

CS 6480: Paper reading summary

HA 13.a

José Monterroso

School of Computing, University of Utah

October 11, 2020

1 TurboEPC: Leveraging Dataplane Programmability to Accelerate the Mobile Packet Core

Paper discussed in this summary is “TurboEPC: Leveraging Dataplane Programmability to Accelerate the Mobile Packet Core” [4].

1.1 First pass information

1. *Category:*

This paper is a description of a research prototype. The research prototype being described is TurboEPC. TurboEPC improves both control plane processing throughput and latency.

2. *Context:*

The technical area of this paper relates to the mobile packet core (EPC) and how handling the signaling messages differently can lead to throughput and latency improvements. This paper uses P4 [1] technology and also mentions that although TurboEPC is meant for 4G it can still work on 5G [3].

3. *Assumptions:*

A big assumption they make is that TurboEPC could also work on the 5G core. From what I have seen of 5G, I think TurboEPC could work

on the NSA 5G infrastructure, but I’m not so sure about the standalone.

4. *Contributions:*

The paper claims three main contributions. The first being TurboEPC itself which is a redesign of the mobile packet core that improves performance. The second is that they built TurboEPC over P4-based programmable software/hardware switches to show feasibility. And third, they provide a quantification of the performance gains of TurboEPC over the traditional CUPS-based EPC design.

5. *Clarity:*

From what I have read this paper appears to be formatted well. However, I think it reads a bit differently from what I am used to in a research paper.

1.2 Second pass information

• *Summary:*

TurboEPC is a redesign of the current mobile packet core. Their main redesign is that they offload a significant portion of the signaling procedures from the control plane to programmable dataplanes. It is because they are moving the “the core” to the edge, we can get higher throughput and lower latency. TurboEPC is based off the traditional CUPS-based EPC.

Specifically, the offload of S1 release and the service request procedures to the edge causes performance increases. Furthermore, parts of the user context can be offloaded to the current eNB because the UE can only connect to one eNB at a time. However, this proposes challenges because a typical core must handle millions of users but switch memory is usually limited. Furthermore, if the switches go down the UE data will be lost. To solve these issues the user propose three methods of solution. The user context is distributed over a set of switches connected in series, the user context is distributed over set of switches on parallel network paths, or a combination of both. TurboEPC delivers significant performance gains over the traditional EPC over realistic traffic mixes which contain a high proportion of offloadable signaling messages.

1.3 Third pass information

- *Strengths:*

I thought that this paper was really well made. It was very detailed in its approach and implementation. The introduction and background do an excellent job of making the reader aware of the problem and provide them with enough information that they understand what the problem is. The paper followed a great format and contain excellent figures that did a great job of illustrating the problem, implementation, and data evaluation. I also like how they explained the issues with TurboEPC and how they propose to fix them.

- *Weaknesses:*

A bit of nit pick but a few of the first figures in the paper were shown a page or two before they were described in the text. This was annoying because I would later need to scroll back up to the figure to understand what they were showing. Overall I think the paper was great, however, I think if they were going to mention 5G and how TurboEPC could apply to 5G, then maybe they should have applied it to 5G also.

- *Questions:*

The overall paper was great but I wonder if telecom companies would every apply something like this to their EPC. It seems like lots of things would need to be remade, and a lot of stress would be put on the switches.

- *Interesting citations:*

The authors mention that MobileStream [2] is similar to TurboEPC. MobileStream is from the University of Utah so I would be curious to read up on the University of Utah's take on mobile packet cores.

- *Possible improvements:*

The only small detail I would change is rearranging some of the figures so they at-least appear on the same page where the text describes them.

- *Future work:*

Since they mention this could work for 5G I can imagine some kind of 5G paper very similar to this that might help speed up the current 5G NSA infrastructure. We can even produce a measurement paper between 5G TurboEPC and the normal 5G. This really isn't my field of interest so I can't really think of any other type of future works.

References

- [1] BOSSHART, P., DALY, D. GIBB, G., IZZARD, M., MCKEOWN, N., REXFORD, J., SCHLESINGER, C., TALAYCO, D., VAHDAT, A., VARGHESE, G., AND WALKER, D. P4: programming protocol-independent packet processors. *ACM SIGCOM Computer Communication Review* (July 2014).
- [2] CHO, J., AND STUTSMAN, R. VAN DER MERWE, J. Mobilestream: a scalable, programmable and evolvable mobile core control plane platform. *CoNext '18: Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies* (Dec. 2018).
- [3] GROUP COMPANIES, N. Making 5g a reality. *NEC White Paper*.

- [4] SHAH, R., KUMAR, V., VUTUKURU, M., AND KILKARNI, P. Turboepc: Leveraging dataplane programmability to accelerate the mobile packet core. *SOSR '20: Proceedings of the Symposium on SDN Research* (Mar. 2020).