PCI Planning and Solution Based on Point Weighted Simulated Annealing Algorithm

Yifan Lai *, Haoran Gao

School of Management NPU, Northwestern Polytechnical University, Xi'an, China, 710072

* Corresponding Author Email: laiyifan98@outlook.com

Abstract. Physical cell identification (PCI) collision, confusion and interference are key problems in modern LTE network optimization, which significantly affect network performance and user experience. Traditional optimization methods that rely on manual adjustment and empirical rules are inefficient and difficult to adapt to complex network environments. This paper presents an optimization method based on simulated annealing to minimize PCI conflict, confusion and interference. Through the collection and pre-processing of user device MR Data and network topology data, combined with feature engineering to extract key features, a hybrid integer programming model with the goal of minimizing the total MR Value is constructed, and the constraint conditions of node unique PCI allocation and adjacent PCI differentiation are designed, and the improved simulated annealing algorithm is used to solve the problem. Experimental results show that the proposed method reduces the total MR Value from 50492016 to 32602231, effectively reduces PCI conflict and interference, significantly improves network stability and performance, and provides a new method with theoretical significance and practical value for LTE network optimization.

Keywords: Vertex Coloring, Multiobjective Optimization, Simulated Annealing, PCI.

1. Introduction

In LTE (Long Term Evolution) networks, the Physical cell identifier (PCI) is a unique identifier for each cell and is used to distinguish and manage different network cells. However, due to the limited PCI resources, it is difficult to achieve the uniqueness of PCI in each cell in the densely deployed network environment, so it is necessary to solve the problem through reasonable reuse. Too close a multiplexing distance will lead to PCI conflicts and confusion, and too far will cause resource waste, which directly affects network performance and user experience. Therefore, how to optimize PCI allocation to improve efficiency and reduce complexity is an important topic in LTE network optimization [1-3].

PCI conflict means that neighboring cells use the same PCI, resulting in handover failure and connection interruption. PCI confusion refers to the use of the same PCI in cells that are not directly adjacent but in the same area, affecting positioning and signal measurement. PCI interference is caused by the multiplexing of demodulation reference signals (DMRS), which affects connection establishment [4-6], as shown in Fig.1 and Fig.2. To avoid these problems, network planning needs to reasonably design PCI allocation strategies to ensure that PCI reuse in different regions is minimized.

Traditional PCI allocation strategies include fixed allocation, dynamic allocation and mixed allocation [7]. Fixed allocation is simple and feasible, but lacks flexibility; Dynamic allocation can adapt to network changes, but requires high computing resources. Hybrid allocation combines the advantages of both, but there is still room for optimization in complex scenarios. Acedo-Hernandez et al. proposed a multilevel graph partitioning algorithm to reduce conflicts and confusion, but its global optimization capability was insufficient [8]. Fairbrother et al. used mixed integer programming (MIP) to model the two-level graph division problem, which lacked sufficient flexibility [9]. Zeljkovic et al. proposed a PCI allocation algorithm ALPACA based on weighted automatic neighbor relationships, but the randomness and adaptability are limited, and it is difficult to achieve global optimization in complex environments [10].

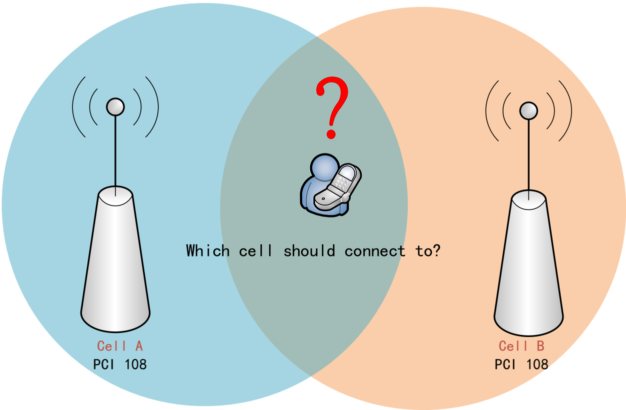

Fig 1. PCI Conflict.

