A Hybrid P4/NFV Architecture for Cloud Gaming Traffic Detection with Unsupervised ML

Joël Roman Ky, Bertrand Mathieu – Orange Innovation Lannion
Philippe Graff, Thibault Cholez – Université de Lorraine, CNRS, Inria, LORIA

IEEE ISCC 2023
Tunis, 11/07/2023
Table of contents

1. Context & Motivation
2. Previous works
3. ML-based CG traffic detection
4. Hybrid P4/NFV Architecture
5. Evaluation
6. P4 Limitations
7. Conclusion
1. Context & Motivation

- Increasing pressure of Cloud Gaming (CG) traffic on current network infrastructures.
- Stringent network requirements, especially in terms of jitter and latency.
- Recent network technologies such as L4S (Low Latency Low Loss Scalable throughput) can reduce the latency of low-latency (LL) traffic.

![Diagram showing flow classification and LL traffic]

IEEE ISCC 2023
2. Previous works

- **Cloud Gaming traffic** from 4 main CG platforms available in Europe (Stadia (STD), GeForceNow (GFN), Xbox Cloud (XC) and PlayStationNow (PSN)):
  - Traffic on normal conditions
  - Traffic with network constraints

- **Non-Cloud Gaming Traffic**:
  - Video conferencing (VC)
  - Video streaming (VS)
  - Live video streaming (LV)
  - Facebook navigation (FN)

- **New datasets**:
  - (new games and new CG platforms (Moonlight, Steam))

- Decision Tree (DT) model (supervised ML) in a set of VNF to detect CG traffic [Graff et al.]

- Performance degradations with new CG traffic (« unseen » CG traffic or new CG platforms).

- Mean packet size
- Average Inter Arrival Times (IAT)
- Total number of packets
- Standard deviation of packet size
- Standard deviation of IATs
- Total size of application data

- 12 network traffic features (computed for uplink and downlink traffic) for a window traffic of 33ms
3. ML-based CG Traffic Detection

- ML models:
  - DT (Decision Tree): CG traffic on normal conditions and NCG traffic (with labels)
  - USAD (UnSupervised Anomaly Detection): CG traffic on normal conditions only (without labels) [Audibert et al.]

- ML models Performance:
4. Hybrid P4/NFV Architecture

- 1 hardware P4 switch, 1 NFV compute node

P4: Programming Protocol-independent Packet Processors

IEEE ISCC 2023
5. Evaluation

### Accuracy & F1-score for all traffic

<table>
<thead>
<tr>
<th>Traffic</th>
<th>Type</th>
<th>Performance with P4</th>
<th>Performance with application</th>
<th>Difference (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Accuracy</td>
<td>F1</td>
<td>Accuracy</td>
</tr>
<tr>
<td>Normal CG</td>
<td>STD</td>
<td>0.965</td>
<td>0.983</td>
<td>0.984</td>
</tr>
<tr>
<td></td>
<td>GFN</td>
<td>0.990</td>
<td>0.995</td>
<td>0.979</td>
</tr>
<tr>
<td></td>
<td>XC</td>
<td>0.903</td>
<td>0.942</td>
<td>0.966</td>
</tr>
<tr>
<td></td>
<td>PSN</td>
<td>0.981</td>
<td>0.990</td>
<td>0.966</td>
</tr>
<tr>
<td>Overall</td>
<td></td>
<td>0.958 (±0.054)</td>
<td>0.979 (±0.03)</td>
<td>0.973 (±0.021)</td>
</tr>
<tr>
<td>CG with network constraints</td>
<td>STD</td>
<td>0.999</td>
<td>0.999</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>XC</td>
<td>0.954</td>
<td>0.977</td>
<td>0.992</td>
</tr>
<tr>
<td></td>
<td>PSN</td>
<td>0.995</td>
<td>0.998</td>
<td>0.961</td>
</tr>
<tr>
<td>Overall</td>
<td></td>
<td>0.983 (±0.035)</td>
<td>0.993 (±0.019)</td>
<td>0.984 (±0.019)</td>
</tr>
<tr>
<td>Non CG</td>
<td>VC</td>
<td>0.938</td>
<td>0.971</td>
<td>0.867</td>
</tr>
<tr>
<td></td>
<td>LV</td>
<td>0.980</td>
<td>0.990</td>
<td>0.978</td>
</tr>
<tr>
<td></td>
<td>VS</td>
<td>0.993</td>
<td>0.996</td>
<td>0.991</td>
</tr>
<tr>
<td></td>
<td>FB</td>
<td>0.988</td>
<td>0.994</td>
<td>0.989</td>
</tr>
<tr>
<td>Overall</td>
<td></td>
<td>0.959 (±0.056)</td>
<td>0.983 (±0.031)</td>
<td>0.918 (±0.114)</td>
</tr>
<tr>
<td>New games CG learned platforms</td>
<td>GFN</td>
<td>0.973</td>
<td>0.986</td>
<td>0.995</td>
</tr>
<tr>
<td></td>
<td>XC</td>
<td>0.969</td>
<td>0.984</td>
<td>0.979</td>
</tr>
<tr>
<td></td>
<td>PSN</td>
<td>0.980</td>
<td>0.990</td>
<td>0.999</td>
</tr>
<tr>
<td>Overall</td>
<td></td>
<td>0.974 (±0.010)</td>
<td>0.987 (±0.005)</td>
<td>0.991 (±0.011)</td>
</tr>
<tr>
<td>New CG platforms</td>
<td>MoonLight</td>
<td>0.999</td>
<td>0.999</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>Steam</td>
<td>0.999</td>
<td>0.999</td>
<td>1.000</td>
</tr>
<tr>
<td>Overall</td>
<td></td>
<td>0.999 (±0.00)</td>
<td>0.999 (±0.00)</td>
<td>1.000 (±0.00)</td>
</tr>
</tbody>
</table>
6. P4 Limitations

- Limited computation capabilities due to:
  - Limited number of operations
  - Variable multiplication/division limited => Standard deviation can not be computed

- Not a packet buffering/copying capability in P4
  - => Impossible to keep in memory packet features value to compute standard deviation

- Processing trigger upon packet arrival
  - => Reports are not sent exactly each 33ms because there is no timer
  - If sporadic traffic, empty reports need to be send to acknowledge the lack of traffic
6. P4 Limitations

- Standard deviation approximation with previous mean value.

- Remove standard deviation from ML features
7. Conclusion

- CG traffic detection implementation on P4 hardware module demonstrates excellent performance.
- Lessons learned from P4 hardware implementation:
  - P4 not suitable for handling complex computational tasks.
  - Time event-based programs are not ideal for P4.
  - Code optimization is required with P4.
- Trade-off between high-speed line-rate packet processing and ease of programming.
- Hybrid P4/NFV architecture is a promising approach to efficiently split processing tasks.
- Evaluation of the solution with operational traffic from network operators will be considered in future work.
Thank you

joelroman.ky@orange.com
https://www.mosaico-project.org/
https://github.com/mosaico-anr/P4_NFV_CG_Detector
8. References


Appendix
A-1. ML Models

- **Decision Tree (DT):**

- **UnSupervised Anomaly Detection (USAD)** [Audibert et al.]:

  ![Diagram of UnSupervised Anomaly Detection (USAD)]