



Teljesítőképeség-vizsgálat

Számítógép-hálózatok

Dr. Lencse Gábor
egyetemi tanár

Széchenyi István Egyetem, Távközlési Tanszék
lencse@sze.hu



Tartalom

- Bevezetés: célok, alapfogalmak
- Módszerek
 - Mérés
 - Analitikus módszer
 - Szimuláció
- Eredmények kezelése
- Eredmények megjelenítése
 - Ábrázolási módok
 - Trükkök
- Gyakorlati példák

BEVEZETÉS

Nem egészen világos kérdések...

- Vonatkozhatnak egyetlen hálózatra, egy szerverre vagy akár egy országos méretű rendszerre is
 - Mire képes a rendszer?
 - Milyen a kihasználtsága (tartaléka)?
 - El fogja-e bírni a forgalmat, ha ...?
 - Milyen késleltetések, illetve késleltetésingadozások lesznek, ha ...?
 - Mekkora és milyen forgalmat engedhetünk még a rendszerre, hogy adott minőségi paramétereket tartani tudjunk?
 - Hol vannak a szűk keresztmetszetek?
 - Mit és hogyan kellene bővíteni?
 - Adott bővítés után hogyan alakulnak a jellemzők?

Néhány fogalom (definíció)

- **Felajánlott forgalom**

- Azon csomagok összessége (időbeli eloszlásukat is beleértve), melyeknek az átvitelét a forgalomforrások kérik.

- **Puffer**

- Ideiglenes tár (memória) a hálózati eszközökben. Ebben tárolják a csomagokat, amíg továbbítani nem tudják őket. Mivel a kapacitásuk véges, ezért adatvesztés történhet.

Néhány fogalom (definíció)

- **Átvitt forgalom**

- Azon csomagok összessége (időbeli eloszlásukat is beleértve), melyeknek az átvitele megtörtént.

- **Felajánlott forgalom $\neq$ átvitt forgalom! Miért?**

- A rendszer nem végtelen kapacitású

- Az átvitel során csomagvesztés lehetséges, de ha nincs is, a várakozások miatt a forgalom időbeli eloszlása megváltozik

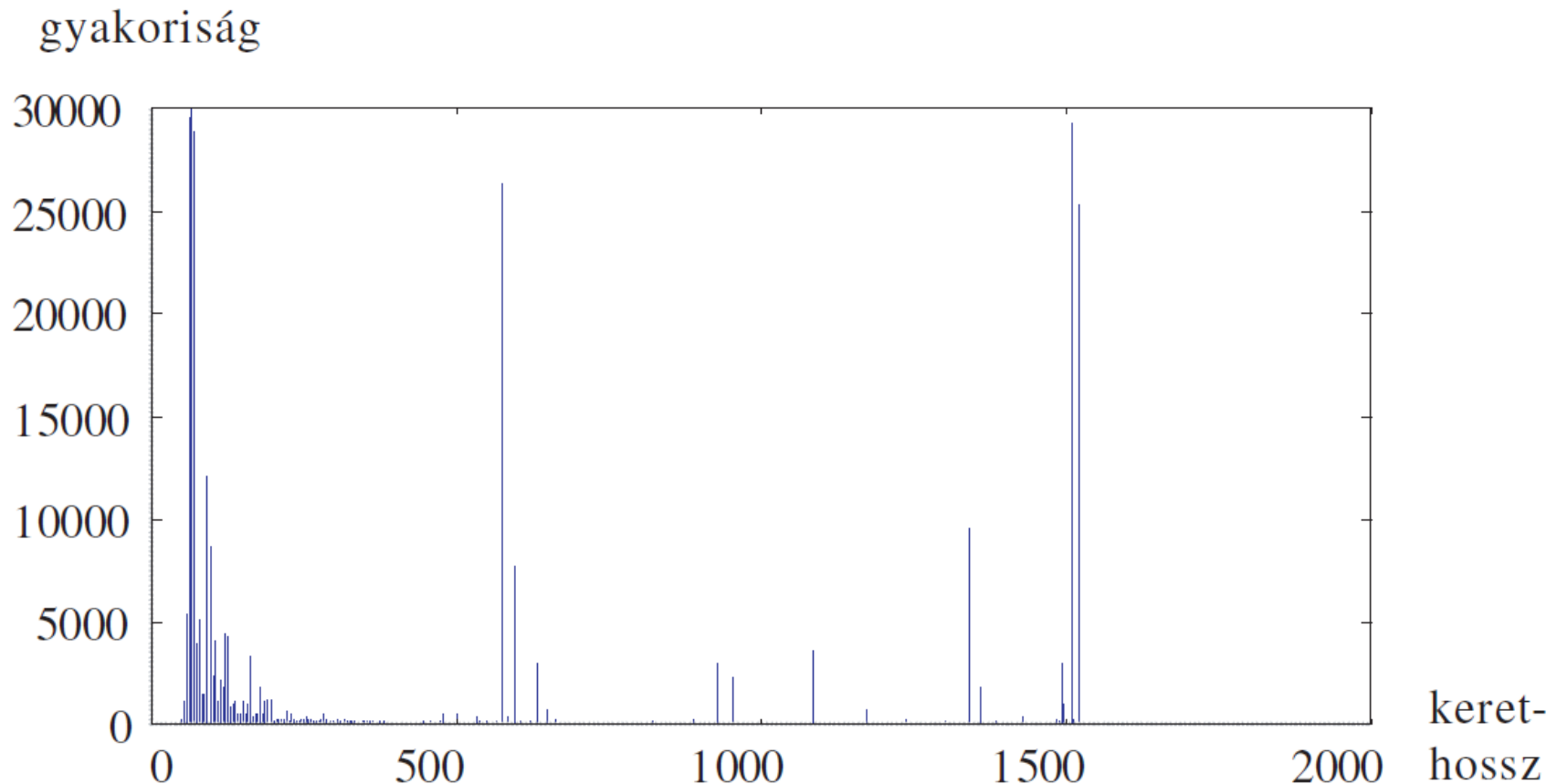
Vajon mivel jellemezhető jól a forgalom?

A forgalom jellemzése

- Egyetlen számérték általában nem elég! Miért?
 - Különböző szempontokból más-más jellemző fontos
 - időegység alatt érkező / átvitt csomagok száma
 - főként az útválasztók terhelését jellemzi
 - időegység alatt érkező / átvitt bitek száma
 - főként az átviteli csatornák / vonalak terhelését jellemzi
 - érkezési időköz eloszlása (inter-arrival time): két egymást követő csomag érkezési időpontjának különbsége.
 - Egyszerű modellekben például exponenciális eloszlásúnak tekintik.
 - csomaghossz eloszlása
 - *burst*össég: a forgalom nem független az előző időszaktól (illetve annak forgalmától), bizonyos „csomósodás” figyelhető meg.

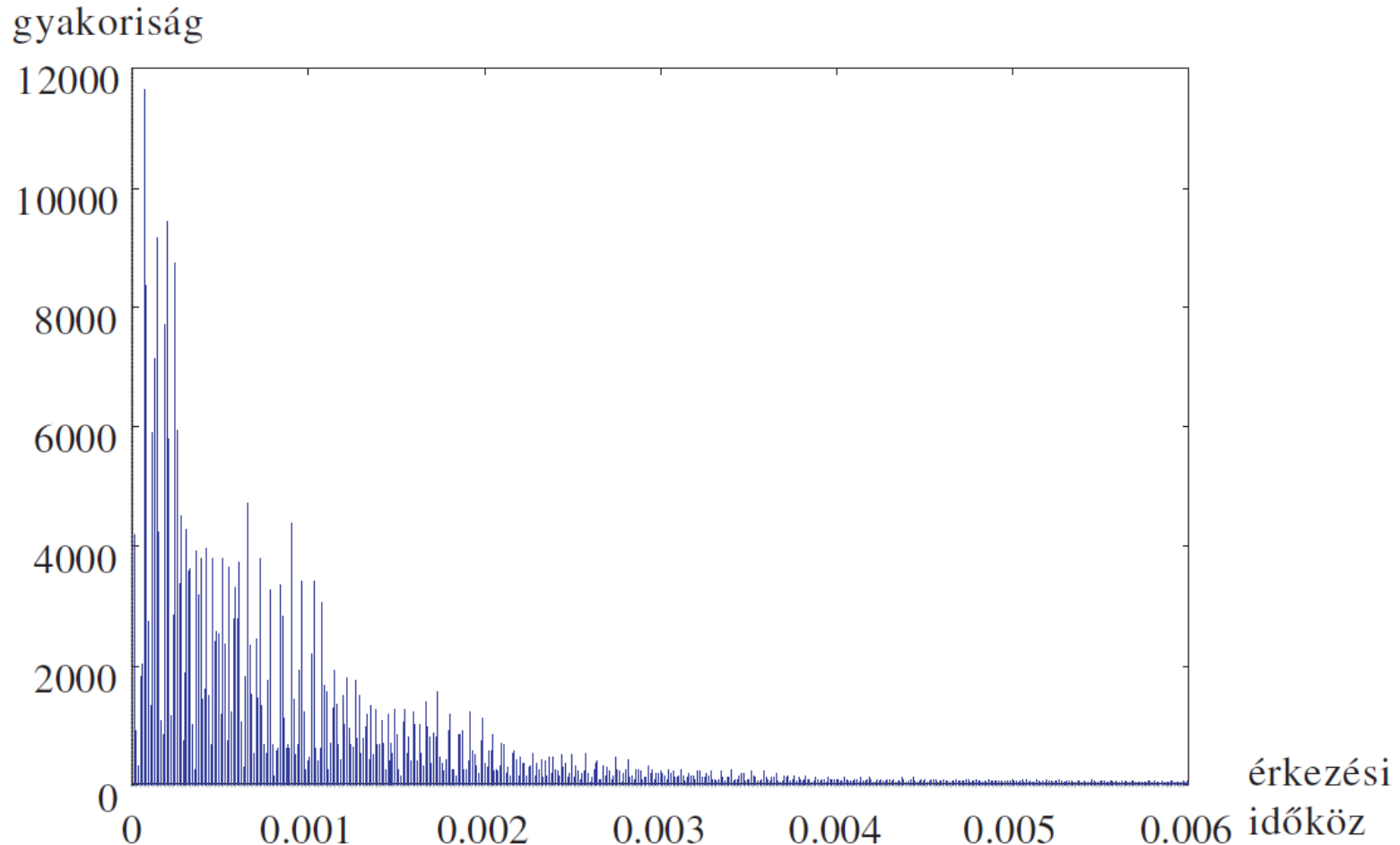
Illusztráció: csomaghossz eloszlása

A BME FDDI gerinchálózatán 1996-ban mért forgalom kerethossz statisztikája
(A 67-es hossz értéknél a gyakoriság értéke 130000 (!) körül van, ezért csonkolt.)



