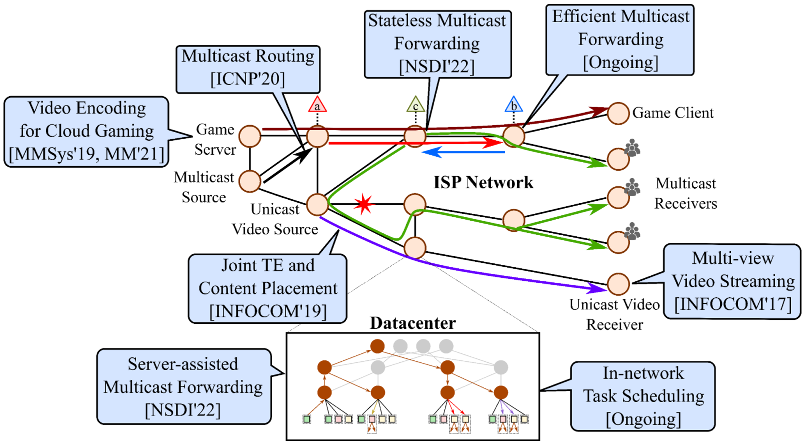
Research Statement

Khaled Diab
kdiab@sfu.ca

My research interests span computer networking, systems, and multimedia. I collaborate with multiple co-PIs and research assistants in various projects in the Network and Multimedia Systems Lab at SFU. I currently lead (or led) the ***Multicast Systems, Datacenter Networking,*** and ***Multimedia Systems*** research programs, and co-lead the ***Cloud Gaming*** and ***Hyperspectral Imaging*** projects. My work has been published in several top-tier venues including USENIX NSDI, IEEE ICNP, IEEE INFOCOM, ACM Multimedia, and ACM Multimedia Systems. In addition, I co-trained multiple High Quality Personnel: I advised two thesis-based MSc students, and I have been working closely with other three PhD candidates. I also collaborate with AMD and Huawei on some of my ongoing research. I describe in the following a high-level overview of my recent and future research.

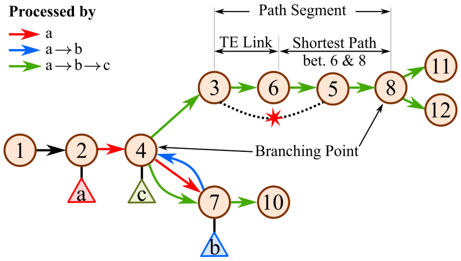
Recent Research

Multicast Systems in ISP Networks. Despite its potential significant bandwidth savings, major network providers have not yet unlocked the full benefits of multicast. This is due to the lack of *efficient* and *scalable* multicast primitives that *(i)* support various routing policies such as traffic engineering and service chaining, where a packet needs to pass through an ordered set of network services *before* reaching the receivers, and *(ii)* forward packets of large numbers of *concurrent* multicast sessions.


Multicast Routing [ICNP'20]
Stateless Multicast Forwarding [NSDI'22]
Efficient Multicast Forwarding [Ongoing]
Video Encoding for Cloud Gaming [MMSys'19, MM'21]
a
c
b
Game Server
Game Client
Multicast Source
ISP Network
Multicast Receivers
Unicast Video Source
Joint TE and Content Placement [INFOCOM'19]
Multi-view Video Streaming [INFOCOM'17]
Unicast Video Receiver
Datacenter
Server-assisted Multicast Forwarding [NSDI'22]
In-network Task Scheduling [Ongoing]

My research laid the foundations of the efficient forwarding and routing of packets of *generalized multicast sessions* that may need to be processed by an ordered set of network services and/or directed through a specific set of network paths within the ISP network. In contrast to supporting simple multicast trees, we enable the support of *multicast distribution graphs* [1], [2]. The key contributions are calculating efficient multicast routes, and addressing the scalability challenges of multicast forwarding by reducing or eliminating the required state to be maintained at routers.

First, we proposed a new system, called Oktopus [1], that supports various traffic engineering and service chaining policies. Using our system, a network operator can build efficient multicast graphs, and route packets through links of these graphs. This is a challenging task due to the large search space. Our key idea is to pre-generate candidate network segments, and efficiently use them to calculate graphs without exceeding the capacity constraints. Our results show that Oktopis is close to the optimal solution, and it increases the number of concurrent sessions by up to 37% compared to the state-of-art algorithm.

