Proceedings of The 6th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/2026.31731

Dynamic Service Chain Deployment Method with High Reliability Awareness

Zhenhao Li[1], Hua Li[1*]

[1]School of Computer Software, Inner Mongolia University, Hohhot, China
**Corresponding Author. Email: cslihua@imu.edu.cn*

Abstract. Aiming at 5G virtual network SFC more faults cause by virtualized resource slicing, and existing studies have low reliability evaluation accuracy, lack differentiated reliability guarantee, this paper put forward a high reliability-aware dynamic SFC deployment scheme; first, the scheme build reliability index based on device usage rate and surrounding reliability coefficient, set differentiated reliability needs according to VNF characteristics, and limit link length to cut backup resource cost, in VNF deployment phase, we propose a node ranking algorithm based on node location and reliability, use Google PageRank core idea to define node importance and guide deployment, link mapping adopt reliability-based method, it sort links with high reliability needs, use K-shortest path algorithm to filter, balance reliability and delay finally; to further ensure reliability, we design a differentiated backup scheme based on VNF and link importance, VNF with importance above average use dedicated backup, below average use shared backup, we introduce unit resource reliability improvement rate index to optimize link backup, it can make backup more efficient without too much resource waste; we build optimization model based on deep reinforcement learning, it interact continuously with virtual layer and underlying environment, maximize cumulative reward to get best deployment strategy, realize joint optimization of load balance and VNF reliability; simulation results show that, the proposed DQN-RDA algorithm in 50 physical nodes network, request acceptance rate keep around 90%, compare with existing algorithms, it perform better in fault rate, failure loss and backup resource consumption, this algorithm can well adapt to virtual network dynamic changes, it can meet high reliability needs of various applications.

Keywords: SFC, Dynamic deployment, high reliability, deep reinforcement learning

1. Introduction

Traditional communication devices couple tightly with services, lack flexibility; 5G uses virtual network tech to fix this, but resource slicing brings more faults, making SFC reliability a research hotspot, existing studies focus on overall reliability via backup or VNF mapping, yet they have flaws [1]: low-precision probabilistic failure models, no differentiated reliability for different VNFs' loss differences, thus this paper propose a high-reliability dynamic SFC scheme for virtual networks: build reliability index by device usage and surrounding reliability, set differentiated VNF needs, limit link length to cut backup cost, realize joint optimization of load balance and VNF reliability

DOI: 10.54254/2755-2721/2026.31731

via deep reinforcement learning and output dynamic deployment, supporting backup algorithm based on VNF and link importance for real needs, 5G builds virtual networks via NFV to decouple hardware and software, support SFC with VNFs for diverse apps, existing studies assume fully reliable infrastructure, which is unrealistic—VNF faults cause SFC interruption, traditional recovery strategies have high delay and cost, heuristic algorithms fail to handle dynamic changes, reinforcement learning can interact with virtual networks in real time, perceive node reliability and resource status, optimize VNF mapping and backup, fit virtual network better, balance resource efficiency and request acceptance while ensuring differentiated reliability [2].

2. Deployment optimization policy

In the VNF deployment phase, this paper put forward a node ranking algorithm based on node location and reliability, and it carry out VNF deployment work according to this ranking result [3].

$$G\left(m\right)=s\left(m\right) \tag{1}$$

$$E\Big(m\Big)=\sum\nolimits_{n\in N^S}\frac{1}{D(m,n)} \tag{2}$$

$$T\left(m\right)=\min_{j,h\in N^S}D\left(j,h\right)j,h\neq m \tag{3}$$

$$\lambda_m=\frac{\mathrm{MTTR}}{\mathrm{MTBF+MTTR}} \tag{4}$$

The VNF deployment algorithm based on node location and reliability learn from Google PageRank algorithm’s core idea, redefine the node importance.

