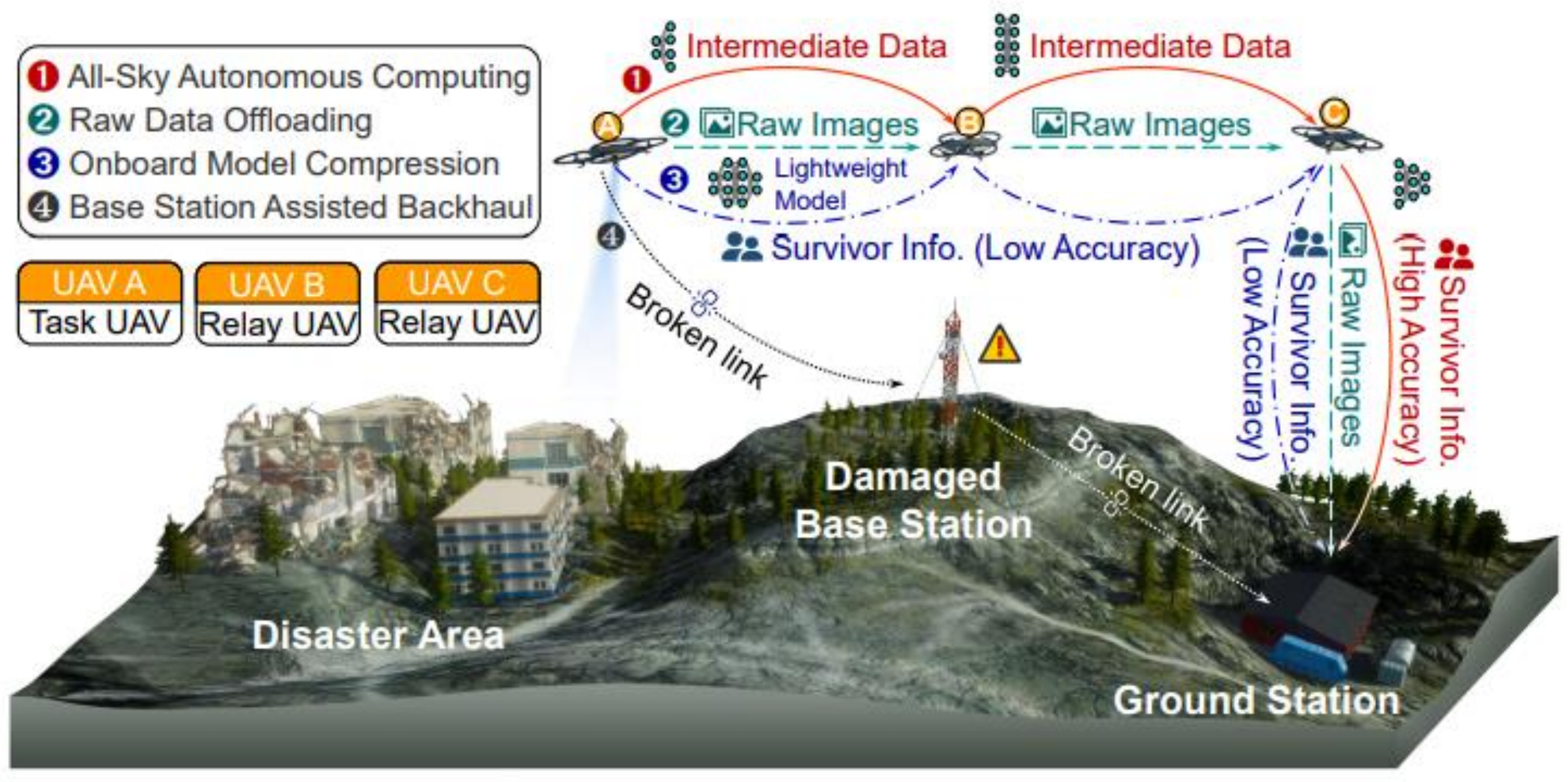
IEEE Hong Kong 6G Wireless Summit
11-12 September 2024
Hong Kong, China
IEEE ComSoc IEEE Communications Society | IEEE

Dynamic Air-Ground Networking and Clustering algorithms for High-Performance Edge Federated Learning

