Chapter 4
Mobile Edge Computing for Beyond 5G/6G

Abstract This chapter first introduces the mobile edge computing (MEC) paradigm in beyond 5G and 6G networks. The motivations, applications, and challenges of integrating MEC into 6G are discussed in detail. We then present a new paradigm, MEC-empowered edge model sharing, as a use case for 6G. Furthermore, the architecture and processes of the MEC-empowered edge model sharing system are provided to show the integration angles of MEC and 6G networks.

4.1 Fundamental Characteristics of 6G

Although still in the early stage, a number of studies have provided visions for 6G [45–47]. Besides considerably improved data rates and communication latency, 6G networks are also considered to be human-centric and connected intelligence. The key features of 6G networks should be as follows.

- *Extremely high data rates and low latency*: Applications in 6G require much higher data rates and much lower latency than in 5G. The data transmission rates are expected to be in the hundreds of gigabytes or even terabytes. The latency should be extremely reduced, and services and applications are thus provided in real time. The extremely high data rates also generate new requirements for more spectrum resources. Hybrid terahertz–visible light communication systems are expected to offer more unexplored bandwidth resources for 6G networks.
- *Low energy consumption*: The increasing number of connected smart devices, such as Internet of Things devices and smartphones, in 6G requires the energy consumption to be low to extend their running time and provide reliable services.
- *High edge intelligence*: Artificial intelligence (AI) is assumed to play a crucial role in 6G networks. The concept of connected devices has evolved into connected intelligence in 6G. Edge intelligence is a key enabler of 6G networks [48]. Network performance will be improved by optimizing the allocation of resources such as the spectrum, computation, and power in the network [49]. Moreover, the integration of AI techniques into edge networks is expected to be greatly improve the quality of service (QoS).

Y. Zhang, *Mobile Edge Computing*, Simula SpringerBriefs on Computing 9,
https://doi.org/10.1007/978-3-030-83944-4_4

- *High security and privacy*: As the number of involved users increases dramatically in 6G, their devices generate a large amount of data. Since the generated data contain users' private information, the risk of data leakage during data transmission and storage is a major threat for 6G networks. Emerging technologies such as blockchain and federated learning are needed to enhance network security and data privacy.

4.2 Integrating Mobile Edge Computing (MEC) into 6G: Motivations, Applications, and Challenges

In cloud-based scenarios, the long-distance transmission of data from end devices or edge servers to the cloud incurs great latency and security risks and consumes a great amount of bandwidth. In 6G systems, a series of emerging applications, such as virtual reality (VR) and real-time video, require ultra low latency performance. Meanwhile, the explosive growth of smart devices in 6G also brings a large amount of distributed computational resources to the edge. In this regard, conventional cloud-based computation can hardly satisfy the expected performance requirements of 6G systems.

4.2.1 Use Cases of Integrating MEC into 6G

MEC enables the computation of applications and services to be executed at the edge of networks, reducing transmission latency and mitigating the threat of data leakage. Moreover, by deploying AI algorithms on edge servers, MEC leverages the distributed computational capabilities of devices and enables edge intelligence to be extensively realized in 6G. Thus, MEC is a key enabling technology for 6G systems. In areas that benefit from MEC, the use cases of MEC can be classified into three categories: consumer-oriented services, operator and third-party services, and network performance and quality of experience improvement services [50, 51].

In the category of consumer-oriented services, end users benefit from MEC by offloading computation to an edge server to run various 6G applications that require high computational capability and low latency performance. For example, in the scenarios illustrated in Fig. 4.1, such as face recognition or smart camera applications, the end devices need to analyze collected images in near real time. In such a case, neither cloud servers nor end devices can satisfy the requirements, due to long transmission distances or constrained computation resources. MEC enables end devices to run such low latency applications by offloading heavy computation to edge servers.