$$r(m)=(1-\gamma)\widehat{C}_m+\gamma\sum\nolimits_{n\in J(m)}\frac{\widehat{B}_{mn}}{\sum_{x\in J(n)}\widehat{B}_{xn}}r(n) \tag{5}$$

$$\widehat{C}_m=\frac{C_mG(m)E(m)T(m)(1-\lambda_m)}{\sum_{n\in N^S}(C_nG(n)E(n)T(n)(1-\lambda_n))} \tag{6}$$

$$r=(1-\gamma)C+\gamma Hr \tag{7}$$

$$r=(1-\gamma)(I-\gamma H)^{-1}C \tag{8}$$

$$\sum\nolimits_{m\in N^S}h\left(m,n\right)=\sum\nolimits_{m\in J(n)}\frac{\widehat{B}_{mn}}{\sum_{x\in J(n)}\widehat{B}_{xn}}=1 \tag{9}$$

The steps of VNF deployment and execution algorithm based on node ranking are as follows: first calculate the importance r(m) of underlying nodes and sort them in descending order, that is sort(m′,descend′); second for SFC, calculate the importance r(u) of each VNF and sort them in descending order, that is sort(u′,descend′) [4]; then traverse each VNF u in the sorted SFC, at the same time traverse the sorted underlying nodes m that have not deployed VNF in the SFC, if the computing resource CSm of node m is greater than the computing resource demand CVu of VNF u,

DOI: 10.54254/2755-2721/2026.31731

map VNF u to node m and jump out of the current loop; if all VNFs of the SFC are successfully deployed to underlying nodes, start link mapping immediately, otherwise reject the SFC request, after successful VNF deployment, need to execute link mapping, first sort according to the reliability requirements of each sub-link in SFC, give priority to handle links with high reliability requirements, delete all underlying links that do not meet the corresponding sub-link requirements, so as to compress search space and improve mapping efficiency; then use K-shortest path algorithm to screen out K paths with the shortest delay and sort them in ascending order of delay; finally select the path with the shortest delay that meets the SFC link reliability constraints, ensure the reliability of SFC request mapping while minimize delay, the time complexity of this link mapping algorithm is O(K｜LV｜ ｜LS｜log｜NS｜) [5].

3. Backup scheme with reliability guarantee

When VNF reliability requirement is high, can choose two deployment nodes based on backup idea, one of them is backup node.

$$\begin{aligned} \ddot{\mathbf{N}}_{kn_i} &= \{n_i | \mathrm{hop}\left(s_k, n_{i-1}\right) + \mathrm{hop}\left(n_{i-1}, n_i\right) \\ &\quad + \mathrm{hop}\left(n_i, d_k\right) \leq u'_k, c^k_{n_I} < c_i\} \end{aligned} \tag{10}$$

$$\begin{aligned} p_{s_k, n_{i-1}} \in \;\; & p_{s_k, v^k_{I-1}}, p_{n_{i-1}, n_i} = \iota_{n_{i-1}, n_i}, p_{n_i, d_k} \\ &= \iota_{n_i, d_k}, n_i \in h\left(v^k_I\right) \end{aligned} \tag{11}$$

$$\sigma^k_{I,J} = \Delta / BW \tag{12}$$

$$\begin{aligned} \Delta &= \prod\nolimits_{l_{i,j} = \mathrm{g}\left(l^k_{I,J}\right), l^k_{I,J} \in L^k_v} \mathrm{RL}\left(l_{i,j}\right) \\ &\quad \cdot \left(1 - \prod\nolimits_{l^*_{i,j} = \mathrm{g}\left(l^{*k}_{I,J}\right), l^{*k}_{I,J} \in L^k_v} \mathrm{RL}\left(l^*_{i,j}\right)\right) \end{aligned} \tag{13}$$

$$\gamma^k_{I,J} = \sigma^k_{I,J} / \prod\nolimits_{l_{i,j} \in \mathrm{g}\left(l^k_{I,J}\right)} \mathrm{RL}\left(l_{i,j}\right) \tag{14}$$

To improve reliability and backup resource use, set different backup ways by normalized VNF importance; VNFs with importance above SFC average use dedicated backup, deploy on high-reliability nodes and backup on low-reliability ones. Those below average use shared backup to cut resource cost. Then choose deployment and backup nodes to finish mapping, if link reliability not meet, need link backup. This paper put forward unit resource reliability improvement rate index, aim to get enough reliability increase with least resource loss, iterate backup from high to low by this index until meet reliability requirement.