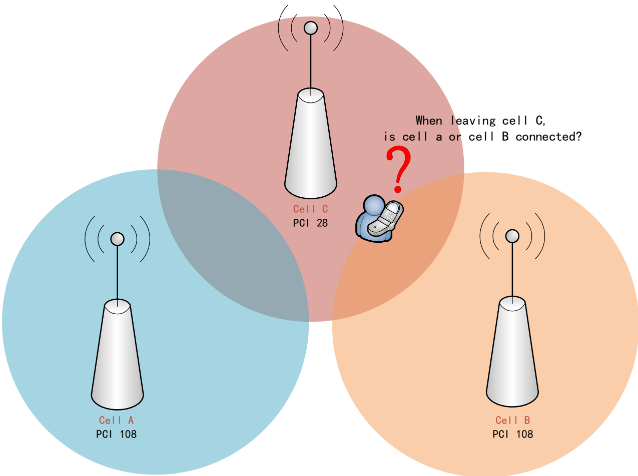

Fig 2. PCI confusion.

In order to overcome the above shortcomings, the simulated annealing algorithm is adopted in this paper, which has strong global optimization ability and high adaptability, and is suitable for multi-objective optimization [11,12]. The successful application of simulated annealing algorithm in optimization problems such as traveling salesman problem and constraint satisfaction problem verifies its potential in PCI allocation. Based on this, this paper proposes a PCI self-configuration algorithm based on simulated annealing to achieve a balance between conflict and confusion minimization through improved optimization. Experiments show that this algorithm can effectively improve PCI allocation efficiency and provide a new idea for LTE network optimization.

2. Data Source and Preprocess

The study uses data from www.saikr.com, specifically a dataset providing detailed node diagrams of cells in a specific area. This dataset includes 2067 selected cells and their relationships with associated cells, represented by collision Mr numbers, confusion Mr numbers, and modulo-3 interference Mr numbers. The PCI assignment is modeled as a graph coloring problem, aiming to minimize colors while ensuring adjacent vertices have different colors.

The dataset's three types of weight information are merged into a comprehensive edge-weight table, with missing or incomplete weights marked as 0. Cells are filtered based on optimization status, focusing on 1613 cells with a frequency of 504990. A total of 454 isolated cells with a frequency of 156510, unaffected by other cells, are excluded from further analysis. These cells are assigned PCI values randomly, assuming their Mr values are always 0.

A new dataset is constructed, including 1613 internal cells and 1279 external cells influencing them, totaling 2892 cells. Edge information is summarized based on the "or" principle, identifying associations involving 504990 frequency cells. The resulting 2892-cell dataset, with detailed node and edge information, serves as the basis for further analysis and problem-solving.

3. PCI Optimization Using PSO and Simulated Annealing

3.1. Diagram and Node Structure

For the PCI allocation problem, this paper constructed a graph structure $G(V,E)$ (where V represents the point set and E represents the edge set) and followed the following rules:

(1) Each cell serves as a node v_i in the graph, with its assigned PCI as the node feature, denoted as$PCI(v_i)$;

(2) If there is one or more situations of conflict, confusion, or interference between two communities, it is considered that there is a relationship between the two communities. In this paper, an edge is drawn between the nodes corresponding to the two communities, denoted as (v_i, v_j), which represents an edge pointing from node i to node j;

(3) Take the conflicting MR, confusion MR, and mode 3 interference MR of each of the two communities as the three weights for that edge. Due to the different weights of (v_i, v_j) and (v_i, v_j) in this question, the graph is a directed graph.

