Towards the 5G Revolution: A Software Defined Network Architecture Exploiting Network Coding as a Service

Dávid Szabó, Felicián Németh,
Balázs Sonkoly*, András Gulyás†
HSN Lab, Budapest Univ. of Technology and Economics
{szabod,nemethf,sonkoly,gulyas}@tmit.bme.hu

Frank H.P. Fitzek
5G Lab Germany, Technische Universität Dresden
frank.fitzek@tu-dresden.de

ABSTRACT

Many networking visioners agree that 5G will be much more than the incremental improvement, in terms of data rate, of 4G. Besides the mobile networks, 5G will fundamentally influence the core infrastructure as well. In our vision the realization of the challenging promises of 5G (e.g. extremely fast, low-overhead, low-delay access of mostly cloudified services and content) will require the massive use of multipathing equipped with low overhead transport solutions tailored to fast, reliable and secure data retrieval from cloud architectures. In this demo we present a prototype architecture supporting such services by making use of automatically configured multipath service chains implementing network coding based transport solutions over off-the-shelf software defined networking (SDN) components.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations - Network management

Keywords

SDN; Network Coding; Mininet; NFV

1. INTRODUCTION

Future communication systems will face dramatic changes compared to state of the art systems. With the Internet of Things (IoT) the number of mobile and wireless devices will increase by one or even two orders of magnitude resulting in 500 billion devices by 2020. This will lead to dramatic bandwidth request for the overall communication system. Furthermore 5G is on the horizon introducing the tactile Internet which impose massive requirements on security, safety, resilience, throughput, and delay reduction. The huge demand for dramatic reduction in delay and increased throughput needs a new area of communication systems. Packet switched networks, the core of today's Internet, usually provision a single path between two communication endpoints on which each router performs store and forward operation for each incoming packet. The single path does not only limit the throughput of the network, it is also vulnerable against hackers and outages. With respect to the tactile Internet the current architecture is problematic as the long distance between the cloud service and the mobile device results in too large delays.

In order to achieve high throughputs, resilience, security and low delays a revolutionary change is currently discussed [3], namely the code centric networks. In the latter, suitable routers may perform a different strategy referred to as compute and forward using random linear network coding. This enables efficient multi path communication over possibly lossy channels while providing inherent algebraic security as hackers would need to tap all involved communication systems. The use of network coding is also enabling another important feature which is a dynamic allocation of distributed clouds on top of each router, placing the cloud in close proximity of the user, which enable low latencies for the tactile Internet.

In order to use network coding efficiently, we have to orchestrate the coders, recoders[1] (a new entity introduced by network coding) and decoders to meet the constraints of the physical topology and we have to take care of steering the traffic over them. Good news is that such orchestration and steering is in perfect agreement with the current practice of Internet Service Providers (ISPs) that services are implemented in the form of appropriately concatenated middleboxes, also known as *service chains*. In this respect we seem to have a lower adoption barrier for network coding at ISPs. However, today's service chains are usually built around special purpose networking hardware elements, configuring and operating these chains is a highly non-trivial task which still requires human interaction. SDN and virtualization can be a promising way out of this managerial trap as these technologies may enable flexible and automated deployment of service chains containing also software middleboxes or Virtualized Network Functions (VNF).

In this demonstration we present a system which enables the definition, configuration and automated deployment of service chains implementing code centric operation over off-the-shelf SDN components. Building on the ESCAPE [1]

*MTA-BME Future Internet research group
†MTA-BME Information systems research group

[1]The application of recoders inside the network can significanlty reduce latency and increase goodput with respect to the approaches coding only at endpoints e.g. in case of Fountain codes.

