Proxy cache szerverek hatékonyság-vizsgálata

BÉRCZES TAMÁS
IFSZ KFT, Debrecen; berczes.tamas@ifsz.hu
SZTRIK JÁNOS
Debreceni Egyetem, Informatikai Kar; jsztrik@inf.unideb.hu

Kulcsszavak: sorbanállási hálózat, proxy cache szerver, teljesítmény vizsgálat

Az Internet használhatóságának egyik elengedhetetlen feltétele, hogy a különböző igények, lekérdezések válaszideje a forgalom bővülésétől függetlenül lehetőleg viszonylag kicsi maradhasson. Jelen dolgozat keretében a proxy cache szerverek hatékonyságát vizsgáljuk a Bose és Chang által felállított nyitott Jackson sorbanállási modellt kiterjesztésével. A módosított hálózati modell figyelembe veszi az összes irányból érkező igényeket, valamint realisztikusan paraméterezi a távoli web-szerver teljesítményadatait. A numerikus eredmények megmutatják, hogy annak eldöntése, hogy érdemes-e proxy cache szervert üzemeltetni, nagyban függ a cég internetezési szokásaitól, nevezetesen, hogy: milyen terheltségű oldalakat látogatnak, milyen gyakorisággal térnek vissza ugyanarra a webhelyre stb. Néhány példán keresztül igyekszünk e modell felhasználásával segítséget nyújtani annak eldöntésére, hogy egy aktuális szituációban megéri-e proxy cache szervert üzemeltetni vagy sem.

