

## Alternatív TCP variánsok és torlódásvezérlő mechanizmusok vizsgálata magas késleltetésű, nagy sávszélességű hálózatokon

Orosz Péter, [oroszp@unideb.hu](mailto:oroszp@unideb.hu)  
Debreceni Egyetem Informatikai Igazgatóság

*Kulcsszavak: TCP variáns, torlódásvezérlés, AIMD, self-clocking, késleltetés, WAN, BDP, átviteli teljesítmény, fairness*

### **Abstract**

Az előadásban megvizsgáljuk számos alternatív TCP variáns teljesítmény paramétereit nagy sávszélességű és magas késleltetésű WAN hálózatokon. A sávszélesség és a késleltetés növekedésével a hagyományos TCP alapú adatátvitel hatékonysága – különösen nagy adatmennyiség átvitelekor – nagymértékben romlik. Válaszként számos TCP variáns született az átviteli teljesítmény fokozására, minthogy az utóbbi években az internetes optikai gerinckapcsolatok sávszélessége jelentősen növekedett (>Gbps). A probléma gyökerét az ablak-alapú (*window-based*) torlódásvezérlés átviteli teljesítmény szabályozásában találjuk. Tudjuk, hogy a TCP torlódási ablak méretét (*congestion window*) az elküldött csomagokra adott nyugták érkezési ideje szabályozza, amely elsősorban a hálózat késleltetésének (*RTT – Round Trip Time*) függvénye. Tehát minél magasabb értékű a sávszélesség és a késleltetés szorzata egy adott kapcsolaton, az algoritmus annál kevésbé képes hatékonyan kihasználni a rendelkezésre álló sávszélességet. Az alkalmazott AIMD (*Additive Increase, Multiplicative Decrease*) algoritmus viszonylag lassan növeli az ablakméretet (*congestion window*), ezáltal az átviteli ráta lassan közelíti a rendelkezésre álló sávszélességet. Ezek mellett fontos megemlíteni, hogy a TCP self-clocking mechanizmusának szabályozó hatása az adott fizikai kapcsolaton egyidejűleg jelenlévő TCP adatfolyamok számától is függ.

Ahhoz, hogy egy TCP variáns széles körben alkalmazható legyen, egyszerre kell megfelelnie mind a nagy átviteli teljesítmény, mind az adatfolyamok közötti méltányosság (*fairness*) kritériumainak. A torlódásvezérlő mechanizmusokat e két szempont egyidejű figyelembevételével, valós hálózati környezetben elemezzük.