

draft-vpolak-bmwg-plrsearch-03

Post IETF-107 Interim BMWG Meeting

Authors: Vratko Polák, Maciek Konstantynowicz

Presented by: Vratko Polák

Draft changes -02 to -03

- Minor update with added FD.io CSIT reference and editing nits.
- Added FD.io CSIT reference

```
FDio-CSIT-PLRsearch:  
  target: https://docs.fd.io/csit/rls2001/report/introduction/methodology\_data\_plane\_throughput/methodology\_plrsearch.html  
  title: "FD.io CSIT Test Methodology - PLRsearch"  
  date: 2020-02
```

Overview

- PLRsearch is a packet “throughput” search algorithm suitable for **probabilistic** (as opposed to deterministic) systems.
- It searches for probabilistically defined **critical load** satisfying given **target loss ratio**.
- It performs **sequential** trial measurements of offered load constant within a measurement.
- It still applies many assumptions on the system behavior, often unrealistic for some systems.
 - It assumes results of trial measurements are **independent** of each other.
 - It assumes possible loss counts follow **Poisson distribution**.
 - It assumes average **loss ratio** does not depend on trial duration.
 - It relies on heuristic **fitting functions** to relate results of trial measurements with different offered loads.
- It uses Bayesian inference computing both trial measurements’ offered load and final estimate.

Sample Implementation

- Implementation of PLRsearch is in Linux Foundation FD.io CSIT project
- Most recent results in CSIT-2001 report:
 - https://docs.fd.io/csit/rls2001/report/vpp_performance_tests/soak_tests/index.html
 - 30 minute search duration is used
 - Some tests give fairly different estimate on repeated run
- CSIT project general information:
 - <https://wiki.fd.io/view/CSIT>
 - <https://git.fd.io/csit/>

Open items

- Similarly to MLRsearch, PLRsearch aims to give result similar to TCP goodput
- PLRsearch works with stateless traffic generators
- No analysis comparing measured TCP goodput with estimated critical rate has been done yet
- Tweaking packet loss distribution can make critical rate estimates more consistent
- Tweaking fitting functions can give more realistic / conservative bounds of the estimate

- Divide the draft to “terms part” applicable for any probabilistic performance testing, and “algorithm part” specific to PLRsearch as one of possible search methods.

Thanks for Comments, here some questions

- Can we move it from the named draft to BMWG workgroup one?
- Next steps

THANK YOU !

draft-vpolak-bmwg-plrsearch-03

Post IETF-107 Interim BMWG Meeting

Authors: Vratko Polák, Maciek Konstantynowicz

Presented by: Vratko Polák