

# YUE FEI SHE/HER

Machine-Learning • Parallel Programming • Signal Processing • Error-Control Codes

EDUCATION	<p><b>University of Toronto</b>  <i>Master of Engineering, Electrical Engineering</i> Sep. 2022 – Jun. 2024</p> <ul style="list-style-type: none"> <li>• Emphasis: Communications</li> <li>• Advisor: Prof. Raviraj Adve</li> <li>• MEng Thesis: “Pilot Training - Angle of Arrival and Channel Estimation in 5G Network”</li> </ul> <p><b>University of Toronto</b>  <i>Bachelor of Applied Science, Electrical Engineering</i> Sep. 2017 – Jun. 2022</p> <ul style="list-style-type: none"> <li>• Capstone Project: Convolutional Neural Network (CNN) NPU Overlay (MobileNetV1) for FPGA (Intel Stratix 10 NX 2100)</li> <li>• Advisors: Prof. Vaughn Betz and Andrew Boutros</li> </ul>
PUBLICATIONS (PEER-REVIEWED CONFERENCE)	<p>1. Arash Ahmadian, Louis S.P. Liu, <u>Yue Fei</u>, Konstantinos N. Plataniotis; Mahdi S. Hosseini. Pseudo-Inverted Bottleneck Convolution for Darts Search Space. <i>IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)</i>, 2023.</p>
AWARDS AND HONORS	<p><b>Dean’s Honour List</b> 2017 Fall, 2018 Winter, 2018 Fall, 2021 Fall, &amp; 2022 Winter  <b>Edward S. Rogers Sr. Department Betz Entrance Scholarship (\$5,000)</b> 2017</p>
CERTIFICATE	<p><b>Certificate in Engineering Business</b> Jun. 2022</p>
INVITED TALKS	<p><b>Panel: Demystifying Machine Learning</b> Mar. 2025</p> <p><i>Talk: From Channels to States — Machine Learning in the Language of Communication and Control</i>  <i>QWomen San Diego — Qualcomm Internal Panel Discussion</i></p>
TECHNICAL SKILLS	<p><b>Programming:</b> Python, Pandas, C/C++, MATLAB, Julia, Arm Assembly, Verilog/SystemVerilog, Unix/Linux Shell, Perl  <b>Data Processing &amp; Automation:</b> CI/CD (Jenkins) pipelines; Makefile  <b>ML &amp; Optimization:</b> Scikit-learn, PyTorch, Stable-Baselines3 &amp; Gymnasium for Reinforcement Learning (RL), GRU-RNN, MLP, Attention mechanism &amp; visualization [Code], Q-Learning (Reinforcement Learning), Convex Optimization (fractional &amp; quadratic-transform), Sampling-based Source Coding  <b>Parallel Programming:</b> POSIX Threads, SIMD (Intel intrinsics), Parallel STL (C++17)  <b>Tools &amp; Environments:</b> Git, Vim/GVim, SimpleScalar simulators</p>

## INDUSTRY EXPERIENCE

### Qualcomm – Design Verification Engineer Markham, Canada

Jun. 2024 – Jul. 2025

- Verified **UWB** receiver path and improved startup performance by reducing LNA charging delay **90%** (20ns → 2ns)."
- Validated **WLAN CP-PLL synthesizer** loop and built **UVM**-compatible test plans spanning 500+ channel indices, ensuring robust coverage across 2G, 5G, and emerging 5G alternative bands.
- Developed **multi-head GRU-based RNN** for receiver gain line-up optimization, where each head learns one analog block (LNA, GM, TIA, BQ, PGA). Transformed a complex combinatorial tuning problem into a scalable learning-based approach, easing designer effort.
- Built a physics-inspired **MLP** (*customized activation function + Pre-Normalization + Post-Normalization*) that predicts **VCO** capacitance from control inputs, removing the need for RF/analog designers to manually tune capacitors for 1000+ frequency targets.

### Alphawave Semi - Digital Verification Engineer Toronto, Canada

May 2020 – Jun. 2021

- Developed **UVM** testbenches to verify **SerDes** (clocking, datapath, **SRAM**), expanding functional coverage across **50+ scenarios**.
- Enhanced **CI/CD** automation to support **15× growth** in regression testing (scaling from **4 to 60+ projects**), improving efficiency and reliability as the company expanded.