In the use cases of operators and third-party services, operators and third parties benefit from MEC systems. In 6G networks, the increasing numbers of smart devices generate huge amounts of distributed data. Directly transmitting these data

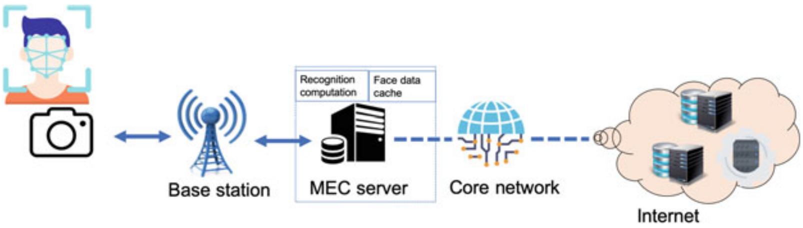


Fig. 4.1 Example application of MEC: Face recognition

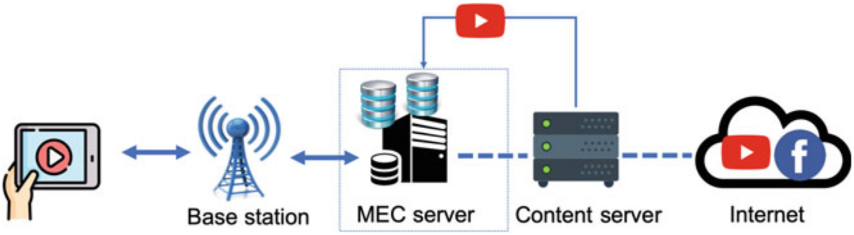


Fig. 4.2 Example application of MEC: video content caching

to the cloud will occupy a great deal of communication resources and lead to the additional consumption of storage and computation resources on cloud servers. In such scenarios, the MEC server operates as the gateway to collect and process generated data in advance. The processed data are then transmitted to cloud servers by the MEC server, which significantly reduces the transmission load from the edge to the cloud and mitigates the computational burden of centralized cloud servers.

In terms of network performance and quality of experience improvement services, MEC alleviates the congested backhaul network by means of content caching and traffic optimization. In content caching, MEC servers store popular content in advance by analyzing the historical records of users in their area, as shown in Fig. 4.2. Once the users request related content, the MEC servers will return the cached content directly to them. Through the content caching of MEC applications, the transmission latency is reduced and the user experience is improved. Moreover, MEC can help to optimally schedule traffic by gathering and analyzing network information and user requirements at the edge.

4.2.2 Applications of Integrating MEC into 6G

Considering the above benefits, MEC can be applied in a series of 6G applications. As shown in Fig. 4.3, the applications can be categorized as follows.

- *Dynamic resource sharing*: In 6G networks, the increasing numbers of connected devices and delay-sensitive applications require tremendous resources to ensure

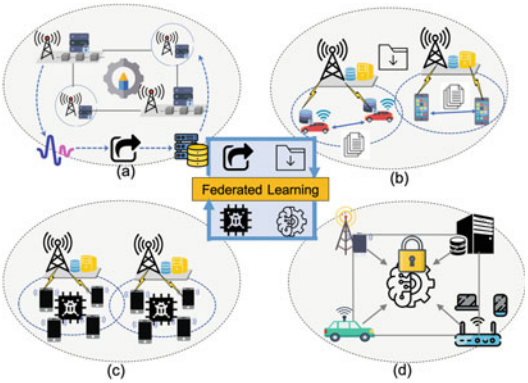


Fig. 4.3 Applications of MEC for 6G

QoS. The types of resources include spectrum resources, computational resources, and even storage resources. The limitation of such resources hinders the wide deployment of delay-sensitive applications in 6G networks. Resource sharing is an effective way to mitigate resource constraints. However, what to share and how to share are two basic issues that must be carefully addressed in resource sharing. MEC provides solutions to these issues by modeling and analyzing the network and optimizing sharing policies.

