

6. gyakorlat – Teljesítménymodellezés és statisztika – Megoldások

Figyelem: Jelen anyag belső használatra készült megoldási útmutató, melyet a ZH felkészülés segítése érdekében publikáltunk. A feladatok részletesebb megoldása magyarázattal gyakorlaton hangzott el.

Dimenzióanalízis

A teljesítménymodellezés feladatok megoldása során érdemes a fizikából ismert dimenzióanalízist¹ elvégezni. Pl. a négyzetes úttörvény:

$$s = v_0 t + \frac{a}{2} t^2 \quad (1)$$

Dimenziókkal:

$$s[\text{m}] = v_0 \left[\frac{\text{m}}{\text{s}} \right] t[\text{s}] + \frac{a}{2} \left[\frac{\text{m}}{\text{s}^2} \right] t^2[\text{s}^2] = v_0 t[\text{m}] + \frac{a}{2} t^2[\text{m}] \quad (2)$$

A dimenzióhasználat fő motivációja, hogy ha a dimenziók nem stimmelnek, akkor a képletet is biztosan elrontottuk valahol.² A dimenzióanalízis gyakran segít a megfelelő képlet kiválasztásában. Fontos, hogy a “darab”, “kérés” stb. jellegű mértékegységek nem számítanak külön dimenziónak, ezért pl. a $\frac{\text{kérés}}{\text{s}}$ és az $\frac{1}{\text{s}}$ dimenziók megegyeznek.

Alapképletek

Little-törvény:

$$N = X \cdot T \quad (3)$$

$$N [1] = X \left[\frac{1}{\text{s}} \right] \cdot T [\text{s}] \quad (4)$$

Kihasznátság intuitíven és a Little-törvényből *egyetlen kizárólagos* erőforráspéldány esetén:

$$U = \frac{X}{X_{\max}} = \frac{T_{\text{busy}}}{T_{\text{measured}}} = N = X \cdot T \quad (5)$$

Átbocsátóképesség végrehajtási időből *egyetlen kizárólagos* erőforráspéldány esetén (az átbocsátóképesség az elérhető legnagyobb átbocsátás, vagyis ilyenkor a kihasználtság 100%):

$$X = \frac{U}{T} \Rightarrow X_{\max} = \frac{1}{T} \quad (6)$$

1. feladat

Egy diszk 50 kérést szolgál ki másodpercenként. Minden kérés kiszolgálása 0,005 másodpercet vesz igénybe. A rendszerben nincs átlapolódás.

- Mekkora a kihasználtság?
- Mekkora a maximálisan kiszolgálható terhelés (érkezési ráta)?

Megoldás

- Az erőforrás kihasználtsága $U = X \cdot T$, ahol X az átlagos átbocsátás és T az átlagos kiszolgálási idő. Tehát $U = 0,25$, így 25%-os a kihasználtság.

A feladat józan ésszel is megoldható: a diszknak másodpercenként 50 kérés $\cdot 0,005 \frac{\text{s}}{\text{kérés}}$ -t kell dolgoznia. Ha másodpercenként 0,25 másodpercet dolgozik, akkor 25% a kihasználtsága.

¹ [Dimenzióanalízis \(Wikipédia\)](#)

² Ajánlott olvasmány: [what if? – Droppings](#)

- b. Ekkor a kihasználtság $U = 1$. Ekkor $X_{\max} = \frac{U}{T} = 200 \frac{\text{kérés}}{\text{s}}$. Vagyis a szabály egyetlen, átlapoló-dásmentes feldolgozó egységre: $X_{\max} = \frac{1}{T} = \frac{1}{0,005 \text{ s}} = 200 \frac{\text{kérés}}{\text{s}}$.