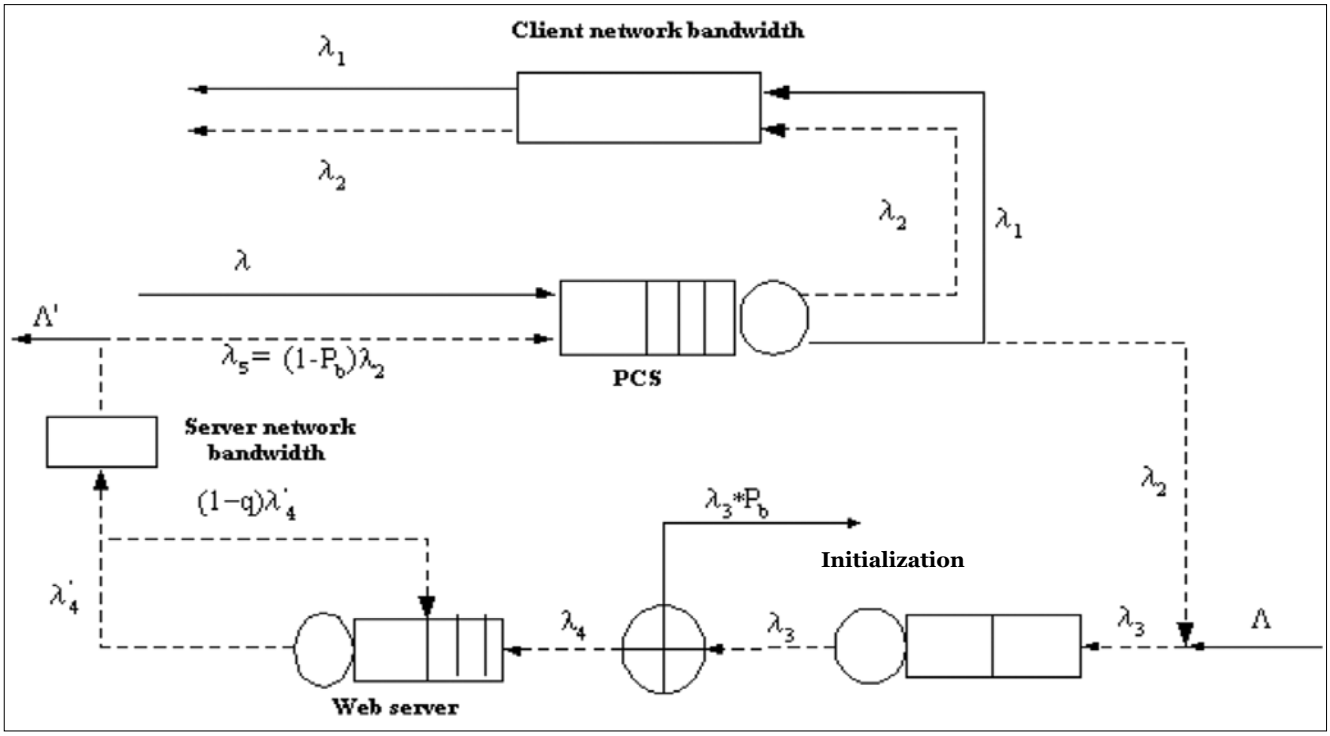
1. Bevezetés

Az internet használata az elmúlt években rohamosan növekedett. A felhasználók száma a 2001-es 474 millióról 2002-re 590 millióra nőtt és becslések szerint 2006-ra eléri a 948 milliót. Figyelembe véve, hogy 1996-ban mindösszesen 40 millióan használták az internetet, a növekedés üteme igen jelentős.

A felhasználók számának növekedésével párhuzamosan növekedett az internet forgalma is. Ennek hatására egyre nagyobb igény mutatkozik a színvonalas és gyors internet elérésre és kiszolgálásra.

Az információ keresése és letöltése közben a válasz a távoli web-szervertől a kliens gépéig gyakran igen sok időt vesz igénybe. A probléma egyik oka, hogy ugyanabban az időben ugyanazt a fájlt más felhasználó is le akarja tölteni. Ebből adódóan ugyanazon fájlok másolatai mennek keresztül a hálózaton, ez pedig a kiszolgálási idő növekedését eredményezi. Természetes megoldásnak mutatkozik az információk tárolása. Ennek egyik megoldási lehetősége a böngésző szoftverben való implementálás. Ebben az esetben a tárolt adatokhoz azonban csak egy személy férhet hozzá. Egy másik lehetőség proxy cache szerver használata.

1. ábra Egy igény lehetséges útja



Jelen dolgozat célja ez utóbbi megoldás hatékonyságának vizsgálata figyelembe véve az idevágó eddigi munkákat, lásd [2,3,4,5].

A felhasználó szemszögéből nézve lényegtelen, hogy az általa keresett fájl fizikailag hol található: egy proxy cache szerveren (PCS) valahol a munkahelyének belső hálózatán vagy a világ túlsó felén egy távoli web-szerveren. A keresett dokumentum érkezhet a web-szervertől vagy a proxy cache szervertől. Kliens oldalról nézve a PCS funkciója ugyanaz mint egy web- szerveré valamint a web-szerver felől nézve a PCS ugyanúgy viselkedik mint egy kliens.

Feltételezzük, hogy a PCS felől érkező belső igények érkezése λ paraméterű Poisson-folyamatot követ, valamint a külső igények paraméterű Poisson folyamat alapján érkeznek a web-szerverhez.

2. A modell

Jelen cikkben a Bose és Cheng [2] által javasolt analitikus modellt módosítjuk. Az eredeti modellhez képest figyelembe vesszük azokat a külső igényeket is melyek nem a PCS irányából érkeznek, hanem bármilyen más felhasználótól is, így ezek jelentősen befolyásolhatják a válaszidőket. A még realisztikusabb vizsgálat érdekében, a Bose- és Cheng-modelltől eltérően, a web-szerver kapacitását végesnek vesszük.

Az *1. ábra* mutatja a módosított modellben egy igény lehetséges útját a felhasználótól kiindulva egészen a visszaérkezésig. A jelölések jelentése megtalálható az *1. táblázatban*.

Proxy cache szervert használva, ha egy fájlt le akarunk tölteni egy távoli web-szerverről először meg kell vizsgálni, hogy a keresett dokumentum egy példánya megtalálható-e a PCS-en (Ennek valószínűségét jelöljük p-vel). Amennyiben megtalálható, egy másolat továbbítódik a felhasználónak, míg amennyiben nem, úgy az igény továbbítódik a web-szerverhez. Miután az igényelt fájl megérkezett a PCS-re, egy másolat azonnal a felhasználóhoz kerül.

