Evaluating the quality of real-time applications using the DCCP/CCID-3 transport protocol.*

J. Van Velthoven, K. Spaey and C. Blondia

University of Antwerp, Department of Mathematics and Computer Science
Performance Analysis of Telecommunication Systems Research Group
Middelheimlaan 1, B-2020 Antwerp, Belgium
{jeroen.vanvelthoven,kathleen.spaey,chis.blondia}@ua.ac.be

Because the reliable service of TCP introduces retransmissions and hence delays that may not always be acceptable for applications like streaming video, internet telephony and on-line games, today these applications use UDP to transport their data. However, the lack of an inbuilt congestion control mechanism in UDP and the growing popularity of the applications using UDP poses a real threat to the overall health of the internet. The Datagram Congestion Control Protocol (DCCP) is a transport protocol recently defined by the IETF to provide a solution to this problem [1]. It combines unreliable flows with built-in congestion control, and makes use of acknowledgement mechanisms and Explicit Congestion Notification (ECN) to discover packet loss and congestion events. The protocol obliges the use of some form of congestion control, but it leaves the choice of the mechanism being used to the application.

Currently there are two congestion control algorithms defined for DCCP, referred to by their congestion control identifier (CCID). CCID-2 [2] is a TCP-like additive increase, multiplicative decrease algorithm, intended for applications that want to transfer as much data as possible in an as short as possible time. CCID-3 [3] is an implementation of the equation based TCP-Friendly Rate Control (TFRC), appropriate for applications requiring a smooth throughput. If a connection uses this CCID, the receiver gives feedback to the sender about the allowed rate at which the packets can be sent. It is designed to be reasonably fair when competing for bandwidth with TCP and TCP-like flows. We focus on this last CCID because many real-time applications would prefer such a smooth behavior of throughput over time above a higher throughput at some particular moments.

Until now, a lot of work is about evaluating the TCP friendliness of DCCP [4, 5]. In our presentation, we will take a look at the influence the use of DCCP has on the quality of the real-time application. For our simulations we used the ns-2 DCCP module implemented by Mattsson [6] to obtain performance measures like throughput, packet loss and delay. These measures can be used together with a quality model, like e.g., the model of Verscheure for MPEG-2 video applications [7], to determine the end-user quality of the application using DCCP. We mainly focus on video sources with infinite granularity in the sense that the source adapts its rate to match the allowed sending rate so that no packets are dropped at the sender due to restrictions on the sending rate.

---

[*] This work was carried out within the framework of the IWT (Flemish Institute for the promotion of Scientific and Technological Research in the Industry) project CHAMP.

Many real-time applications are very sensitive to packet loss and therefore it is preferable to reduce this loss as much as possible. This can be done by using ECN in combination with active queue management so that packets can be marked instead of dropped to indicate network congestion. In our simulations we considered how the use of ECN influences the monitored performance measures.

The presentation will be structured as follows. First, we will discuss the need of a new transport protocol like DCCP to provide congestion control to real-time applications. We will give a brief overview of the DCCP mechanism, the CCID-3 congestion control algorithm and ECN. This will be followed by the presentation of the performance measures of interest obtained from different simulation scenarios. The results show us among others that by using ECN the number of lost packets can be reduced significantly while maintaining approximately the same average throughput. The use of ECN also results in a smaller average delay experienced by the packets. Further, we find that DCCP/CCID-3 flows receive a fair share when competing for bandwidth with TCP flows on the same bottleneck link, under the condition that the applications above both transport protocols use the same packet size.

References

[1] E. Kohler, M. Handley, S. Floyd, "Datagram Congestion Control Protocol (DCCP)", IETF draft-ietf-dccp-spec-11, http://www.ietf.org/internet-drafts/draft-ietf-dccp-spec-11.txt, March 2005.

[2] S. Floyd, E. Kohler, "Profile for DCCP Congestion Control ID 2: TCP-like Congestion Control", IETF draft-ietf-dccp-ccid2-08, http://www.ietf.org/internet-drafts/draft-ietf-dccp-ccid2-10.txt, March 2005.

[3] S. Floyd, E. Kohler, J. Padhye, "Profile for DCCP Congestion Control ID 3: TFRC Congestion Control", IETF draft-ietf-dccp-ccid3-11, http://www.ietf.org/internet-drafts/draft-ietf-dccp-ccid3-11.txt, March 2005.

[4] M. Vojnovi, J-Y Le Boudec, "On the long-run behavior of equation-based rate control", SIGCOMM '02: Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications, ACM Press, 2002.

[5] S. Floyd, M. Handley, J. Padhye, J. Widmer, "Equation-based congestion control for unicast applications", SIGCOMM '00: Proceedings of the conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, ACM Press, 2000.

[6] N-E Mattsson, "A DCCP module for ns-2", http://epubl.luth.se/1402-1617/2004/175/LTU-EX-04175-SE.pdf, September 2004.

[7] O. Verscheure, P. Frossard and M. Hamdi, "User-Oriented Qos Analysis in MPEG-2 Video Delivery", Journal of Real-Time Imaging (Special Issue on Real-Time Digital Video over Multimedia Networks), Vol. 5, N. 5, October 1999.