

**Dynamic, Fine-Grained Data Plane Monitoring with Monocle**

**Abstract:**

Ensuring network reliability is important for satisfying service-level objectives. However, diagnosing network anomalies in a timely fashion is difficult due to the complex nature of network configurations. We present Monocle — a system that uncovers forwarding problems due to hardware or software failures in switches, by verifying that the data plane corresponds to the view that an SDN controller installs via the control plane. Monocle works by systematically probing the switch data plane; the probes are constructed by formulating the switch forwarding table logic as a Boolean satisfiability (SAT) problem. Our SAT formulation quickly generates probe packets targeting a particular rule considering both existing and new rules. Monocle can monitor not only static flow tables (as is currently typically the case), but also dynamic networks with frequent flow table changes. Our evaluation shows that Monocle is capable of finegrained monitoring for the majority of rules, and it can identify a rule suddenly missing from the data plane or misbehaving in a matter of seconds. In fact, during our evaluation Monocle uncovered problems with two hardware switches that we were using in our evaluation. Finally, during network updates Monocle helps controllers cope with switches that exhibit transient inconsistencies between their control and data plane states.

**Existing System:**

We argue that these methods are insufficient – ping/traceroute and other similar tools do not determine what packet header values can test for data plane correspondence. They are also often not capable of sending arbitrary packets that are required in the SDN context. ATPG provides end-to-end data plane monitoring and can quickly localize problems, however it is designed to batch-generate probes for all network rules at the same time and as a consequence it requires substantial time to pre-compute its probes after each network change. This delay is too long for modern SDNs where the ever-increasing amount and rate of change demand a quick, dynamic monitoring tool that is the focus of this paper. In particular, a major reason behind SDN getting traction is that it makes

it easy to quickly provision/reconfigure network resources. New network demands created by Amazon EC2 spot instances, more control being given to the applications, and more frequent routing recomputation make it even harder to ensure data plane correspondence.

**Proposed System:**

We present the design and implementation of Monocle, a data plane correspondence monitoring tool that can operate on fine-grained timescales needed in SDN. In particular,Monocle goes beyond the state-of-the-art in its ability to quickly recompute the required monitoring information during a rule update.

We formulate a set of formal constraints the monitoring packets must satisfy. We handle unicast, multicast, ECMP, drop rules, rule deletions and modifications. When necessary, we provide proofs that our theoretical foundation is correct. This formal treatment of the rule generation problem is the key advancement over our earlier work on RUM. We also optimize the conversion of the constraints a probe needs to satisfy into a form presented to an off-the-shelf SAT solver.

We describe our experience of using Monocle showing that it can reveal switch problems that were previously unknown to us. This is one of the major advancements of this article over our earlier conference paper