2. feladat

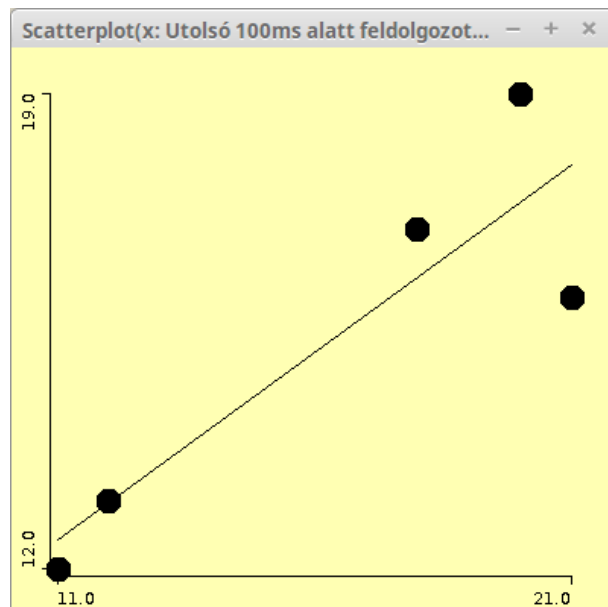
Egy szerveren az alábbi teljesítményjellemzőket mértük:

Mintavétel időpontja [ms]	500	600	700	800	900
Utolsó 100ms alatt feldolgozott kérések száma [darab]	11	12	21	18	20
Utolsó 100ms átlagos kiszolgálási ideje [ms]	15	20	21	25	27
Utolsó 100ms CPU kihasználtság [%]	12	13	16	17	19
Utolsó 100ms HDD I/O kihasználtság [%]	55	63	87	61	73

- Ábrázoljuk a feldolgozott kérések számát és a CPU kihasználtságot pontfelhő (scatterplot) diagramon! Értelmezzük a diagramot!
- A rendelkezésre álló adatok alapján a szervert melyik erőforrása tűnik a szűk keresztmetszetnek?
- Az első mintavétel idején mekkora az átbocsátási ráta értéke? Az 5 mintavétel alapján mekkora az átbocsátási ráta tapasztalati átlaga és mediánja? Mi tartozik a 40%-os kvantilisbe?
- Ezen 5 mérés alapján milyen becslést tudunk adni az egyszerre kiszolgálás alatt lévő kérések átlagos számára?

Megoldás

- Két klaszter (csoportosulás) látszik, nagyjából pozitív a korreláció (kb. arányosak az adatok), de nem direkt monoton (valami más is befolyásolhatja az adatokat, ezért ingadozik). Értelmezve a látottakat a CPU átlagos kihasználtsága a feldolgozott kérések számával nő. A bal alsó csoport kisebb terhelésű pillanatokot tartalmaz, míg a jobb felső nagyobbakat. Ez a megfigyelés adott esetben jó alapja lehet a terhelés vizsgálatának (pl. az egyes csoportokhoz tartozó pontok időben is közel vannak-e egymáshoz).



- A HDD kihasználtsága a legnagyobb. A terhelés felskálázásával először a HDD fog telítődni.
- A mintavételi időkből látszik, hogy két mintavétel között 100 ms telik el. Ebből

$$X_1 = \frac{k_1}{\Delta t} = \frac{11 \text{ kérés}}{100 \text{ ms}} = \frac{11 \text{ kérés}}{100 \text{ ms}} \left[\frac{1000 \text{ ms}}{1 \text{ s}} \right] = 110 \frac{\text{kérés}}{\text{s}}. \quad (7)$$

A tapasztalati átlag kiszámítása történhet a másik négy átbocsátás kiszámításával és átlagolással, vagy a következő módon (kihasználva, hogy Δt végig 100ms):

$$\bar{k} = \frac{\sum_{i=1}^n k_i}{n} = \frac{11 + 12 + 21 + 18 + 20}{5} = 16,4 \quad (8)$$

Ebből az átlagos átbocsátás $\bar{X} = \frac{\bar{k}}{\Delta t} = \frac{16,4}{0,1} = 164 \frac{\text{kérés}}{\text{s}}$

Az elemek sorba állítva 11, 12, 18, 20, 21, ebből rögtön látszik, hogy a medián 18.

