Load Balancing Mechanism Based on Sparse Matrix Prediction in C-RAN Networks

Yang Liu(✉), Zhanjun Liu, Ling Kuang, and Xinrui Tan

School of Communication and Information Engineering,
Chongqing University of Posts and Telecommunications, Chongqing, China
liuzj@cqupt.edu.cn

Abstract. In order to solve the problem that the existing algorithms in large-scale networks have high complexity in adjusting power iteratively, a load balancing mechanism based on sparse matrix prediction is proposed to achieve load balancing in C-RAN architecture. In order to minimize the correlation degree of load transfer and the balance of load transfer, the optimal sparse matrix block is obtained combined with Ncut cutting algorithm to realize dimension reduction and zero removal of the load transfer matrix. After the block, the load transfer matrix of each block is recalculated, and the load transfer matrix is used to predict the load. Finally, combined with the predicted load, the power adjustment step size is determined, and the pilot signal power of each block is adjusted in parallel to achieve load balancing. The simulation results show that the load balancing mechanism can reduce the complexity of load balancing.

Keywords: Sparse matrix · Matrix block · Load balancing

1 Introduction

The scale of the mobile network based on C-RAN architecture tends to expand and complexity [1]. Followed by, the network load balancing technology faces new challenges. The existing load balancing algorithms based on iterative adjustment of transmission power achieves the maximum throughput but the complexity is very high [2]. As a essential method of reducing the complexity, sparse matrix technology is an important research direction of wireless network technology in the future and has a wide range of applications. The similarity matrix describing image pixels in image segmentation algorithm is a large-scale sparse matrix. The dimension of sparse Matrix can be reduced by establishing the corresponding

This work is supported by the National Natural Science Foundation of China (NSFC) (61801064), Chongqing Education Commission Project KJQN (201801908), and PHD Initialted Fund Project (A1029-17).

H. Gao et al. (Eds.): ChinaCom 2019, LNICST 312, pp. 720–728, 2020.
https://doi.org/10.1007/978-3-030-41114-5_54

relationship between large-scale sparse matrix and undirected graph and combined with the Normalized cutting Criterion (Ncut) in Graph Theory [3]. The sparse matrix is used in the spectral clustering image segmentation algorithm. The similarity matrix created is sparse and the theoretical analysis and the image segmentation experiment results show that the algorithm can effectively reduce the computational complexity of the spectral clustering, and improve the accuracy and the robustness of the segmentation [4]. With the increasing scale of the network, not all the nodes in the network are related. Although the order of the matrix describing the relationship between the network nodes is very high, the matrix has sparse characteristics. Therefore, the sparse matrix can be used to solve the problem of the high complexity of the network [5]. In order to predict the future load better, the commonly used time series models are the autoregression prediction model (AR), neural network prediction model, Markov prediction model and so on [6]. Load balancing can be realized by adjusting the power of the pilot signal. However, the power adjustment of the pilot signal directly relate to the performance of the system. To find the best adjustment power, absorbing game theory is used to establish the market buying and selling model in economics. To the same end, reinforcement learning is applied to the load balancing of the ad hoc network to determine the optimal transmission power by iterating. Moreover, markov decision can be utilized to adjust power [7–9]. C-RAN, as an access network of cloud processing model, contains a large number of RRU due to its distributed wireless network architecture [10]. However, it is not able to adapt to the development trend of a mobile communication network in the future because its high complexity of load balancing technology. In order to solve the problem of the high complexity of the above load balancing algorithm, a load balancing mechanism based on sparse matrix prediction is proposed inspired of load transfer matrix obtained in reference [11]. In this mechanism, the predicted load is obtained by the load transfer matrix predicted by sparse matrix, and then the power adjustment step size is determined. The pilot signal power of each block is adjusted in parallel to realize the load balance of the block cell. The simulation results show that the proposed load balancing mechanism can reduce the complexity of load balancing.

The paper is organized as follows. The system model of this paper is introduced in Sect. 2, including mathematical formulas and theoretical analysis. Load balancing optimization model based on sparse matrix prediction is specialized in Sect. 3. The new algorithm and the mechanism of innovation are proposed in Sect. 4. The performance of the proposed mechanism and comparisons with the existing algorithms is analyzed in Sect. 5. Finally, this paper is conclued in Sect. 6.