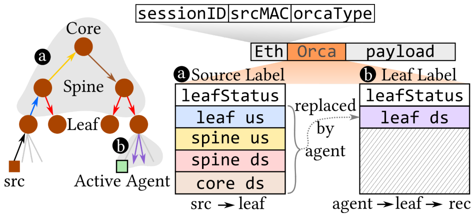


Second, we designed two systems that forward multicast packets on *arbitrary* links in the network while reducing or even eliminating the maintained state at switches. This is critical for ISPs to handle their traffic engineering and service chaining requirements. The first system, Yeti [2], *completely* moves the forwarding state to packets of the session as labels. Designing and processing such labels, however, pose key challenges that need to be addressed. First, we need to efficiently encode the graph forwarding information in as few labels as possible. Second, the processing overheads and hardware usage at routers need to be minimized. This is to support many concurrent multicast sessions, and to ensure the scalability of the data plane. Third, forwarding packets should not introduce ambiguity at routers. That is, while minimizing label redundancy and overheads, we must guarantee that routers will forward packets *on and only on* the links of the multicast graph. Our results show that unlike Yeti which does not maintain state at any core router, previous approaches maintain state at 20–100% of the routers. Yeti reduces the label overhead by an average of 65.3% compared to the state-of-art label-based system.

The second system, Helix [4], splits the forwarding state between switches and labels attached to packets. The key idea is to enable the operator to balance between the label and state overheads in a principled approach. Helix uses probabilistic set-membership data structures to encode links of multicast sessions in *fixed-size* labels, and applies a novel recursive encoding algorithm to further reduce the required state. We implemented Yeti and Helix in testbeds using programmable network devices such as NetFPGA and Intel Tofino, and assessed their performance in large-scale simulations.

Datacenter Networking. The current focus of my research is to design various *networking primitives* to efficiently manage the datacenter network resources. Specifically, I have been designing systems that efficiently forward multicast traffic, and quickly schedule tasks *within* the network.

We present a novel architecture, named Orca [3], to realize efficient multicast forwarding that can support millions of concurrent multicast sessions in datacenter networks [15], [19]. The idea of our system is to offload some of the state maintained at network switches to end servers (or agents). To achieve this idea, our system computes *fixed-size* and *compact* labels and attaches them to packets of multicast sessions. These labels effectively enable shifting some of switches' tasks to servers. As a result, Orca significantly reduces the state at switches, minimizes the bandwidth overhead, incurs small and constant processing overhead, does not limit the sizes of multicast sessions, and eliminates redundant traffic.

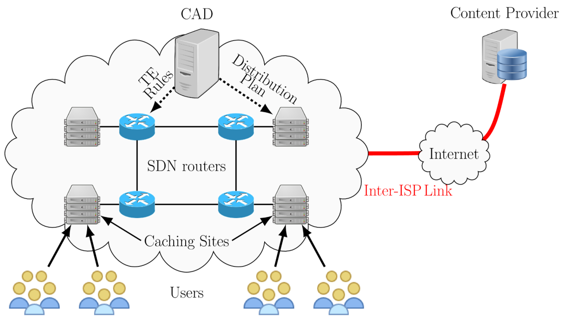


My second research direction is to enable the efficient and fast scheduling of μ-scale tasks. For these short-running tasks, application-layer schedulers incur high latency overheads and fail to scale to tens of millions of tasks per second. Our insight is that programmable data planes enable offloading various applications to the network to achieve higher throughputs and lower latencies [20]. We designed Horus [7], a new *in-network scheduler* that enables efficient *datacenter-wide* scheduling of short tasks. Horus efficiently tracks and distributes the worker state at leaf and spine layers, which enables the switches to schedule tasks at line-rate while making near-optimal decisions. We propose a new distributed scheduling policy that minimizes the state and communication overheads by adaptively utilizing information about idle workers and load values. Our results show that Horus can reduce the tail response time by up to 85%, increase the throughput by up to 2.17X, and reduce the processing load on switches by up to 2.5X, compared to the state-of-the-art.