Illusztráció: érkezési időköz eloszlása

A BME FDDI gerinchálózatán 1996-ban mért érkezési időköz gyakoriságok statisztikája



Néhány fogalom (definíció)

- **Modellezés** (modell alkotás)
 - Olyan emberi tevékenység, amely során egy valóságos (létező vagy elképzelt) rendszernek egy valamilyen eszközkészlettel kezelhető, általában egyszerűsített változatát hozzuk létre. A modell valamilyen számunkra fontos tulajdonságban hasonlít a valós rendszerre.
 - Példa:
 - Az ember modellje lehet:
 - bábu egy áruház kirakatában (szempont: testi felépítés)
 - egér (szempont: bizonyos gyógyszerek hatása)

MÓDSZEREK

Módszerek

- Kommunikációs rendszerek teljesítőképesség-vizsgálatára alkalmazható módszerek:
 1. Mérés (a valóságos rendszeren való mérés)
 2. Analitikus módszer (matematikai számítás)
 3. Szimuláció (a rendszer modelljén végrehajtott kísérlet)

Mérés

- A legkézenfekvőbb megoldás: mérjük meg, amit tudni szeretnénk.
- Egy jól végzett méréssel (mérés sorozattal) pontos, hiteles adatokhoz juthatunk.
- Gyakran a másik két módszer esetén is szükségünk van arra, hogy a bizonyos adatokat méréssel szerezzünk meg.
- Miért nem elég?
 - Milyen hátrányai, akadályai lehetnek a mérésnek?

A mérés hátrányai, akadályai

- A mérés beavatkozást jelent a rendszerbe
 - Ez néha problémát okozhat. Például forgalmat okoz, erőforrásokat foglal, stb.
- A méréshez a rendszernek léteznie kell
 - A rendszer megépítése drága lehet, vagy a rendszert esetleg nem is lehet megvalósítani, mert például még a tervezett rendszer elemei sem léteznek.
- A mérésnek jogi akadályai vannak
- A mérésnek biztonsági akadályai vannak
 - A mérés a mérést végzőre, a mért rendszerre, vagy a rendszer adataira veszélyt jelenthetne.
- A mérést időben nem lehet kivitelezni
- A mérés túl drága lenne
 - Emberi erőforrások, műszerek ára miatt.

Néhány RFC a méréssel kapcsolatban

- RFC 2544 – „Benchmarking Methodology for Network Interconnect Devices”
 - Célja: trükközés helyett objektív mérés
 - Konkrét leírásokat ad: mit, hogyan mérjünk
 - Boltban kaphatók a szabványt megvalósító eszközök
- RFC 5180 – „IPv6 Benchmarking Methodology for Network Interconnect Devices”
 - IPv6 specifikus kiegészítés
 - Az IPv6 áttérési technológiákkal nem foglalkozik
- RFC 6349 – „Framework for TCP Throughput Testing”
 - A felhasználó tapasztalataihoz közelebb álló mérések

Néhány RFC a méréssel kapcsolatban

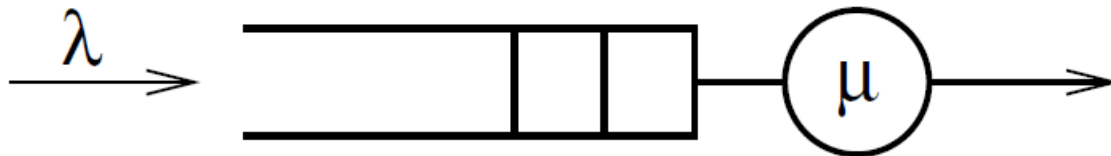
- RFC 8219 – „Benchmarking Methodology for IPv6 Transition Technologies”
 - Célja: IPv6 áttérési technológiákra vonatkozó mérési módszerek
 - A nagy számú technológiát kis számú csoportba sorolja
 - egyszeres fordítás, kétszeres fordítás, beágyazás, DNS64
 - Csoportonként ad meg vizsgálati módszereket
 - Érdeklődőknek: az RFC-t bemutató angol nyelvű előadás:
<http://seminar-materials.iijlab.net/iijlab-seminar/iijlab-seminar-20171010.mp4>
Az előadás fóliái:
https://www.tilb.sze.hu/tilb/targyak/NGD_MD096_1/iijlab-seminar-20171010.pdf

Analitikus módszer

- DEFINÍCIÓ: A vizsgált rendszer *matematikai modelljének* segítségével végzünk számításokat.
- Példa:
 - egy várakozás sor modellezése *Markov lánc*cal.
 - A várakozási sor maga is modellje lehet valamilyen hálózati eszköznek, például egy routernek!

Várakozási sor modellezése

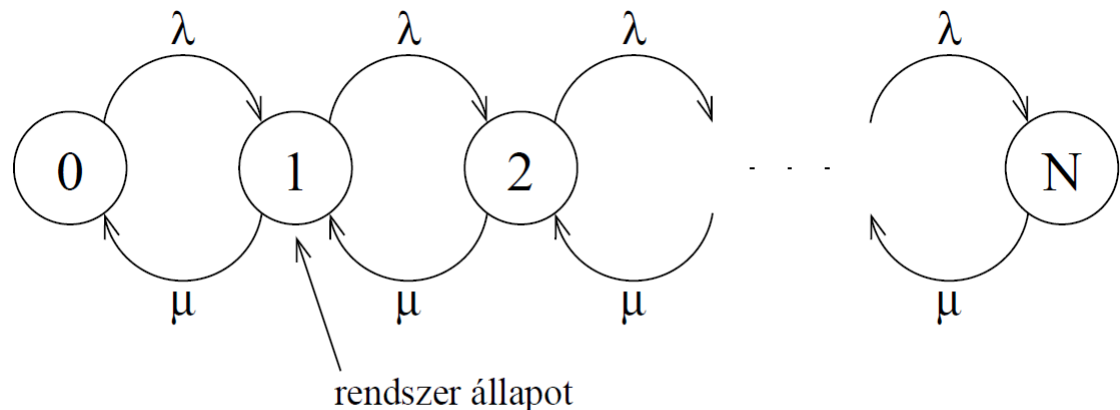
- Érkezési-kiszolgálási feladat:



- Az érkezés: λ paraméterű *Poisson folyamat* szerinti.
- A kiszolgálás ideje exponenciális eloszlás szerinti, μ paraméterrel.
- A rendszer *állapotát* a sorban álló feladatok (csomagok) száma jelenti.
 - Ez minden érkezéskor eggyel nő, és minden kiszolgálás befejezésekor eggyel csökken.
 - Kérdés lehet például, hogy állandósult állapotban mekkora a sorhossz várható értéke, vagy a csomagok hány százaléka veszik el adott (véges) méretű sor esetén

Modellezés Markov-lánccal

- A *Markov-lánc* egy matematikai objektum, számunkra most elég, hogy:
 - állapotai vannak
 - ezeket bizonyos valószínűséggel veszi fel,
 - egyik állapotából egy másik állapotába bizonyos valószínűséggel kerül át.
- A feladathoz rendelt Markov-lánc:



λ - érkezési intenzitás
 μ - kiszolgálás sebessége

Modellezés Markov lánccal

- Jelölje p_i annak a valószínűségét, hogy a rendszer az i . állapotban van (a sorban várakozó csomagok száma: i), formálisan:

$$\Pr \{a rendszer az i. állapotban van\} = p_i$$

- Állandósult állapotban ugyanakkora valószínűséggel megy át a rendszer a 0-s állapotból az 1-es állapotba, mint fordítva:

$$p_0 \cdot \lambda = p_1 \cdot \mu$$

$$\Downarrow$$

$$p_1 = p_0 \frac{\lambda}{\mu}$$

Modellezés Markov lánccal

- Az 1-es és a 2-es állapot közötti állapot átmenetekre hasonlót felírva kapjuk, hogy:

$$p_2 = p_1 \cdot \frac{\lambda}{\mu} = p_0 \cdot \left(\frac{\lambda}{\mu}\right)^2$$

- Általános kifejezéssel:

$$p_n = p_0 \cdot \left(\frac{\lambda}{\mu}\right)^n$$

Modellezés Markov lánccal

- Elvileg a sorhossz lehet végtelen, ekkor a valószínűségek összegére felírhatjuk, hogy:

$$\sum_{i=0}^{\infty} p_i = 1$$

⇓

p_0 kiszámítható.

Modellezés Markov lánccal

- Gyakorlati esetben persze a sorhossz mindig véges, jelölje N . Ha a puffer tele van, és új igény (csomag) érkezik, akkor az el fog veszni. Hasonlóképpen a valószínűségek összege:

$$\sum_{i=0}^N p_i = 1$$

⇓

p_0 ekkor is kiszámítható.

Modellezés Markov láncsal

- A módszer segítségével megválaszolható kérdések például:
 - Mekkora a sorhossz várható értéke?
 - Mekkora puffer kell, hogy a veszteség 1 % alatt legyen?

Az analitikus módszer korlátai

- az analitikus módszer pontatlan
 - Mert a modell a rendszerhez képest túl sok leegyszerűsítést, elhanyagolást tartalmaz.
- az analitikus módszer nem számolható végig
 - Mert nem létezik zárt alakú megoldás, vagy mert a közelítő megoldás számításigénye sem elégíthető ki.

Kellően nagy komplexitású rendszerek esetén az analitikus módszer alkalmatlan a rendszer viselkedésének kielégítő vizsgálatára.

Szimuláció (definíciók)

- **Szimuláció**

- Számítógép által végrehajtható modellen végzett kísérlet.

- **Emuláció**

- Valamilyen hardvert vagy szoftvert más hardverrel vagy szoftverrel helyettesítünk. Fekete dobozként (kívülről nézve) ugyanúgy működik, mint az eredeti, belső működése lehet teljesen más.

- **Röviden:**

- Szimuláció: kísérletezés
 - Emuláció: üzemszerű használat

Magyarázó példák

- Repülőgép szimulátor: nem tudunk vele repülni, csak a pilóta gyakorolhat rajta.
- Commodore 64 emulátor: Futnak rajta a régi játékprogramok, de azokat nem egy *MOS Technology 6510* CPU hajtja végre, hanem a számítógépünk processzora.

További fogalmak

- **Verifikáció**

- A szimulációs modell vagy az eredmények helyességének ellenőrzését jelenti.

- Tipikus kérdés: „Ez a modell jó?”

- **Validáció**

- Annak a vizsgálata, hogy a modell, illetve az eredmények valóban leírják-e a valóságos rendszert, amit vizsgálunk.

- Tipikus kérdés: „Ez a jó modell?”

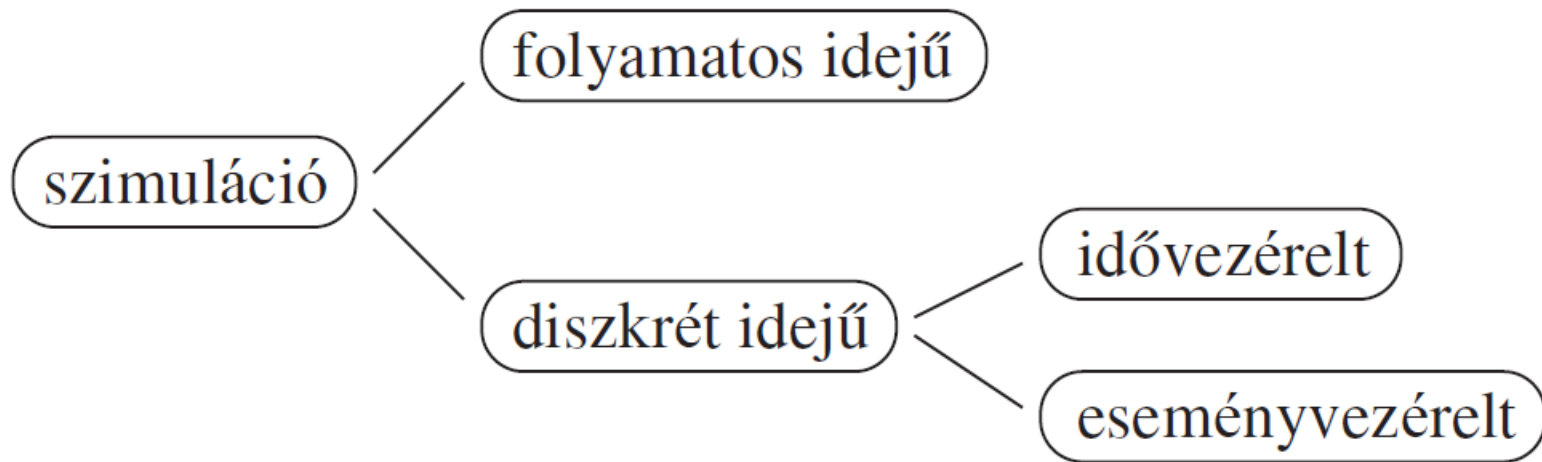
További fogalmak

- **Folytonos idejű rendszer**
 - állapota időben folyamatosan változik
 - például: víz áramlása egy csőben
- **Diszkrét idejű rendszer**
 - állapota *diszkrét* (egymástól elkülöníthető) időpontokban változik
- **Diszkrét idejű szimuláció**
 - A rendszer állapotváltozásai egymástól elkülöníthető időpontokban történnek (vagy úgy vesszük figyelembe őket).