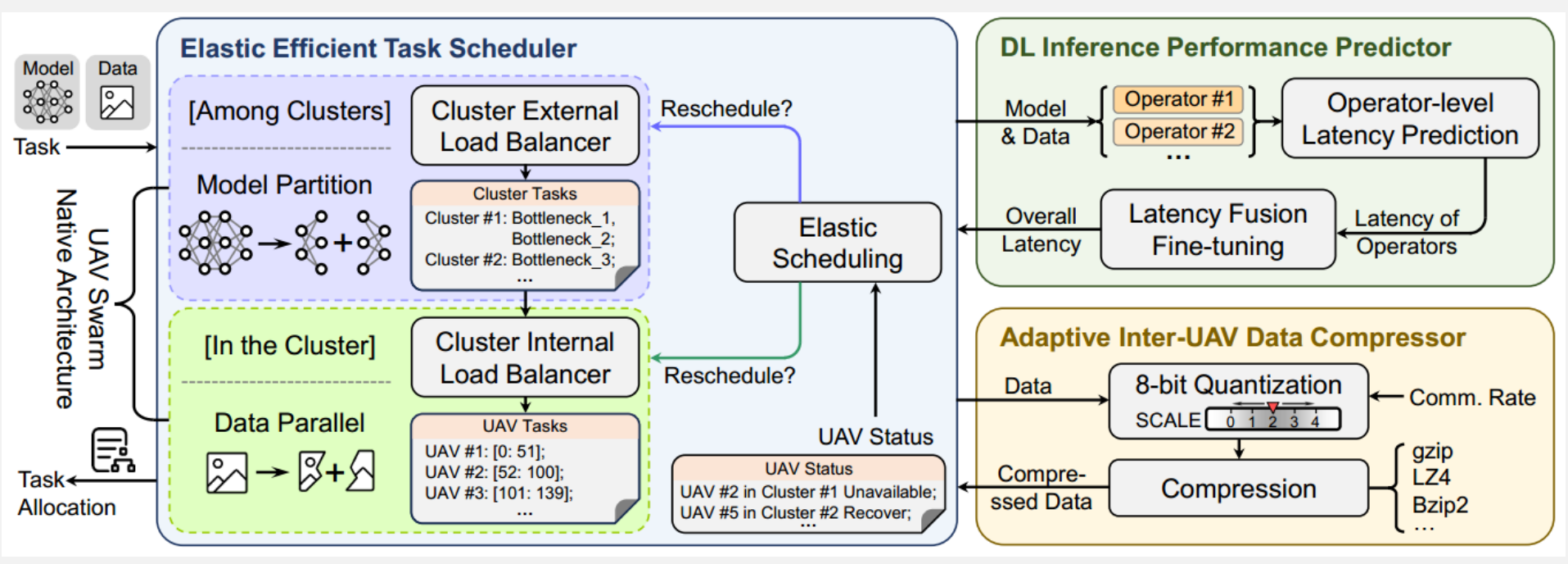
Hao Sun, Yuben Qu,Chao Dong, Haipeng Dai, Zhenhua Li, Lei Zhang, Qihui Wu, and Song Guo

Abstract Unmanned aerial vehicles (UAVs) play an essential role in emergency cases and adverse environments for applications like disaster detection and mine exploration. To process the massive volume of sensing data generated by various sensory payloads in these applications, existing works either compress deep learning (DL) models to conduct onboard computing, or offload raw data back to the resourceful ground station with the help of relay UAVs due to base station damage. However, the former sacrifices the inference accuracy of DL models (up to 10% accuracy loss), while the latter achieves high accuracy at the cost of significant latency, due to limited wireless communication resources in the multi-hop transmission. To address the problem, exploiting the resources of the UAV swarm including both task UAVs and relay UAVs, we build up an all-sky autonomous computing (ASAP) system to autonomously conduct collaborative computing in the swarm, to achieve both high accuracy and low latency of sensing data processing. In detail, we first propose a novel UAV swarm-native collaborative computing architecture, considering the general hierarchy and clustering structure of UAV swarms, as well as the characteristic of DL model execution. We then design an elastic efficient task scheduler to allocate computing tasks for UAVs, and update the scheduling scheme online when some UAVs are unavailable, with the aid of a lightweight and accurate DL inference performance predictor. Finally, we design an adaptive inter-UAV data compressor, to adapt to the limited and dynamic communication resources between UAVs. Experiment results on 24 airborne computers and five real-world UAVs show that, the proposed system can perform collaborative computing in a timely manner and effectively deal with situations when some UAVs become unavailable.



SYSTEM OVERVIEW

The goal of ASAP is to *conduct efficient and reliable collaborative computing in the UAV relay swarm*,which relieves the burden of air-ground communication by transmitting valuable data results instead of raw data. The system is composed of three modules, i.e., elastic efficient task scheduler, lightweight accurate DNN block performance predictor, and adaptive inter-UAV inference data compressor . First, when there is a task to be processed, i.e., a DL model and a sequence of data, the elastic efficient task scheduler partitions the task to UAV clusters and UAVs inside to conduct efficient collaborative computing. Besides, the allocation scheme can be updated by the task scheduler when the status of UAVs varies, e.g., some UAVs become unavailable or rejoin the swarm. Second, the task allocation scheme is developed with the help of the DL inference performance predictor, which can estimate the inference latency of various models and data partitions on different UAVs at a clip. Finally, the intermediate data transmitted between UAVs will be further compressed by the adaptive inter-UAV data compressor to save the inter-UAV communication resource.