- *Distributed device-to-device caching*: In 6G networks, massive amounts of high-quality low latency applications, such as online games and real-time multimedia, generate huge amounts of content on edge devices. Instead of storing these contents on the cloud server (e.g., a macro base station), caching these contents at the edge considerably reduces the transmission costs and centralized storage burden. In 6G, since the computational and storage capabilities of smart devices will be significantly improved, caching content with end users can better leverage the distributed resources to reduce transmission latency and improve the QoS. End users with constrained resources are caching requesters, while end users with sufficient resources are caching providers. The device-to-device caching system is illustrated in Fig. 4.3b. The MEC server collects the information of end users under its coverage and optimally determines the caching strategy by analyzing and predicting content popularity among distributed users. The analysis can be conducted through optimization algorithms and AI algorithms that jointly consider the latency requirements and current information on the demands and offers from end users.
- *Joint edge computation offloading*: Since blockchain maintenance and the aggregation of updates require intensive computation, it is a challenging task for edge servers to execute computations within the applicable constraints, especially with large numbers of participating nodes. To alleviate the computational pressure, MEC completely utilizes the distributed computing resources by splitting the computation task into shards and offloading these onto other computing servers with sufficient computing resources. Moreover, the offloaded computation tasks can

also leverage the target user's data to complete the computation, which can further reduce the transmission overhead. An overview of the computation offloading scheme is shown in Fig. 4.3c.

- *Secure and private data analysis*: In 6G networks, a large amount of network data must be processed and analyzed to improve the QoS. With increasing concerns of data security and privacy, conventional cloud-based mechanisms raise serious threats of the leakage of user data. MEC allows the data to be analyzed at the edge of networks or even at the side of end users. Empowered by emerging paradigms such as federated learning [52], MEC will considerably enhance data privacy in the data analysis of 6G applications.

4.2.3 Challenges of Integrating MEC into 6G

Although integrating MEC into 6G has a series of benefits, new challenges also arise. Considering the characteristics of 6G networks and connected devices, the main challenges can be summed in three points: the heterogeneity of distributed resources, the high mobility of end users such as vehicles and mobile devices, and increasing security and privacy concerns.

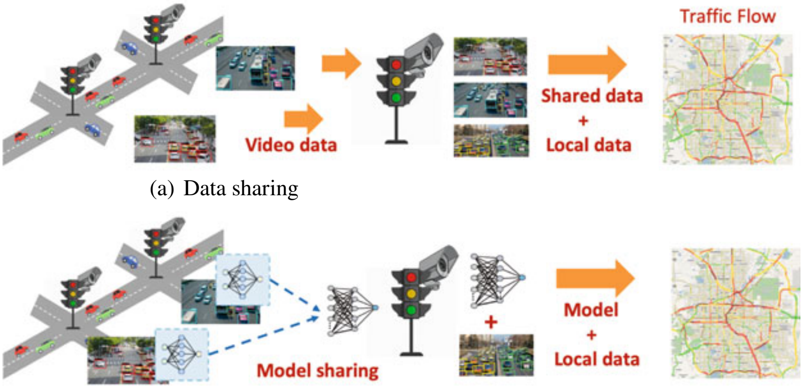
- *Distributed heterogeneous resource management*: In 6G networks, a huge amount of multidimensional heterogeneous resources have emerged as the number of smart devices has increased. In addition, as the capabilities of mobile devices improve, many resources are distributed among these devices. To improve the QoS and utility of distributed resources, heterogeneous resources need to be optimally allocated in real time. MEC plays a crucial role in edge resource management. However, the heterogeneity of the distributed resources, the dynamic system states, and critical latency constraints raise new challenges to integrating MEC into 6G for real-time resource management. Ways to improve the intelligence of an MEC system to address resource heterogeneity and to improve latency performance for real-time resource allocation require further investigation.
- *Reliability and mobility*: There are many fast-moving scenarios in 6G network, such as vehicular networks and mobile networks. In these scenarios, end users are continuously moving in the network. The network topology therefore varies, since the times and communication channels between users and base stations are unstable. However, the demand for low latency and high-reliability services also exists among end users. In a conventional MEC system, the MEC server executes computation tasks or caches content to reduce the transmission delay and improve computational capability. New MEC schemes must therefore be developed for 6G to guarantee the continuity of services for moving users in dynamic networks.
- *Security and privacy*: The increase in the number of end devices also generates huge amounts of user data. Leveraging these data for analysis can improve the QoS. For example, the accuracy of advertising recommendations can be further improved by learning the behaviors of users. Moreover, using AI algorithms to

learn the network running data can help to improve the network performance to satisfy the requirements of 6G networks. However, these data can contain sensitive user information. The risks of data leakage increase in this process. To integrate MEC into 6G, concerns of user privacy and security need to be addressed. More privacy-preserving machine learning algorithms and security collaboration mechanisms are required to enhance the security and privacy of MEC systems.

