Hierarchical Resource Sharing and Queuing in OMNeT++ and INET framework

Authors: *Atacan Iyidogan* (presenter), *Marcin Bosk*, *Filip Rezabek*

To manage the ever increasing amount of traffic in today's computer networks, new queueing disciplines using hierarchical scheduling algorithms are being introduced. One of these novel methods is the Hierarchical Link Sharing (HLS) [1] queuing discipline (qdisc), a successor to the Hierarchical Token Bucket (HTB) [2]. HLS supports minimum rate guarantees and sharing of excess resources to classes specified by a pre-determined hierarchy. HLS is implemented as a kernel module, but the qdisc is not yet fully integrated into the Linux kernel.

To better understand the differences between HLS and HTB qdiscs, we design a set of experiments and metrics comparing them. We perform our comparisons in the discrete simulator OMNeT++ [3] and INET framework [4]. We further evaluate the results against hardware experiments performed using Linux implementations [5,6] of the qdiscs. While both qdiscs are partially available in Linux, an implementation of HLS is missing in OMNeT++. Therefore, as one of the main contributions of this work, we introduce **HLSQueue**, an HLS implementation intended as an extension of the INET framework.

Further, we validate that behaviour of **HLSQueue** corresponds to its Linux implementation. Similarly, the behavior of previously introduced HTBQueue [7] is also validated to confirm its correct functionality. We then compare bandwidth sharing properties of HLS and HTB using OMNeT++. We show that HTB has more customization options for bandwidth sharing, while HLS better isolates bandwidth between subtrees containing multiple classes. We additionally compare the delay and jitter caused by the schedulers. When the network is not overloaded, the delays generated by HTB are comparable to those of HLS. The jitter of HTB, however, is considerably larger when excess bandwidth is shared between classes. On the other hand, with an overloaded network HTB shows a much higher delay for well behaving flows compared to HLS. In those cases the delay introduced by HTB rises by up to 260% compared to non-overloaded case. The HTB delay is also higher than that of HLS in an overloaded network, with the average increase for each well-behaving class being 66%.

References:

[1] N. Luangsomboon and J. Liebeherr. A Round-Robin Packet Scheduler for Hierarchical Max-Min Fairness. 2021. doi: 10.48550/ARXIV.2108.09864. url: https://arxiv.org/abs/2108.09864.

[2] M. Devera. Hierarchical Token Bucket. 2003. url: http://luxik.cdi.cz/~devik/qos/htb/.

[3] OMNeT++ Discrete Event Simulator. url: https://omnetpp.org/.

[4] INET Framework for OMNeT++. url: https://inet.omnetpp.org/.

[5] Linux qdisc implementation of HTB. url: https://github.com/torvalds/linux/blob/master/net/sched/sch_htb.c.

[6] N. Luangsomboon. Linux qdisc implementation of HLS. 2021. url: https://github.com/Lantua/HLS.

[7] M. Bosk, M. Gajić, S. Schwarzmann, S. Lange, and T. Zinner. HTBQueue: A Hierarchical Token Bucket Implementation for the OMNeT++/INET Framework. 2021. doi: 10.48550/ARXIV.2109.12879. url: https://arxiv.org/abs/2109.12879.