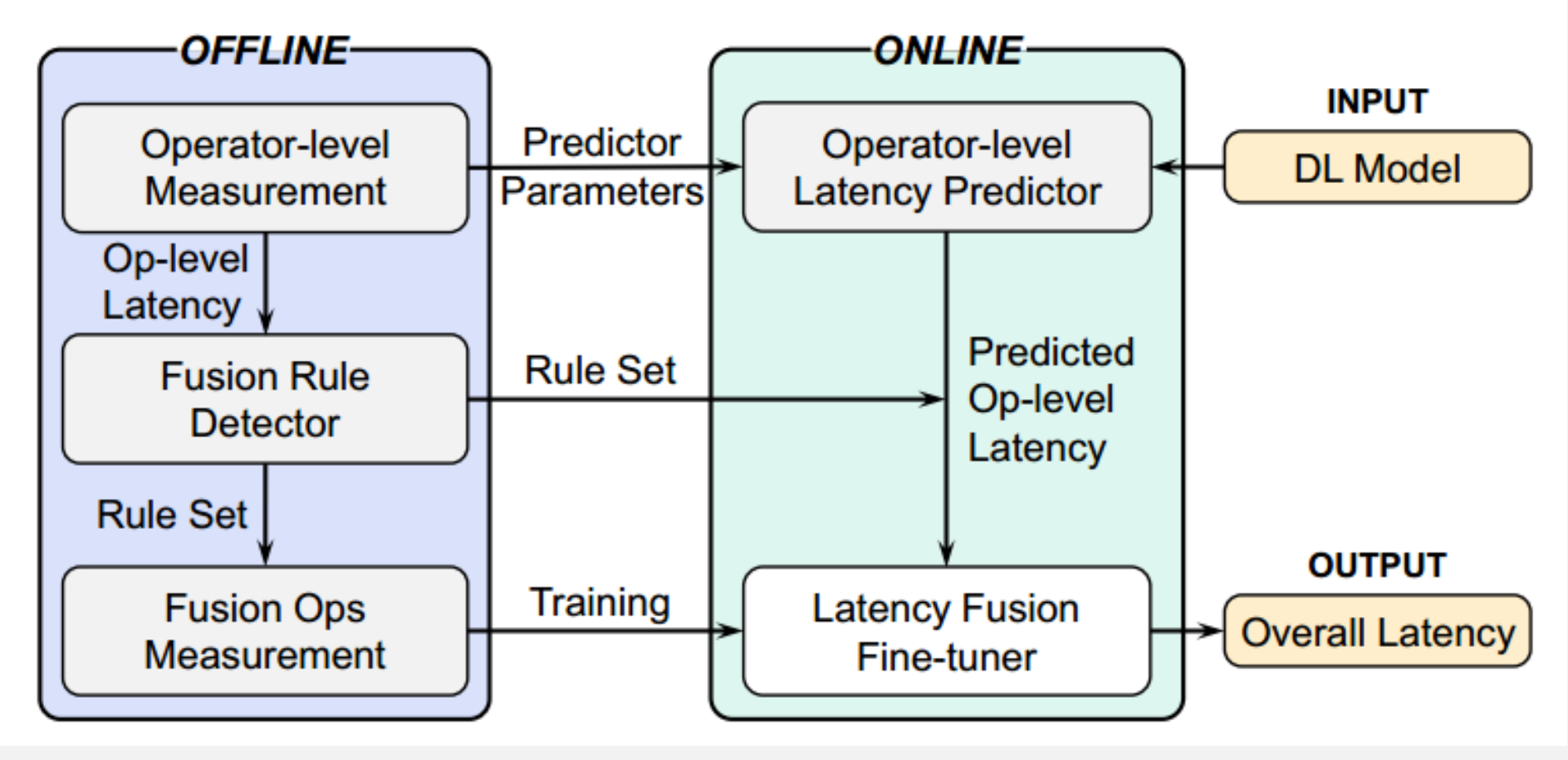
Overview of ASAP design

ELASTIC EFFICIENT SCHEDULER

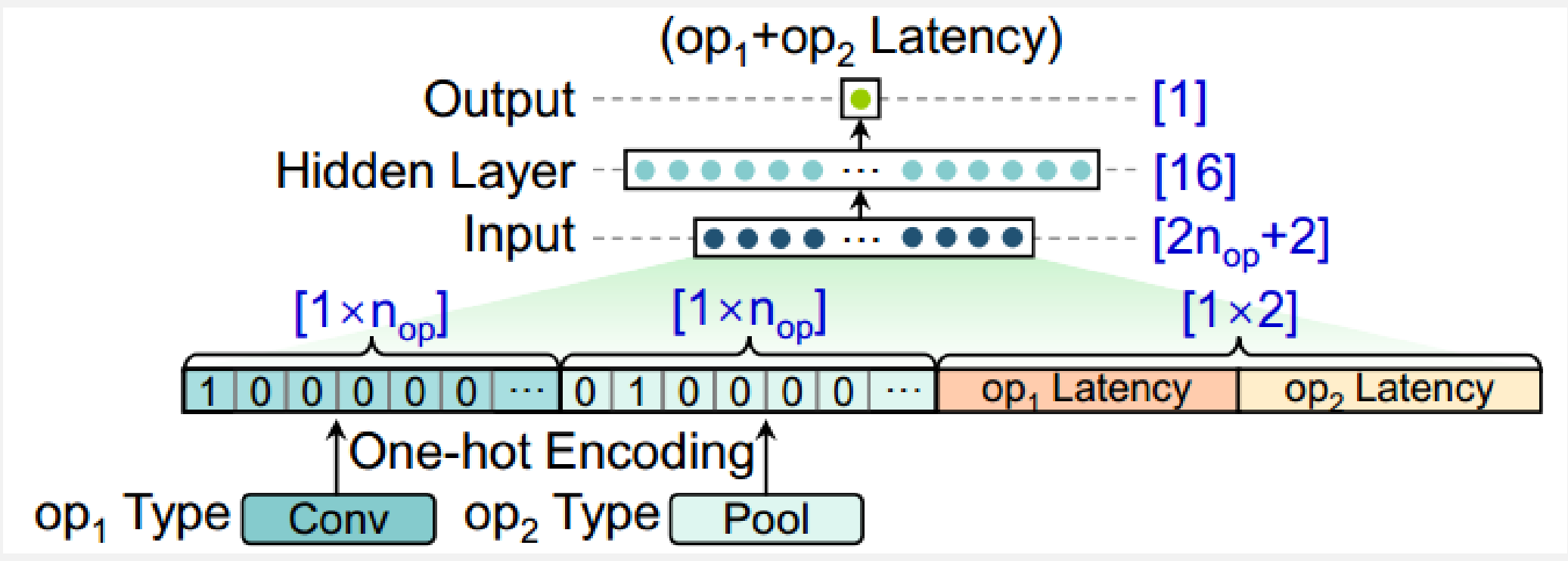
The scheduler mainly consists of three key components: *cluster external load balancer, cluster internal load balancer, and an elastic scheduling module*. First, the cluster external load balancer partitions the entire model into submodels according to the capability of clusters in the swarm, which avoids the collaborative efficiency being encumbered by the slowest UAV. Second, the cluster internal load balancer is responsible for generating data partition schemes for UAVs in the same cluster, which reduces the efficiency reduction induced by data synchronization. Finally, the elastic scheduling module conducts some mechanisms to deal with unexpected situations, which controls the scheduler to reschedule tasks for each UAV.

Lightweight Accurate DL Inference Performance Predictor

The proposed predictor decomposes DL inference latency prediction into *operator-level prediction and latency fusion fine-tuning*. The operator-level prediction is based on the relation between latency performance and hyperparameters of each operator, and the latency fusion fine-tuning uses a very small model to learn the fusion rules of DL frameworks. The architecture of the predictor has two phases: offline phase and online phase. In the offline phase, the inference latency of operators is measured to generate parameters of the operator-level latency predictor, and the operator fusion rule set is obtained from the latency of operator pairs. The latency of operators is collected to train the latency fusion fine-tuner. In the online phase, the latency fusion fine-tuner adjusts the operator-level latency of operators and adds them up to get the DL model latency.



