

PHD CANDIDATE, COMPUTER SCIENCE, THE UNIVERSITY OF HONG KONG

教育背景

南方科技大学 (SUSTech)

通信工程 - 学士

2014 - 2018

专业 GPA: 3.77/4.00 排名: 3/30.

香港大学 (HKU)

计算机科学 - 博士

2018 - 2022 (预计)

研究方向

边缘计算, 面向应用 (Application-centric) 的网络优化, 马尔科夫决策过程 (MDP) 应用。

科研技能

核心课程: 凸优化, 多项式优化, 随机过程; 无线通信, 信息论。

编程语言: Python, Rust, C/C++, Nodejs, TypeScript, and LabVIEW。

仿真工具: Numba + SciPy + Numpy, PyTorch/TensorFlow。

个人成就

Awarded **Second-class Scholarship and “Excellent Student”** from 2016 to 2017.

Awarded **Best Track Paper of Algorithm Track** in IEEE MSN, 2020.

代表论文

[1] **Y. Hong**, B. Lv, R. Wang, H. Tan, Z. Han, H. Zhou, F. C. M. Lau, “Online Distributed Job Dispatching with Outdated and Partially-Observable Information,” in *Proc. IEEE MSN, 2020 Best Track Paper Award*.

[2] **Y. Hong**, B. Lv, R. Wang, H. Tan, Z. Han, F. C. M. Lau, “Distributed Job Dispatching in Edge Computing Networks with Random Transmission Latency: A Low-Complexity POMDP Approach,” in *IEEE Internet of Things Journal*.

[3] L. Zhou, **Y. Hong**, S. Wang, R. Han, D. Li, R. Wang, and Q. Hao, “Learning centric wireless resource allocation for edge computing: Algorithm and experiment,” *IEEE Transactions on Vehicular Technology*, Jan. 2021.

学术研究

基于部分与过时信息 (Outdated-and-Partial Information) 的边缘计算任务调度

指导导师: 王锐, 谈海生, *Prof. Francis C. M. Lau*.

Nov '18 - Feb '21

- 在这个课题中, 我们将边缘计算系统中, 具有过时和部分信息的分布式任务调度问题, 表述为一个 POMDP 问题。
- 为了解决 MDP 求解过程中的维度爆炸 (curse of dimensionality) 问题, 我们提出了一个低复杂度的解决方案, 替代策略迭代 (alternative policy iteration), 并得出了解析性能下限和一个更严格的半解析下限。
- 基于 Google Cluster 的公开数据集, 我们进行了大量的仿真并将我们的方法和三个启发式算法进行比较。结果显示, 我们提出的算法可以实现减少高达 20.67% 的平均响应时间, 并且在信息滞后性的各种参数设置下始终表现良好。

在移动车辆上部署用于 3D 物体检测的在线联邦学习

指导导师：王锐，*Prof. Francis C. M. Lau*。

Nov '19 - 现在

- 在这个课题中，我们首先给出了一个场景，多个配备 SECOND 网络的车辆将会合作收集道路上的 3D 物体信息到边缘服务器。
- 在这个应用场景下，我们开发了一个半监督（semi-supervised）的联合学习框架，边缘服务器上的融合去噪的物体信息返回给车辆，车辆可以通过这种知识提炼（Knowledge Distillation）的方法进一步提高模型精度。
- 我们在高保真自动驾驶模拟器 CARLA 的基础上进行了实验，并生成了符合 KITTI 数据集格式的数据集。源代码可以在 Github 上找到 [Github](#)。
- 在后续的扩展工作中，我们提出了上述场景下的资源分配问题，以实现在线联邦学习中的通信和计算效率最大化。

项目经历

Linux 内核中网络堆栈的实践

Oct '16 - Oct '18

- **MAC Layer (L2)**: ([Github Link](#)) 在这个项目中，我们提供了对于无线网卡驱动控制更快、更可靠的访问。我们设计了一个内核模块来劫持驱动程序的函数入口，并使用“mmap”来建立内核和用户空间之间的共享内存通信。实验结果表明，对于一组 **IEEE 802.11e** 参数（与信道访问优先级有关）的 1000 次操作，我们可以在 10ms 内完成。
- **IP Layer (L3)**: ([Github Link](#)) 该项目实现了基于 Linux Netfilter 子系统的网络数据加解密试验（类似于 IPsec）。它使用 Linux 内核提供的异步加密方法 AES-128，对 L3 的有效载荷（即 IP 数据包的内容）进行加密，并在接收方进行相应解密。这个项目的扩展工作中会实现一个识别机制，允许 Tx/Rx 协商启用加解密，以及添加基于 Linux Netlink 子系统的用户空间工具来管理密钥。

VLC-WiFi 融合通信平台

Sep '17 - Dec '18

在这个项目中，我们实现了一个混合 VLC-WiFi 通信系统，其中 VLC 链接承担高吞吐量的下行链路，Wi-Fi 链接服务器作为上行链路。我们基于 **NI 仪器** 和 USB Wi-Fi 网卡实现了该系统，其中重传机制通过 IP 层实现。系统实现的源代码在 [Github](#) 上提供。该平台上还拥有一项授权专利（CN110429979B）产出。

基于 SerDe 机制的 IPC (Inter-Process Communication) 框架

Apr '21 - Present

- 该 IPC 框架使用 Rust 语言编写，源代码可以在 [Github](#) 上找到。其中“SerDe”是“serializing-and-deserializing”的意思。
- 该 IPC 借鉴了来自 *Android Binder* 和 *gRPC* 的优秀实践，旨在提供对于**任何语言中任意函数签名的统一 FFI 访问**（需要有稳定的 ABI），目前仅支持 Rust/CPP/Python 的服务。
- 该 IPC 框架采用 C/S 模型设计，并且后端是模块化的。理论上它可以支持不同的 **IPC 协议**（例如 socket, binder）和不同的 **SerDe 机制**（例如 json, ProtoBuf）。