1. táblázat Az alkalmazott jelölések

λ	A klienstől érkező igények intenzitása
Λ	A külső igények érkezési intenzitása
F	Az igényelt fájl mérete
P	A PCS találati valószínűsége
B_{xc}	A PCS kimenő puffere
B_s	A Web szerver kimenő puffere
I_{xc}	A PCS –en való keresési idő
Y_{xc}	A szerver statikus ideje a PCS esetén
R_{xc}	A PCS dinamikus szerver ideje
I_s	Egyszeri kapcsolat inicializálási idő
Y_s	A Web-szerver statikus ideje
R_s	A Web-szerver dinamikus szerver ideje
N_c	A kliens sávszélessége
N_s	A szerver sávszélessége

A proxy cache szerver hatékonysága a következő tényezőktől függ:

– a találati arány (a kért dokumentum milyen valószínűséggel található meg a PCS-en)
– a PCS sebessége
– a kliens oldali sávszélesség
– a szerver oldali sávszélesség
– a külső igények intenzitása
– a Web szerver karakterisztikája

Legyen F a keresett dokumentumok átlagos mérete. Az alábbiakban definiáljuk az 1. ábrán szereplő változókat.

$$\lambda_1 = p^*\lambda; \quad (1)$$

$$\lambda_2 = (1-p)^*\lambda; \quad (2)$$

$$\lambda_3 = \lambda_3 + \Lambda; \quad (3)$$

Az egyenes vonal (λ_1) reprezentálja azt az esetet, mikor a keresett dokumentum egy példánya megtalálható a PCS-en. λ_2 jelöli azon igények útját (szaggatott vonallal rajzolva), melyek nem találhatóak a proxy-szerveren, így ezen igények továbbítódnak a távoli web-szerverhez. λ_3 reprezentálja a web-szerverhez érkező összes igény érkezési intenzitását.

A web-szerverhez érkező igényeknek először fel kell állítaniuk egy TCP kapcsolatot. Legyen I_s ezen egyszeri inicializáláshoz szükséges idő. A várakozó igények tárolására szolgáló puffer kapacitását jelöljük K-val. Annak a valószínűsége, hogy a beérkező igényt a szerver elutasítja legyen P_b.

A web-szerver hatékonyságát a következő három jellemzővel írhatjuk le, lásd [2,4]: a szerver kimenő pufferének kapacitása B_s, a statikus szerver idő Y_s valamint R_s a dinamikus szerver arány. Az M/M/1/K sorbanállási modell alapján meghatározható a P_b blokkolási valószínűség, vagyis annak a stacionárius valószínűsége, hogy egy érkező igény a rendszerben K igényt talál, lásd [1],

$$P_b = \frac{(1-\rho)\rho^K}{1-\rho^{K+1}} \quad (4)$$

ahol

$$\rho = \frac{\lambda_3 F\left(Y_s R_s + B_s\right)}{R_s B_s}. \quad (5)$$

Így a web-szerverhez érkező igények Poisson-folyamatot alkotnak

$$\lambda_4 = (1-P_b)^*\lambda_3 \quad (6)$$

intenzitással. Az előzőekhez hasonlóan a proxy cache szerver karakterisztikáját a B_{xc}, Y_{xc}, R_{xc} paraméterhármassal határozhatjuk meg.

Ha a felhasználó által kért fájl mérete nagyobb, mint a szerver kimenő puffere, akkor egy visszacsatolási ciklus kezdődik, mely addig tart, míg az igény kiszolgálása be nem fejeződik. Legyen

$$q = \min\left(1, \frac{B_s}{F}\right)$$

annak a valószínűsége, hogy a szerver az igényt elsőre ki tudja szolgálni és nem következik be visszacsatolási ciklus. Ezt felhasználva az egyensúlyi egyenleteket kapjuk:

$$\lambda_4 = q\ \lambda_4' \qquad (7)$$

ahol λ_4' a web-szerver kiszolgáló egységéhez érkező igények intenzitása, figyelembevéve az esetleg bekövetkező visszacsatolást.

Jelölje T_{xc} valamint T a válaszidőt PCS használata esetén, illetve annak hiányában.