To store the graph, two main classes are used in the code for this paper: Graph and Vertex. The Vertex class stores information about nodes in a small cell and provides some methods. Its object represents a vertex in the graph, and each vertex represents a network cell. It contains the following attributes:

Key: Unique identifier of the community;

Frequency: The frequency used by the community;

Current_pci: The PCI value of the current network in the community;

Is_optimized: Mark whether the cell can be optimized, with a value of 0 or 1;

Neighbors: A dictionary type that stores the weights of adjacent vertices and their relationships (conflicts, confusion, interference). By using the neighbor's dictionary of each node, unidirectional weights can be stored to achieve edge data in a directed graph.

The Graph class stores all Vertex objects. During the modeling process, due to the large number of nodes, only the topology of the subgraph is shown here. This subgraph selects five nodes from the directed graph and draws their interrelationships. The logical relationship is shown in Fig.3:

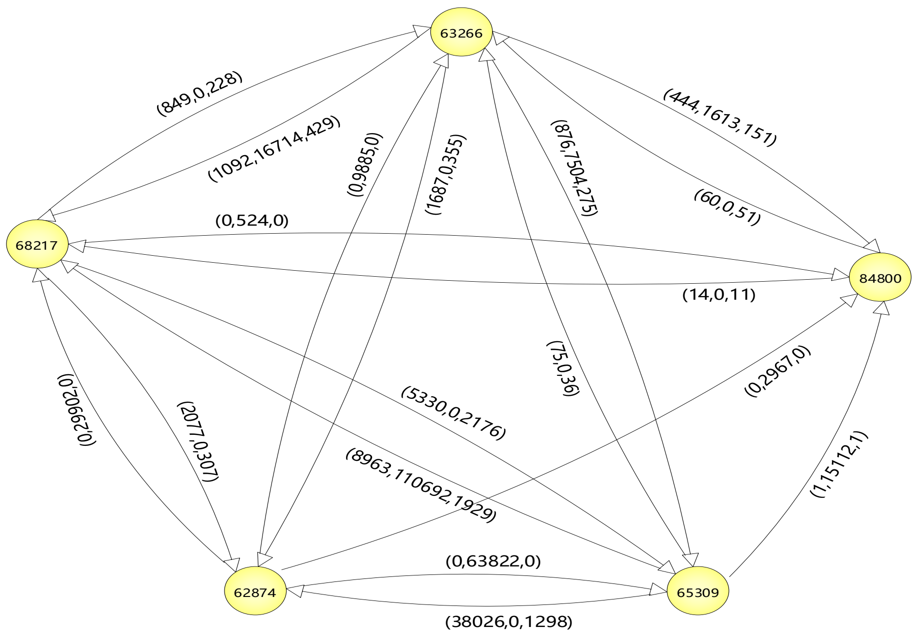

Fig 3. Schematic diagram of subnet topology structure.

It can be seen that the arrow in the figure indicates the relationship between the master node and the node in the adjacent area, and the node indicated by the arrow indicates the adjacent area. The data marked on the edge respectively represents the MR number of conflict, confusion, and mode 3 interference when the corresponding node is the master node and the adjacent node. The data clearly revealed the interaction and complexity of more than 310,000 paths between 1,613 nodes.

3.2. The Model Building

The research problem can be expressed as a single objective mixed integer programming model. Since there is no priority among the collision Mr number, confusion Mr number, and modulo-3 interference Mr number between given cells, the objective function of this problem can be obtained by simply adding them:

$$\mathrm{Min}\ Z = y_1 + y_2 + y_3 \tag{1}$$

Among them, the total number of conflicting MRs in the network corresponds to y_1, the total number of confusing MRs in the network corresponds to y_2, and the total number of mode 3 interfering MRs in the network corresponds to y_3.