További fogalmak

- **Eseményvezérelt szimuláció**
 - A szimuláció végrehajtása során csak azokkal az időpontokkal foglalkozunk, amelyekben a rendszer állapotát megváltoztató *esemény* történik.
- **Idővezérelt szimuláció**
 - A modellben az idő mindig valamilyen előre meghatározott Δt értékkel nő, az így adódó időpontokban meg kell vizsgálni, hogy az előző időpont óta történt-e valami a rendszerben.

A szimuláció fajtái



A diszkrét idejű szimulációt angolul úgy nevezik, hogy: *Discrete Event Simulation* (rövidítve: DES).

Idő elnevezések

- **Virtuális idő** (*virtual time*)
 - A vizsgált rendszer modelljében használt idő, éppen ezért kifejezőbb (de ritkábban használt) a *modellidő*.
- **Végrehajtási idő**
 - A szimulációt végrehajtó gép valós idő órája által mért idő, ezt angolul úgy nevezik, hogy *wall clock time* (falióra idő).

Eseményvezérelt DES

- **Jövőbeli események halmaza (Future Event Set)**
 - A szimuláció során ebben tároljuk azokat az eseményeket, amelyekről már tudjuk, hogy be fognak következni.
 - Az eseményeket időbélyeggel látjuk el; az időbélyeg megmutatja, hogy mely virtuális időpontban fog az esemény bekövetkezni.
 - Példa

időbélyeg	esemény leírása
0.012	keret adása befejeződik
0.018	keret eleje a vevőhöz megérkezik

DES algoritmus

Inicializálás, azaz bizonyos események felidőzítése
a FES-be;

REPEAT

Legkisebb időbélyegű esemény kivétele (és törlése)
a FES-ből;

MOST := a kivett esemény időbélyege;

Esemény feldolgozása, eközben esetleg újabb
esemény(ek) generálása (és behelyezése a
FES-be);

UNTIL (elfogytak az események) v (MOST > mint egy
beolvasott határ) v (egyéb ok miatt
meg kell állni);

Hálózati szimulátorok

- Kommunikációs hálózatok teljesítőképességének vizsgálatára alkalmas szimulátorok:
 - OMNeT++
 - Oktatási, kutatási célra szabadon használható
 - Érdeklődőknek: <https://omnetpp.org/>
 - OMNEST: kereskedelmi verzió, több kiegészítő eszközzel
 - Riverbed Modeler (eredetileg: OPNET Modeler)
 - IT Guru Academic Edition: 6 hónapos licence kérhető
 - NS-3
 - szabad szoftver, GPLv2 licenc alatt

Teljesítőkéesség-vizsgálat szimulációval

- Lépések

- Elkészítjük a rendszer modelljét az alkalmazott szimulátor eszközkészletének felhasználásával.

- Lehet, hogy a rendszerünk elemeinek a modelljét megtaláljuk a szimulátor elemkönyvtárában, de lehet, hogy magunknak kell megírnunk.
 - A szimulátor által nyújtott módon leírjuk a rendszer topológiáját.
 - Lehet, hogy grafikus felületen összerakhatjuk a hálózatot, de az is lehet, hogy valamilyen szöveges topológia leíró nyelvet kell használnunk.
 - Modelleznünk kell a hálózatot használó alkalmazásokat, pontosabban azok forgalmát is!

Teljesítőképeség-vizsgálat szimulációval

- További lépések
 - Ha már kész a modell
 - Meghatározzuk, hogy milyen feltételek mellett szeretnénk elvégezni a vizsgálatot, és ennek megfelelően állítjuk be a modell paramétereit.
 - Meghatározzuk, hogy milyen jellemzők érdekesek számunkra, és ennek megfelelően állítjuk be a statisztikagyűjtést.
 - Végrehajtjuk a szimulációt, közben a szimulátor statisztikát gyűjt az általunk kívánt jellemzőkről.

Teljesítőképeség-vizsgálat szimulációval

- További lépések
 - Kiértékeljük a gyűjtött statisztikákat.
 - Ennek alapján dönthetünk úgy, hogy a modellt vagy a terhelést módosítjuk, illetve szükség lehet további jellemzők megfigyelésére is.
 - Ilyen esetekben újabb szimulációt végzünk.
 - Amikor végre választ kaptunk a kérdéseinkre, akkor az eredményeinket olyan formára hozzuk, hogy az mások számára is emészthető legyen.
 - Ennek részleteivel mélyebben is foglalkozunk!

EREDMÉNYEK KEZELÉSE

Eredmények kezelése

- A „mérési” eredmények származhatnak
 - Valódi mérésből
 - Szimulációból
- Alapelv:
 - Egy mérés nem mérés!
 - A mérések/kísérletek száma függ a feladattól.
 - Például az RFC 8219 legalább 20 mérést ír elő.

Mérési eredmények „összegzése”

- Több mérés eredményét valahogyan összesítjük:
 - Átlag
 - Medián
- A kapott jellemzőtől való eltérés minősítésére
 - Szórás (a tapasztalati szórás képletét használjuk)
 - Minimum/maximum értékek
 - Percentilisek (pl. 10 percentilis és 90 percentilis)
- A teljes eloszlást is jellemezhetjük
 - Például hisztogrammal
 - Ezt meg is fogjuk tanulni! 😊

Egy egyszerű számpélda

minimum ----->	1370	137 1	1100
10 percentilis --->	1390	138 9	1390
	1410	141 1	1410
	1430	142 9	1430
	1450	145 1	1450
medián ----->	1500	1499	1500
	1550	155 1	1550
	1570	156 9	1570
	1590	159 1	1590
90 percentilis --->	1610	160 9	1610
maximum ----->	1630	163 1	1630
	átlag: 1500	1500.09	1475.455
	szórás: 94.34	94.35	150.16

A számpélda tanulságai

- Átlag (és tipikusan a szórás is)
 - Zajra érzéketlen
 - Kiugró értékre érzékeny
- Medián (és tipikusan a percentilisek is)
 - Zajra érzékeny
 - Kiugró értékre érzéketlen

Eredmények „pontossága”

- Hány tizedesjegyet tüntessünk fel?
- Példa
 - Valamilyen mennyiségre azt kaptuk, hogy:
 - átlag: 8,4786
 - szórás: 1,2345
 - Mi az utolsó két tizedesjegy értelme?
 - SEMMI!
 - Kerekítsük bátran, például így: 8,48 és 1,23

EREDMÉNYEK MEGJELENÍTÉSE

Miért érdekes?

- Számos esetben kell eredményeket mások számára megmutatnunk, például:
 - Mérési jegyzőkönyv
 - TDK dolgozat, szakdolgozat
 - Tudományos publikáció
 - Termék leírása

Milyen eszközeink vannak?

- Ha néhánynál több számérték van:
 - ábrák
 - táblázatok

Egy ábra alaki kellékei

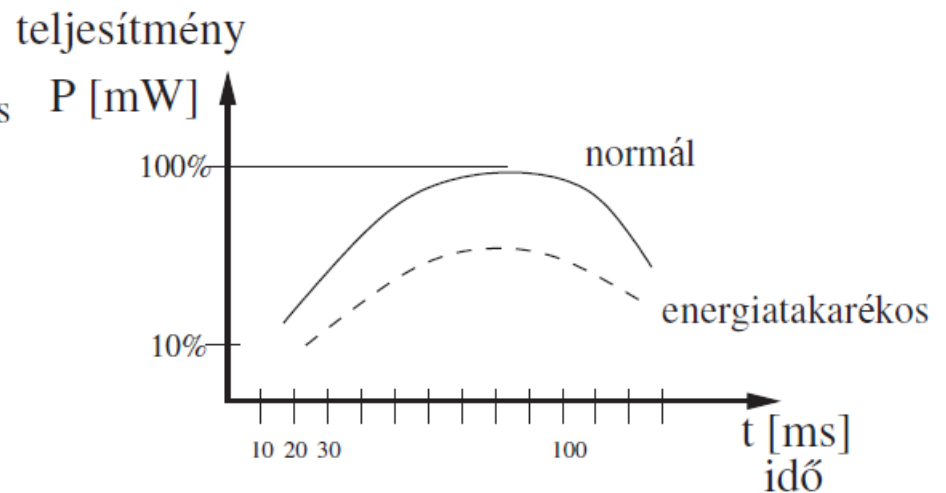
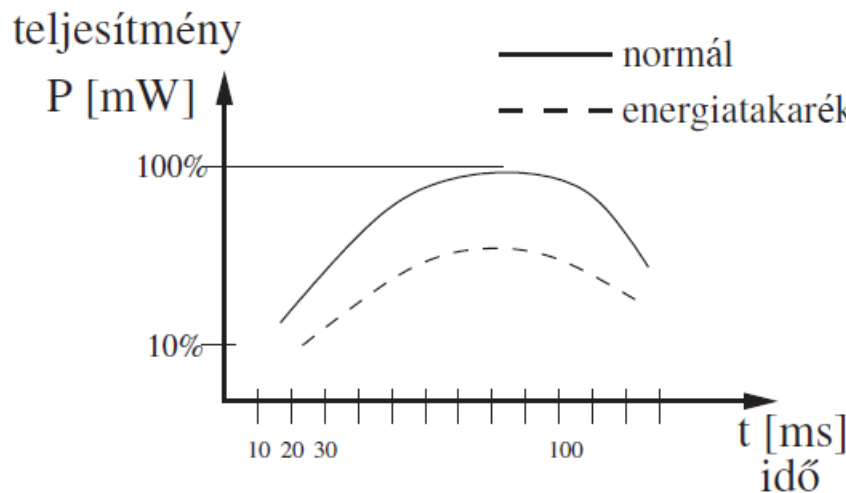
- Például egy grafikon esetén
 - cím (képaláírás): mit látunk az ábrán
 - tengelyfeliratok, mértékegységek
 - tengelyeken egységek bejelölése; derüljön ki, hogy lineáris vagy logaritmikus skálát használunk
 - ha több jellemzőt is ábrázolunk: melyik grafikon mit jelent
 - hasznos a *nevezetes értékek* (min/max hely, inflexiós pont, stb.) koordinátáinak feltüntetése.