Bose és Cheng [2] gondolatmentét követve meghatározhatjuk a T_{xc} valamint T értékeit, nevezetesen (8):

$$T_{xc} = \frac{1}{\frac{1}{I_{xc}} - \lambda} + p\left(\frac{1}{\frac{B_{xc}}{F\left(Y_{xc} + \frac{B_{xc}}{R_{xc}}\right)} - \lambda_1} + \frac{F}{N_c}\right) +$$

$$+(1-p)\left(\frac{1}{\frac{1}{I_s} - \lambda_3} + \frac{1}{\frac{B_s}{F\left(Y_s + \frac{B_s}{R_s}\right)} - \frac{\lambda_4}{q}} + \frac{F}{N_s} + \frac{1}{\frac{B_{xc}}{F\left(Y_{xc} + \frac{B_{xc}}{R_{xc}}\right)} - \lambda_5} + \frac{F}{N_c}\right)$$

valamint,

$$T = \frac{1}{\frac{1}{I_s} - (\lambda + \Lambda)} + \frac{1}{\frac{B_s}{F\left(Y_s + \frac{B_s}{R_s}\right)} - \frac{(1-P_b)(\lambda+\Lambda)}{q}} + \frac{F}{N_s} + \frac{F}{N_c}$$

A fenti formulákhoz az alábbi magyarázatot fűzzük. A T_{xc} válaszidő három részből tevődik össze: az első annak az időtartama, míg eldől, hogy a proxy-szerver tartalmazza-e az igényelt fájlt. Ez a sorbanállás elméletből jól ismert M/M/1 folyamat várakozási idejéből adódik, ahol λ az érkezési intenzitás valamint $1/I_{xc}$ a kiszolgálási idő.

A képlet második tagja annak a válaszideje, amikor az igény megtalálható a PCS-en, ahol a proxy-szerver kiszolgálási ideje $\frac{B_{xc}}{F^*\left(Y_{xc} + \frac{B_{xc}}{R_{xc}}\right)}$,

valamint F/N_c az „utazási” idő míg a dokumentum keresztül jut a kliens hálózatán (N_c a kliens sávszélessége).

A képlet harmadik tagja reprezentálja annak az igénynek a válaszidejét, mely nem található meg a PCS-en. Ez további három részre bontható. Az első az egyszeri TCP inicializáláshoz szükséges idő, a második a Web-szervernél töltött idő, ahol a szerver kiszolgáló egységéhez érkező igények érkezési intenzitása $\lambda_4' = \lambda_4/q$.

A harmadik tag harmadik része, a PCS-hez visszaérkező igények kliens felé való továbbításának az időtartamát reprezentálja.

PCS nélkül a modellünk a fentebb tárgyalt esetnek a leegyszerűsített változata.

3. Numerikus eredmények

A numerikus számításokhoz a Bose és Cheng [2] cikkben közölt paraméter értékeket használtuk: $I_s = I_{xc} =$ 0.004 másodperc, $B_s = B_{xc} =$ 2000 byte, $Y_s = Y_{xc} =$ 0.000016 másodperc, $R_s = R_{xc} =$ 1250 Mbyte/s, $N_s =$ 1544 kbit/s és $N_c =$ 128 kbit/s.

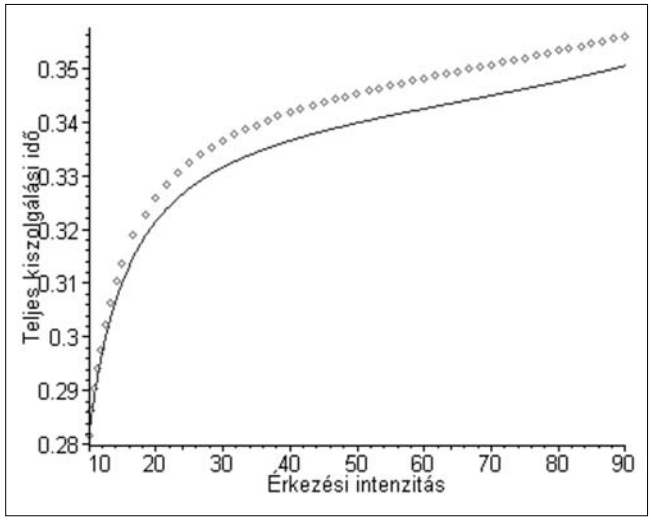
Az ábrákon a teljes válaszidőt a belső igények érkezési intenzitásának függvényében ábrázoltuk. Az összes tárgyalt grafikonon szaggatott vonal jelenti a teljes válaszidőt PCS létezésekor, míg a folyamatos vonal a PCS nélküli válaszidőt mutatja.

A *2. és 3. ábra* esetén a PCS találati valószínűsége 0.1, a keresett dokumentum mérete 5000 byte míg a web-szerver kapacitása 100 igény volt.