4. Key algorithms and simulation

Deep reinforcement learning combine deep learning's strong perception ability into reinforcement learning's decision process, seek optimal strategy by maximizing cumulative reward; the SDN/NFV architecture proposed in this paper use GS-O to finish business information collection, analysis and reliable mapping strategy execution; besides, since VNF sharing degree is not clear, need

DOI: 10.54254/2755-2721/2026.31731

continuously interact with virtual layer and underlying environment, explore the environment to define suitable VNF reliability requirements, and get the optimal deployment scheme at the same time.

$$\begin{aligned} N_{kn_i}^{'} &= \{n_i | \mathrm{hop}\,(s_k, n_{i-1}) + \mathrm{hop}\,(n_{i-1}, n_i) \\ &\quad + \mathrm{hop}\,(n_i, d_k) \leq u_k^{'}, R_{n_i} \geq \left(R_N^k\right)^{\omega_I}, \\ &\quad c_{n_I}^k < c_i \Big\} \cdot_{s_k, n_{i-1}} \in p_{s_k, v_{I-1}^k}, p_{n_{i-1}, n_i} \\ &\quad = \ell_{n_{i-1}, n_i}, p_{n_i, d_k} = \ell_{n_i, d_k}, n_i \in h\left(v_I^k\right) \end{aligned} \tag{15}$$

$$\begin{aligned} r_t &= \Big(-\phi \mathrm{gap}_{v_I^k} - (1-\phi)\Big[\mathrm{hop}\,(i,j) b_{I,J}^k / b_{i,j}\Big) \\ &\quad + c_{n_v}^k / c_{n_i}\Big] \Big| s = s_t, a = \mathbf{a}_t\Big) \end{aligned} \tag{16}$$

$$Q\left(s_t, a_t\right) = E\left[\sum_{t=1}^{\mathrm{D}} \alpha^t G\left(r | s = \mathbf{s}_t, a = \mathbf{a}_t\right)\right] \tag{17}$$

$$\pi^*\left(s\right) = arg\max_{a^{'}} Q\left(s_t, \mathbf{a}_t^{'}\right) \tag{18}$$

$$\begin{aligned} \widetilde{Q}\left(\mathbf{s}_t, \mathbf{a}_t | \theta^-\right) = &\quad G\left(r_t | s = \mathbf{s}_t, a = \mathbf{a}_t\right) \\ &+ \alpha \max_{a^{'} t+1} Q\left(\mathbf{s}_{t+1}, \mathbf{a}_{t+1}^{'} \middle| \theta^-\right) \end{aligned} \tag{19}$$

$$Loss = \left(\widetilde{Q}\left(s_t, \mathbf{a}_t | \theta^-\right) - Q\left(\mathbf{s}_t, \mathbf{a}_t | \theta\right)\right)^2 \tag{20}$$

$$\theta \leftarrow \theta - \lambda_{\mathrm{SGD}} \nabla L\left(\theta\right) \tag{21}$$

Continuous training make neural network converge and learn knowledge, then generate SFC deployment scheme by Q estimation; the algorithm get optimal results, but may lack physical nodes meet VNF reliability in mapping range, link mapping need backup to meet reliability and improve success rate. Simulation design: 50 physical nodes, CPU and bandwidth random in [40,90]; each SFC has 4–7 virtual nodes, CPU and link rate are 10; virtual network lifetime random from [200,250,300]. Virtual nodes have 3 parameter groups (70%,25%,5% proportion), links 2 groups. Monte Carlo method use 20 tests' average, DQN-RDA compare with CCI-RA and PARD. Results: Figure 1 shows DQN-RDA's request acceptance rate keep around 90% for 500 requests in 5000 time units; Figure 2 shows it use least backup nodes via priority deployment and shared backup; Figure 3 shows its lower failure rate and loss by ensuring VNF reliability; Figure 4 shows lower learning rate slows convergence but stabilizes it, need set by actual needs.

