ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/1147-1152
S Tejaswi *et. al.,* / International Journal of Engineering & Science Research

Secure Computation Offloading and Service Caching in Mobile Edge Computing Networks.

Vinod Kumar, S Tejaswi, G sivani
[1] Associate Professor, Department of CSE, Bhoj Reddy Engineering College for Women, India.
[2,3]B. Tech Students, Department of CSE, Bhoj Reddy Engineering College for Women, India.

ABSTRACT

Mobile edge computing (MEC) enables the caching of various services close to users, thereby reducing service delay for emerging applications. However, realizing efficient and secure computation offloading is challenging due to the limited storage capacity of MEC servers and the offloading security issue arising from the open nature of wireless channels. In this letter, we investigate a MEC network with an eavesdropper, where the MEC server caches service programs required for task execution. Moreover, we propose a cooperative jamming-assisted scheme to enhance the security of computation offloading, in which a friendly jammer actively transmits signals to disrupt eavesdropping. Considering the limited caching capacity of the MECserver and the security concerns of computation offloading, we formulate a delay minimization problem by jointly optimizing service caching placement, transmit power, and devices' computation offloading decisions. To tackle this problem, we develop an efficient layered algorithm using successive convex approximation and the cross entropy-based technique. Numerical results demonstrate the effectiveness of our proposed scheme in reducing the overall delay in completing devices' computation tasks compared to baseline schemes.

Key Words—Mobile edge computing, service caching, secure computation offloading, cooperative jamming.

I. INTRODUCTION

the commercialization of 5G and the proliferation of mobile devices, various computation-intensive applications such as interactive gaming, augmented reality (AR), and medical care are constantly emerging However, some devices are trapped in the dilemma of limited computing capabilities, which impedes the execution of these applications. Benefiting from the advent of mobile edge computing (MEC), the issue can be addressed by pushing computation resources down to the network edge . As a result, the computation-intensive tasks of devices can be offloaded to the MEC server, thereby reducing the execution latency and the power consumption.

Despite the potential of empowering devices with enhanced computation performance, MEC networks face security challenges due to the broadcast nature of wireless communications. Considering that the cryptography-based security technique requires high-complexity key management, physical layer security (PLS) has been proposed as an alternative to guarantee secure transmission . In particular, cooperative jamming

(CJ) as a kind of PLS technology has attracted extensive attention due to its ability to inject jamming signals to confuse eavesdroppers (Eves) . The authors in investigated a secure offloading problem in an unmanned aerial vehicle (UAV)-assisted MEC system by optimizing resource allocation and UAV's trajectory. In a secrecy-based MEC network was studied, where non-offloading devices form a jammer group and provide CJ to impede eavesdropping. Also, by exploring inter-user interference caused by non-orthogonal multiple access, a secure computation offloading scheme was developed in to achieve security provisioning.

While the above works have addressed PLS-aided secure offloading, they have overlooked the heterogeneity of computation tasks and the issue of service caching.[1] This is particularly important when devices deal with computation-intensive applications like AR and medical care, which require different service programs. With the support of service caching, a MEC server can cache specific service programs and directly process computation tasks related to those services . However, due to the limited caching space of the MEC server, it is impractical to store all of service programs. Considering the relationship between service caching and computation offloading, a weighted-sum delay minimization problem was studied in . To reduce computation delay and energy consumption, the authors in jointly designed service caching and computation offloading schemes in a single-user MEC network. Also, the work further investigated the problem of offloading dependent tasks with service caching in MEC networks. It is worth noting that the security issues related to computation offloading were not considered in , , and . In order to realize efficient and secure offloading, it is imperative to address both secure computation offloading and service caching placement.

Motivated by the above discussions, this letter investigates a secrecy-based design of computation offloading and service caching in MEC networks. To enhance the security of computation offloading, we design a CJ-assisted scheme in which a friendly jammer sends jamming signals to decrease

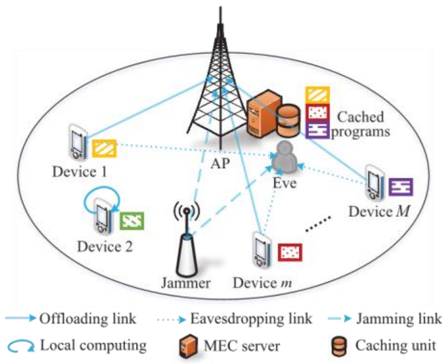

Fig. 1.Service caching-assisted MEC network.

Eve's eavesdropping ability. Our objective is to minimize the task completion delay of devices by jointly optimizing service caching placement, transmit power, and devices' computation

ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/1147-1152
S Tejaswi *et. al.,* / International Journal of Engineering & Science Research

offloading decisions subject to the secrecy offloading rate constraint and the limited caching capacity constraint. Since the formulated problem is a mixed-integer nonlinear programming (MINLP) problem, we develop an efficient layered algorithm by decomposing the formulated problem into two subproblems, which are handled by successive convex approximation (SCA) and the cross entropy (CE)-based approach. Simulation results demonstrate the superiority of our scheme compared with baselines in terms of the task completion delay.

