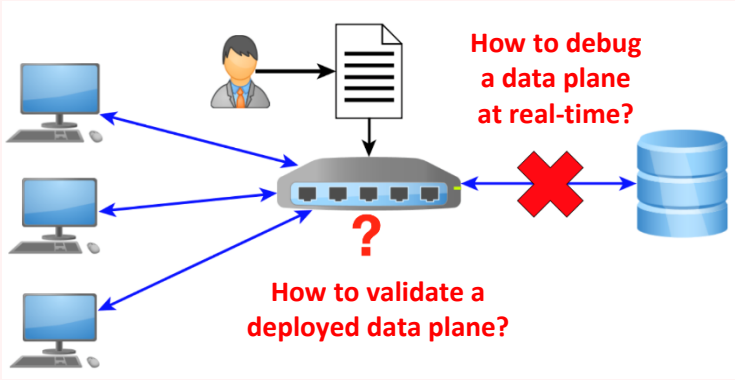
A Programmable Framework for Validating Data Planes

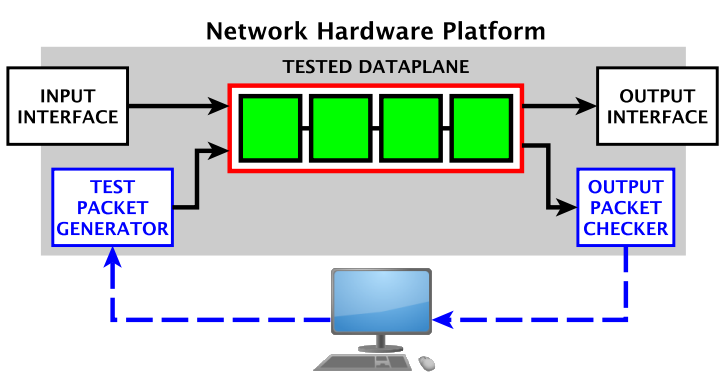
Pietro Bressana[1], Noa Zilberman[2], Robert Soulé[1]
[1] Università della Svizzera italiana, [2] University of Cambridge

Context & Problem

- Emerging trend of In-Network Computing
- Programmable network hardware accessible to users



System Overview



- Deployed in parallel to live traffic
- Internally connected to the data plane under test
- Framework managed by software agent

Our Solution

- Framework for data plane validation
- Fully programmable using P4 language
- Independent from the language of the DUT
- Hardware/Software solution
- Embedded into the network platform
- Debugging at line rate in real-time

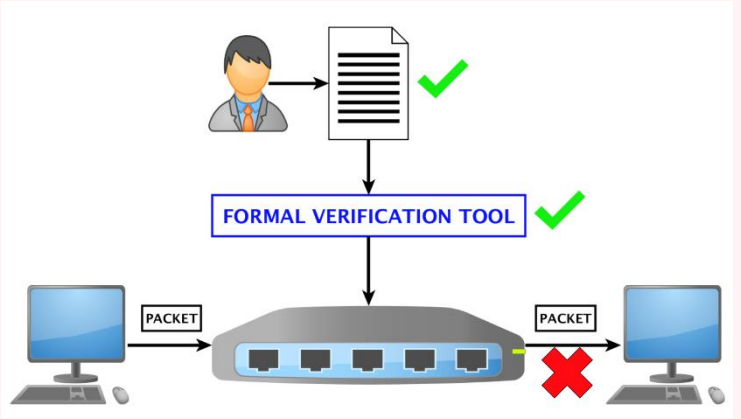
Use-cases

Use-case	NetDebug	Software formal verifiers	External network testers
Functional testing	✓	✓	partial
Performance testing	✓		partial
Compiler check	✓		partial
Architecture check	✓		partial
Resources quantification	✓		
Status monitoring	✓		
Comparison	✓	partial	partial

- Compiler Check: limitations in the compiler
- Architecture Check: Limitations in the architecture

Evaluation

- Several data plane programs validated
- NetDebug detects bugs hidden from verification tools
- Found bug in SDNet: parser's reject state not implemented



Conclusions

- NetDebug addresses a urgent need for data plane debugging and verification
- Visibility into network devices
- Leveraging both P4 language and hardware design
- Prototype implementation using NetFPGA SUME framework and Xilinx SDNet

Acknowledgements

SNSF award 407540_167173, Doc.Mobility fellowship P1TIP2_178657, the Leverhulme Trust (ECF-2016-289), the Isaac Newton Trust.