A p kvantilis definíció szerint az a szám, amelynél az elemek p -ed része kisebb vagy egyenlő³. A p kvantilisba azok az elemek tartoznak, amelyek kisebb vagy egyenlők a p kvantilisnál. A kvantilis speciálisabb változata a percentilis, amely egész százalékokkal dolgozik, valamint a kvartilis, amely “negyedeli” az adatot. Pl. a 35. percentilis a 35%-os kvantilisnak felel meg (a kvantilis lehetne pl. 35,7% is!), a második kvartilis pedig az 50%-os kvantilisnak.

Itt az elemek legkisebb 40%-a a 11 és a 12, ezért a 40%-os kvantilis értéke a 12 lesz, és a 11, illetve 12 elemek tartoznak bele.

- d. Az utolsó 100 ms alatt feldolgozott kérések számából és az átlagos kiszolgálási időből adódik. Mivel az átlagos kiszolgálási idő különböző elemszámú adathalmazokból került kiszámításra, egyszerű átlagolásuk helyett a feldolgozott kérésekkel súlyozott átlagukat kell vennünk.

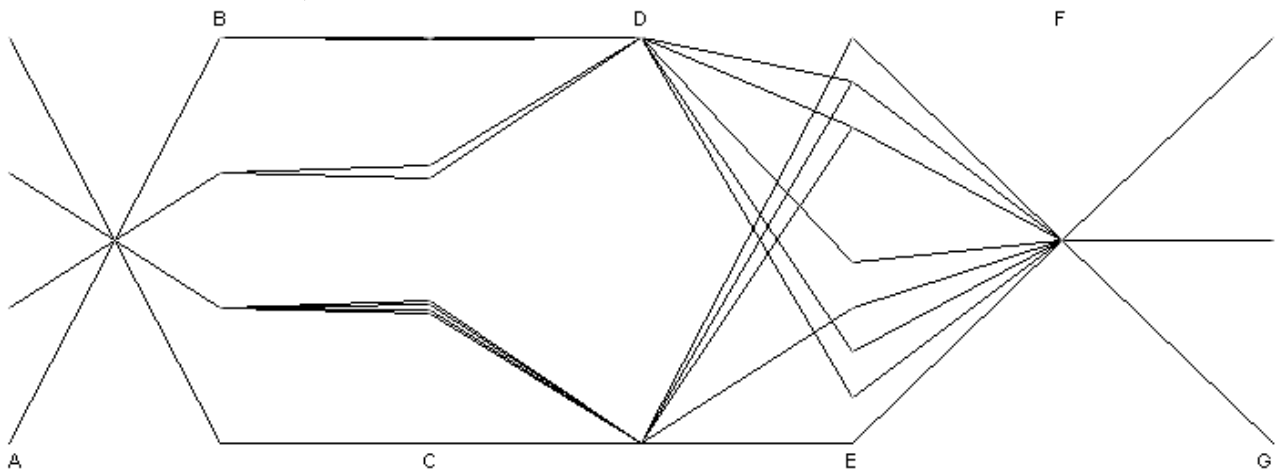
$$T = \frac{\sum_{i=1}^n k_i t_i}{\sum_{i=1}^n k_i} = \frac{11 \cdot 15 + 12 \cdot 20 + 21 \cdot 21 + 18 \cdot 25 + 20 \cdot 27}{11 + 12 + 21 + 18 + 20} = 22,39 \text{ ms} \quad (9)$$

A rendszer egyensúlyi állapotban van, ezért a c) feladatban kiszámolt átlagos átbocsátással alkalmazhatjuk a Little-törvényt:

$$N = \bar{X} \cdot T = 164 \frac{1}{\text{s}} \cdot 22,39 \text{ ms} = 164 \frac{1}{\text{s}} \cdot 0,02239 \text{ s} = 3,67196 \quad (10)$$

3. feladat

Alternatív algoritmusok teljesítményjellemzőit mérjük az $A-G$ algoritmusok többszöri futtatásával. Az alábbi ábrán ezen algoritmusok átlagos memóriagigéjét hasonlítjuk össze 10 különböző benchmark (töröttvonallal ábrázolva) segítségével.⁴



- a. Hogy hívják ezt a diagramot?

