

Machine Learning-based End-to-End QoE Monitoring Using Active Network Probing

Gilson Miranda Jr.*[†], Esteban Municio*, Johann M. Marquez-Barja*, Daniel Fernandes Macedo[†]

*University of Antwerp - imec, IDLab, Faculty of Applied Engineering - Antwerp, Belgium

[†]Universidade Federal de Minas Gerais - Computer Science Department - Minas Gerais, Brazil

E-mail: {gilson.miranda, esteban.municio, johann.marquez-barja}@uantwerpen.be, damacedo@dcc.ufmg.br

Abstract—Video on Demand (VoD) is responsible for a significant amount of traffic on IP networks. To meet users' expectations, network operators need means to monitor and to identify when service quality is degraded in order to take actions to avoid customer churn. Many proposals in the literature correlate network Quality of Service (QoS) metrics with indicators of user Quality of Experience (QoE). However, most solutions cannot monitor end-to-end conditions without modification on video player applications or require deep packet inspection techniques, which may raise privacy issues. In previous work, we proposed a method to estimate QoE using active ICMP probing, which is widely supported by network devices and can be used for end-to-end network measurements. In this work, we improve our previous method by adding a secondary model that operates over

literature lack the ability to monitor the last-mile link, which in many cases is the network bottleneck, especially in wireless networks [6]. Therefore, QoE monitoring methods that neglect the last mile may be unable to detect degraded QoE.

User satisfaction is highly influenced by events such as video playback stalls, initial buffering delay, and oscillations in video quality [7]. Using client-side information to estimate user QoE is one of the approaches found in the literature [8], [9]. The problem with client-side QoE monitoring is that it requires changes in video players or the deployment of additional software on clients or servers. With network-only techniques on the other hand providers do not need to have