2 System Model

Consider a multi-cell multiuser system, which has a base station and serves a single antenna user. According to reference [12], a load of each base station over time is represented as a first-order markov prediction model:

$$L(n+1) = R(n) * L(n) \tag{1}$$

Where $L(n) = [l_1(n), l_2(n), \cdots, l_M(n)]^T$ represents the load condition of each base station at n time, and $R(n)$ represents the load state transition matrix of the network at n time, as follows:

$$R(n) = \begin{bmatrix} r_{11}(n) & r_{21}(n) & \ldots & r_{M1}(n) \\ r_{12}(n) & r_{22}(n) & \ldots & r_{M2}(n) \\ \cdots & \cdots & \cdots & \cdots \\ r_{1M}(n) & r_{2M}(n) & \ldots & r_{MM}(n) \end{bmatrix} \tag{2}$$

Where $r_{ij}(n)$ represents the load transfer rate of the base station i at n time to the base station j at $n+1$ time. The $M+1$ network states are obtained by formula (1) to obtain $R(n)$. The load state of the next network is predicted by the load transfer matrix, and the load transfer matrix $R(n)$ is obtained by formula (3).

$$[L(n), L(n-1), \cdots, L(n-M+1)] = R(n) * [L(n-1), L(n-2), \cdots, L(n-M)] \tag{3}$$

The load transfer matrix $R(n)$ in large-scale C-RAN networks is a matrix with large sparsity. The division of zero and dimension reduction of $R(n)$ needs to be divided into blocks. Reasonable segmentation enhances the load transfer relationship within the block and weakens the load transfer relationship between blocks.

3 Load Balancing Optimization Model Based on Sparse Matrix Prediction

The above block problem can be transformed into the optimal segmentation problem of the undirected graph in graph theory. In order to find the most reasonable sparse matrix block, the following optimization model is established. The optimization goal is to minimize the load transfer correlation degree and load transfer balance. Therefore, the optimization model is established by means of joint optimization.The load transfer correlation degree is defined as the coupling degree of load transfer between blocks of an undirected graph, and the load transfer balance degree is defined as the mean square error of each block load after block segmentation of undirected graph.

Assume the base station set is $B = U_{m=1}^{K} * B_m$,$B_m \cap B_k = \Phi$,$\forall m \neq k (k = 1, 2, \cdots, K)$. K is divided into blocks, and the number of cells per block is represented by $N_k (k = 1, 2, \cdots, K)$. By using the parameters α and β, the load transfer correlation degree and load transfer balance degree are optimized jointly. And assume $V_j \in B_k$ if j cell is assigned to k block, then $x_{kj} = 1$, otherwise takes $x_{kj} = 0$. In order to meet reality, the number of cells in the block are not too small, so $N_k (k = 1, 2, \cdots, K) \geq 3$ and $K \geq 2$, The optimization model is established as follows:

$$
\begin{aligned}
G(x, \alpha, \beta) = \min[\alpha \cdot \sum_{k=1}^{K} \sqrt{(B_k - \bar{B})/K} + \beta \cdot \sum_{k=1}^{K} \frac{cut(B_k, B - B_k)}{vol(B_k)}] \\
\bar{B} = (\sum_{k=1}^{K} B_k)/K(K \geq 2) \\
B_k = \sum_{j=1}^{M} l_j \cdot x_{kj}(1 \leq k \leq K) \\
cut(B_m, B_k) = \sum_{i \in B_m, j \in B_k} r'_{ij} \\
vol(B_k) = \sum_{i \in B_k, j \in B} r'_{ij} \\
\text{s.t.} \quad \sum_{j=1}^{M} x_{kj} \geq 3 \\
\sum_{k=1}^{K} x_{kj} = 1 \\
x_{kj} = 0 \cup x_{kj} = 1 \\
\alpha + \beta = 1(\alpha \geq 0, \beta \geq 0)
\end{aligned} \tag{4}
$$

Where r'_{ij} represents the load transfer from the i cell to the j cell, corresponding elements in $R' = D - W$. W is $< R + R^T >$, R is the load transfer matrix and $< X >$ represents the zero-setting transformation of the main diagonal elements of the X matrix. $D = diag(d_1, d_2, \cdots, d_M)$ and $d_i = \sum_j w_{ij}$ represents the extent to which point i is associated with other nodes. w_{ij} is the element in an adjacent matrix W. $cut(B_m, B_K)$ represents the weighted sum of the associated edges of the subgraph B_m and the subgraph B_K, and $vol(B_K)$ represents the sum of the weights of the B_k edges of the subgraph. N_k is the total number of cells in the block k, K is the number of blocks, l_i is the load before cell i prediction, and r_{ij} is the element in the load transfer matrix after dimension reduction. B_k is the predicted load of the k block, and $\bar{B}$ is the average load of all blocks. M is the total number of system cells, l'_j is the load value of the predicted cell j.