³ A matematikai statisztika elemei

⁴ Egy népszerű benchmark halmaz pl. a [The Computer Language Benchmarks Game](#).

- Mi az F algoritmus jellegzetessége?
- Hogyan viszonyul egymáshoz az A és B algoritmusok memóriaigénye?
- Az F algoritmuson kívül mely algoritmus(ok) viselkedése tér el a többitől?
- Elegendő információt szolgáltat-e ez a diagram ahhoz, hogy válasszunk az algoritmusok között?

Megoldás

- Párhuzamos koordináták diagram (*parallel coordinate plot*).
- Konstans memóriával dolgozik, mivel minden benchmarkhoz tartozó vonal egy pontban metszi az F -hez tartozó tengelyt. A párhuzamos koordináták jellegzetességei miatt erről az ábráról nem leolvasható, hogy pontosan mennyivel.
- Fordítottan arányos, mert a két tengely között futó vonalak egy pontban metszik egymást. Ha a vonalak meghosszabbításai a két tengelyen kívül metszik egymást, vagy teljesen párhuzamosak, akkor egyenes arányosságot állapíthatnánk meg (mint pl. B és C között).
- Az A algoritmus memóriaigénye kb. fordítottan arányos a B , C , D algoritmusokkal. Az E és a G algoritmusok futásidőjének elemzéséhez interaktív elemzésre lenne szükségünk (pl. Mondrian), mivel most nem látszik, hogy (pl. az F -beli kereszteződés után) melyik szakasz melyik benchmarkhoz tartozik.
- Általában nem, pl. a futásidőt nem mutatja – ezt pl. egy scatterplottal lehetne megvizsgálni.

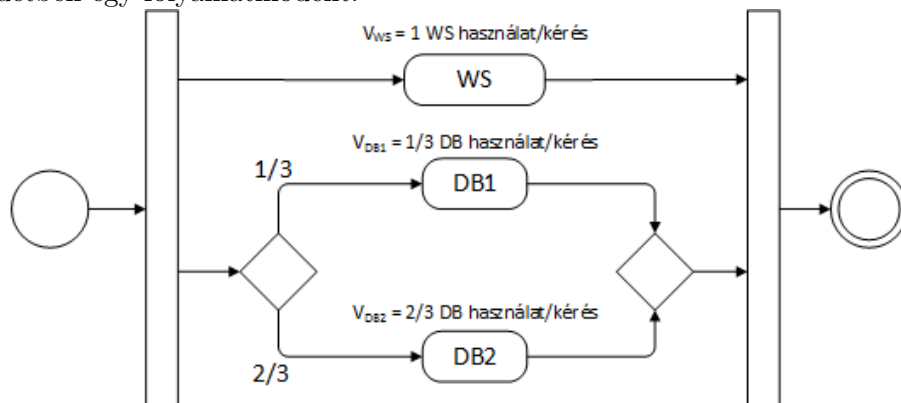
4. feladat

Adott egy webszerver (WS) és két fürtözött adatbázisszerver (DB1, DB2). A két adatbázis szerver közt súlyozott round robin terheléelosztás alapján választunk, 1:2 arányban. Minden felhasználói kérés kiszolgálása során mindkét fajta erőforrást használjuk. A csúcsidőszakban 30 percig monitorozzuk a rendszert, ezalatt 9000 kérést szolgál ki. A szervereken mért foglaltsági idők: WS – 1350 s CPU idő; DB1 – 810 s, DB2 – 1320 s diszk IO idő.

- Mekkora az egyes szerverek jelenlegi átbocsátása?
- Mennyi időt töltenek egy-egy kérés kiszolgálásával a szerverek?
- Mekkora a rendszer maximális áteresztőképessége?
- Miért nem egyféle foglaltsági időt vettünk figyelembe a két erőforrástípusnál?
- Hol csal még így is a modell?