The problem requires optimizing the communication load in the network by assigning reasonable values to PCI variable cells. Therefore, the variable PCI_k can be set to represent the PCI value of the k-th cell node, where k is the key value of the cell node corresponding to the adjustable PCI:

$$0 \le PCI_k \le 1007\ ,\ k, PCI_k \in N \tag{2}$$

Meanwhile, PCI_i and PCI_j are used to represent the PCI values of the i-th and j-th cell nodes, where i and j correspond to the key values of the cell nodes in the current network graph.

a_{ij},b_{ij}, and c_{ij} represent the collision value, confusion value, and interference value on the edge from the i-th cell node to the j-th cell node, respectively. Based on the fact that the PCI values of the i-th and j-th cell nodes are the same, the conflict MR values and confusion MR values in the network graph increase; The PCImod3 values of the i-th and j-th cell nodes are the same, and the interference MR value of the network graph increases. Here, the Kronecker function is used:

$$\delta(PCI_i, PCI_j) = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \ne j. \end{cases} i, j \in N \tag{3}$$

The Kronecker δ function represents that when i and j are equal, it is 1, and otherwise it is 0. This will result in three judgment coefficients that are 0-1 variables: α_{ij},β_{ij}, and γ_{ij}, used to control whether to include the collision value, confusion value, and interference value on the edge from the i-th cell node to the j-th cell node in the total conflict MR value, confusion MR value, and interference MR value of the network:

$$\alpha_{ij} = \delta(PCI_i, PCI_j) \tag{4}$$

$$\beta_{ij} = \delta(PCI_i, PCI_j) \tag{5}$$

$$\gamma_{ij} = \delta(PCI_i \bmod 3, PCI_j \bmod 3) \tag{6}$$

The total conflict MR value, confusion MR value, and interference MR value of the network graph are represented byy_1, y_2 and y_3. The calculation method is to traverse the entire network and multiply the conflict value a_{ij}, confusion value b_{ij}, and interference value c_{ij} on the edge in the direction from the i-th cell node to the j-th cell node by the judgment coefficients α_{ij},β_{ij}, and γ_{ij} of the three values:

$$y_1 = \sum_i \sum_j a_{ij}\, \alpha_{ij} \tag{7}$$

$$y_2 = \sum_i \sum_j b_{ij}\,\beta_{ij} \tag{8}$$

$$y_3 = \sum_i \sum_j c_{ij}\,\gamma_{ij} \tag{9}$$

The final form of the single objective mixed integer programming model is as follows: $Min\ Z = y_1 + y_2 + y_3$

$$s.t.\begin{cases} y_1 = \sum_i \sum_j a_{ij}\,\alpha_{ij} \\ y_2 = \sum_i \sum_j b_{ij}\,\beta_{ij} \\ y_3 = \sum_i \sum_j c_{ij}\,\gamma_{ij} \\ \alpha_{ij} = \delta(PCI_i, PCI_j) \\ \beta_{ij} = \delta(PCI_i, PCI_j) \\ \gamma_{ij} = \delta(PCI_i \ mod\ 3, PCI_j \ mod\ 3) \\ 0 \le PCI_k \le 1007 \\ i, j, k, PCI_k \in N \end{cases} \tag{10}$$

3.3. Algorithm settings

The large data prediction model for the user's electricity consumption is implemented in the Clementine software.

3.3.1. PSO

Particle swarm optimization (PSO) is derived from the simulation of bird predation behavior. Through the cooperation between particles and information transmission, the position and speed are dynamically adjusted according to the historical optimal, and the global optimal solution is gradually approached [13].

When dealing with a complex network optimization problem with more than 1600 nodes and 300000 edges, set the number of particles to 50, which is a trade-off between the computational complexity of the algorithm and the expected exploration ability. A higher number of particles can enhance the global search ability, but it will also significantly increase the computational cost. In a network with more than 1600 nodes, a moderate number of particles can ensure the performance of the algorithm while avoiding excessive consumption of computing resources.

The inertia weight (W) is set to 0.8, which helps to maintain a high momentum in the search process, so as to achieve a good balance between exploration and development. High inertia weight is conducive to the extensive search of particles in the solution space, which is particularly important for dealing with complex optimization problems in large-scale networks.