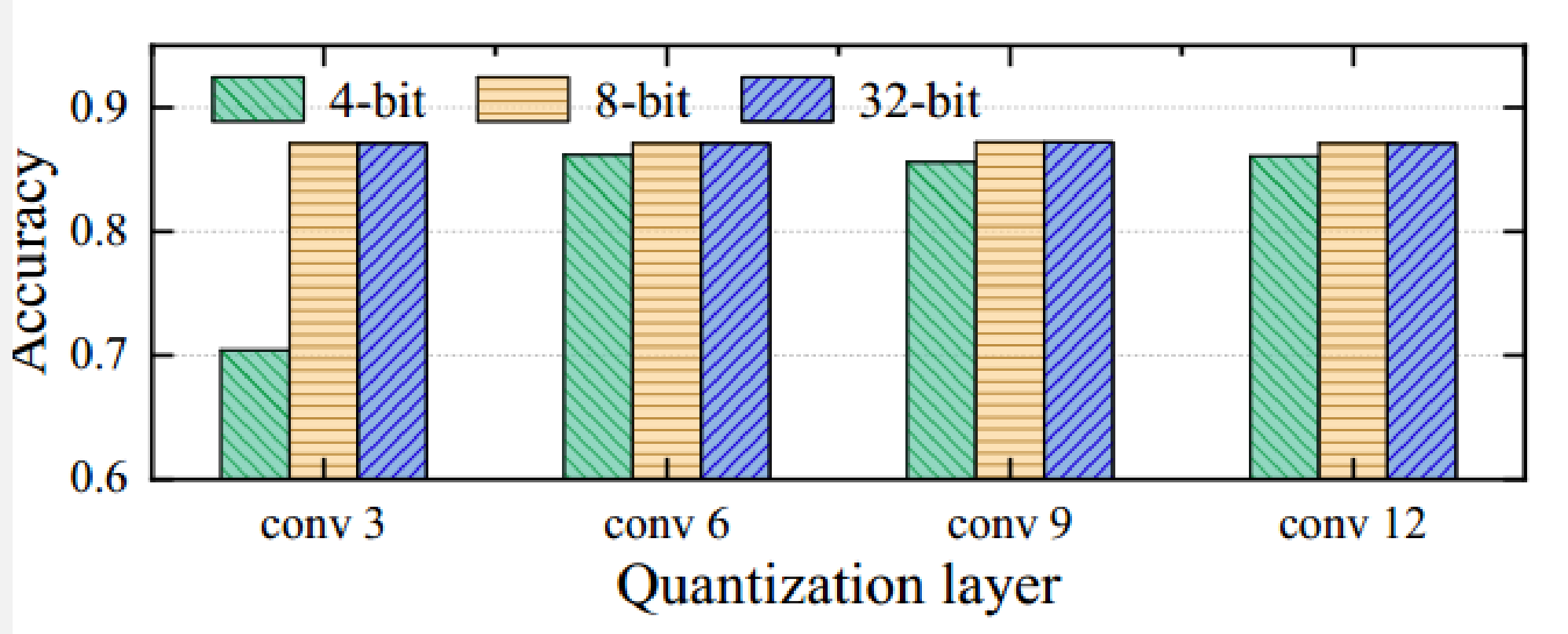
Architecture of lightweight accurate DL inference performance predictor



The training of latency fusion fine-tuner

Adaptive Inter-UAV Data Compressor

The adaptive data compressor *conducts 8-bit quantization and compression algorithms* (e.g., gzip, LZ4, Bzip2.) on the intermediate data transmitted between UAVs to decrease communication overhead. Instead of a fixed quantization scale, the adaptive data compressor changes compression degree according to data size and communication rate between UAVs, which saves communication resources and guarantees the accuracy of tasks at the same time.

