# Raymond Mogg

raymondmogg@gmail.com | raymogg.com | linkedin.com/in/raymogg | github.com/raymogg

## PROJECTS

- **Foragers RL** (JAX): Built a JAX environment for simulated population control using RL. Agent uses PPO to keep the population of "foragers" within a controlled band. Experimented with how the control agents performance changed based on its target band. Results and writeup: https://raymogg.com/projects/foragers-world/
- **Home Lab** (vLLM): Deployed local LLM infrastructure, running open source models on an RTX 5000 Pro with vLLM. Wrote custom kernels supporting my GPU for newer models (Flint).
- **Self-Guided Research**: Independent research into open-endedness, continual learning and autoresearch. Explored Karpathy's autoresearch, building a multi-agent meta-harness to compare actor-critic inspired agent topologies. Designed and built experiments measuring LLM performance on niche tasks, including Tailbench, a p99 tail latency optimisation benchmark over procedurally generated train/test environments.

## PROFESSIONAL EXPERIENCE

**Senior Quantitative Developer** · Auros, Hong Kong & Brisbane — April 2026 – September 2026

- Led new desk build-out. Architected and built a distributed, async trading system (Rust, Tokio, Docker).
- Built the desk's data pipeline (Kafka, ClickHouse), capturing transactions and market events for pricing and execution research.
- Researched and implemented spot pricing models for low-liquidity, long-tail assets (Python, Rust).
- Modelled quote fill probability on Polymarket using logistic regression on historical order and fill data (Python, NumPy).

**Non-Compete Period** · Company-mandated 12 months out of trading — April 2025 – April 2026

- Built an end-to-end research and trading stack solo (Python, NumPy, GCP Cloud Run) allocating capital across on-chain lending markets against a modelled fair value. Ran $150k USD of own capital, returning $\sim$18.3% annualised over 4 months live.

**Software Engineer** · Flow Traders, Amsterdam — May 2023 – April 2025

- Architected and built a modular async trading engine (Rust, Tokio). Reduced tick-to-trade latency by 90%, from 10ms+ to $\sim$800$\mu$s, through profiling, benchmarking, and cache-level optimisation.
- Built simulation environments replaying historical pricing and on-chain event data, used to test algorithm and trading strategy changes against real market conditions before production.
- Redesigned the hedging system to update trading parameters automatically in response to real-time hedging cost, improving trade sizing and realised performance.
- Built observability and alerting tooling (Kafka, ClickHouse, Grafana, Python) giving oversight of execution quality, used to monitor and debug regressions in live systems.

**Co-Founder / CTO** · Pier Two (prev. Mycelium), Brisbane — Feb 2019 – April 2023

- Built and operated large-scale distributed infrastructure (Kubernetes, Docker, GCP) serving 712 pricing feeds across 9 blockchains.
- Ran the system against 24/7 uptime requirements, with redundancy and automated failover.
- Led core protocol development across research and engineering, and grew and managed a team of 20+ engineers.
- Designed and built a novel on-chain tokenised derivatives protocol (Solidity, Python, NodeJS). Traded over $700m USD of notional value.
- Company acquired by Bitmine (NYSE:BMNR).

**Software Engineer Intern** · Atlassian, Sydney — Nov 2019 – Feb 2020

**Research Intern** · UQ Robotics Design Lab — July 2016 – Nov 2016

## PUBLICATIONS

**A Framework for Generating Evasion Attacks for Machine Learning Based Network Intrusion Detection Systems** WISA 2021

- Trained decision-tree intrusion detection classifiers on NSL-KDD (126k samples, 41 features), chosen for interpretability so that model decisions could be traced feature by feature.
- Automated the production of adversarial payloads used to red-team the decision tree classifier, ensuring the produced payloads remained valid attacks.
- Achieved 43.3–87.3% evasion rates across four attack classes, and showed the highest rates were an artefact of overfitting to the seed rather than genuinely viable attacks.
- Traced evasions to specific decision boundaries, using model interpretability to explain why each adversarial sample succeeded.
- First author, with S.Y. Enoch and D.S. Kim. Springer LNCS vol 13009.

## EDUCATION

**University of Queensland** · MSc, Statistics July 2026 – Current

**University of Queensland** · BE (Hons), Software Engineering Feb 2016 – June 2020

- Honours Thesis (Grade 7/7): *Evasion attacks against machine learning based intrusion detection systems using genetic algorithms* - supervised research, published at WISA 2021 (see Publications).
- Overall grade 86% (Honours IIA).
- Key coursework: Machine Learning, Artificial Intelligence, Probability Models & Analysis, Multivariate Calculus & ODEs, Calculus & Linear Algebra I–II, Advanced Algorithms & Data Structures.

## SKILLS

**Languages:** Python, Rust, SQL, Solidity

**ML:** PyTorch, JAX, NumPy, vLLM, Local Inference, Agent Harness Engineering, Evaluation & Benchmark Design, RL Environment Design, Trajectory Analysis, Statistical Modelling

**Concepts:** Distributed Computing, Low Latency Infrastructure, Async Programming (Tokio, asyncio), Kubernetes, Containerisation (Docker), Sandboxed Code Execution, Profiling & Benchmarking, Data Pipelines (Kafka, ClickHouse), Observability (Grafana), GCP