The individual learning factor (C1) and social learning factor (C2) were set to 0.5 and 0.3, respectively. These two parameters control the tendency of particles to update their positions according to their own experience and the optimal experience of the group. In large-scale network optimization, properly adjusting these parameters can help prevent the algorithm from premature convergence to local optimum, so as to enhance the diversity of solutions and the exploration ability of the algorithm.

3.3.2. Random sampling SA algorithm based on BIS (Better Initial Solution)

The simulated annealing algorithm is derived from the physical model of solid annealing process, taking temperature as the control parameter, through global search of high temperature stage, local convergence of gradual cooling, exploration and development of dynamic equilibrium solution, and finally approximates the global optimal solution [14].

In this paper, a simulated annealing algorithm based on weighted probability is proposed to get a better initial solution. In this algorithm, a function is constructed to select and replace the PCI value of the node. When a node is taken as a key value, the greater the sum of its Mr values, the higher the importance of the node in the network graph, and therefore the greater the probability of being extracted. For each selected node, the algorithm traverses 0 to 1007 PCI values and calculates the change of the objective function value. The PCI value with the largest reduction of the objective function value will be assigned to the node and the updated graph class instance will be returned. In this way, the whole network diagram can be optimized iteratively.

Based on the above steps, this paper designs an evaluation function of the annealing algorithm and sets an initial temperature and decay rate to simulate the process of metal annealing. And by intermittently using the normalized influence score as the selection probability to randomly select nodes, try to change their PCI value. Specifically, a completely random selection is made every 100 iterations, the PCI of one point is updated and solved, and the normal single-node update is carried out in other cycles. This strategy ensures that nodes with high influence (i.e. nodes connected to multiple high-weight conflicts or interferences) have a higher probability of being optimized, to improve the network performance on the whole. When a single node is updated, the domain is defined. The algorithm allows fine-tuning in a local range based on the current PCI value so that each iteration can finely explore the solution space.

When there is no improvement in successive iterations, the temperature recovery operation shall be carried out. That is, when the set cumulative number of invalid iterations is reached, the PCI of one node is randomly updated and rewritten. This setting helps the algorithm to "heat up" again when it falls into the local optimum and regain the ability to jump out of the local optimum. With proper rewarming, the algorithm can avoid premature convergence and increase the possibility of global exploration. The specific steps are as follows:

Specifically, the main process of the random sampling simulated annealing algorithm consists of 8 steps.

First define the objective function as follows:

$$y = \sum_{(v_i, v_j)\in E} \left[\delta\left(PCI(v_i), PCI(v_j)\right) \times MR(v_i, v_j)\right] \tag{11}$$

Where δ is the indicator function, if the two parameters are the same, return 1, if they are not the same, return 0?

Secondly, 1613 nodes are randomly sampled with weighted importance and put back. The importance of the node is determined by the sum of the maximum MR values possible for the node, that is, assuming that the node triggers all the penalties;

Then for each selected node v, the PCI value of that node is updated by:

$$PCI(v) = aTgmin_{pci' \in \{0,1,\cdots,1007\}} y_{new}(v, pci') \tag{12}$$

Where $y_{new}(v, pci')$ represents the new objective function value calculated when the PCI value of node v is updated to pci';

Then set the old objective function value to y_{old}, if y_{new} is less than y_{old}, then the probability of accepting the new graph class instance is 1; otherwise, the following formula is used to calculate the acceptance probability:

$$P = \frac{y_{old} - y_{new}}{\sqrt{T}} \tag{13}$$

Where T represents the current temperature;

Further, a random number $p \in [0, 1]$ is generated. If $p < P$, the update is accepted; Otherwise, keep the current state; At the same time, let the attenuation rate be α, and $T_{new} = \alpha \times T_{old}$ at each iteration; Based on the current iteration result, determine whether a temperature-back operation is required;

Finally, repeat the above steps until $T_{new} < 1$, ending the iteration.

