

# Hsiang-Jui (Jerry) Lin

## Academic Record

- 2016** *National Taiwan University, Bachelor of Computer Science and Information Engineering* + Undergraduate thesis: *Solving graph domination problem on DH graph using split decomposition*
- 2021** *National Taiwan University, Master of Computer Science and Information Engineering*
- 2021** *National Taiwan University, PhD program in Computer Science and Information Engineering* under the supervision of Professor Chi-Sheng Shih. The research focuses on autonomous vehicles, multi-modal sensor fusion and vehicle-to-everything communication.

## Work Experience

- 2010 - 2016** **Software Developer at CAVEDU Education (CAVEDU 教育團隊)**  
Developer of ApplInventor project, Authored LeJOS on LEGO NXT book.
- 2018 Jun. - 2019 Jun.** **R&D Assistant at Microsoft Taiwan**  
Bing's Reverse Geocoder backend development
- 2021 Sep. - 2024 May.** **R&D Assistant at ADLINK Technology Inc.**  
Develop distributed consensus and traffic control using Zenoh

## Skills

### Advanced System Programming

Know C/C++, Rust, CUDA and OpenCL. Can implement advanced data structures like [ziptree](https://github.com/jerry73204/ziptree-rs) (<https://github.com/jerry73204/ziptree-rs>). Experienced in building parallel and concurrent programs.

### Machine Learning

Experienced in PyTorch, TensorFlow and several visualization tools. Worked on a deep learning project in my thesis.

### Computer Security and Cryptography

Studied cryptography mathematics and practical computer security lessons during college, and was ranked #2 as a hacker in National 金盾獎 competition..

### Digital Circuits and Robotics

Ever worked as a Maker promoter in CAVEDU and learned 3D printing and circuit design. The [hacky-arm] project is the work in CAVEDU.

## Publications

### (2013) 機器人程式設計與實作：使用 Java / *Robotic Programming Design and Practices: Using Java*

A book introduces leJOS framework on LEGO NXT robots. + ISBN: 9789862768228  
([http://isbn.ncl.edu.tw/NCL\\_ISBNNet/main\\_DisplayRecord.php?PHPSESSID=c8kchinqo5ncq873i47c4sqkb0&Pact=Display&Pstart=1](http://isbn.ncl.edu.tw/NCL_ISBNNet/main_DisplayRecord.php?PHPSESSID=c8kchinqo5ncq873i47c4sqkb0&Pact=Display&Pstart=1))

### (2016) Undergraduate thesis: *Solving graph domination problem on DH graph using split decomposition*

The study focuses on total and paired domination problems on distance-hereditary graph. Discovered an alternative optimal solution to total domination based on split decomposition. The handout can be found in the [link](https://drive.google.com/file/d/18H1fvSZ7td3vArSJaeoTKkLjJ-DqApaP/view?usp=sharing) (<https://drive.google.com/file/d/18H1fvSZ7td3vArSJaeoTKkLjJ-DqApaP/view?usp=sharing>).

### (2021) Master thesis: *Occlusion resistant tracking based on idempotent adjunction relationship between cyber and physical objects*

This work combines deep learning and formal methods to tackle the occlusion in image tracking-by-detection problem. It developed a method to establish strong relationship among sensor data and learned representations, and achieved short-term prediction of vehicle movements based on the material derivative method. The model is trained on completely labeled data is evaluated on partially occluded data. It shows 0.3 meter displacement error in average.

### (RTSCA 2022) *Scalable and Bounded-time Decisions on Edge Device Network using Eclipse Zenoh*

This work developed an collaborate decision algorithm based on the *cheap consensus* concept. By stacking CRDT, Reliable Broadcast and the data exchange framework *Zenoh* together, the method was able to achieve consistent decision among several peers within a bounded period of time. It also evaluated the performance on Zenoh and CycloneDDS.

### (2023) *A Performance Study on the Throughput and Latency of Zenoh, MQTT, Kafka, and DDS*

This work evaluated the performance of popular pub/sub communication frameworks in terms of latency and throughput under various scenarios.

## Lectures

### (COSCUP 2023) *Highly Performant Dataflow in Stream-Oriented Programming in Rust*

This lecture showed off the par-stream, a demonstration of stream-oriented programming in Rust. The proposed framework leveraged async/.await syntax to mix I/O-bound and CPU-bound operations written in iterator style. It made it simple and straightforward to build an efficient data pipeline. ( [COSCUP](https://coscup.org/2023/zh-TW/session/JLZM37) (https://coscup.org/2023/zh-TW/session/JLZM37))

## Open-Source Contributions

The following lists the contribution of my own work.

### TorFS (2019)

The project exploited free storage by abusing Tor's Onion service. It provided a virtual file system that stored data blocks as RSA public keys in disguise, and distributed them to the rest of Onion network. ( [github](https://github.com/jerry73204/cns-final-tor-store) (https://github.com/jerry73204/cns-final-tor-store))

### hacky-arm (2020)

Build a dirty-cheap robotic arm solution that performs object grabbing tasks using computer vision based SLAMs. ( [github](https://github.com/jerry73204/hacky-arm/) (https://github.com/jerry73204/hacky-arm/))

### par-stream (2020)

It developed building blocks for asynchronous parallel data flow for Rust. It was developed to serve image data processing that mix heavy I/O and computation tasks. It provided iterator-style stream combinators to distribute data and scale on workers, and is designed to support Rust's async/await in nature. ( [github](https://github.com/jerry73204/par-stream) (https://github.com/jerry73204/par-stream))

### ddshark (2023)

Live monitoring tool like htop for RTPS protocol. It is compatible with various DDS implementations and was tested with Cyclone DDS. ( [github](https://github.com/NEWSLabNTU/ddshark) (https://github.com/NEWSLabNTU/ddshark))

### misc.

[carla-rust](https://github.com/jerry73204/carla-rust) (https://github.com/jerry73204/carla-rust), [rust-tfrecord](https://github.com/jerry73204/rust-tfrecord) (https://github.com/jerry73204/rust-tfrecord), [rust-cv-convert](https://github.com/jerry73204/rust-cv-convert) (https://github.com/jerry73204/rust-cv-convert), [fast-yuv442-to-rgb24](https://github.com/jerry73204/fast-yuv442-to-rgb24) (https://github.com/jerry73204/fast-yuv442-to-rgb24) and more.

The following lists the cooperation works and contributed projects.

### autoware\_carla\_launch (2023)

This is an cooperated work with ADLINK's evshary, which develops the Zenoh communication interface among CARLA simulator and Autoware, one of the most comprehensive vehicle driving software.