4 Load Balancing Mechanism Based on Sparse Matrix Prediction

4.1 Solution Algorithm

The above optimization problem is a 0–1 integer programming problem. Using exhaustive method to obtain the optimal matrix block is with high complexity, so load balancing algorithm based on sparsity (SLBA) is designed combining with Ncut cutting algorithm to obtain the best matrix block. SLBA is as follow:

Algorithm 1. Load Balancing Algorithm Based on Sparsity (SLBA)

1: **Initialization:**input load transfer matrix R^{M*M}, and initial block number $k = 2$
2: **Output:**The block with the least load balance$\sigma = min(\sigma_2, \sigma_3, \cdots, \sigma_{[M/3]})$
3: Calculate adjacency matrix W and degree matrix D with $R + R^T$.
4: Eigenvalue decomposition of $D^{\frac{1}{2}} W D^{\frac{-1}{2}}$ and find out the eigenvector of the corresponding eigenvalue.
5: The feature vector $v_1, v_2, \cdots, v_k$ of the k maximum characteristic values are selected to form V^{M*k} by column
repeat.
6: Cluster $y_i (i = 1, 2, \cdots, M)$ by k-means algorithm where y_i is the row vector of V^{M*k} to obtain the partition of graphs $C_1, C_2, \cdots, C_k$.
7: Calculate the load transfer matrix for each block $R_1, R_2, \cdots, R_k$ according to block case.
8: According to the load transfer matrix of each block, the load of the cell in the block is predicted and calculate load transfer balance σ_k by $\sigma = \Sigma_{k=1}^{K} \sqrt{(B_k - \bar{B})^2 / K}$.
9: $k = k + 1$, then go to step 4, and calculate the load transfer balance under this block.
until $k > [\frac{M}{3}]$.
10: Select the first block with the lowest load balance $\sigma = min(\sigma_2, \sigma_3, \cdots, \sigma_[M/3])$.

4.2 Load Balancing Mechanism

According to the proposed algorithm (SLBA), the above optimization problems are solved. Then the solution with the least load transfer balance is found in the finite feasible solution. Finally, the solution with the lowest load transfer balance and the least load transfer correlation degree is selected as the optimal solution. After the optimal cell block is obtained, the load balancing of each block is carried out to realize the load balancing of the whole network.

5 Simulation and Result Analysis

The simulation scene is mainly aimed at the isomorphism network under C-RAN architecture, and the occupation rate of PRB in the cell is taken as the load measurement index, and the initial network users are randomly uniform in each cell. In order to simulate the real load imbalance scene, Three cells are selected as hot spots. The speed of users is 5 km/h, 30 km/h and 60 km/h, respectively, which represents the speed of walking, bicycle and car. The algorithm, the load balancing mechanism of the iterative method and the mechanism which can't adopt load balancing compared and analyzed. Detailed configuration of system simulation parameters is shown in Table 1.

According to the parameters in the above table and the corresponding scenes, the proposed algorithm is simulated and compared with the existing algorithms in terms of system capacity, Jain's fairness index, PRB occupation ratio and so on.

Table 1. System simulation parameter configuration.

Parameter name	Parameter values
Number of base stations	19
Base station coverage radius (m)	500
Base station transmission power (w)	40
System bandwidth (MHZ)	10
Load balancing cycle (ms)	200
Background noise power (dBm)	−104
Service rate (kbit/s)	512
Overload threshold	0.9
Motor pattern (km/h)	speed [5,30,60] direction (0, 360)
Road loss model (dB)	$L = 128.1 + 37.6 * lg(d(km))$

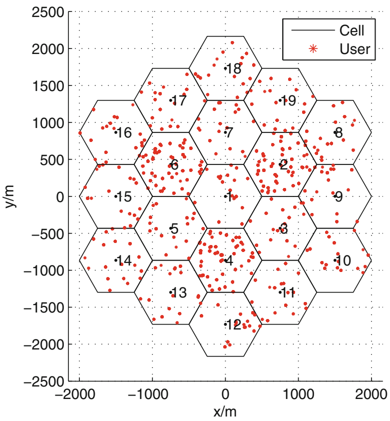

Fig. 1. Topology diagram of the base station and initial user distribution

Figure 1 is a scenario diagram for establishing a load balancing mechanism. It can be seen that the user density of each cell is different, so a reasonable load balancing mechanism is necessary to maximize throughput.