Multimedia Systems. Streaming videos over the Internet is a multi-billion-dollar industry and it is rapidly growing. This is due to the proliferation of new types of multimedia content such as 4K videos and immersive content such as multiview and 360-degree videos. These new multimedia contents are becoming popular because they offer high-quality and unprecedented experiences to users. For example, multiview videos allow users to explore scenes from different angles and perspectives, whereas 360-degree videos put users in virtual scenes that can be navigated. As the popularity of immersive videos increases, the amount of multimedia traffic distributed over the Internet increases at high rates. Thus, major ISPs need to efficiently carry and manage larger volumes of multimedia traffic through their networks.

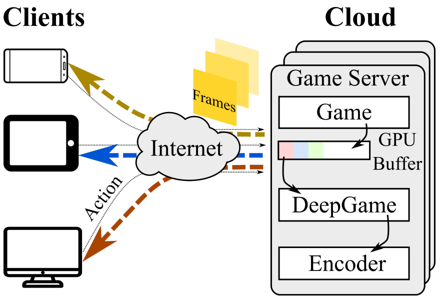
My recent work focused on two aspects of multimedia systems: *(i)* the adaptive streaming of multiview videos, and *(ii)* the traffic engineering and content placement of multimedia content.

We proposed the Multiview Adaptive Streaming over HTTP (MASH) algorithm [5]. MASH introduces a new perspective to the rate adaptation problem in multiview video streaming systems: it constructs probabilistic view switching models and utilizes these models in dynamically selecting segments of various views at different qualities, such that the quality of the videos are maximized while the network bandwidth is not wasted. We conducted extensive empirical study to compare MASH versus the algorithm used by YouTube for multiview videos. The results show substantial improvements across multiple performance metrics and in different network conditions. For example, MASH achieves up to 3X improvement in the average quality and up to 2X improvement in the prefetching efficiency compared to the YouTube algorithm.



Next, we designed the CAD (Cooperative Active Distribution) system [6] to serve user requests from the emerging telco-CDNs. CAD allocates the storage and processing resources at each caching site to the different video representations as well as directs the traffic flows of the video sessions through the ISP network such that the cost incurred by the telco-CDN is minimized (in terms of the amount of inter-ISP traffic) and the user-perceived quality is optimized (in terms of end-to-end latency). Managing telco-CDNs is, however, a complex task, because it requires concurrently managing resources at the network layer (TE) and system layer (processing and storage capacities of caches), while supporting the adaptive nature and skewed popularity of multimedia content. We proposed an algorithm to efficiently solve the resource management problem in telco-CDNs. Compared to the closest system in literature, CAD achieves up to 64% reduction in the inter-domain traffic and 14% improvement in the quality of experience for users.

Cloud Gaming. Cloud gaming offloads the main components of games from clients/consoles to cloud servers. This leads to several advantages such as facilitating new game updates, avoiding potential piracy, and reducing the computation and storage resources at the clients. The cloud server executes the game logic, renders the game, and encodes and transmits the frames to end users. This introduces fundamental tradeoffs between latency, bandwidth consumption, and perceived quality. Our objective is to reduce the bandwidth usage of cloud gaming while achieving a high perceived quality. The main idea of our research is to create computational models to understand how players interact with various game genres, and encode various regions in the frame proportional to their *importance* to the player. We designed and implemented CAVE [8] that encodes the regions of interest differentially given that these regions are given to our system. Although CAVE is bandwidth-efficient, it requires slight modifications to the source code of existing games. Thus, we proposed a learning-based approach, called DeepGame [9], to automatically understand the spatial and temporal context of a game, *without* any modification to the game or encoder. We implemented our learning approach in a real testbed, and showed that it reduces the required bandwidth by up to 36%, achieves high quality, and imposes negligible delays.



Hyperspectral Imaging. Unlike traditional RGB imaging systems that capture information in the visible electromagnetic spectrum, hyperspectral imaging systems capture information in the visible and near infrared spectrum. These cameras include special hardware such as narrow-band filters to capture the information across many bands. Thus, these systems can produce unique fingerprints (*aka* spectral signatures) for different materials and objects. These fingerprints enable a new range of applications such as food quality monitoring and fraud detection. However, these cameras are expensive, time consuming, heavy, and hard to deploy. Our current research enables the wide adoption of hyperspectral imaging on mobile devices. We propose a new system [10] that turns regular mobile devices into hyperspectral cameras. Our idea is to design a new camera sensor, and reconstruct the required hyperspectral bands by employing a learning-based approach.

