YeahTcp

h1. Yeah-TCP Heuristic TCP protocol gained from experimental evaluation of [CUBIC|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBCubicTcp],[HSTCP|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBHighSpeedTCP], [H-TCP|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBHamiltonTCP], Africa, [Compound TCP|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBCompoundTCP]. Algorithm Design Goals: * Network capacity should be exploited efficiently. This is the most obvious goal, which can be achieved by modifying the congestion window update rules; [YeAH-TCP|YeahTcp] can exploit anyone of the increment rules of other proposals \(e.g. [STCP|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBScalableTcp],[H-TCP|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBHamiltonTCP], etc.\). * The stress induced to the network should be less or equal than that induced by [Reno TCP|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBTcpReno]. * TCP friendliness with [Reno|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBTcpReno] traffic. A "politically" acceptable algorithm should be able to compete fairly with Reno flows, avoiding starvation of competing flows, and simultaneously exploiting the link capacity. * The algorithm should be internally and RTT fair. * Performance should not be substantially impaired by non-congestion related \(random\) packet loss events; random packet loss cannot be ruled out even in case of high speed optical backbones. Reasonable values of this loss depend on the technological context, but we verify that even a loss rate in the order of 10^-7 can give rise to sensitive performance degradation. * [Small link buffers|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBNetworkBufferSizing] should not prevent high performance. It is not feasible to design buffer size equal to the bandwidth-delay product in [high BDP links|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBLongFatNetworks] as required by standard Reno congestion control. This goal can be achieved by adopting a decrease policy in case of packet loss similar to the Westwood algorithm. [YeAH-TCP|YeahTcp] has two different operating modes, "Fast" and "Slow", like Africa TCP. During the "Fast" mode, [YeAH-TCP|YeahTcp] increments the congestion window according to an aggressive rule \(e.g. [STCP|%SCRIPTURL{"view"}%cgibintwikiviewPERTKBScalableTcp] rule\). The state is decided according to the estimated number of packets in the bottleneck queue \(inferred from the measured RTTs\). h2. References * _YeAH-TCP: Yet Another Highspeed TCP,_ A. Baiocchi, A. Castellani, F. Vacirca, PFLDnet 2007, Feb. 2007 \([PDF|pfldnet2007:paper^YeAH_TCP.pdf]\) -- Main.ChrisWelti - 26 Feb 2007