Általános szempontok

- A grafikus megjelenítés jobban áttekinthető, mint a táblázatos, de nem helyettesíti azt.
- Az ábrába a maximális mennyiségű információt vigyük be.
- Ugyanakkor az olvasótól minimális erőfeszítést követeljen meg az eredmények áttekintése.
- Az eredmény érvényességére vonatkozó adatok is ott szerepeljenek (az ábrán, vagy az ábra címében).

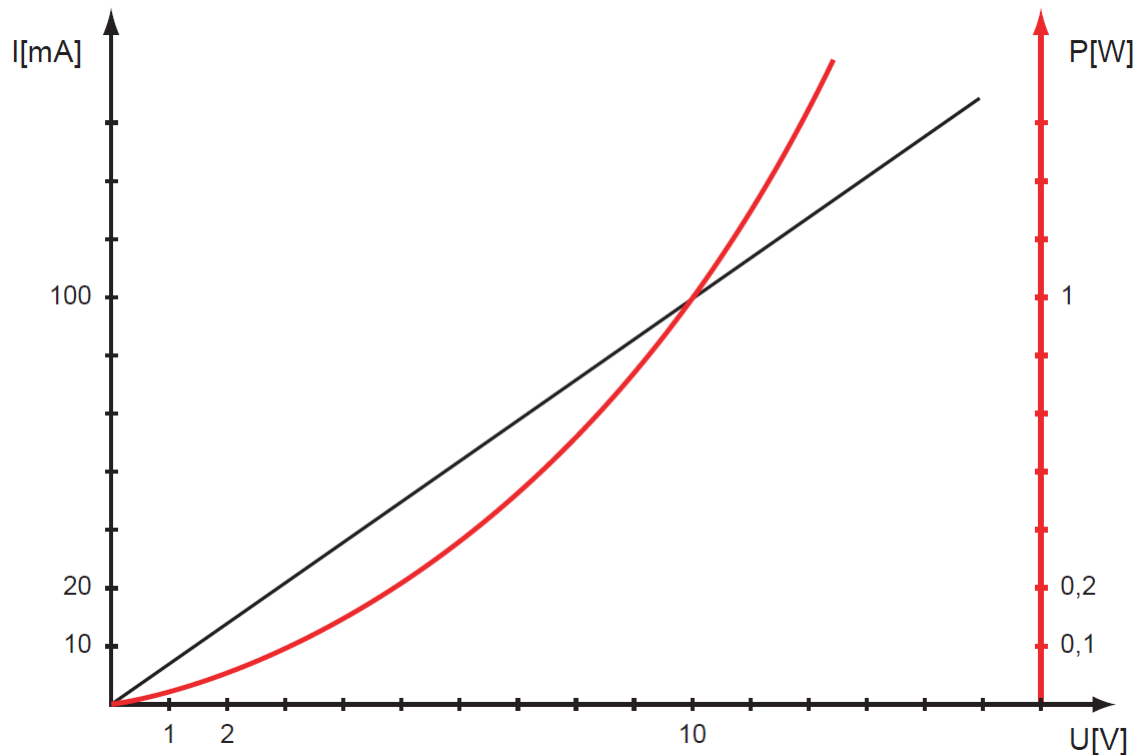
Általános szempontok

- A grafikonok jelentését külön magyarázzuk meg, vagy még jobb, ha az ábrába írjuk.



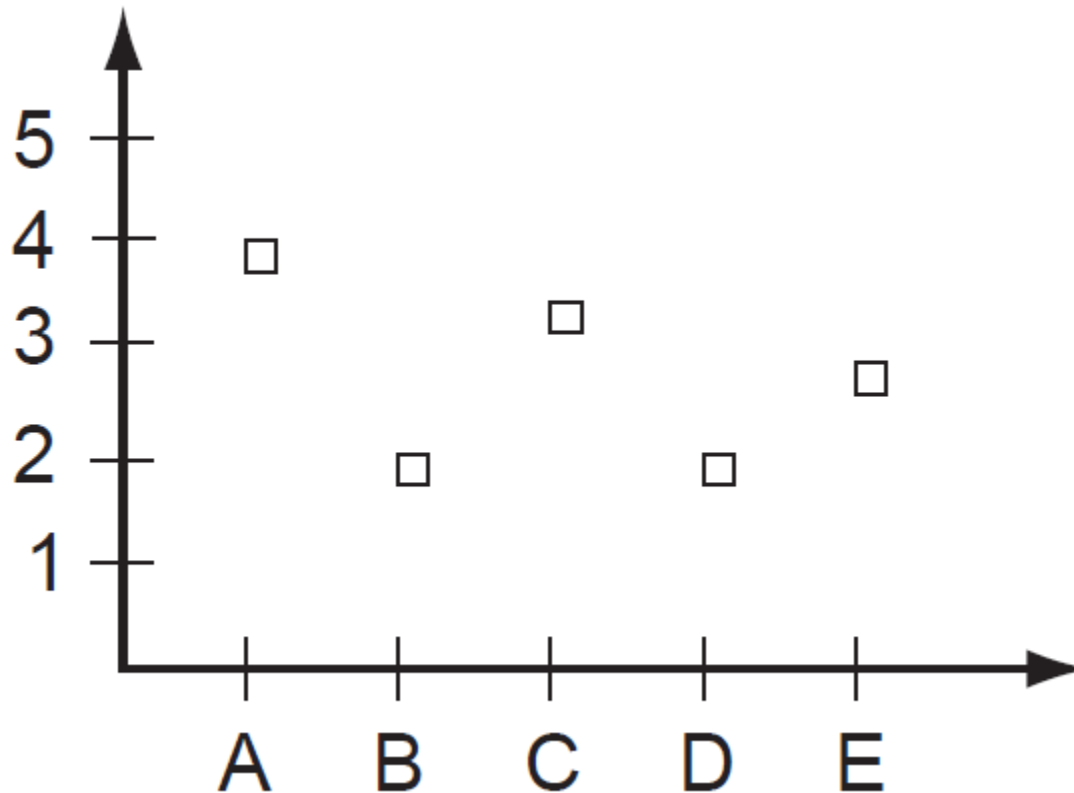
Általános szempontok

- Adott független változó függvényében akár eltérő függő változókat is ábrázolhatunk egy ábrában, de ne legyen az y tengelyen sok skála, legfeljebb kettő.



Általános szempontok

- Ne kössünk össze nem folytonos dolgokat.

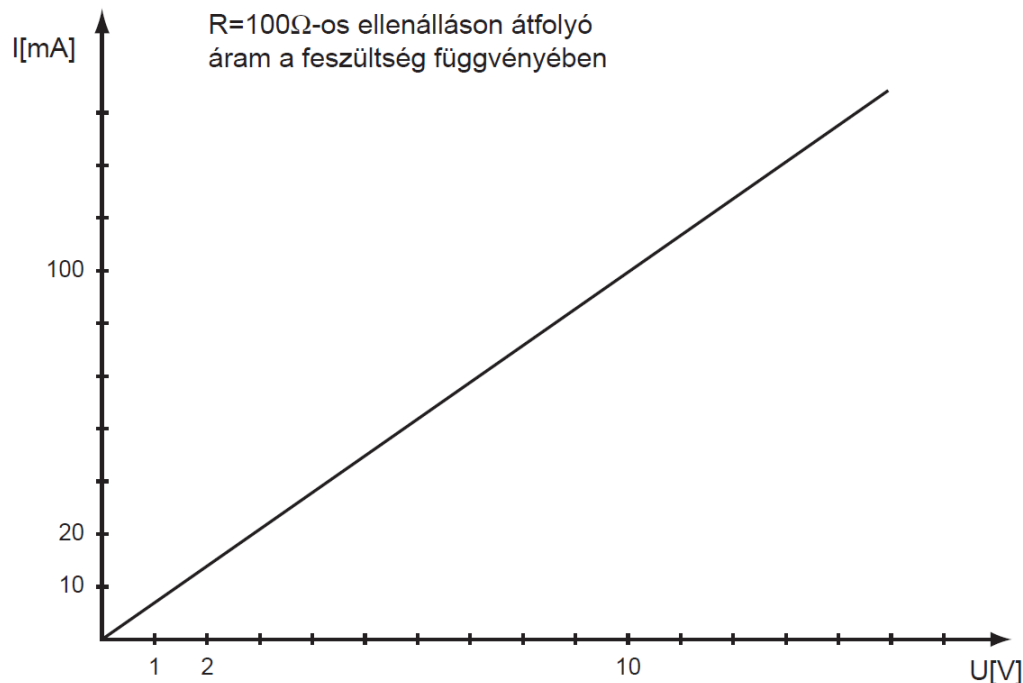


Általános szempontok

- Érdemes színeket használni, ha erre lehetőség van.
- Színválasztás: háttér és szöveg (rajz) fényereje erősen eltérő legyen.
 - Például sötét háttéren világos ábra **vagy fordítva**
 - **Különböző projektoros kivetítésnél gondokjaink lehetnek**
- Összeillő színeket válasszunk.
 - Színkörbe szabályos sokszöget írunk.
- Ügyeljünk arra, hogy fekete-fehér nyomtatás / fénymásolás után is értelmezhető maradjon a munkánk.
 - Például eltérő vonalvastagság, szaggatott vonal, stb.

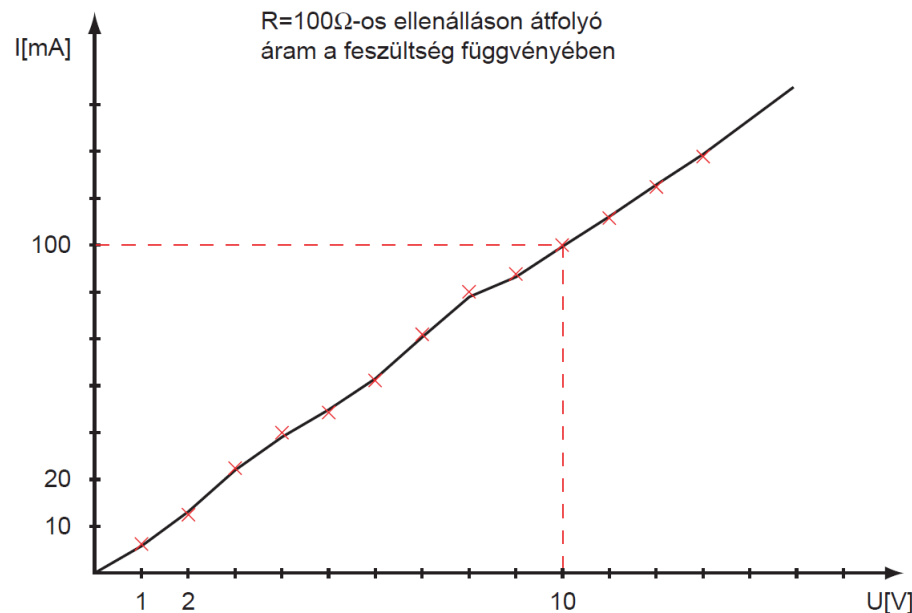
Ábrázolási módok

- Függvény grafikonja
 - Amennyiben az ábrázolni kívánt függvény értelmezési tartománya és értékkészlete (a vizsgált tartományban) *folytonos*, akkor így ábrázolhatjuk:



Függvény grafikonja – folytatás

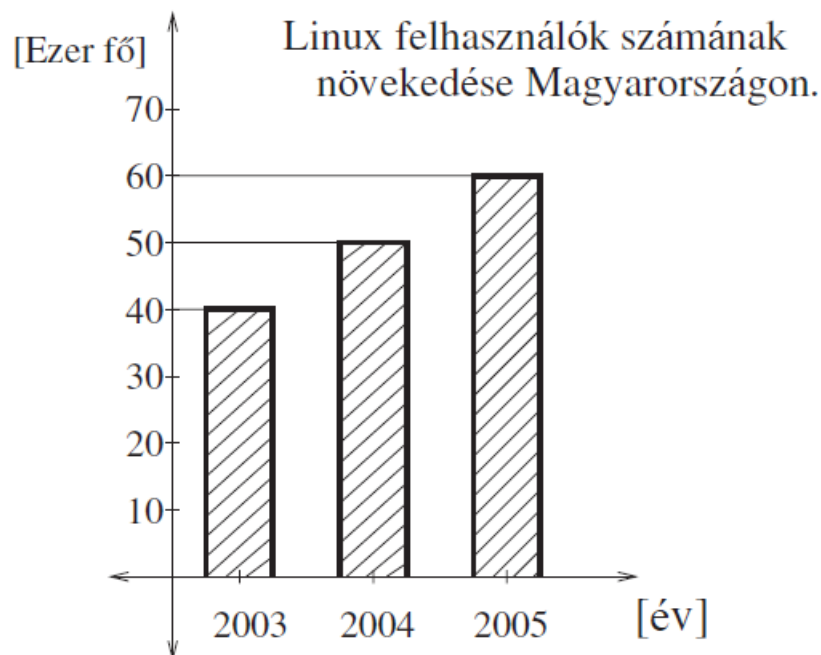
- Ha viszont mérési eredményeket ábrázolunk, akkor csak véges, esetleg kis számú független változó értékre van mért értékünk.
 - Ebben az esetben alkalmazhatunk törtvonalas közelítést, vagy illeszthetünk rá görbét.
 - Ilyenkor hasznos lehet a mérési pontokban kapott értékek megkülönböztetése a többi értéktől.



Ábrázolási módok

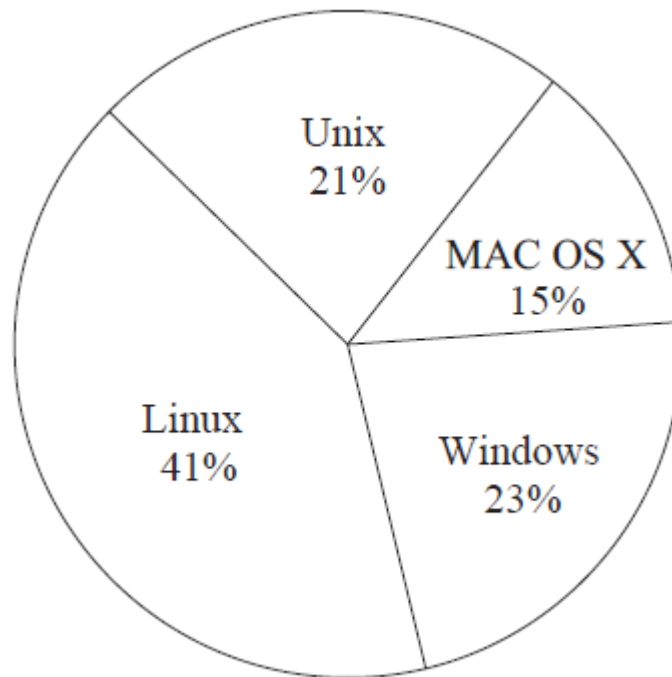
- Oszlopdiagram

- Az oszlopdiagramot olyan függvényeknél célszerű alkalmazni, amikor az értelmezési tartomány nem folytonos, a független változó egymástól egyenlő távolságban található, viszonylag kis számú értéket vesz fel.



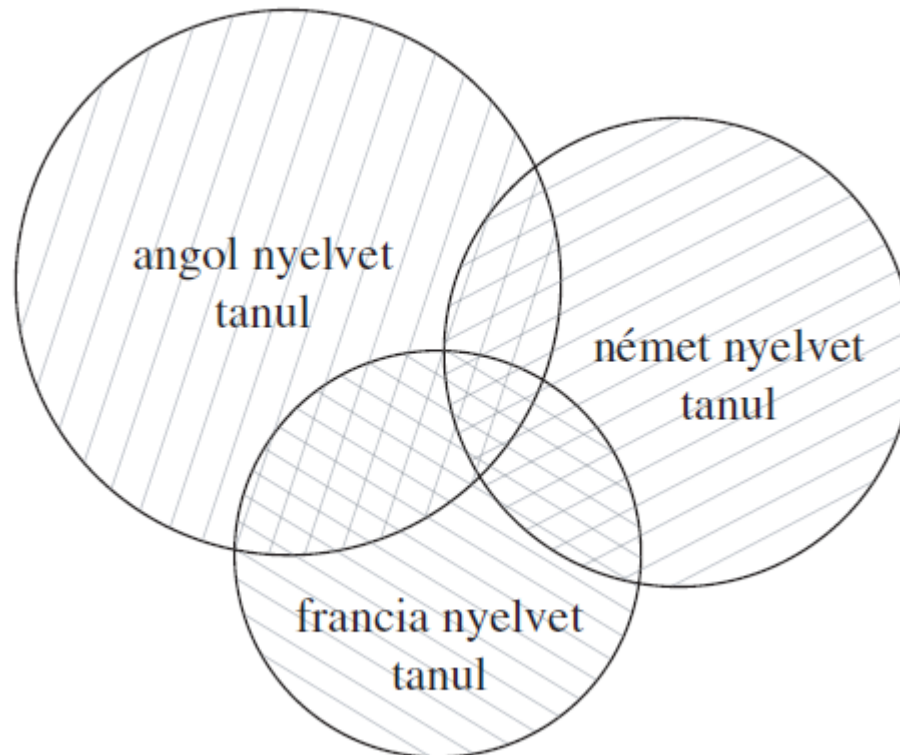
Ábrázolási módok

- Kördiagram
 - A kördiagramot százalékos megoszlás ábrázolására használhatjuk. Az egész kör jelenti a 100%-ot.



Ábrázolási módok

- Venn-diagram
 - Halmazok és a köztük levő relációk ábrázolására használhatjuk a Venn-diagramot.



Ábrázolási módok

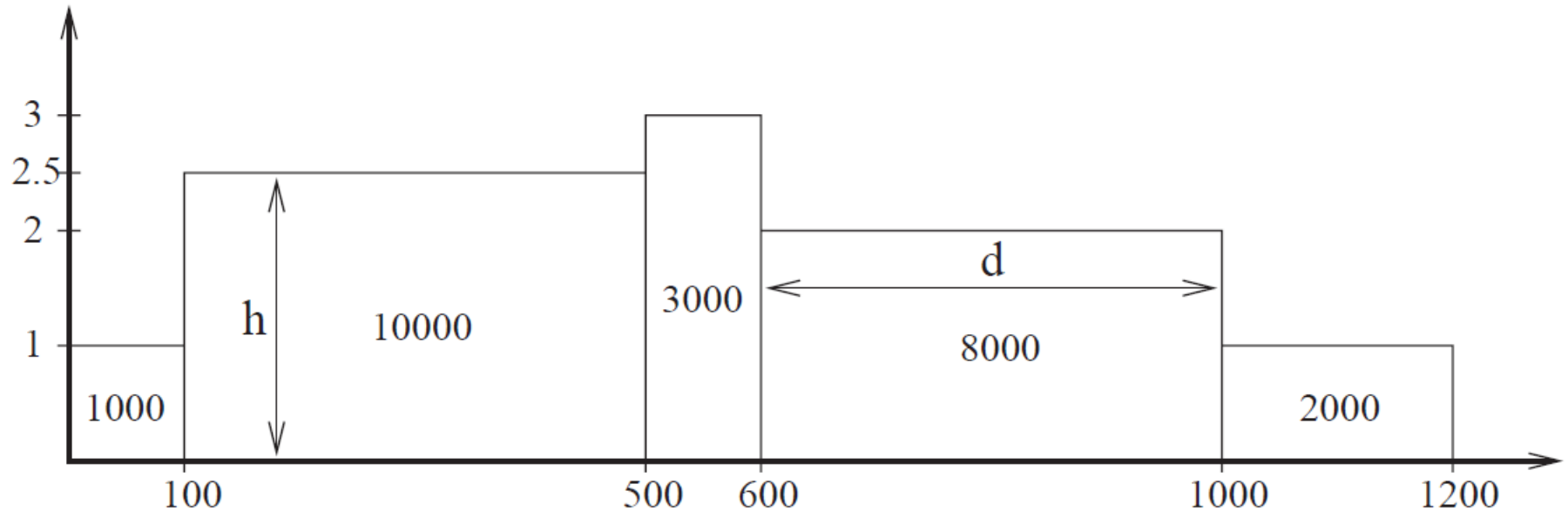
- Hisztogram

- Eloszlásokat szemléletesen jellemezhetünk a valószínűségi sűrűségfüggvényükkel.
 - Amennyiben az eredményeink megfigyelésből származnak, tipikus, hogy az eloszlás jellemzésére egy adott számosságú minta áll rendelkezésünkre.
 - A minta elemeinek egyenkénti megjelenítésénél sok esetben jobb megoldás, ha a mintákból hisztogramot készítünk.
- Hisztogram készítéséhez a megfigyelések lehetséges értékkészletét diszjunkt, és az értékkészletet teljesen lefedő intervallumokra bontjuk.
- Ezeket az intervallumokat *celláknak* nevezzük. A legegyszerűbb esetben a cellák szélessége azonos.
 - Ez nem mindig célszerű választás, sokszor jobb (az eloszlást pontosabban jellemző) eredményt kaphatunk, ha az eloszlás közelítő ismerete alapján a cellahatárokat úgy választjuk meg, hogy az egyes cellákba eső megfigyelések száma közel egyenlő legyen.