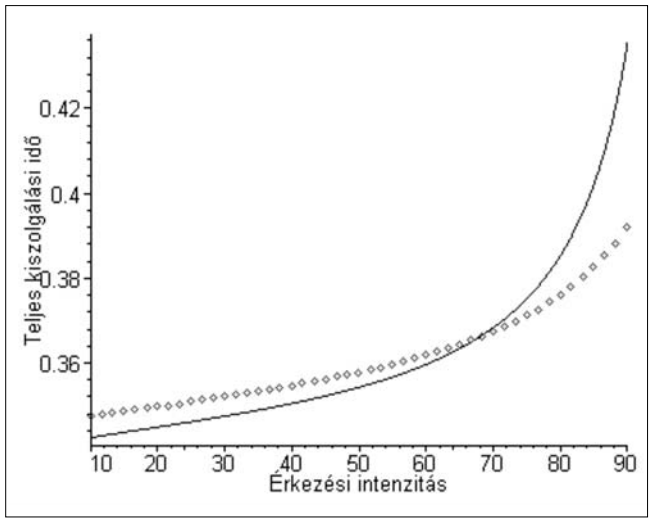
Mint látható, amennyiben a külső érkezési intenzitás 100 igény/másodperc (2. ábra), úgy a PCS beiktatása nagyobb válaszidőket eredményez. Azonban ha megnöveljük a külső érkezési intenzitást 150 igény/másodpercre (3. ábra) érdekesebb válaszidőket kapunk:

Kis belső érkezési intenzitás esetén (λ<70) a proxy-szerver használata nagyobb válaszidőket eredményez. Viszont ha a belső igények érkezési intenzitása nagyobb mint 70 igény/másodperc a PCS haszna egyértelművé válik.

2. ábra Teljes kiszolgálási idő, p=0.1, F=5000, Λ=100, K=100



3. ábra Teljes kiszolgálási idő, p=0.1, F=5000, Λ=150, K=100

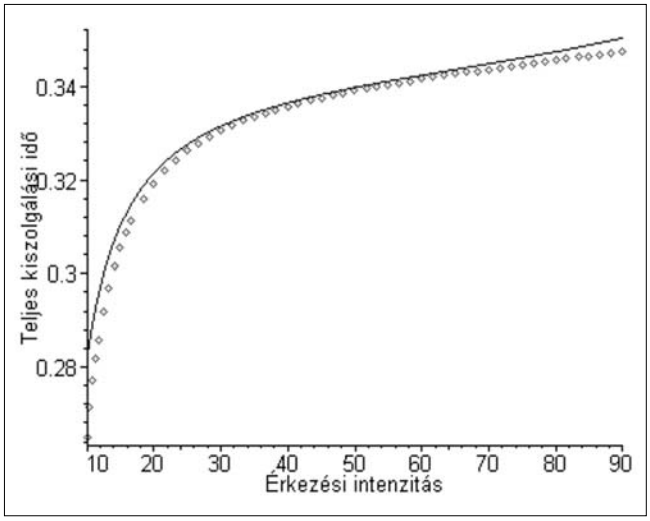


A következő két ábrán *(4. ábra, 5. ábra)* minden paramétert változatlanul hagytunk, kivéve a találati valószínűséget, melyet mindkét grafikon esetében 0.25-re emeltünk.

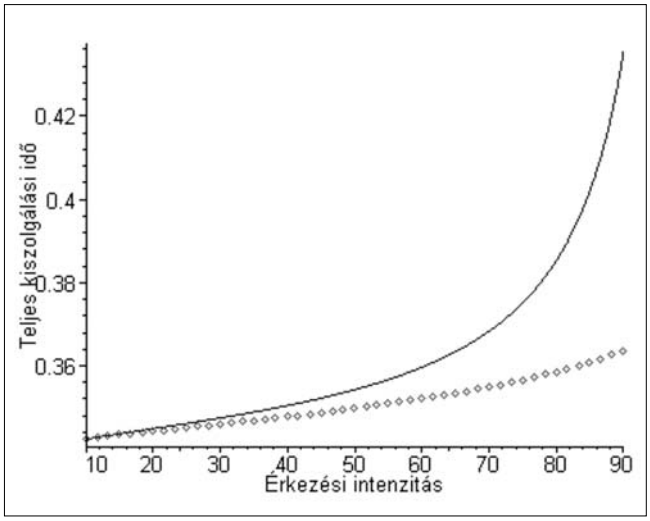
Összehasonlítva a 2. és 3. ábrákat, láthatjuk, hogy a találati valószínűséget növelve kis külső érkezési intenzitás esetén is minimális előny mutatkozik PCS használata esetén. Az 5. ábrát megvizsgálva láthatjuk, hogy nagyobb külső intenzitást és nagyobb találati valószínűséget használva (Λ=150, p=0.25) a PCS jelenléte minden esetben kisebb válaszidőt eredményez.