Future Research

My research vision for the next 1–2 years is to design *network-assisted datacenters* by taking advantage of programmable network devices such as switches and SmartNICs. That is, my research will focus on deciding *(i)* what tasks to be offloaded to the network, and *(ii)* how these tasks can be offloaded efficiently. In contrast to the current Infrastructure Processing Units (IPUs), my research shall go beyond these chips to support a wider range of infrastructure operations. In the following, I describe a high-level overview of two of these potential infrastructure operations.

In-network Service Chaining. Running chained services is critical to the security and operations of datacenter applications [18], especially with the emerging trend of network micro-segmentation [21]. With the near end of network

scaling [11] due to DRAM latencies, the current network function virtualization (NFV) platforms cannot support interfaces beyond 100 GbE without significantly increasing the number of CPU cores [18]. To overcome this potential limitation, my research will offload the execution and dynamic scaling of *chained network services* to programmable network devices that can process packets at high speeds. This can lead to significant performance gains such as running stateful services at Tbps rates, reducing end-to-end processing delays, and relaxing the load on CPUs. However, the limited processing resources and lack of full flexibility of these devices pose severe challenges to realize this idea.
I will focus on two research directions: chain synthesis and chain placement. In the first direction, I will design a system that synthesizes a single “Big Function” that matches the behaviour of the service chain. This eliminates redundant operations and reduces processing overheads by grouping read and write instructions. Few prior systems attempted to achieve efficient service chaining by splitting services among CPUs and switches [12], [13], or constructing a pipeline of services in individual switches [14]. These systems, however, do not *(i)* handle the stateful nature of services, *(ii)* automatically support all synthesized chains (even if they exist), *(iii)* verify the correctness of the synthesized chains given the input rulesets, or *(iv)* consider the fine-grained operations of services. Similar to a compiler in an operating system that produces architecture-dependent artifacts, my vision is to create a *service compiler* that is aware of the available programmable network devices and their resources and limitations, while addressing the aforementioned challenges. Given a running synthesized chain, the second direction will focus on the dynamic placement, migration, and scaling of chained network services. The key challenges are to *(i)* build a fresh view of the distributed services, *(ii)* manage the state of chained services [22], and *(iii)* ensure the correctness of the updated chain.

In-network Scheduling of Granular Applications. Although Horus [7] can schedule μ-scale tasks efficiently, the next-generation granular applications [15], [16] run a combination of parallel and sequential tasks, exhibit bursty patterns, and suffer from cold-start latency [16], [17]. Scheduling these applications without considering their nature may reduce the resource utilization of the datacenter and increase the end-to-end delay. My research will focus on designing a new in-network scheduler that handles tasks dependencies, addresses the challenges of data locality, and considers multiple resource usage requirements such processing, networking, and storage.

Industrial Collaboration and Service

Industrial Collaboration. I have been collaborating with AMD and Huawei on the cloud gaming and hyperspectral imaging projects, respectively. In particular, I co-manage these projects with the SFU co-PI, Prof. Hefeeda, to train HQPs, design new systems, and propose efficient algorithms. For example, our collaboration with AMD has resulted so far in a MSc thesis and two publications, where one of them [8] had received the Best Student Paper Award.

Service. I have been an active member in the research community. I was a member of the technical program committee of IEEE INFOCOM (2021 and 2022) and ACM NOSSDAV (2021). I was awarded as a **Distinguished Member** of the 2022 IEEE INFOCOM TPC. I am regularly invited to be an external reviewer for ACM Multimedia, ACM Multimedia Systems, and ACM Multimedia Asia. I frequently review articles in the following transactions/journals: ACM TOMM, IEEE/ACM ToN, IEEE TNSM, and IEEE TMM. At SFU, I am a committee member of the Master of Science in Professional Computer Science (MPCS) program. My main tasks include co-designing the new cybersecurity concentration, working with our industrial partners, and reviewing MSc applications per year.