4.3 Case Study: MEC-Empowered Edge Model Sharing for 6G

4.3.1 Sharing at the Edge: From Data to Model

In conventional data sharing scenarios, the data providers share original data directly with the data requesters, which incurs a large amount of data transmission and increases the risk of data leakage. For example, a conventional traffic prediction application scenario is depicted in Fig. 4.4a. Distributed cameras share their video data with others and the cloud server to obtain overall traffic flow conditions. The traffic analysis and prediction are executed on the cloud server and then sent back to the end users. In the model sharing scenario, shown in Fig. 4.4b, end users equipped with MEC servers train locally based machine learning models with their collected video data. The trained machine learning models are shared with other users requesting the sharing of traffic data. Requesters then run the received machine learning model on their local data and build a new model for predicting real-time traffic conditions. By leveraging MEC to share the computing model instead of original data at the edge, response latency is reduced and data privacy is considerably enhanced.



(a) Data sharing

(b) Model sharing

Fig. 4.4 From data sharing to model sharing

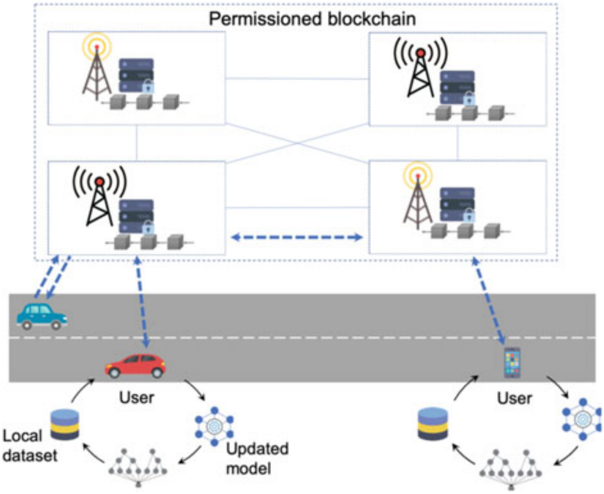


Fig. 4.5 The architecture of MEC-empowered model sharing

4.3.2 Architecture of Edge Model Sharing

The architecture of edge model sharing is illustrated in Fig. 4.5. We introduce blockchain into the proposed architecture to construct a secure sharing mechanism among end users who lack mutual trust. In the proposed sharing scheme, the providers register to a permissioned blockchain with their data profiles and run local training on their data to build the machine learning models. The permissioned blockchain runs on the base stations or roadside units as the parameter server. The registration information of users, the model parameters, and the sharing events are recorded in the blockchain. The requesters retrieve the blockchain for potential providers and request multiple users to provide the models. Through blockchain, the providers can be rewarded for sharing their models with requesters.

4.3.3 Processes of Edge Model Sharing

Based on the proposed architecture, edge sharing applications can be performed in MEC systems. The overall edge sharing procedures are shown in Fig. 4.6, which shows all the processes of MEC-empowered model sharing. The detailed processes are as follows.