Mint ahogyan a numerikus eredményekből látszik, annak eldöntése is, hogy megéri-e egy proxy cache szervert üzemeltetni, nagyban függ az Internetet használók szokásaitól. Amennyiben a proxy-szervert használók nagy valószínűséggel ugyanazokat a dokumentumokat akarják letölteni, vagy olyan oldalak iránt érdeklődnek, melyek igen leterheltek, a PCS használata számottevő javulást eredményezhet a válaszidők tekintetében.

4. ábra Teljes kiszolgálási idő, p=0.25, F=5000, Λ=100, K=100



5. ábra Teljes kiszolgálási idő, p=0.25, F=5000, Λ=150, K=100



4. Összefoglaló

A Bose és Cheng [2] cikkben tárgyalt sorbanállási modellt úgy módosítottuk, hogy figyelembe vettük a webszerverhez érkező azon igényeket is, melyek nem a vizsgált proxy cache szerver felől érkeznek, valamint a web szerver realisztikusabb vizsgálata érdekében feltételeztük, hogy a szerver véges kapacitású.

A proxy cache szerver hatékonyságának vizsgálatához valós paraméterek felhasználásával [2,4] kiszámoltuk a válaszidőt arra az esetre, amikor használtunk PCS-t illetve arra, amikor nem. A numerikus eredmények alapján látszik, hogy annak eldöntése, hogy egy cég vagy intézmény számára érdemes-e proxy cache szervert üzemeltetni, nagyban függ az internetezési szokásoktól: milyen terheltségű oldalakat látogatnak, milyen gyakorisággal térnek vissza ugyanarra a webhelyre stb.

Általánosságban elmondhatjuk, hogy az érkezési intenzitást növelve a válaszidők is nőni fognak függetlenül attól, hogy telepítettünk-e PCS-t vagy nem. Abban az esetben, ha a PCS találati valószínűsége kicsi, valamint a látogatott web-szerver kis terheltségű, egyértelműen látszik, hogy PCS-t használva nagyobb válaszidőket kapunk (2. ábra). Még abban az esetben sem egyértelmű a PCS használatának előnye, ha a külső érkezési intenzitást 50%-al növeltük. Ebben az esetben ha a cég felől érkező igények intenzitása nagyobb mint 70 igény/másodperc, PCS-t használva kisebb válaszidőket kapunk. (3. ábra). Ha nagyobb a találati valószínűség valamint a külső igények érkezési intenzitása legalább 150 a proxy cache szerver használatának előnye egyértelmű.

A numerikus eredményeket vizsgálva láthatjuk, hogy a külső igények figyelembe vétele nagyban befolyásolja a kapott válaszidőket. Ezen igények növelésével a web-szerver terheltsége nő, ezáltal a PCS használatának előnye jobban megmutatkozik, főleg akkor amikor a találati valószínűség legalább 0.25.

Irodalom

[1] Bolch, G.–Greiner, S.–de Meer H.–Trivedi K.S.:
Queueing Networks and Markov Chains.
John Wiley and Sons, New York, 1998.

[2] Bose, I.–Cheng, H.K.:
Performance models of a firms proxy cache server.
Decision Support Systems and Electronic Commerce, 29 (2000), pp.45–57.

[3] CacheFlow Inc.: CacheFlow White Papers (1999).
http://cacheflow.com/technology/

[4] Menasce, D.A.–Almeida, V.A.F.:
Capacity Planning for Web Performance:
Metric, Models and Methods.
Prentice Hall., 1998.

[5] Slothouber, L.P.:
A model of Web server performance.
5th International World Wide Web Conference,
Paris, France, 1996.