DOI: 10.54254/2755-2721/2026.31731

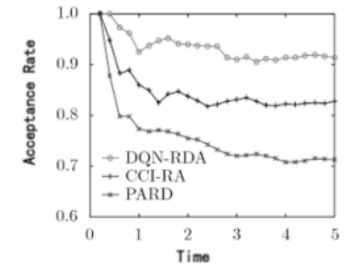

Figure 1. Comparison of SFC request acceptance rate

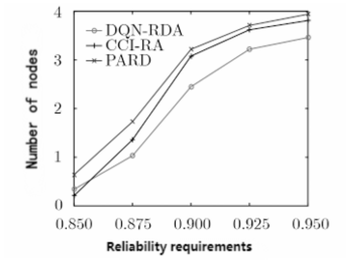

Figure 2. SFC backup consumption comparison

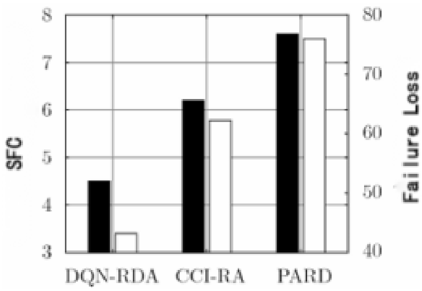

Figure 3. Comparison of SFC failure loss

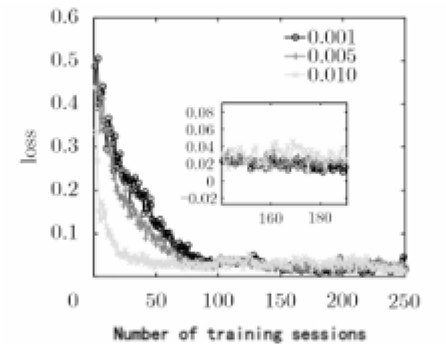

Figure 4. Impact of learning rate on convergence performance

5. Conclusion

Based on network virtualization architecture, aiming at high failure probability and loss even when overall reliability meet, this paper propose a reliable deployment algorithm via deep reinforcement

DOI: 10.54254/2755-2721/2026.31731

learning by ensuring VNF reliability to achieve SFC reliability. The scheme determine each VNF's reliability demand according to failure loss, aim to avoid reliability waste and balance load, use deep reinforcement learning to get deployment plan; add node (link) backup, ensure reliability and improve request acceptance rate. Simulation results show that this scheme maximize request acceptance rate, and effectively reduce virtual service loss caused by underlying network failure.

Funding information

This work was supported by the National Natural Science Foundation of China (No. 61862047, No. 62262047), the Inner Mongolia Science and Technology Program (No. 201802028, No. 2020GG0186), Inner Mongolia Key Research and Development and Achievement Transformation Project (No. 2025YFHH0101), and the fund of Supporting the Reform and Development of Local Universities (Disciplinary Construction) and the special research project of First-class Discipline of Inner Mongolia A. R. of China (No. YLXKZX-ND-036)

References

[1] Cao Haotong, Zhu Yongxu, Zheng Gan, et al. A novel optimal mapping algorithm with less computational complexity for virtual network embedding [J]. IEEE Transactions on Network and Service Management, 2018, 15(1): 356–371. doi: 10.1109/TNSM.2017.2778106.

[2] Zhang Xin, Qian Zhuzhong, Zhang Sheng, et al. Pattern-aware reliable virtual network function chain deployment [C]. 2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC), Guangzhou, China, 2017: 335–342. doi: 10.1109/ISPA/IUCC. 2017.00055.

[3] Shahriar N, Ahmed R, Chowdhury S R, et al. Generalized recovery from node failure in virtual network embedding [J]. IEEE Transactions on Network and Service Management, 2017, 14(2): 261–274. doi: 10.1109/TNSM.2017.2693404.

[4] Pourvali M, Bai Hao, Crichigno J, et al. Multicast virtual network services embedding for improved disaster recovery support [J]. IEEE Communications Letters, 2018, 22(7): 1362–1365. doi: 10.1109/LCOMM.2018.2822739.

[5] Sun Jian, Zhu Guangyang, Sun Gang, et al. A reliabilityaware approach for resource efficient virtual network function deployment [J]. IEEE Access, 2018, 6: 18238–18250. doi: 10.1109/ACCESS.2018.2815614.