

draft-ietf-bmwg-mlrsearch-03

IETF-115 London, BMWG Meeting

Authors: Maciek Konstantynowicz, Vratko Polák

Draft Status

- draft-ietf-bmwg-mlrsearch-03 posted on 9th of November 2022 (post IETF-115 deadline)
- Still work in progress.

Draft changes -02 to -03

- Major updates in all sections
 - Addressing most/all points from BMWG feedback (thanks to Al, Gabor, Vladimir)
 - Clearer articulation of the problem statement
 - Clearer description of MLRsearch methodology and strategies to address listed problems
 - Stricter references to methodologies and terms specified in RFCs 1242, 2285, 2544
 - Updated Terminology section
 - Many details intentionally not specified (for now)
- Sections describing sample implementations
 - Outdated and got removed to avoid confusion
 - Will be added in future revision

MLRsearch – What is it, Motivation and Objectives

- Multiple Loss Ratio search (MLRsearch)
 - A new test methodology aiming to improve network throughput search specified in RFC 2544
- Motivation:
 - Address problems posed by testing NFV systems and other software based network data planes.
- Objectives:
 - minimize the total test duration
 - search for multiple loss ratios
 - improve results repeatability and comparability

Problems

- Long Test Duration
 - Vanilla bisection is too slow
- DUT within SUT
 - DUT – software program processing packets, device of interest
 - SUT – server hardware and operating system, server resources shared
 - DUT is effectively “nested” within SUT
 - SUT performance is a spectrum, “noiseless” end is more stable and of interest
- Repeatability and Comparability
 - Need for search procedure that reports more stable results across iterations and environments
- Throughput with Non-Zero Loss
 - Searching for multiple loss ratio goals helps to describe SUT performance better
 - Non-Zero loss performance is more stable
- Inconsistent Trial Results
 - Is a zero-loss trial a throughput if there was non-zero-loss trial with smaller load?

Terminology update (highlights)

- Throughput
 - Defined as in RFC2544, must be zero loss
- Conditional throughput
 - For non-zero loss
 - Coincides with throughput when the loss ratio goal is zero
 - Refers to forwarding rate (instead of load) for non-zero loss ratio goal
- Lower / upper bound:
 - Highlighted they are relative to the current search phase
 - Otherwise vague, to avoid “inconsistent results” issues

MLRsearch Approach

- Main inputs
 - One or more loss ratio goals, Target trial duration, Target precision
- Sequence of phases
 - Initial phase, Middle phases, Final phases
- Initial phase
 - Establish “starting” bounds for first middle phase
 - Use 3 trials with following Oloads: i) MOL, ii) FRMOL [terms in RFC2285], iii) FRFRMOL [new]
 - By default FRMOL used as upper bound, FRFRMOL used as lower bound
- Middle phases
 - Explicit 4 actions within a phase
 - Turns “starting” bounds into “ending” bounds
- Final phases
 - One for each goal
 - Final phase for first goal is done before middle phases for second goal
 - (FR of) Lower ending bound becomes the conditional throughput

MLRsearch - Running code

Versions

ver.	Goals	IOrder	Track	trial	DV	(reference to patch in <i>gerrit.fd.io</i>)
1	NDR+PDR	GiP	2pG	1	-01	(30954)
2	Multiple	GiP	AFS	1	-02	(36061)
3	Multiple	PiG	AFS	1	-03	(37418)
4	Multiple	PiG	AFS	n		(none yet)
5	Multiple	PiG	2	n	(-04)	

Legend

- Goals: How many loss ratio goals are supported.
 - NDR+PDR: Unconditional NDR goal, optional configurable PDR goal.
 - Multiple: One or more goals, sorted from low to high loss.
- IOrder: Iteration order of phases.
 - GiP: Goal in phase, meaning all goals at 1s are bounded before 5s.
 - PiG: Phase in goal, meaning first goal to 30s before second goal starts.
- Track: How much data from shorter load results is tracked.
 - 2pG: Upper and lower bounds from previous phase are tracked for each goal separately.
 - AFS: All results are tracked, but Forgetting Short if load is re-measured.
 - 2: Only upper and lower bounds from immediately preceding phase are tracked.
- Trial: Whether "sub-trials" are supported.
 - 1: No. Load time is always spent in a single trial.
 - n: Yes. Multiple trials are added (behaves as "1" if load time is short).
- DV: IETF draft version applicable the most.

Next steps

- BMWG review
 - Scrutiny and validation of listed problems
 - Review of proposed MLRsearch approach
- MLrsearch draft work
 - Compatibility with Binary Search with Loss Verification is coming (not in -03)
 - Inconsistent trial handling is important, but currently not described
 - Non-ambiguous definitions are needed for comparability between MLRsearch implementations
 - At least for “final phase goal” and “lower bound”

THANK YOU !

draft-ietf-bmwg-mlrsearch-03

IETF-115 London, BMWG Meeting

Authors: Maciek Konstantynowicz, Vratko Polák