1. *Initialization*: When data providers join the system, local similarity clustering is performed to classify these datasets, as well as the providers, into various categories. The similarity between different datasets is quantified by their logical distances, such as cosine similarity and Jaccard similarity. For a specific data

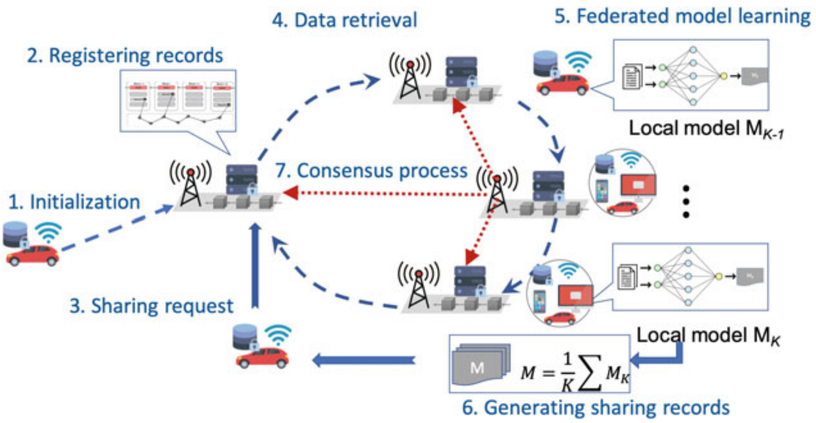


Fig. 4.6 The processes of MEC-empowered model sharing

provider P_i, nearby blockchain nodes will search the blockchain network to find similar data records. Then the ID of the dataset from P_i is generated based on the hash vectors of similar records, to ensure that similar datasets hold close IDs. The participants are divided into different communities according to their ID distances, that is, data similarity.

2. *Registering retrieval records*: Data provider P_i is required to register in the blockchain by sending a public key PK_r and its data profiles to a nearby blockchain node (e.g., MEC server). The blockchain node then generates a data retrieval record for provider P_i and broadcasts it to other nodes in the network for verification. The nodes in the blockchain verify their received records and pack them into candidate blocks. The candidate blocks are then verified through a consensus protocol and written to the permissioned blockchain if they are verified.
3. *Launching data sharing requests*: Data requester P_r submits a sharing request $Req = \{f_1, f_2, ..., f_x\}$ that contains the requester ID, the requested data category, and the time stamp to a nearby blockchain node SN_{req}. The sharing request Req is signed by P_r through the requester's private key SK_r.
4. *Data retrieval*: When the blockchain node near P_r receives the sharing request, it first validates that the identity of P_r is legal. If P_r is an authorized user, the blockchain node searches for the sharing records in the permissioned blockchain to check whether the request has been processed before. If there is a hit, the cached model will be returned to requester P_r directly. Otherwise, the blockchain node will carry out a data retrieval process among registered providers according to their ID distances, to find related data providers.
5. *Data model learning*: Data providers related to the request Req work together to train a collaborative data model $\mathcal{M}$. The local training samples consist of a request query f_x and its corresponding query results $fx(D)$, $D^T =< f_x, f_x(D) >$. The local models are trained on dataset D^T and aggregated into a global model $\mathcal{M}$.

The learned global model is then returned to the requester P_r as the result, which is also cached by the system for future requests. The requester can obtain the exact results it required based on the received model and its local data.

6. *Generating sharing records*: Data sharing events are recorded in the blockchain as transactions and broadcast to other blockchain nodes for verification. These records are collected by blockchain nodes and packed into candidate blocks.
7. *Carrying out consensus*: Candidate blocks consisting of data sharing records are verified by blockchain nodes participating as data providers. The blockchain nodes compete for the opportunity to generate blocks of the blockchain though consensus protocols such as proof of work or delegated proof of stakes. Nodes that obtain the right to generate blocks add their candidate blocks to the blockchain. The sharing records in the blockchain are traceable and tamper proof.

The combination of blockchain and federated learning enables secure intelligent data sharing in 6G networks. Based on federated learning, the data sharing among mobile devices is transferred to model sharing, which avoids the transmission of original data and reduces the risks of data leakage. Moreover, integrating the training process in federated learning with the blockchain consensus process improves the utilization of computing resources and the efficiency of data sharing. This edge model sharing case shows the great potential of integrating MEC into 6G networks to improve QoS and applications. MEC brings edge intelligence to wireless edge networks and enhances the connected intelligence among end devices in 6G networks.