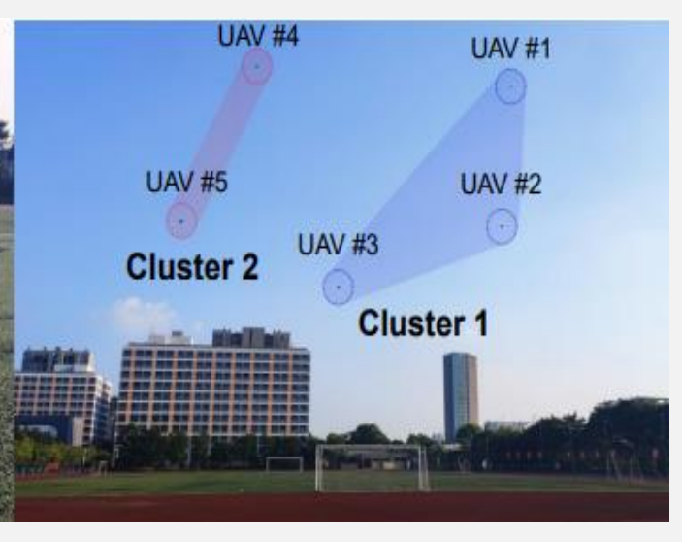


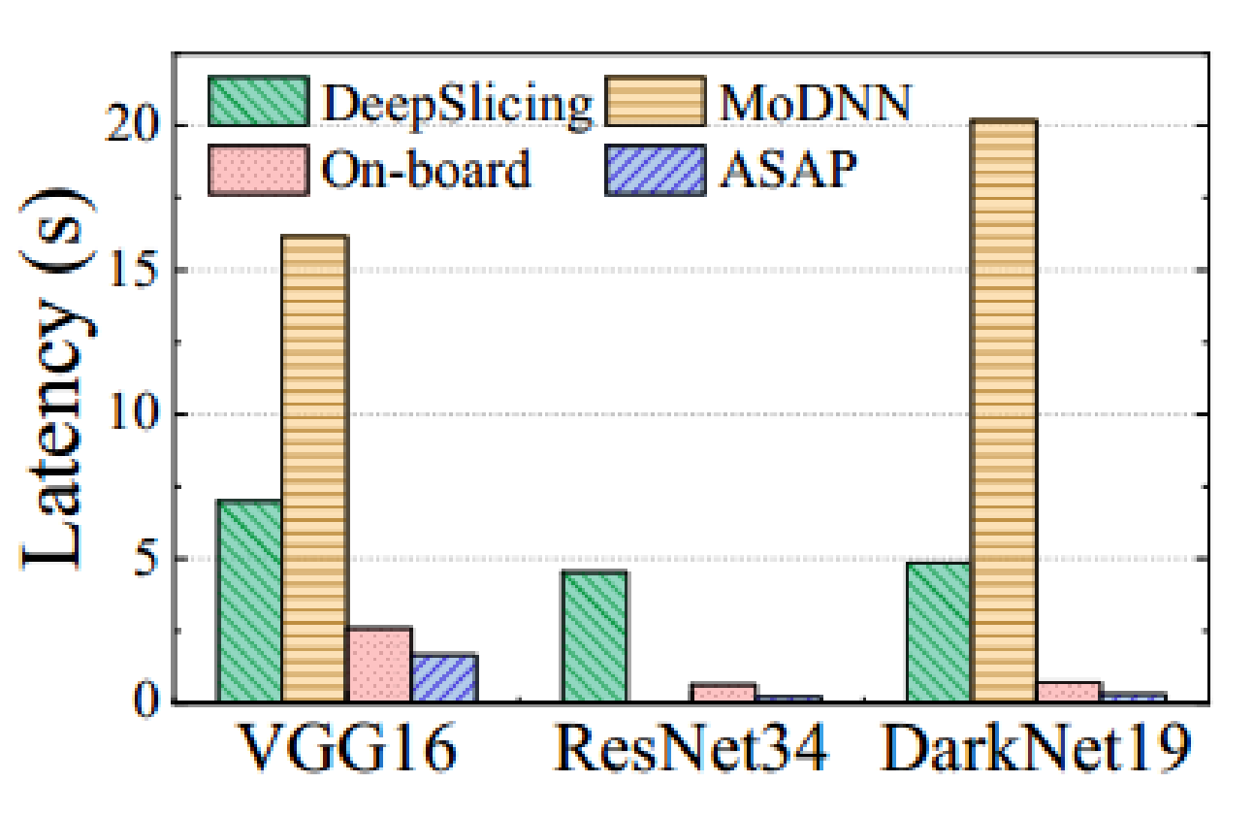
Inference accuracy under different quantization levels of VGG16