References

[1] **K. Diab**, C. Lee, and M. Hefeeda. *Oktopus: Service Chaining for Multicast Traffic*. In Proc. of IEEE ICNP'20

[2] **K. Diab** and M. Hefeeda. *Yeti: Stateless and Generalized Multicast Forwarding*. In Proc. of USENIX NSDI'22

[3] **K. Diab**, P. Yassini, and M. Hefeeda. *Orca: Server-assisted Multicast for Datacenter Networks*. In Proc. of USENIX NSDI'22

[4] **K. Diab** and M. Hefeeda. *Efficient Multicast Forwarding* (To be submitted)

[5] **K. Diab** and M. Hefeeda. *MASH: A rate adaptation algorithm for multiview video streaming over HTTP*. In Proc. of IEEE INFOCOM'17

[6] **K. Diab** and M. Hefeeda. *Joint Content Distribution and Traffic Engineering of Adaptive Videos in Telco-CDNs*. In Proc. of IEEE INFOCOM'19

[7] P. Yassini, **K. Diab**, and M. Hefeeda. *Horus: Granular In-Network Task Scheduler for Datacenters* (Under Submission)

[8] M. Hegazy, **K. Diab**, M. Saeedi, B. Ivanovic, I. Amer, Y. Liu, G. Sines, and M. Hefeeda. *Content-aware Video Encoding for Cloud Gaming*. In Proc. of ACM MMSys'19. [Best Student Paper Award]

[9] O. Mossad, **K. Diab**, I. Amir, and M. Hefeeda. *DeepGame: Ecient Video Encoding for Cloud Gaming*. In Proc. of ACM MM'21

[10] N. Sharma, P. Moghadam, **K. Diab**, and M. Hefeeda. *MobiSpectral: Turning Mobile Devices into Hyperspectral Cameras* (To be submitted)

[11] S. Thomas, R. McGuinness, G. M. Voelker, and G. Porter. *Dark Packets and the end of Network Scaling*. In Proc. of ACM ANCS'18

[12] G. P. Katsikas, T. Barbette, D. Kostic, R. Steinert, and G. Q. M. Jr. *Metron: NFV Service Chains at the True Speed of the Underlying Hardware*. In Proc. of USENIX NSDI'18

[13] G. P. Katsikas, T. Barbette, D. Kostić, JR. G. Q. Maguire, and R. Steinert. *Metron: High-performance NFV Service Chaining Even in the Presence of Blackboxes*. ACM Trans. Comput. Syst., Jul. 2021

[14] D. Wu, A. Chen, T. S. E. Ng, G. Wang, and H. Wang. *Accelerated Service Chaining on a Single Switch ASIC*. In Proc. of ACM HotNets'19

[15] C. Lee and J. Ousterhout. *Granular Computing*. In Proc. of ACM HotOS'19

[16] L. Ao, L. Izhikevich, G. M. Voelker, and G. Porter. *Sprocket: A Serverless Video Processing Framework*. In Proc. ACM SoCC'18

[17] K. Kaffes, N. J. Yadwadkar, and C. Kozyrakis. *Centralized Core-granular Scheduling for Serverless Functions*. In Proc. of ACM SoCC'19

[18] P. Zave, F. B. Carvalho, R. A. Ferreira, J. Rexford, M. Morimoto, and X. K. Zou. *A Verified Session Protocol for Dynamic Service Chaining*. IEEE/ACM Transactions on Networking, Feb. 2021

[19] M. Li, D. G. Andersen, J. W. Park, A. J. Smola, A. Ahmed, V. Josifovski, J. Long, E. J. Shekita, and B. Y. Su. *Scaling distributed machine learning with the parameter server*. In Proc. of USENIX OSDI'14

[20] J. Li, E. Michael, and D. R.K. Ports. Eris: Coordination-free consistent transactions using in-network concurrency control. In Proc. of SOSP'17.

[21] A. Bremler-Barr, Y. Harchol, and D. Hay. *OpenBox: A Software-Defined Framework for Developing, Deploying, and Managing Network Functions*. In Proc. of ACM SIGCOMM'16

[22] D. Kim, J. Nelson, D. R. K. Ports, V. Sekar, and S. Seshan. *RedPlane: enabling fault-tolerant stateful in-switch applications*. In Proc. of ACM SIGCOMM'21