3.4. Model solving

Through the previous data preprocessing, 1279 peripheral cells related to 1613 cells with the same frequency are taken into account to form a coloring problem and optimize the PCI value allocated to the remaining 1613 cells.

Table 1. PSO Algorithm Solution.

	MR total	Conflict MR	Confusing MR	interference MR
Initial target value	50492016	47838	0	50444178
Optimize target value	50267840	47827	512	50219501

Based on the above results in Table.1, since the impact of peripheral cells was included in the evaluation system, it was found that the total Mr value decreased from 50492016 to 50267840, with a small decrease, and the change directions of the three MR values were more complex. The decline of the total Mr value was due to the decline of the interference Mr value, but it was necessary to offset the rise of the confusion Mr value. See the following figure for specific data dynamic changes:

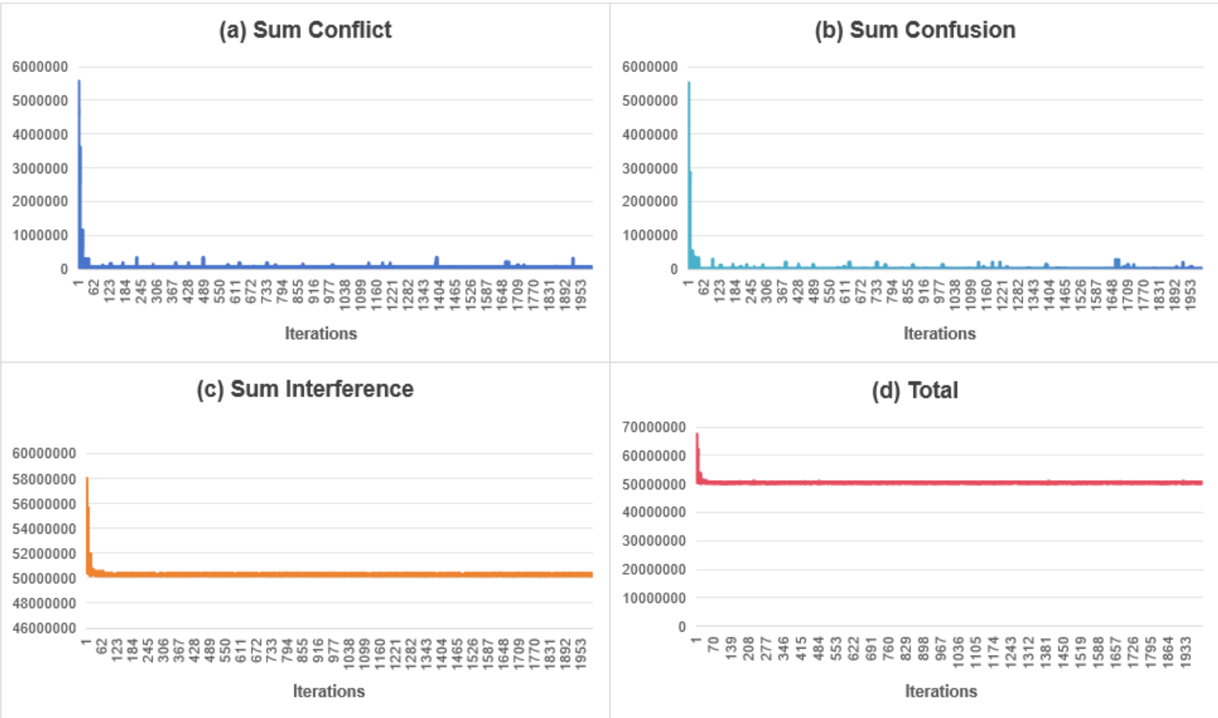

Fig 4. PSO algorithm - MR value variation diagram.

The above Fig.4 shows that under the operation of the PSO algorithm, the convergence speed of each MR value is faster, and it falls into the local optimal solution after about 36 iterations, and it is unable to break the dilemma to find a new optimal solution later.