Overall Performance

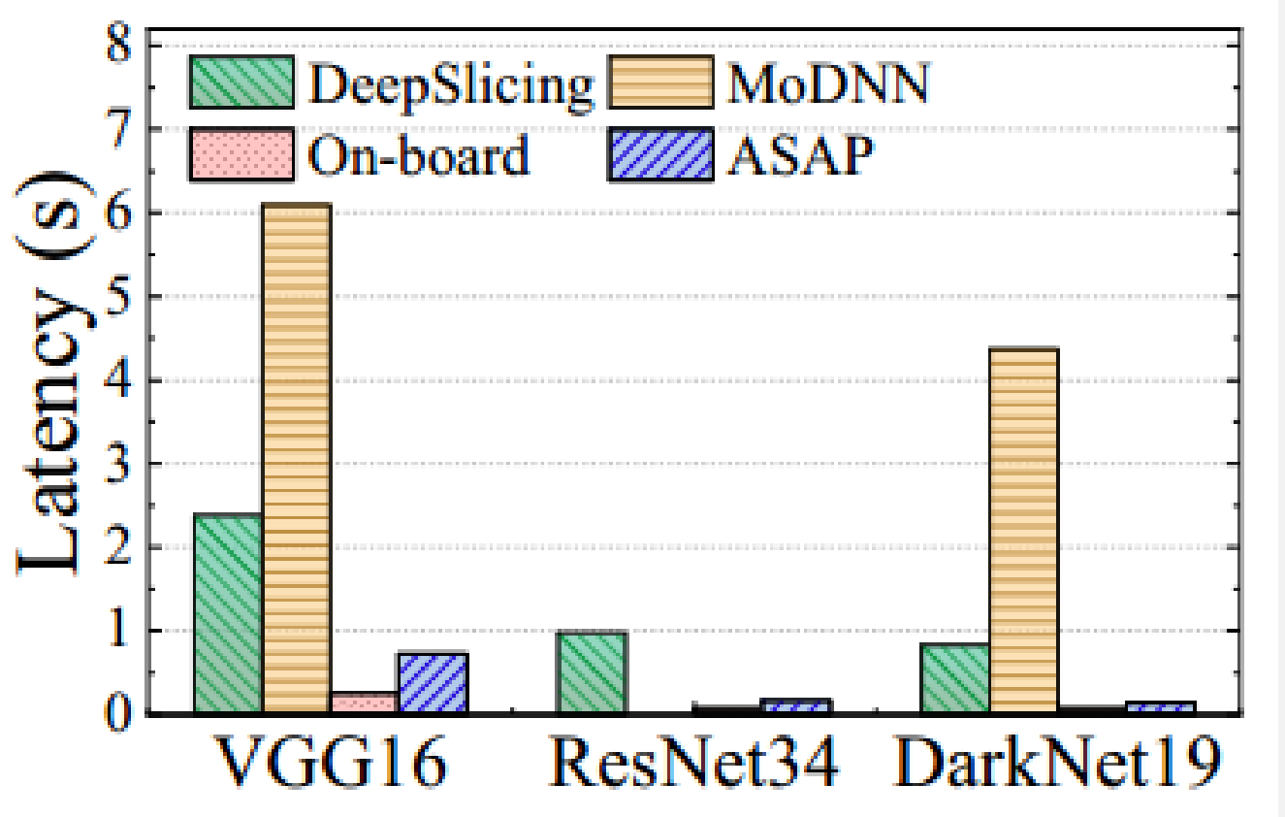
ASAP vs. Existing terrestrial collaborative edge computing. We compares the average computing latency of ASAP with DeepSlicing and MoDNN under three medium DNN models. Note that Deepslicing and MoDNN are designed for terrestrial IoT systems and not suitable for UAV swarm, we enable communication between all nodes for the two systems. Our system keeps the computing latency at a low level in contrast with two baseline systems in both scenarios.

ASAP vs. On-board computing. when it comes to the system overhead, ASAP reduces the computing and memory overhead per node by up to 90.56% and 90.02%, respectively, which greatly releases the burden of a single device. Although DeepSlicing and MoDNN show worse latency performance compared with on-board computing, it does not mean that they do not have latency advantages in any situation. In detail, they are more suitable for very weak devices, where the computing latency is much higher than transmission latency, not for the GPU enabled devices in our experiment, and ASAP can achieve more performance gain on these devices.