Megoldás

Rajzoljunk kezdetben egy folyamatmodellt!



Mivel a feladatban nem volt egyéb megkötés, azt feltételeztük, hogy a kérések kiszolgálása a különböző erőforrásokon párhuzamosan történik. Ehelyett a modell lehetne szekvenciális is (az átbocsátás szempontjából nincs különbség, *de a végrehajtási időben igen!*), viszont az előbbi általánosabb, hiszen a WS használata átlapolódhat az adatbázis használatával. A valóságban persze a WS az adatbázishívás előtt és után is dolgozik, sőt, időnként még közben is. A mostani modell azt fejezi ki, hogy – pontos információ híján – ezeket a szakaszokat aggregáljuk és elfelejtjük, hogy milyen sorrendben futottak (absztrakció!).

Emlékeztető: A vizitációs számmal (többek között) a rendszer és a komponensek átbocsátása és átbocsátóképessége között tudunk váltani. Ha átbocsátással dolgozunk, akkor rendszerint a rendszer átbocsátásából számítjuk a komponensek átbocsátását – ilyenkor a vizitációs számmal szorozni kell, hiszen minden rendszerbe belépő tokent átlagosan annyszor kell feldolgoznia a komponenseknek, mint amennyi a vizitációs szám. Ha átbocsátóképességet szeretnénk számolni, akkor rendszerint a komponensek (egyszerűen számítható) átbocsátóképességéből kiindulva határozzuk meg a rendszer átbocsátóképességét – ilyenkor a vizitációs számmal osztani kell, hiszen ha minden belépő tokent annyszor kell feldolgoznia a rendszernek, mint amennyi a vizitációs szám, akkor annival kevesebb token érkezik a rendszerbe túltelítődés nélkül. Ne feledjük, hogy (többek között a szűk keresztmetszetek miatt) ebben az irányban nem elegendő a vizitációs számmal számolni, gyakran szükség van a számított értékeken végzett egyéb számításokra (pl. minimumképzésre)

a. Számoljunk először a rendszerre, aztán az erőforrásokra! A feldolgozott kérések száma $C = 9000$ (“Count”), a mérés ideje $T_m = 30$ min.

- $X_{\text{rendszer}} = \frac{C}{T_m} = \frac{9000 \text{ kérés}}{30 \text{ min}} = \frac{9000}{1800} \frac{\text{kérés}}{\text{s}} = 5 \frac{\text{kérés}}{\text{s}}$
- $X_{\text{WS}} = X_{\text{rendszer}} \cdot v_{\text{WS}} = 5 \frac{\text{kérés}}{\text{s}} \cdot 1 = 5 \frac{\text{kérés}}{\text{s}}$
- $X_{\text{DB1}} = X_{\text{rendszer}} \cdot v_{\text{DB1}} = 5 \frac{\text{kérés}}{\text{s}} \cdot \frac{1}{3} = 1,666 \frac{\text{kérés}}{\text{s}}$
- $X_{\text{DB2}} = X_{\text{rendszer}} \cdot v_{\text{DB2}} = 5 \frac{\text{kérés}}{\text{s}} \cdot \frac{2}{3} = 3,333 \frac{\text{kérés}}{\text{s}}$

b. Az egyes erőforrásokra (B a mért foglaltsági idő, “Busy time”, az egyes szerverek pedig $C \cdot v_i$ kérést dolgoznak fel):