II. SYSTEM MODEL AND PROBLEM FORMULATION

System Model

We consider a service caching-assisted MEC network as shown in Fig. 1, which consists of an access point (AP) integrated with a MEC server, a jammer, an Eve, and multiple devices. In this system, devices generate computation-intensive tasks that require service programs or databases to support their executions. We denote the sets of devices and service programs as M ={1,...,M} and N ={1,...,N}, respectively. Accordingly, these computation tasks can be offloaded if the MEC server has cached the service programs. Owing to the broadcast nature of wireless links, the task offloading process is vulnerable to being overheard by the Eve. To enhance security, a PLS-assisted secure offloading scheme is developed, in which the friendly jammer transmits jamming signals to degrade the information reception quality of the Eve. Similar to and , we consider that channel state information (CSI) is available for the service caching-assisted MEC network.

In this network, all channels experience quasi-static flat fading so that channels remain constant over a time block but vary across different time blocks. We focus on a time block and assume that the computation task of device m∈M needs to be supported by service program n∈N. To characterize the mapping relationship between device m's computation task and service program n, we adopt n = φ(m). For device m's computation task, let S_m and C_m respectively represent the data size of the task and the number of central processing unit (CPU) cycles required for executing one-bit task data. Also, each time block comprises local computing, computation offloading, and computing result feedback. Similar, we neglect the time spent on feedback transmission because the size of computation results is usually small. Next, service caching, secure computation offloading, and computing delay models. SIMULATION RESULTS

This section performs simulations to verify the secure computation offloading and service caching scheme. The simulations are conducted in a square area of 60 m × 60 m. We consider that there are 5 devices randomly distributed around the AP. The system parameters are set following [5] and [9]. Specifically, the channel gains are modeled as $|h_{a,b}|^2 = \theta_{a,b} d^{-}_{a,b}{}^{\beta}$ for (a,b) ∈ {(m,A),(m,E),(J,A),(J,E)}, where $\theta_{a,b}$ is an exponentially distributed random variable with a unity mean, $d_{a,b}$ represents the distance between nodes, and β denotes the path-loss exponent. Unless other specified, we set B = 1 MHz, N = 5, $\sigma_A{}^2 = \sigma_E{}^2 = -110$ dBm, β = 3, $P_J{}^{max} = 1$ W, $P_m{}^{max} = 0.23$ W, $S_m = [0.3,0.4]$ Mb, $c_n = [1,2]$ GB, $R_{min} = 2 \times 10^4$ bps, $C_{max} = 4$ GB, $f_m^{MEC} = 10$ GHz, $f_m^{loc} = 0.7$ GHz, and

$C_m=10^3$ cycles/bit.

We first present the convergence behaviors of the proposed CE-based algorithm in Fig. 2(a), which shows the values of $\{\phi_n,\psi_m\}_{\forall n\in N,\forall m\in M}$. It is obvious that the algorithm yields

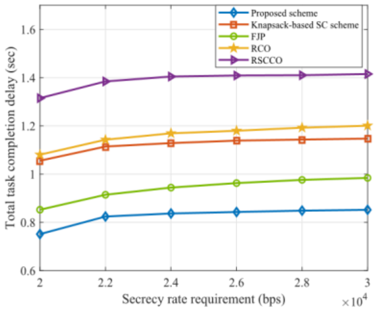

Fig. 3. Total task completion delay w.r.t. the secrecy rate requirement.

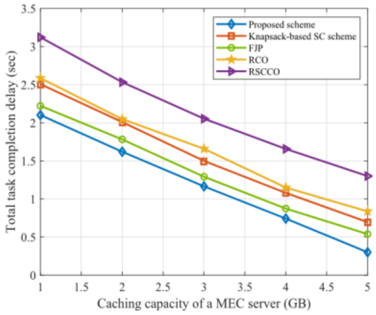

Fig. 4.Total task completion delay w.r.t. MEC server's caching capacity.

stable values of $\{\phi_n,\psi_m\}_{\forall n\in N,\forall m\in M}$ after 8 iterations. It is worth noting that when ϕ_n is close to 1, it signifies that service program n is cached on the MEC server. Also, ψ_m close 1 indicates that devices offload their computation tasks to the MEC server. Besides, we evaluate the convergence behavior of the SCA-based algorithm in Fig. 2(b). It is clear that SCA converges within 3-4 iterations, which demonstrates its rapid convergence.

To quantify the effectiveness of the proposed scheme, we compare against several baseline schemes. Knapsackbased service caching (SC) scheme by the 0-1 knapsack algorithm [13]. Fixed jamming power (FJP) scheme adopts fixed transmit power for the jammer to interfere with the Eve. Random computation offloading (RCO) scheme adopts random offloading decisions for devices. Random service caching and computation offloading (RSCCO) scheme randomly caches service programs on the MEC server with limited caching capacity and randomly selects task execution modes for devices.

