Summary of Session 4: Trustworthiness and Protection

Karthik Pattabiraman

University of British Columbia (UBC)

Two Talks

- *Issues of Trust and Trustworthiness for Dependable Machine Learning Systems* (Flavio Figueiredo, Federal University of Minas Gerais)

- *Protecting autonomous operation with high-assurance operating systems* (Gernot Heiser, University of New South Wales)

Flavio's Talk - 1

- When is a ML system dependable ? Definitions of dependability metrics.
- What is Trust in the context of ML ? Distinction between trust and trustworthiness (Onara O'Niell)
- Trust and trustworthiness changes over time (e.g., politicians)
- What does fairness mean ? Fair to whom. Larger discussion involving politics or society
- We can't ignore that "The systems we build no longer impact society (anymore)".
- We can't think of the human as the problem: humans can reduce dependability

Flavio's Talk - 2

- 21 different definitions of fairness. Statistical definition Vs. causality
- Definitions of causality, fairness - different in different communities
- Interpretability, Transparency and accountability
- Definition of "fair elections" is about process, not a single metric.
- Fairness is a hard problem – no silver bullet. Still an open

Gernot’s Talk - 1

- There’s no safety without cybersecurity. Helicopters and military vehicles.
- Retrofit technology to military vehicles. Professional hackers couldn’t hack it.
- SEL4 protected mode OS with sound and complete WCET analysis.
- Mixed criticality systems: high-critical process cannot depend on low-critical process
- Scheduling contexts - time caps. Scheduling budgets. Capability-based system

Gernot's Talk - 2

- Shared server - scheduling context (client gets charged for server time)
- High-assurance OS supporting mixed criticality w/o sacrificing utilisation
- Boeing retrofitted the system onto their aerospace system. Took 2 people 3 years to do the task (total of 6 person years)
- Component middleware: CamkES. Architecture specification language for ACL4. Enforcing the architecture (under certain conditions)
- Cogent - Code-generation of proof and code. Reduce the cost of verified systems code. Smaller TCB compared to other systems.
- Dependability-cost tradeoff: Model checking to remove (some) bugs.

Takeaways and Open Questions

- **Need think about Trustworthiness at both algorithmic and systems levels**

- **Algorithmic solutions: achieving fairness, transparency etc.**
  - Can't ignore that the systems we build have significant societal impact
  - Need to involve actors in the social and political space (outside CS/Engineering)
  - Provide quantitative measures for fairness – 21 definitions in literature (at least)

- **Systems-level solutions: Verified and Secure OS**
  - Can't have safety without cyber-security – sel4
  - Need to have verified Oses capable of supporting multiple-criticality systems
  - Verified tool flow starting from high-level specifications to the running code