Paper Evaluation, Composing Software-Defined Networks

Jiani Jiang <jianij@kth.se>

1. Paper summary

Software-Defined Networking (SDN) platforms lacks support for creating modular applications, which forces programmers to reason manually about the low-level dependencies between different parts of codes. This paper introduces an abstraction to solve the problem that the previous works about modularity in software system don't research on building a single application out of multiple, independent modules. The authors use parallel composition operator to carry out multiple policies operating concurrently on the same set of packets and sequential composition operator to allow one module on the packets produced by another module. Also, they operate an abstract topology of the network object for each policy to constrain the module and define an abstract packet model for further extension. The abstractions are realized using Pyretic language, so the authors focus on evaluating the language and the applications running on Pyretic.

2. Top 3 contributions

a. It proposes the Pyretic programming model to achieve the modularity of applications. Pyretic allows programmers to build a large, sophisticated controller out of small, self-contained modules.

b. It contributes the new idea of network objects which can allow policies to operate on abstract locations that map to the sets of physical network elements.

c. It make it possible to decompose functionality using the parallel and sequential composition operators and is much easier to program the network.

3. Problems

a. After reading this paper, the biggest problem I find is that there is no evaluation for the performance. Only the codes of Pyretic language which is not enough to show the result.

b. Also, the paper lacks the analysis and related mechanisms of network failures.