## SELECTED PROJECTS

Highlighted academic, research, and technical projects spanning ML, optimization, signal processing, HPC, and architecture.

### ML & Optimization

Modeling and optimization for wireless networks and semantic coding

#### Convex & Fractional Programming for Multi-Cell MIMO Beamforming

Sep. 2023 – Dec. 2023

- Applied **fractional-programming** and **quadratic-transform** optimization to improve multi-cell **MIMO beamforming**, boosting convergence and power-constrained sum-rate performance.

#### Sampling-Based Semantic Source Coding (One-Shot Info Theory)

Jan. 2023 – May. 2023

- Implemented **Poisson functional representation**, **rejection sampling**, **importance sampling**, and hybrid Poisson + dithered-quantization for 6G semantic source-coding in MATLAB. [Code]

#### Transformer & Embedding Visualization (Research Assistant)

Jul. 2021 – Sep. 2021

- Explored **RNN-based Transformer** models and **Attention mechanisms** for NLP tasks .
- Applied **PCA-based embedding visualization**—as used in **GloVe**—to project high-dimensional embeddings into **2D/3D space** using **Python** (**NumPy**, **Matplotlib**), enabling intuitive inspection of semantic clusters.

### Parallel & HPC

Multi-threaded acceleration and memory-coherence design

#### Parallel Beamforming & Cache-Coherence Foundations for Scalable Compute

Jan. 2024 – May. 2024

- Accelerated medical-imaging by **7×** (17 s → 2.5 s) via **16-thread data-parallel** ultrasound beamforming with **SIMD** intrinsics and memory optimizations (**restrict**, **single-write**); validated correctness (RMS error < 1e-16) and scalability (1–16 threads) — a paradigm relevant to **data-parallel LLM training**. [Code]
- Designed and verified a **3-hop directory cache-coherence protocol** (MSI/MESI) in **Murφ**; optimized with **Exclusive (E)** state to eliminate bus transactions (**0 MB** overhead vs 80 MB baseline). [Code]

**Computer Architecture**  
Speculation mechanisms relevant to LLM decoding

**Computer Architecture Coursework**

*Sep. 2021 – Dec. 2021*

- Developed a 5-stage pipelined CPU with hazard detection & forwarding, implemented a perceptron-based branch predictor, Tomasulo out-of-order execution[Code], Bouquet prefetcher, and MSI-directory cache-coherence protocol; these experiences deepened my understanding of pipeline parallelism for distributed LLM training and how speculative execution in CPUs parallels LLM speculative decoding for faster inference.

**Coding Theory**  
Error-correcting codes for reliable communications

**Graph-Based Error-Correcting Codes**

*Sep. 2023 – Dec. 2023*

- Implemented LDPC, fountain/LT, and Polar encoders/decoders over binary-erasure channel; developed custom simulators in Julia and MATLAB. [Code]

**Convolutional Codes & Viterbi Decoder**

*Jan. 2023 – May. 2023*

- Built a rate- $\frac{1}{2}$  convolutional encoder and Viterbi decoder in MATLAB with custom trellis structures; validated against built-in tools and demonstrated 2-bit-error correction on a noisy BSC channel. [Code]

**BCH Codes & Extended Euclidean Algorithm**

*Jan. 2023 – May. 2023*

- Developed a binary BCH encoder/decoder over  $GF(2^m)$ , applying the extended Euclidean algorithm to solve the key equation for syndrome decoding and implementing Chien search for error location. [Code]

**Signal Processing**  
Estimation and detection for modern wireless systems

**Angle of Arrival (AoA) & Channel Estimation**

*Jan. 2023 – May. 2023*

- Implemented MUSIC, DFT, and Matrix-Pencil eigenvalue methods in MATLAB for AoA estimation under large-scale and Rayleigh-fading channels; Matrix-Pencil delivered  $\approx 2$  dB gain in low-SNR regimes with 16-antenna arrays, outperforming MUSIC and DFT for highly-correlated signals.

**LTE Signal Processing**

*Jan. 2022 – Apr. 2022*

- Processed captured LTE signals for time/frequency sync, OFDM demodulation, and pilot-power analysis in MATLAB; resampled 40 MHz front-end data to 30.72 MHz LTE rate and validated PSS/SSS detection.