Figures 2 and 3 show the Jain fairness index and the relationship between system capacity and the total number of users, respectively. It can be seen from the diagram that the system capacity and fairness index of this mechanism are improved compared with the load balancing mechanism, and the system capacity and fairness index of this mechanism are slightly lower than those of the iterative load balancing mechanism. The capacity of the system is reduced by about $0.02\% - 0.1\%$, and the maximum reduction of fairness index is about 1.6%. This is because the load balancing mechanism is based on the established load forecasting, and there are always a small number of errors in the load forecasting. On the other hand, the sparse matrix block algorithm of the load

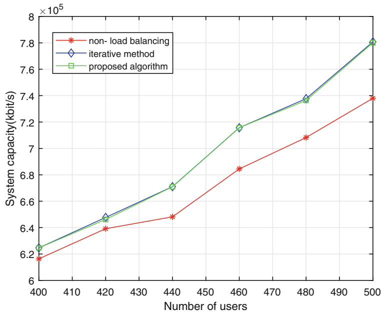

Fig. 2. System capacity and number of users

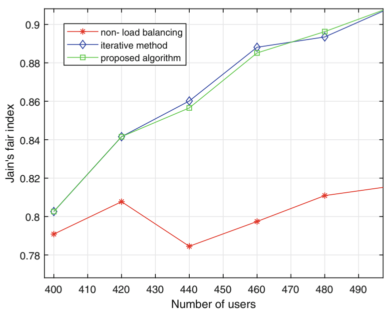

Fig. 3. Jain fairness index and number of users

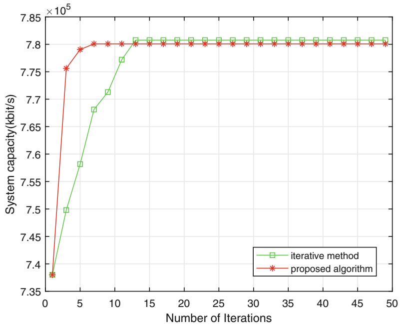

Fig. 4. System capacity and number of iteractions

balancing mechanism is the smallest load transfer correlation degree. But the load transfer between blocks still exists.

Figure 4 is a diagram of the system capacity varying with the number of iterations. It can be seen from the diagram that with the increase of the number of iterations, the load balancing mechanism of the iterative method and the proposed load balancing mechanism converge to the desired system capacity. However, the convergence speed of this load balancing mechanism is faster than that of iterative load balancing mechanism, which is because the load balancing mechanism is based on sparse matrix block. The number of cells in the block is smaller than the total number of cells in the network.

Figure 5 is a schematic diagram of PRB occupation ratio under different algorithms in each cell, and a straight line with a longitudinal coordinate of 0.9 represents the overload threshold. In this scenario, the total number of network users is 500. It can be seen from the diagram that compared with the unload balancing mechanism, the iterative method and the network of this algorithm do not exceed the overload threshold, and the proportion of PRB occupied by the iterative method is basically similar to that of the proposed algorithm.

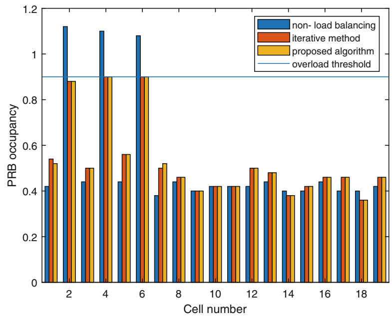

Fig. 5. Schematic diagram of PRB occupation ratio under different algorithms in each cell