Table 2. Simulated Annealing Algorithm Solution.

	MR total	Conflict MR	Confusing MR	Interference MR
Initial target value	50492016	47838	0	50444178
Interim target value	44181981	11719	3	44170259
Optimize target value	32602231	7153	5651	32589427

Based on the above PSO algorithm, the simulated annealing algorithm is used to solve the problem, and the results are shown in Table.2. It is found that the effect of this algorithm is better than that of PSO. The total Mr value decreases from 50492016 to 32602231, with a decrease of 35.43%. There are two major factors that show significant optimization effects.

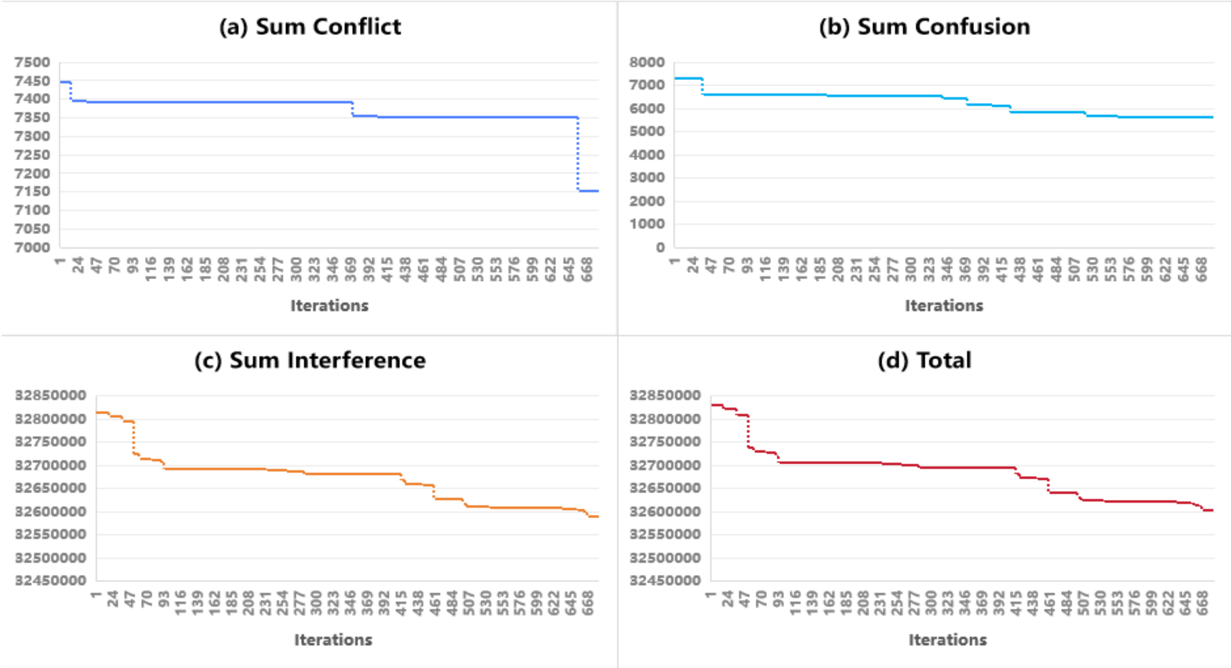

Fig 5. SA algorithm - MR value variation diagram.

The running results of the simulated annealing algorithm in Fig.5 show that the MR value fluctuates and declines, but the change range of conflict and confusion Mr value is small, and the change of MR total value is consistent with the change of interference Mr value, so it can be attributed to this.