Ábrázolási módok

- Hisztogram (folytatás)
 - A hisztogram celláit oszlopokkal (téglalapokkal) ábrázoljuk úgy, hogy az oszlop alapja a hisztogram celláját jelentő szakasz (a cella szélességét jelölje d), magassága pedig h .
 - Az i . cella esetén: $h_i = \frac{n_i}{d_i}$
 - ahol:
 - h_i = az i . cellát ábrázoló oszlop magassága
 - n_i = az i . cellába eső elemek száma
 - d_i = az i . cella szélessége

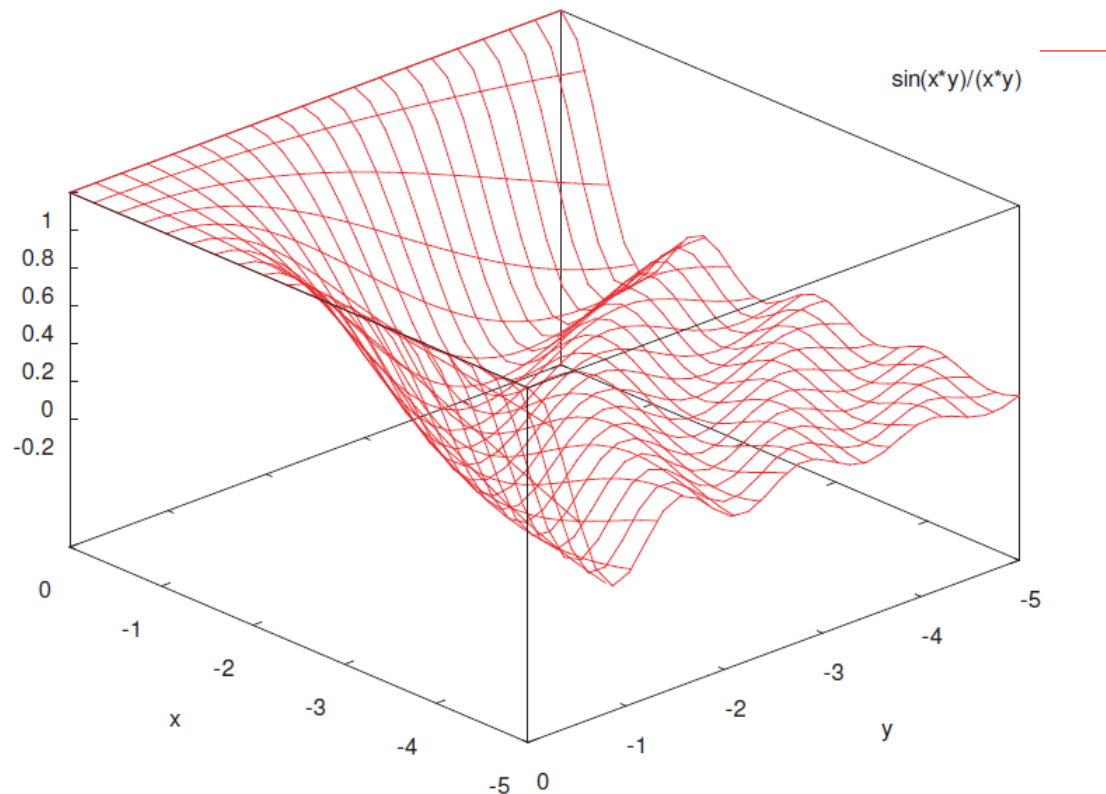
Példa hisztogramra



- Ha egy hisztogramot *normálunk*, azaz a cellák magasságát elosztjuk az összes megfigyelés számával, akkor hisztogram területe 1 lesz: megkapjuk a valószínűségi sűrűségfüggvény közelítését.

Ábrázolási módok

- Kétváltozós függvény ábrázolása felülettel
 - A független változóit egy négyzetháló mentén tekintjük, és ezekre ábrázoljuk a függvény értékét.

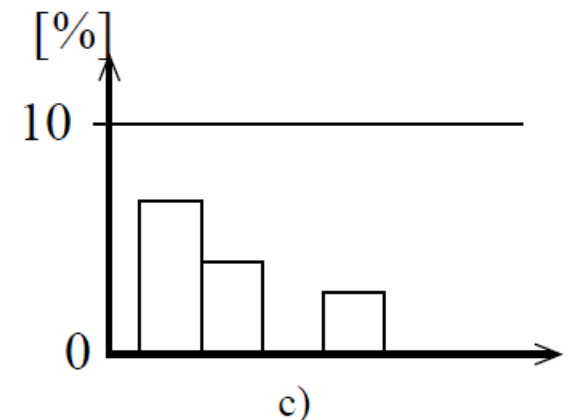
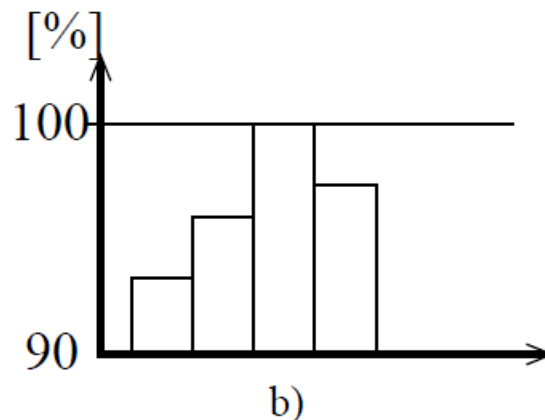
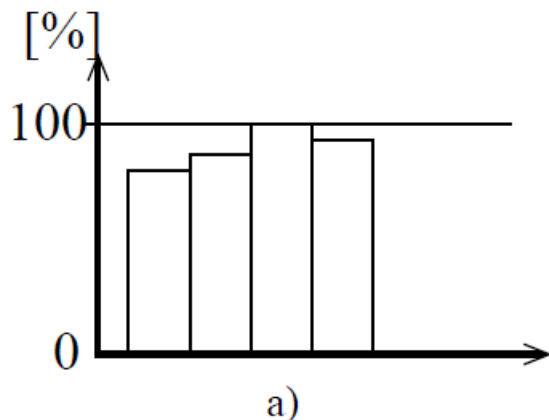


Trükkök

- Mire használhatók?
 1. Valós tartalmú mondanivalók hangsúlyozására
 2. Az olvasó/néző megtévesztésére
- Mi az előbbit javasoljuk
 - És azt is, hogy vegyék észre, amikor mások csálnak!

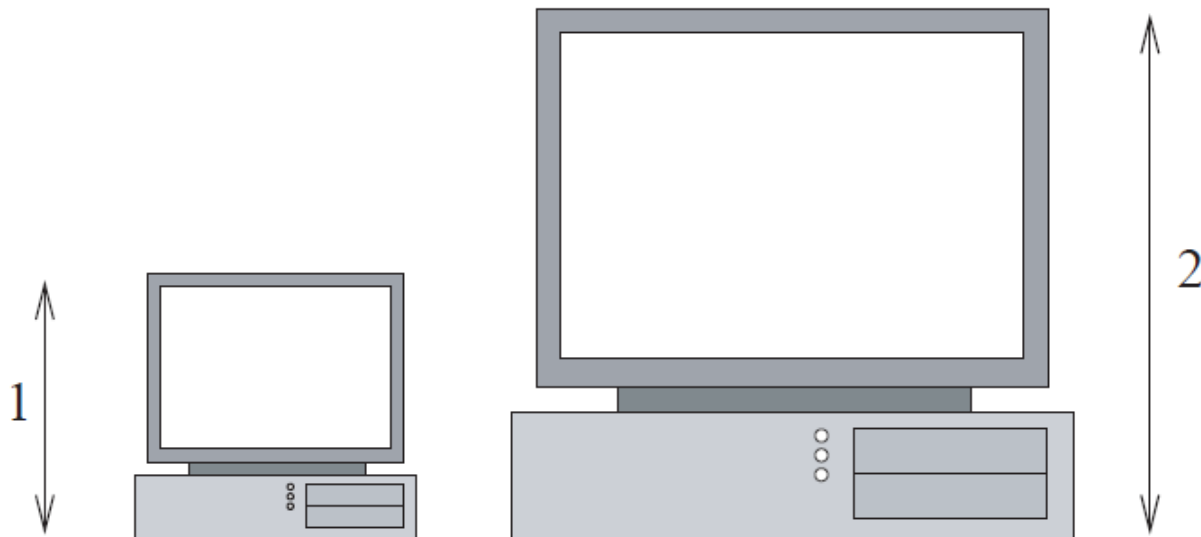
Trükkök

- Ha azt szeretnénk hangsúlyozni, hogy:
 - minden majdnem 100%-os, akkor. ábra „a” részében látható módon a skála 0%-tól 100%-ig terjedjen,
 - ha a különbséget kívánjuk hangsúlyozni, akkor az ábra „b” részében látható módon a skálát nem 0%-tól célszerű indítani.
 - A különbséget hangsúlyozza az ábra „c” része is, ahol azt ábrázoltuk, hogy mennyi kellene a 100%-hoz.



Trükk: különbség felnagyítása

- Összehasonításnál egyenlő szélességű oszlopok helyett az ábrázolandó mennyiséggel lineáris méretben arányos, két dimenziós ábrákat használunk
 - Becsapás, a szemünk területet érzékel!
 - 2x -> 4x arány!



Trükk: relatív jellemzők használata

- A relatív jellemzők használata kifejezetten alkalmas a becsapásra. Például:
 - kerekek száma / ütemek száma