- $T_{\text{WS}} = \frac{B_{\text{WS}}}{C \cdot v_{\text{WS}}} = \frac{1350 \text{ s}}{9000 \text{ kérés}} = 0,15 \frac{\text{s}}{\text{kérés}}$
- $T_{\text{DB1}} = \frac{B_{\text{DB1}}}{C \cdot v_{\text{DB1}}} = \frac{810 \text{ s}}{3000 \text{ kérés}} = 0,27 \frac{\text{s}}{\text{kérés}}$
- $T_{\text{DB2}} = \frac{B_{\text{DB2}}}{C \cdot v_{\text{DB2}}} = \frac{1320 \text{ s}}{6000 \text{ kérés}} = 0,22 \frac{\text{s}}{\text{kérés}}$

c. A rendszer maximális átbocsátóképessége az a legnagyobb átbocsátás, amivel egyik komponensbe sem érkezik több kérés, mint annak átbocsátóképessége. Ennek megfelelően pl. a DB1 ágra

$$X_{\text{rendszer}} \cdot v_{\text{DB1}} \leq X_{\text{DB1}}^{\max} \Rightarrow X_{\text{rendszer}} \leq \frac{1}{v_{\text{DB1}}} \cdot X_{\text{DB1}}^{\max}. \quad (11)$$

Ugyanígy DB2-re és WS-re:

$$X_{\text{rendszer}} \leq \frac{1}{v_{\text{DB2}}} \cdot X_{\text{DB2}}^{\max} \quad (12)$$

$$X_{\text{rendszer}} \leq \frac{1}{v_{\text{WS}}} \cdot X_{\text{WS}}^{\max} = X_{\text{WS}}^{\max}. \quad (13)$$

Mivel DB1 és DB2 *kötött arányú választás* (hosszú távon gyakorlatilag olyan, mintha minden “munkát” 1:2 arányban szétbontanánk és továbbküldenénk, tehát ilyen szempontból a fork-join és a szabad választás⁵ közé tehető), ezért a számított értékek minimuma érkezik meg a decision csomóponthoz túltelítés nélkül:

$$X_{\text{rendszer}} \leq \min \left(\frac{1}{v_{\text{DB1}}} \cdot X_{\text{DB1}}^{\max}, \frac{1}{v_{\text{DB2}}} \cdot X_{\text{DB2}}^{\max} \right). \quad (14)$$

A fork mindig mindkét irányba továbbküldi a kérést, és mindkét irányba a “teljes munkát” továbbítja, tehát az elágazásra számított érték és a WS-re számított érték közül a kisebb lehet a rendszer átbocsátóképessége. Ez alapján a (13) és (14) egyenlőtlenségekből a maximális átbocsátás, vagyis az átbocsátóképesség képlete:

⁵. A szabad választású döntés akármelyik irányba továbbküldheti a kérést, tehát ha az egyik ág telítésben van, nyugodtan választhatja a másikat (a kötött arányú nem). Emiatt szabad választásnál az átbocsátóképességek összeadódnak.

$$X_{\text{rendszer}}^{\max} = \min \left(X_{\text{WS}}^{\max}, \frac{1}{v_{\text{DB1}}} X_{\text{DB1}}^{\max}, \frac{1}{v_{\text{DB2}}} X_{\text{DB2}}^{\max} \right) \quad (15)$$

A feladat megoldásához tehát a komponensek átbocsátóképességeit kell kiszámolnunk:

- $X_{\text{WS}}^{\max} = \frac{1}{T_{\text{WS}}} = \frac{1}{0,15 \frac{\text{s}}{\text{kérés}}} = 6,666 \frac{\text{kérés}}{\text{s}}$
- $X_{\text{DB1}}^{\max} = \frac{1}{T_{\text{DB1}}} = \frac{1}{0,27 \frac{\text{s}}{\text{kérés}}} = 3,704 \frac{\text{kérés}}{\text{s}}$
- $X_{\text{DB2}}^{\max} = \frac{1}{T_{\text{DB2}}} = \frac{1}{0,22 \frac{\text{s}}{\text{kérés}}} = 4,545 \frac{\text{kérés}}{\text{s}}$

A rendszer maximális átbocsátóképessége ezekből:

$$X_{\text{rendszer}}^{\max} = \min \left(6,666 \frac{\text{kérés}}{\text{s}}, 3 \cdot 3,704 \frac{\text{kérés}}{\text{s}}, \frac{3}{2} \cdot 4,545 \frac{\text{kérés}}{\text{s}} \right) = X_{\text{WS}}^{\max} = 6,666 \frac{\text{kérés}}{\text{s}}$$

