARAWIJITR, Krittin, IIDA, Katsuyoshi, KOGA, Hiroyuki. Simulation Study of Low Latency Network Architecture Using Mobile Edge Computing[J]. Ieice Transactions on Information & Systems, E100.D(5):963-9