$$4/2=2 > 4/4=1$$



- egy főre jutó űrhajósok száma

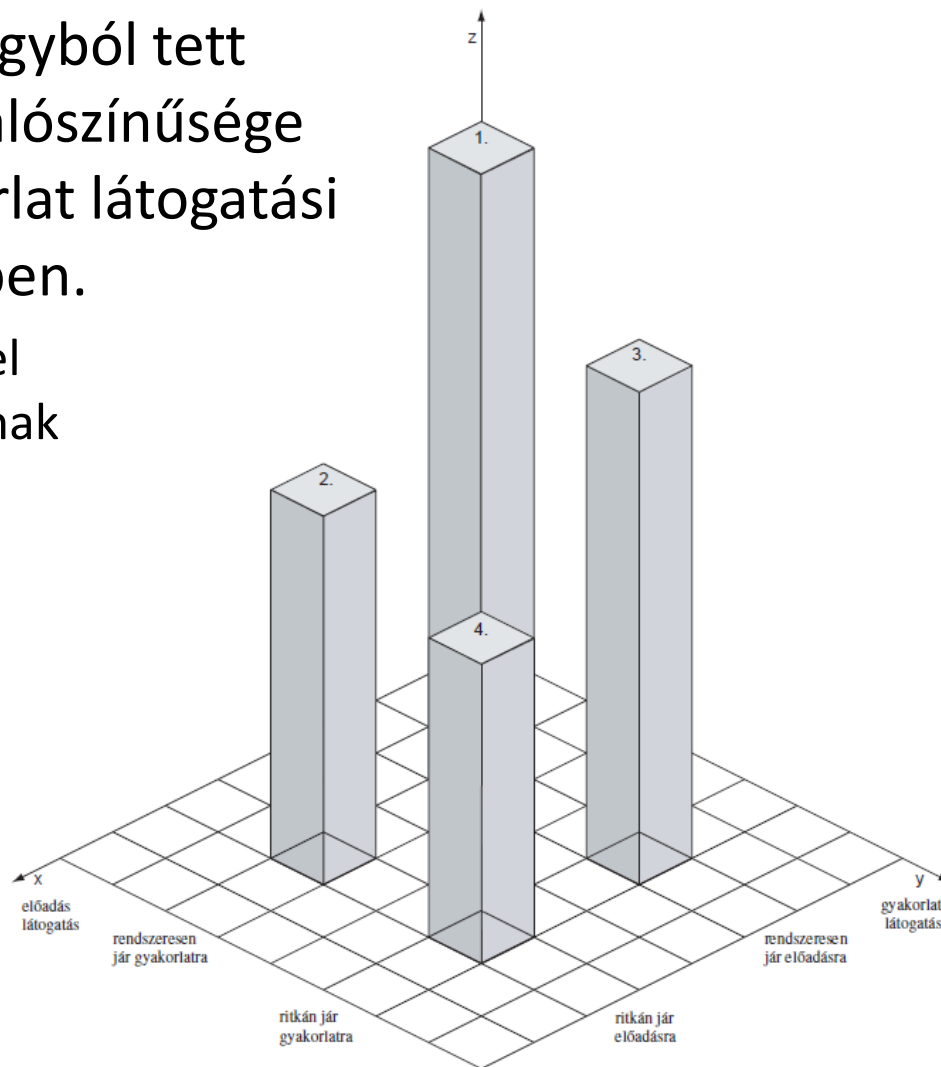


$$1/10,000,000 > 1/20,000,000$$



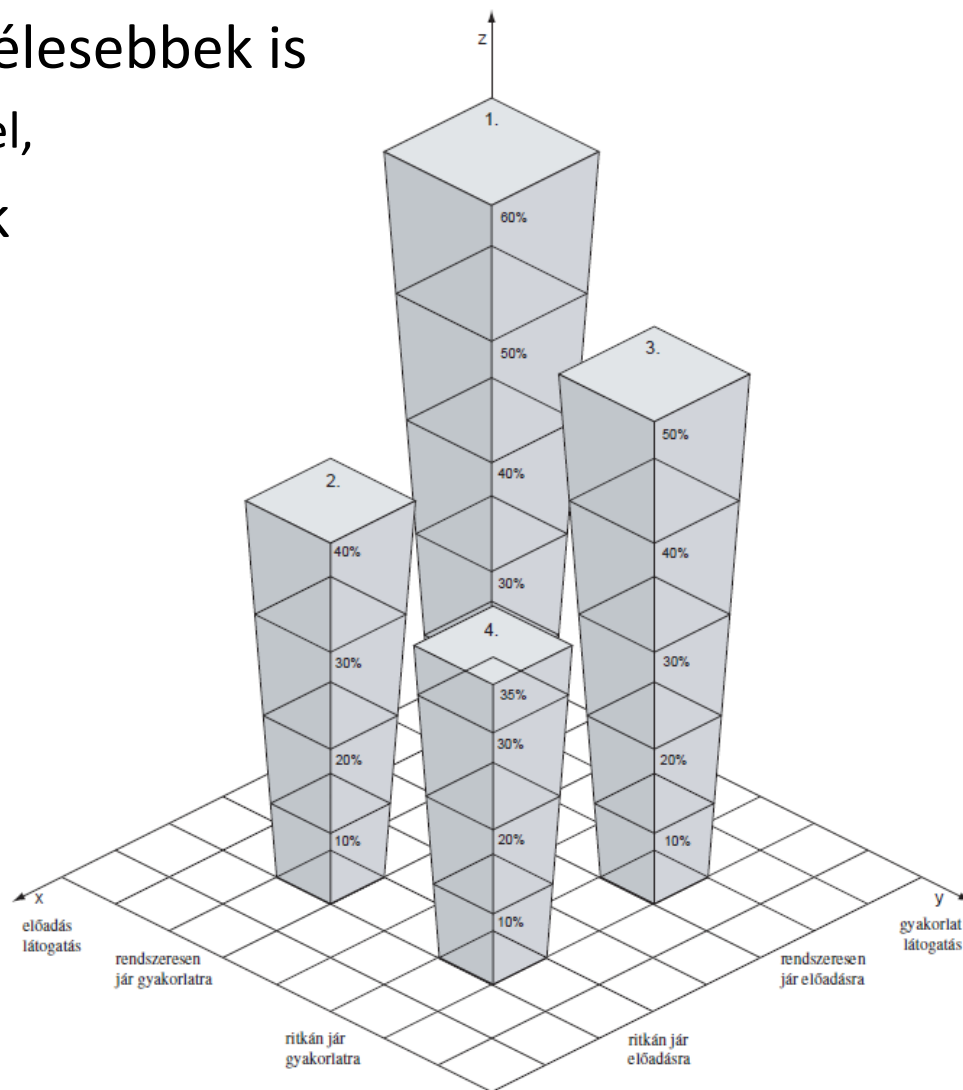
Trükk: oszlopok elhelyezése

- Számítógép-hálózatok tantárgyból tett első vizsga sikerességének valószínűsége a hallgatók előadás és gyakorlat látogatási hajlandóságának függvényében.
 - Akik mindkét helyre becsülettel járnak, toronymagasan kiugranak a többiek közül.
 - Trükk: az ő oszlopuk eleve magasabban kezdődik!



Trükk: perspektíva használata

- A magasabb oszlopok így szélesebbek is
 - és a szemünk területet érzékel,
- A függő változó tengelyének a skálája nem lineáris:
 - a magasabb oszlopok még magasabbnak látszanak!



GYAKORLATI PÉLDÁK

1. példa: átlag +/- szórás és min/max

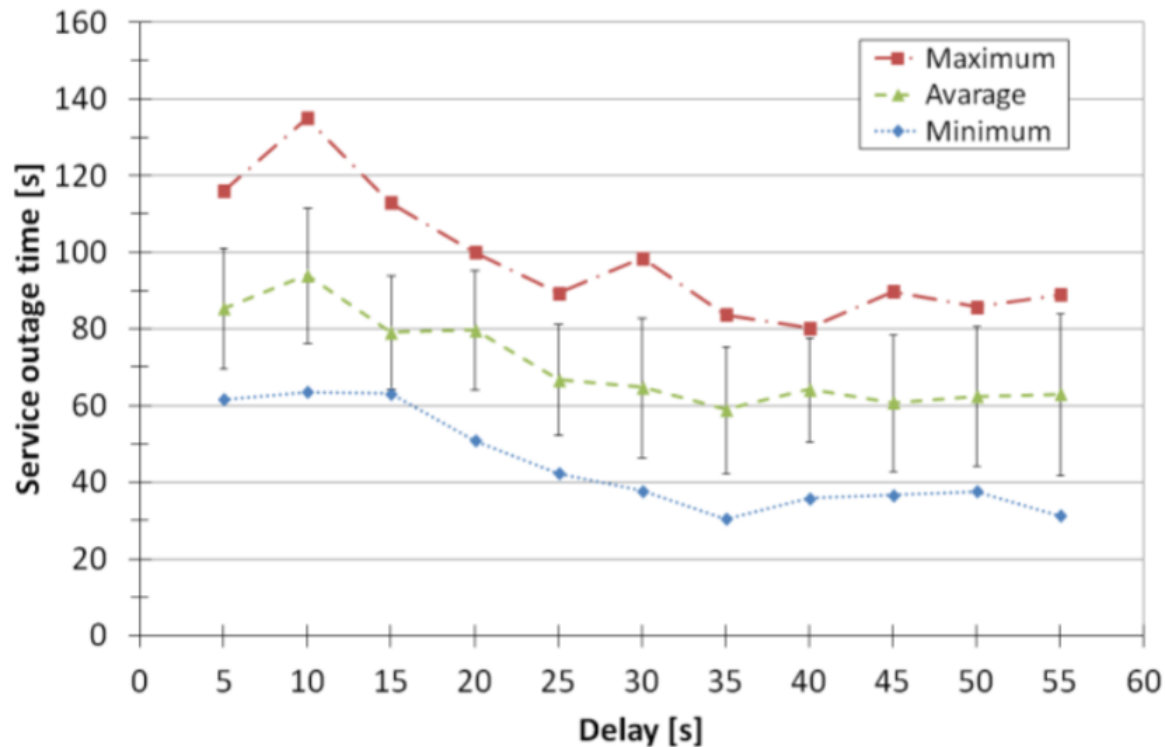


Fig. 5. Service Outage Times in the Function of the Delay from the Last C-RP-Adv Message to the Stopping of the RP on xorp2 Router

G. Lencse and I. Derka, "Investigation of the Fault Tolerance of the PIM-SM IP Multicast Routing Protocol for IPTV Purposes", *Infocommunications Journal*, Vol. V, No. 1. (March, 2013) pp. 21-28.

2. példa: medián és átlag viselkedése

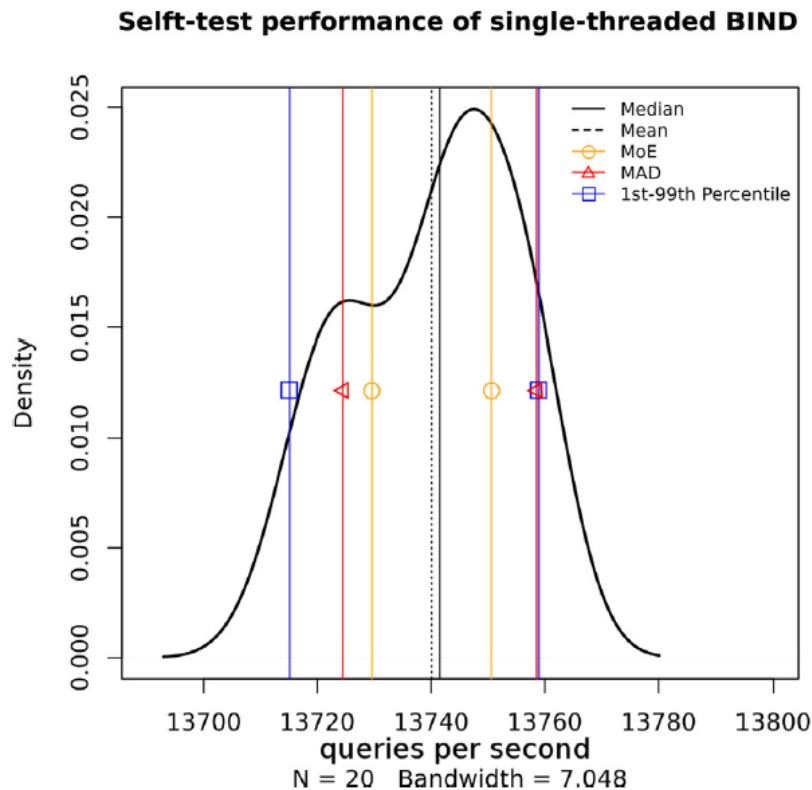


Fig. 4. BIND9 authoritative DNS server performance results: number of successfully answered AAAA record requests per second (**single-threaded**, 800 MHz CPU clock frequency).

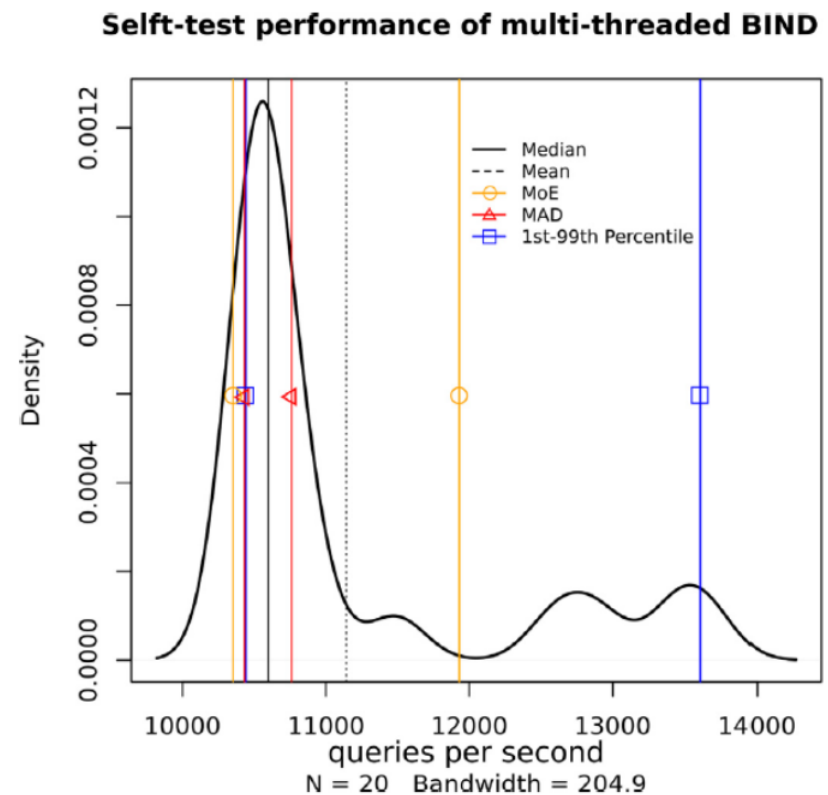


Fig. 5. BIND9 authoritative DNS server performance results: number of successfully answered AAAA record requests per second (**multi-threaded**, 800 MHz CPU clock frequency).