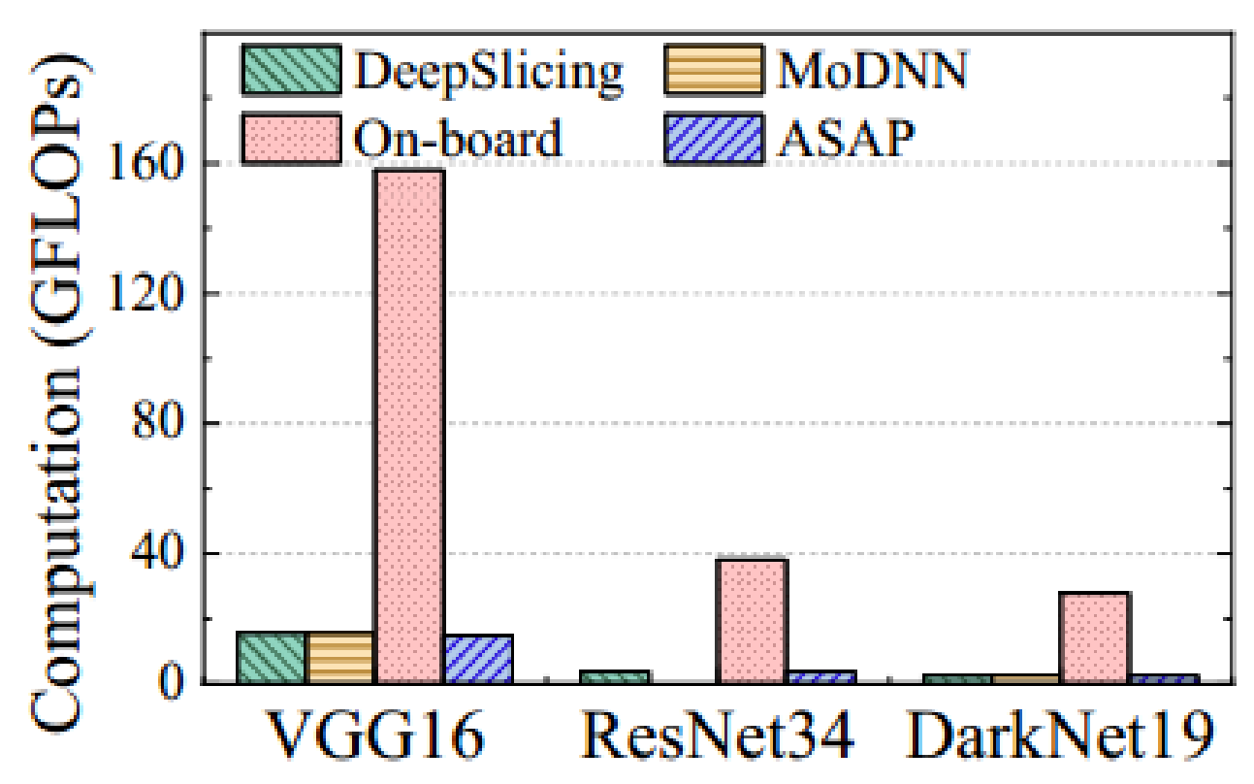




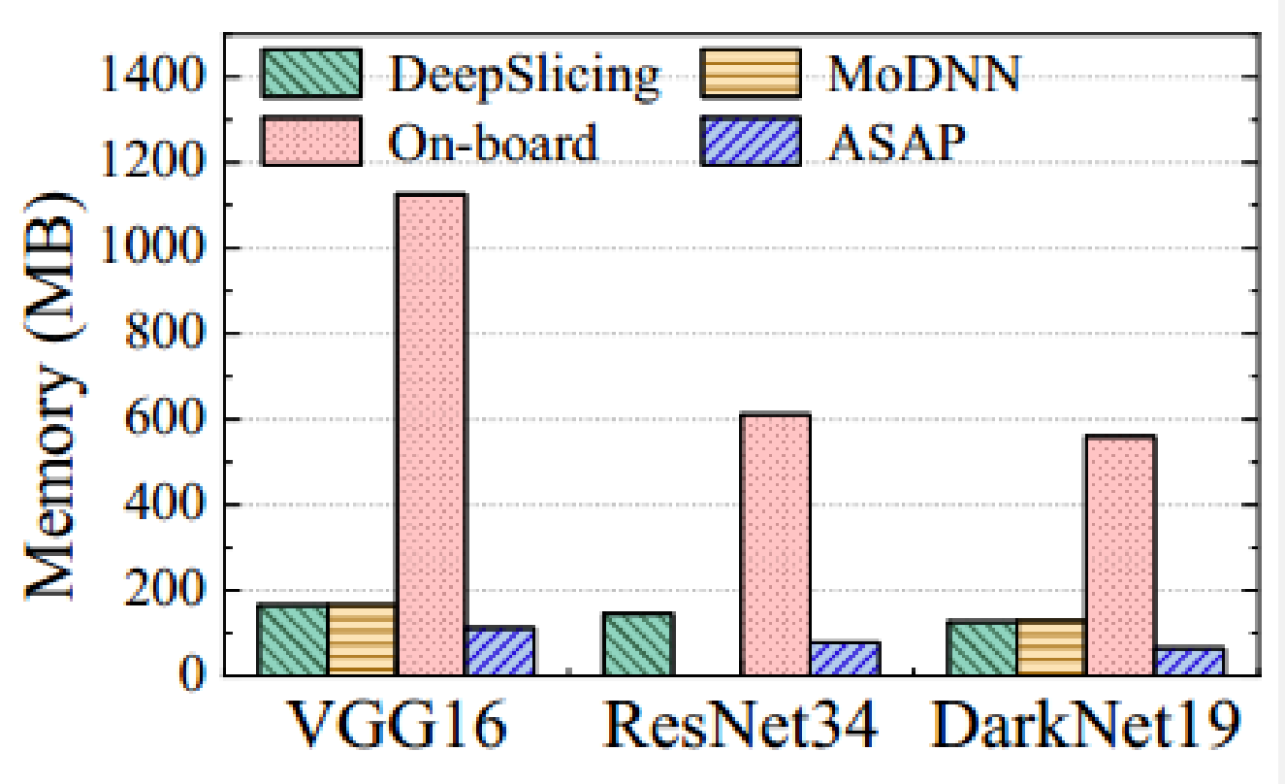
11 UAV airborne computers



5 real-world UAVs



Computing overhead



Memory overhead