Fig. 3 compares devices' task completion delay with respect to (w.r.t.) different secrecy rate requirements. Two observations can be drawn from this figure. Firstly, when the secrecy rate requirement becomes more stringent, there is an overall increase in the task processing delay. This phenomenon occurs as a larger proportion of devices select to perform local execution to meet the secrecy requirement, resulting in prolonged task execution delay. However, as the secrecy rate requirement continually increases, the growth rate of the task completion delay slows down. This trend can be attributed to the fact that the offloading decisions of devices are no longer the predominant factor influencing the task completion delay. To satisfy the secure offloading requirement, it is necessary to coordinate the transmit power for computation offloading and the jamming power for CJ, which impacts devices' task completion delay.

Fig. 4 illustrates the simulation results for the total task completion delay against the caching capacity of the MEC server in different schemes. It can be seen that the task completion delay of all devices decreases with the increasing cache capacity. This trend can be attributed to the fact that a larger cache capacity allows the MEC server to store more service programs. On this basis, devices can be provided with more choices to offload their tasks to the MEC server or perform local computing, ultimately reducing the total delay. Moreover, the proposed scheme outperforms the baseline schemes. This demonstrates that the proposed scheme's ability to allocate communication resources flexibly and make effective offloading decisions for devices.

III. CONCLUSION

In this letter, we have investigated secure computation offloading in service caching-assisted MEC networks. To minimize the total task completion delay of devices, we have optimized service caching placement, transmit power, and devices' computation offloading decisions. Since the formulated problem is a non-convex MINLP, we have developed an efficient layered algorithm by exploring the SCA and the CE approach. Numerical results have illustrated that the proposed scheme outperforms the baseline schemes. Designing secure computation offloading and service caching schemes under imperfect CSI would be an interesting future work.

REFERENCES

[1] Z. Ning, X. Wang, and J. Huang, "Mobile edge computing-enabled 5G vehicular networks: Toward the integration of communication and computing," IEEE Veh. Technol. Mag., vol. 14, no. 1, pp. 54–61, Mar. 2019.

[2] L. P. Qian, H. Zhang, Q. Wang, Y. Wu, and B. Lin, "Joint multi-domain resource allocation and trajectory optimization in UAV-assisted maritime IoT networks," IEEE Internet Things J., vol. 10, no. 1, pp. 539–552, Jan. 2023.

[3] Z. Zhu et al., "Robust beamforming design for IRS-aided secure SWIPT terahertz systems with non-linear EH model," IEEE Wireless Commun. Lett., vol. 11, no. 4, pp. 746–750, Apr. 2022.

[4] Y. Xu, T. Zhang, D. Yang, Y. Liu, and M. Tao, "Joint resource and trajectory optimization for security in UAV-assisted MEC systems," IEEE Trans. Commun., vol. 69, no. 1, pp. 573–588, Jan. 2021.

[5] L. Qian, W. Wu, W. Lu, Y. Wu, B. Lin, and T. Q. S. Quek,

"Secrecy-based energy-efficient mobile edge computing via cooperative non-orthogonal multiple access transmission," IEEE Trans. Commun., vol. 69, no. 7, pp. 4659–4677, Jul. 2021.

[6] K. Wang, H. Li, Z. Ding, and P. Xiao, "Reinforcement learning based latency minimization in secure NOMA-MEC systems with hybrid SIC," IEEE Trans. Wireless Commun., vol. 22, no. 1, pp. 408–422, Jan. 2023.

[7] K. Wang, W. Chen, J. Li, Y. Yang, and L. Hanzo, "Joint task offloading and caching for massive MIMO-aided multi-tier computing networks," IEEE Trans. Commun., vol. 70, no. 3, pp. 1820–1833, Mar. 2022.

[8] S. Bi, L. Huang, and Y. A. Zhang, "Joint optimization of service caching placement and computation offloading in mobile edge computing systems," IEEE Trans. Wireless Commun., vol. 19, no. 7, pp. 4947–4963, Jul. 2020.

[9] Q. Shen, B.-J. Hu, and E. Xia, "Dependency-aware task offloading and service caching in vehicular edge computing," IEEE Trans. Veh. Technol., vol. 71, no. 12, pp. 13182–13197, Dec. 2022.

[10] A. D. Wyner, "The wire-tap channel," Bell Syst. Tech. J., vol. 54, no. 8, pp. 1355–1387, Oct. 1975.

[11] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[12] X. Huang, W. Xu, G. Xie, S. Jin, and X. You, "Learning oriented crossentropy approach to user association in load-balanced HetNet," IEEE Wireless Commun. Lett., vol. 7, no. 6, pp. 1014–1017, Dec. 2018.

[13] X. Dai et al., "Task offloading for cloud-assisted fog computing with dynamic service caching in enterprise management systems," IEEE Trans. Ind. Informat., vol. 19, no. 1, pp. 662–672, Jan. 2023.