G. Lencse, M. Georgescu, and Y. Kadobayashi, “Benchmarking methodology for DNS64 servers”, *Computer Communications*, vol. 109, no. 1, pp. 162-175, September 1, 2017, DOI: 10.1016/j.comcom.2017.06.004

Feladatok önálló gondolkozásra

- Milyen trendet látunk az ábrán?
- Hány grafikon van összesűrítve?
- Vajon mit jelentenek a hibasávok?

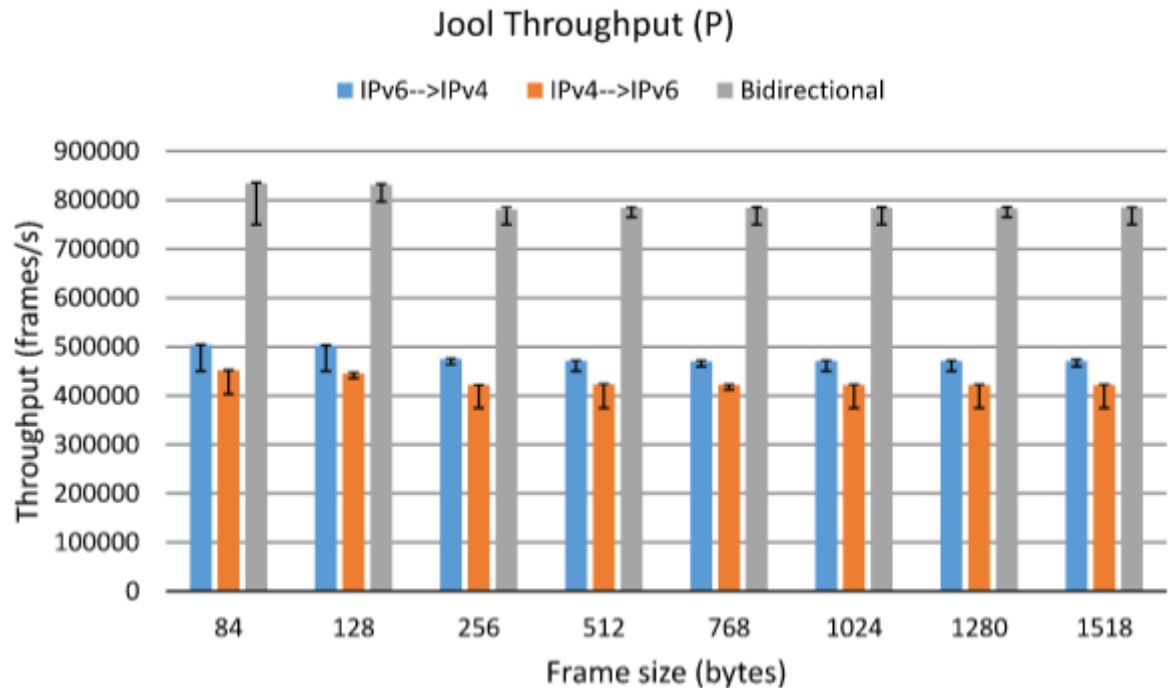


Fig. 4. Throughput results of Jool, TS2.

G. Lencse, K. Shima, "Performance Analysis of SIIT Implementations: Testing and Improving the Methodology", *Computer Communications*, vol. 156, no. 1, pp. 54-67, April 15, 2020, DOI: 10.1016/j.comcom.2020.03.034

Feladatok önálló gondolkozásra

- Milyen trendet látunk az ábrán?
- Hány bemeneti paramétert látunk az ábrán?
- Milyen más tipp alkalmazását látjuk az ábrán?

G. Lencse, "Benchmarking Authoritative DNS Servers", *IEEE Access*, vol. 8. pp. 130224-130238, July 2020. DOI: 10.1109/ACCESS.2020.3009141

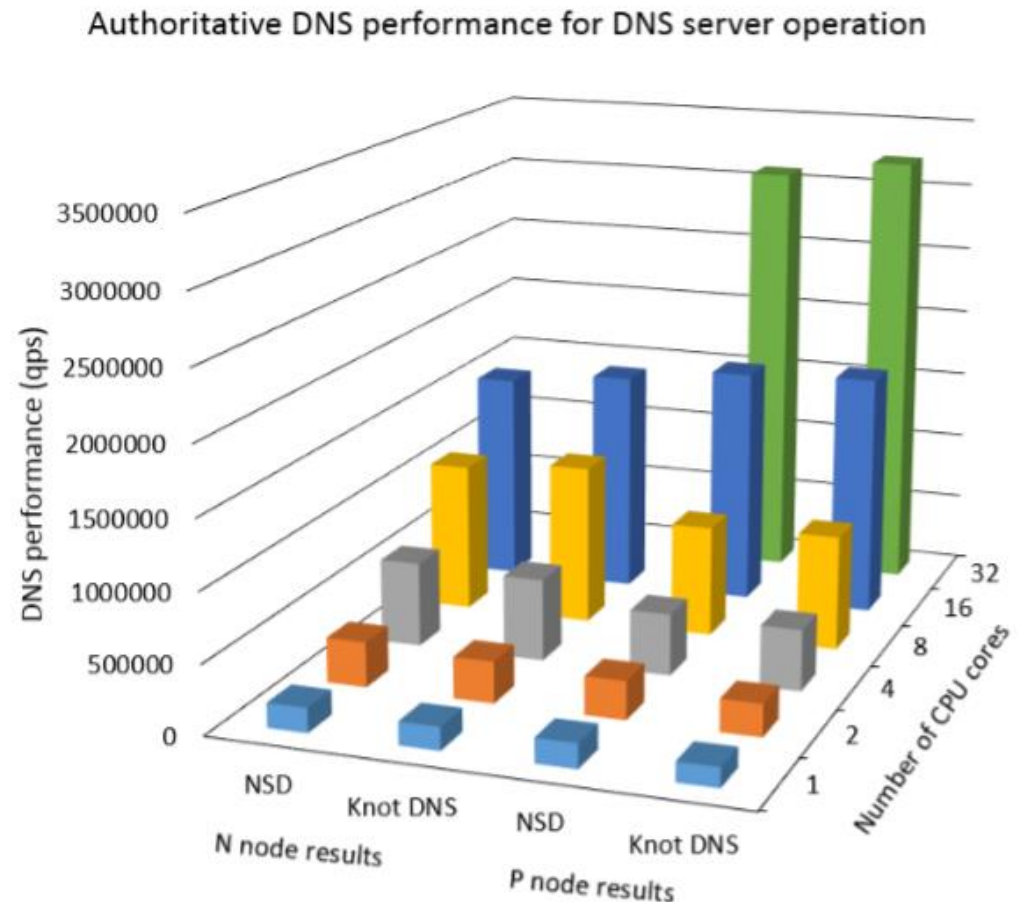


FIGURE 3. Comparison of NSD and Knot DNS for DNS server operation. (The N node result of Knot DNS at 16 cores are limited by Tester performance.)

Feladatok önálló gondolkozásra

- Az alábbi esetben meg tudjuk-e indokolni a táblázat használatát? (Miért nem grafikont használtak a szerzők?)

Table 13 DNS64 Performance: BIND, Forwarder, Opteron

1 Operating System		Linux				OpenBSD				FreeBSD			
2 Number of clients		1	2	4	8	1	2	4	8	1	2	4	8
3 Exec. time of	average	0.067	0.098	0.213	0.409	0.094	0.188	0.382	0.783	0.082	0.121	0.239	0.479
4 256 DNS	std. dev.	0.005	0.007	0.017	0.018	0.006	0.005	0.007	0.015	0.010	0.005	0.008	0.012
5 queries (s)	maximum	0.140	0.170	0.290	0.530	0.180	0.260	0.440	0.860	0.490	0.160	0.290	0.540
6 CPU utiliza-	average	57.91	72.20	63.00	69.92	26.15	27.22	27.42	27.28	66.88	87.82	88.68	89.52
7 tion (%)	std. dev.	0.95	1.38	3.02	2.45	0.85	0.87	0.98	0.93	2.10	1.76	1.76	1.74
8 Memory cons. (MB)		49.992	80.691	147.262	277.242	38.324	67.652	123.117	233.848	62.609	94.406	169.414	303.613
9 Number of requests served in a second (req./s)		3838	5208	4816	5003	2721	2724	2682	2615	3130	4215	4290	4272

G. Lencse and S. Répás, "Performance Analysis and Comparison of Four DNS64 Implementations under Different Free Operating Systems", *Telecommunication Systems* (Springer), vol 63, no 4, pp. 557-577, DOI: 10.1007/s11235-016-0142-x

Összefoglalás

- Bevezetés: célok, alapfogalmak

- Módszerek

- Mérés
- Analitikus módszer
- Szimuláció

- Eredmények kezelése

- Eredmények megjelenítése

- Ábrázolási módok
- Trükkök

- Gyakorati példák

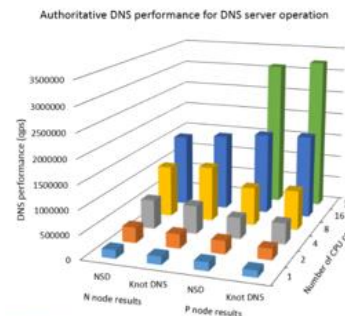
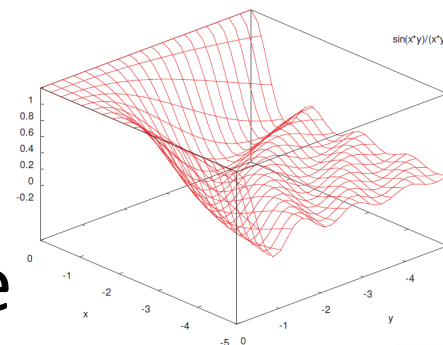
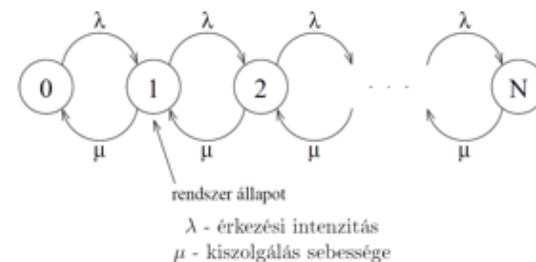
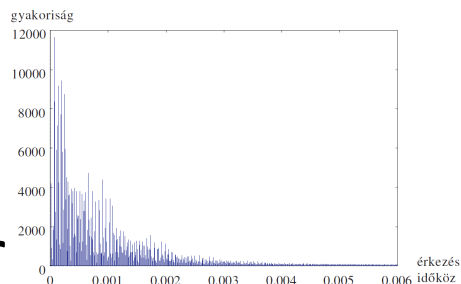


FIGURE 3. Comparison of NSD and Knot DNS for DNS server operation. (The N node result of Knot DNS at 16 cores are limited by Tester performance.)



Kérdések?

KÖSZÖNÖM A FIGYELMET!

Dr. Lencse Gábor
egyetemi tanár
Széchenyi István Egyetem, Távközlési Tanszék
lencse@sze.hu

