Towards evolution-based autonomy in large-scale systems

Presented by: Damien Anderson
Authors: Damien Anderson\textsuperscript{1}, Paul Harvey\textsuperscript{3}, Yusaku Kaneta\textsuperscript{2}, Petros Papadopolous\textsuperscript{1}, Philip Rodgers\textsuperscript{2}, Marc Roper\textsuperscript{1}

\textsuperscript{1}University of Strathclyde, Glasgow, United Kingdom
\textsuperscript{2}Rakuten Mobile, Tokyo, Japan
\textsuperscript{3}University of Glasgow, Glasgow, United Kingdom
The Problem

Issues/problems with Networks
- Unreliable connections
- Change in user needs
- Real-time unexpected events
How we plan to achieve this

Evolution

• Adapt to unseen and unimagined scenarios
• Self optimising capabilities
• Efficient search space exploration for configurations
• Scalability
Challenges

• Generic architecture
• Decouple Evolution from Operational
• Fitness function decomposition
• Problem Encoding
• Generation of realistic traffic
Evolution-based Autonomy

- Evolution layer is platform agnostic
- Operation layer is tailored to the platform/technology
The Experimental Setup

Start with a simpler use case

- Develop a working version of the system on a particular type of network
  - Content Delivery Network (CDN)

CDN optimisation is an ongoing area of research. CDNs have many uses for a wide variety of content types. Can we develop an autonomous CDN which can adapt itself for the particular type of content?
The Experimental Setup

- **Master Evolution Controller**: Responsible for overall CDN Performance (KPIs, Health etc)
- **Meta Evolution Controller**: Responsible for specific CDN KPIs (Geographical, Throughput, Latency etc)
- **Local Evolution Controller**: Responsible for individual Varnish Cache/VCL performance

- **Evolutionary/Operational Boundary**

- **Operation Controller (Backend)**
- **Operation Controller (Directors)**
- **Operation Controller (TTL)**

- **VCLWriter**: Translates the Operation Controller configurations into a VCL File
- **VCLDeployer**: Deploys a VCL File onto a target Varnish Cache
Observations

- Granularity of Operation Controllers is challenging
- Dependencies between Operation Controllers
- Need to decouple the platform from the Operation Controllers
Future Work

- Build more use cases
- Different platforms
- Different Evolutionary approaches
Thank you for listening!

Any questions?
Damien.anderson@strath.ac.uk