SIGCOMM '15 August 17-21, 2015, London, United Kingdom

ACM ISBN 978-1-4503-3542-3/15/08.
DOI: http://dx.doi.org/10.1145/2785956.2790025

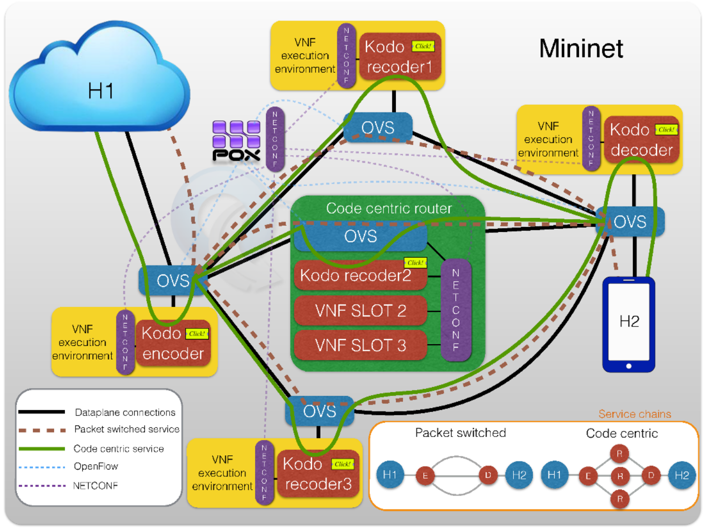

Figure 1: The prototype architecture.

prototyping framework, we combine a Mininet emulated OpenFlow 1.0 network controlled by POX with VNFs implementing network coding in the Click [2] modular router platform using the Kodo [4] network coding software library for prototyping a novel communication infrastructure paving the future of 5G networking.

2. ARCHITECTURE AND DEMO

Fig. 1 presents the architecture of our prototype and an outline of the demo scenario. We build our OpenFlow network in Mininet using Open vSwitch (OVS) instances and hosts (H1/H2) which can represent any sources of traffic e.g. a simple smartphone or a server in a data center. The switches are controlled by a POX controller which contains our steering module handling the flow tables according to the configuration of the running service chains. Besides the switches and hosts we have virtualized network functions implementing encoders, recoders and decoders in the click router platform using the Kodo library. Kodo implements random linear network codes which allows flexible recoding of the packets anywhere in the network. These VNFs can be run in two ways. VNF execution environments (yellow boxes) are advanced Mininet hosts running a NETCONF (OpenYuma) agent. Our NETCONF client implemented as a POX module can connect to these agents and can start a given VNF process on the host. This solution stands for the case when we run VNFs outside the routers e.g. in a near OpenStack data center having built-in NETCONF support. This solution may slightly increase latency but provides the possibility of scaling out in the presence of massive network loads combined with more complex coding schemes (e.g. more efficient random linear codes using larger field size). Alternatively, we have also built a visionary prototype of the code centric router (green box). This router consist of a standard OVS (no modifications in OpenFlow) instance but has the capability to execute VNFs and connect them to virtual ports of the switch through NETCONF. Using code centric routers adds less delay but there is no way to scale out beyond the router's hardware resources. Our POX orchestrator module receives service chains as inputs (can be given by GUI) and configure the switches and start the appropriate VNFs accordingly.

During the course of the demo we define two multipath service chains over our network with 50% loss configured for each link. The first chain (bottom right corner of Fig. 1) uses end-to-end codes (like Reed Solomon or Raptor) and ordinary packet switching as packets follow the dashed brown paths between H1 and H2. Then we start a VLC server/client on H1/H2 respectively configured to send and receive a real-time video stream using UDP. The video has nice quality but at the cost of +115% redundancy at the encoder, which has to flow through the network. At this point we upgrade our service by adding the recoder elements (as shown explicitly in the bottom right corner of Fig. 1). Our system reconfigures the switches and launches the required VNFs thus transiting the service to the code centric area (the green path). Without noticeable interruption in the video stream we witness perfect video quality at H2 but with a 30% less total number of sent packets in the network and reduced latency due to the recoder elements. Our demo[2] presents a plausible way to combine network coding with SDN and realizing code centric services on top of off-the-shelf SDN components. Our framework comprising OpenFlow, Click and Kodo enables experimenting with code centric networks and may promote their transition from a vision to the reality of future 5G networks.

Acknowledgements – The research leading to these results has received funding from the European Union Seventh Framework Programme under grant agreement № 619609.

3. REFERENCES

[1] A. Csoma et al. Escape: extensible service chain prototyping environment using mininet, click, netconf and pox. In *ACM SIGCOMM*. ACM, 2014.

[2] E. Kohler et al. The click modular router. *ACM Trans. Comput. Syst.*, 18(3):263–297, Aug. 2000.

[3] M. Medard et al. Network coding mythbusting: Why it is not about butterflies anymore. 2012.

[4] M. V. Pedersen et al. Kodo: An open and research oriented network coding library. In *NETWORKING 2011 Workshops*, pages 145–152, 2011.

[2] A screencast showing a simplified version of the demo is available at: https://youtu.be/TG3xduqpD_4