Érdemes megfigyelni, hogy a minimum a WS-en esett, de a DB2-höz tartozó érték ($6,818 \frac{\text{kérés}}{\text{s}}$) szintén nagyon közel van. A szűk keresztmetszet tehát jelenleg a webszerver, de csak ennek a komponensnek a fejlesztésével vagy többszörözésével csak korlátozott mértékben növelhető a teljesítmény, mert nagyon hamar a DB2 válik majd szűk keresztmetszetté.

- d. Azért, mert mind a DB szerver, mind a WS egy-egy kis rendszer önmagában is, és belül a diszk I/O, ill. a CPU bizonyul szűk keresztmetszetnek jelen esetben. Más rendszerben, más feladatot végrehajtva lehet, hogy az egyik erőforrás hálózati linkje, míg a másik erőforrás RAM sávzélessége fog szerepelni. Vegyük észre, hogy ez egy absztrakció, melynek célja a számítások egyszerűsítése a nem (vagy kevésbé) releváns adatok eltávolításával, ami abból indul ki, hogy az elhanyagolt adatok hatása a megtartott adatokénál jóval kisebb (itt: a webszerver memóriája vagy merevlemez sávzélessége sokkal később telítődne, mint a processzora, de ezt már el sem érzjük, ha a processzor miatt vergődik a rendszer). Egyúttal emlékezzünk vissza a 2. feladat b) részére, ahol adatelemzéssel állapítottuk meg a potenciális szűk keresztmetszetet, vagyis a skálázódás és telítődés szempontjából legmeghatározóbb adatot.
- e. Több egyszerűsítéssel is éltünk, pl.

- lineáris skálázódást feltételeztünk, holott a valós rendszerek ennél általában rosszabbul skálázódnak (ráadásul telítődés közelében hajlamosak leromlani),
- nem vettük figyelembe a valódi rendszerben előforduló összes erőforrást (lásd előző feladat),
- feltételeztük, hogy a kéréseket statikus módon elosztva tökéletes terheléelosztást kapunk, holott ez általában nem igaz: az átlagos értékek hosszú távon a számított módon alakulnak, de rövidebb időszakokra nézve egy átlagosnál hosszabb végrehajtási idejű kérés például rövid időre telítésbe viheti a rendszert.

5. feladat

Egy sziget lakói minden reggel munkába menet átkelnek a szigetet ölelő tavon. Észak felé híd vezet, dél felé autósomp. Az irányonként egysávos híd 200 m hosszú, és 60 km/h sebességgel szabad rajta haladni, a követési távolság (hátsó lámpától hátsó lámpáig 30 m) betartása mellett. A négy komphajó egyenként 15 percnként teszi meg a sziget-szárazföld-sziget kört, és így óránként négyen együtt legfeljebb 800 autót tudnak átvinni a szárazföldre.

- a. Mekkora a híd átbocsátóképessége (észak felé)?
- b. Hány autó fér el egy komphajón?
- c. A reggeli csúcsforgalomban mekkora a szigetet elhagyó két útvonal együttes átbocsátóképessége?
- d. Ha délben a szárazföldi főutat baleset miatt lezárták, és a szigeten keresztül (a hídon, majd a kompon átkelve) terelik a forgalmat, mekkora a terelőútvonal átbocsátóképessége?

- e. Valamelyik reggel 7:00 és 8:30 között 900 autó hagyta el a szigetet komppal. Mennyi volt ebben az időszakban a kompok átbozsátása és kihasználtsága?
- f. A fenti mérésben átlagosan hány autó állt sorba egyszerre a parton, ha az autók jól időzítve, átlagosan fél perccel a beszállásuk előtt érkeztek kompkikötőhöz?