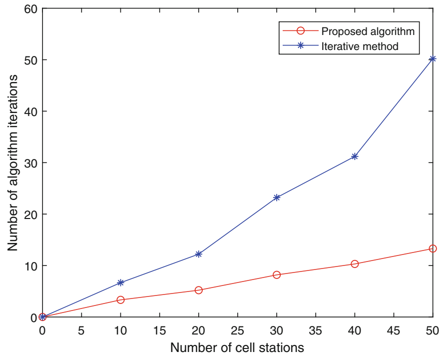

Fig. 6. Number of iterations and cell base stations

Figure 6 is the relationship between the number of iterations and the number of cell base stations. It can be seen that with the increase of the number of cell base stations, the number of iterations to achieve load balancing by the proposed algorithm (SLBA) is less than that by the iterative method. Moreover with the increase of the cell size, the number of iterations under the proposed algorithm increases more slowly, that is to say, the complexity of the proposed algorithm is lower.

From the above load balancing mechanism, it can be seen that if the load transfer matrix is not preprocessed, the complexity of the power adjustment algorithm is $O(M * N)$ directly by iteration, where M represents the total number of cells and N represents the number of iterations. If the sparseness of the load transfer matrix is used to block it, then load balancing is carried out in parallel. The complexity of the algorithm is $O(m * N)$, in which $m = max(N_1, N_2, \cdots, N_K)$ represents the number of cells of the largest block after block, and N represents the number of iterations of the algorithm. Obviously, m is smaller than M, especially in the case of large scale and a large number of blocks.

6 Conclusion

In this paper, a load balancing mechanism based on sparse matrix block is proposed. The sparse characteristics of the load transfer matrix and the Ncut cutting algorithm are considered to obtain the best matrix block, and then the load transfer matrix of each block is calculated to predict the load. Finally, the power adjustment step size is determined to achieve load balance. The simulation results show that the proposed load balancing mechanism can reduce the complexity of the load balance of the existing iterative algorithm under the condition that there is no difference between the existing iterative algorithm in ensuring the system capacity, jain's fair index and PRB.

References

1. Chai, T.S.: Brief introduction of C-RAN technology and its application scenario in mobile network. Chin. New Commun. **19**(12), 107–107 (2017)
2. Hao, Y.W., Wang, J.N., Li, X.X., et al.: Research on storage load balancing algorithm based on distributed environment. Inf. Technol. **09**, 55–58 (2016)
3. Cimorelli, F., Priscoli, F.D., Pietrabissa, A., et al.: A distributed load balancing algorithm for the control plane in software defined networking. In: 2016 24th Mediterranean Conference on Control and Automation (MED), pp. 1033–1040. Athens (2016)
4. Liu, Z.M., Li, Z.M., Li, B.H., et al.: Spectral clustering image segmentation algorithm based on sparse matrix. J. Jilin Univ. (Eng. Ed.) **47**(4), 1308–1313 (2017)
5. Zhang, Y., Roughan, M., Willinger, W., et al.: Spatio-temporal compressive sensing and internet traffic matrices. In: ACM SIGCOMM Conference on Data Communication, pp. 267–267. Barcelona (2009)
6. Ding, G., et al.: Spectrum inference in cognitive radio networks: algorithms and applications. IEEE Commun. Surv. Tutor. **20**(1), 150–182 (2018)
7. Liu, H.R.: Coverage control algorithm for wireless sensor networks based on noncooperative game. J. Commun. **40**(1), 71–78 (2019)
8. Li, Y., Chen, Z.H., et al.: Research on LTE load balancing algorithm based on game. Telev. Technol. **37**(19), 141–144 (2013)
9. Zhang, X., Nakhai, M.R., Ariffin, W.N.S.F.W.: Adaptive energy storage management in green wireless networks. IEEE Signal Process. Lett. **24**(7), 1044–1048 (2017)
10. Duan, T.T., Zhang, M., Wang, Z.L., Song, C.: Inter-BBU control mechanism for load balancing in C-RAN-based BBU pool. In: 2016 2nd IEEE International Conference on Computer and Communications (ICCC), pp. 2960–2964. Chengdu (2016)
11. Kuan, L., Liu, Z.J., Tan, X., et al.: Prediction matrix algorithm based on sparsity in C-RAN networks. Telecommun. Technol. **59**(3), 255–259 (2019)
12. Ma, Q.C., Liu, Z.J., Peng, X., et al.: Brief introduction of C-RAN technology and its application scenario in mobile network. Multidimensional Markovian model of traffic forecast of wireless network in architecture of C-RAN. Video Eng. **39**(3), 148–152 (2015)