4. Conclusions

In the single-object problem in this paper, based on the consideration of peripheral cells, PCI reallocation minimizes the sum of the three MR Types of conflict, interference and confusion in 2067 cells. Due to the disadvantages of the classical simulated annealing algorithm, such as high computational complexity, slow iteration, and easy to fall into local optimal, this paper proposes a point-to-point weighted random sampling simulated annealing algorithm based on the better initial solution, and will carry out the temperature-back operation when there is no improvement in successive iterations. Firstly, the algorithm finds the initial feasible solution through the simulated annealing algorithm of the point weighted PCI value traversal. In order to avoid falling into the local optimal, the algorithm is used to search for the optimization based on the initial feasible solution. With the deepening of iteration, it can be seen that the algorithm has achieved good optimization effect.

In summary, by developing a new method based on simulated annealing algorithm, this study successfully optimized the PCI conflict, confusion and interference problems in LTE networks, solved the local optimal solution problems existing in traditional methods, and provided a more intelligent and efficient solution. Future research can further optimize the parameter setting of the simulated annealing algorithm on the basis of this study, or combine with other optimization algorithms to improve its adaptability in more complex and variable network environments. At the same time, the method of this study is also expected to be applied to other types of wireless communication network optimization, and provide reference and guidance for a wide range of network optimization problems.

References

[1] Gan T ,Wang S ,Ma Q , et al.A Survey on Technologies and Challenges of LTE-U[J].COMPUTER SYSTEMS SCIENCE AND ENGINEERING,2022,41(1):321-337.

[2] Andrade C E, Pessoa L S, Stawiarski S. The physical cell identity assignment problem: a practical optimization approach[J]. IEEE Transactions on Evolutionary Computation, 2022, 28(2): 282-292.

[3] You Y H, Jung Y A, Lee S H, et al. Complexity-Effective Joint Detection of Physical Cell Identity and Integer Frequency Offset in 5G New Radio Communication Systems[J]. Mathematics, 2023, 11(20): 4326.

[4] Andrade C E, Pessoa L S, Stawiarski S. The physical cell identity assignment problem: a practical optimization approach[J]. IEEE Transactions on Evolutionary Computation, 2022, 28(2): 282-292.

[5] Krishnaswamy D. Quasi-Quantum PCI optimization in 5G Networks[C]//2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring). IEEE, 2021.

[6] Tunali A İ, Çirpan H A. Impact of Imperfect Channel Estimation on 5G-NR[C]//2021 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom). IEEE, 2021: 1-6.

[7] 3GPP. NR; Physical Channels and Modulation: Version 16.6.0, 3GPP Standard (TS) 38.211 [EB /OL]. (2021-06). [2024-10-30]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2440.

[8] Acedo-Hernández R, Toril M, Luna-Ramírez S, et al. A PCI planning algorithm for jointly reducing reference signal collisions in LTE uplink and downlink[J]. Computer Networks, 2017, 119: 112-123.

[9] Fairbrother J, Letchford A N, Briggs K. A two-level graph partitioning problem arising in mobile wireless communications[J]. Computational Optimization and Applications, 2018, 69: 653-676.

[10] Zeljković E, Bajić D, Perić S, et al. ALPACA: A PCI Assignment Algorithm Taking Advantage of Weighted ANR [J]. Journal of Network and Systems Management, 2022, 30(2): 33.

[11] Zhou Y, Xu W, Fu Z H, et al. Multi-neighborhood simulated annealing-based iterated local search for colored traveling salesman problems [J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(9): 16072-16082.

[12] Westbroek M J E, King P R, Vvedensky D D, et al. Pressure and flow statistics of Darcy flow from simulated annealing [J]. Journal of Physics A: Mathematical and Theoretical, 2020, 54(3): 035002.

[13] Sag T, Ihsan A. Particle Swarm Optimization with a new intensification strategy based on K-Means [J]. Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 2023, 29(3): 264-273.

[14] Niewiadomski S, Mzyk G. Optimization of Procurement Strategy Supported by Simulated Annealing and Genetic Algorithm[C]//International Conference on Dependability of Computer Systems. Cham: Springer Nature Switzerland, 2024: 196-205.