Megoldás

- a. Little törvényében az átbozsátás szerepel, nem az átbozsátóképesség – de abban a speciális esetben, amikor pont telítve van a rendszer, a kettő megegyezik:

- $N = X \cdot T \rightarrow X = \frac{N}{T}$;
- $N = \frac{200 \text{ m}}{30 \text{ m/kocsi}} = \frac{20}{3}$ kocsi;
- $T = \frac{200 \text{ m}}{60 \text{ km/h}} = \frac{0,2 \text{ km}}{60 \text{ km/h}} = \frac{0,2}{60} \text{ h}$; tehát
- $X = \frac{20/3}{0,2/60} = 2000 \frac{\text{kocsi}}{\text{h}} = X^{\max}$.

- b. Az előzőhöz hasonlóan Little törvényéből az átbozsátóképesség:

- $N = X \cdot T$; $X = 800 \frac{\text{kocsi}}{\text{h}}$;
- $T = 15 \text{ min} = 0,25 \text{ h}$; ekkor $N = 200$, tehát egyszerre 200 autó utazik. Mivel 4 hajó van, ezért egy hajóra 50 kocsi fér fel.

- c. Az együttes átbozsátóképesség a két átbozsátóképesség összege. A hídon egy irányba óránként 2000 kocsi haladhat át, tehát $2000 \frac{\text{kocsi}}{\text{h}}$ a híd átbozsátóképessége. A kompok óránként 800 autót visznek át, tehát az átbozsátóképesség $2800 \frac{\text{kocsi}}{\text{h}}$ egy irányba.

- d. A terelőút átbozsátóképessége (soros kompozíció): $X = \min(X_{\text{híd}}, X_{\text{komp}}) = 800 \frac{\text{kocsi}}{\text{h}}$.

- e. Átbozsátás: $X = \frac{K}{T} = \frac{900}{1,5} = 600 \frac{\text{kocsi}}{\text{h}}$.

Kihasználtság: $U = \frac{X}{X^{\max}} = \frac{600 \frac{\text{kocsi}}{\text{h}}}{800 \frac{\text{kocsi}}{\text{h}}} = 0,75 = 75\%$.

- f. Komphoz sorbanállásra Little-törvény: $N = X \cdot T = 0,5 \text{ min} \cdot 600 \frac{\text{autó}}{\text{h}} = 5 \text{ autó}$.

6. feladat

Vállalatunk nyilvános szakmai tudástára egymásra is hivatkozó szócikket kínál a cég termékeit világszerte használó ügyfeleknek. Egyetlen szócikk lekérésének kiszolgálásához a szervert átlagosan 60 ms-ig veszi igénybe. A szócikk megtekintése után az olvasó csak az esetek 30%-ában hagyja el az oldalt, többnyire ugyanis egy újabb szócikkre mutató hivatkozásra kattint.

- a. Egy olvasó összes tudásszomjának kielégítéséhez átlagosan mekkora szerveridő szükséges?
- b. Tekintsük úgy, hogy az egyes kérések a szerveren nem párhuzamosíthatóak. Óránként hány egyedi látogatót képes kiszolgálni a szerver?

Megoldás

- a. Egy szócikk lekérésének kiszolgálása átlagosan 60 ms, egy felhasználó pedig átlagosan $v = \frac{1}{0,3}$ szócikket tekint meg,⁶ tehát $T = 60 \frac{\text{ms}}{\text{szócikk}} \cdot \frac{1}{0,3} \frac{\text{szócikk}}{\text{felhasználó}} = 200 \frac{\text{ms}}{\text{felhasználó}}$. A v most is a vizitációs szám.

- b. Maximális eset, amikor a kihasználtság 100%, azaz $U = 1$. Ekkor $U = X \cdot T \rightarrow X = \frac{U}{T} = \frac{1}{0,2} = 5 \frac{\text{látogató}}{\text{s}}$. Óránként $3600 \text{ s} \cdot 5 \frac{\text{kéréslátogató}}{\text{s}} = 18000 \text{ látogató}$.

6. Geometriai eloszlás várható értéke (Wikipédia)