



Pázmány Péter Katolikus Egyetem Információs Technológiai Kar

Önálló laboratórium

**Turisztikai alkalmazás készítése GSM alapú
helymeghatározás**

Készítette: Farkas Csanád

Konzulens: Tihanyi Attila

Tartalomjegyzék

Bevezetés.....	3
Rendszer:	3
Előzmények.....	4
Első lépések:.....	4
Adatbázis felépítése:.....	6
SQL adatbázisunk:	6
Táblák felépítése:	6
Adatbázis felépítése, táblák kapcsolata	7
Több utas terjedés és Fading	12
Rendszerintegritások:	14
GPS hibák:.....	15
Torony láthatósági zavarok	17
Összegzés:	19
Konklúzió:	20
Források:.....	21

Bevezetés

Jelen dolgozatomban egy mobil turisztikai alkalmazás fejlesztésével foglalkoztam, amely az egyetemen implementált GSM alapú helymeghatározásra épül. A helymeghatározási feladatokat a munkacsoportban egymás között szétosztottuk. Az egész rendszer egy GPS alapon felvett referencia hálózat és egy GSM vételi jelszint térkép összevetésére alapul. Az adatgyűjtés során különböző eszközökből és különböző módokon keletkezett adatok kerültek egy "mobilhely" nevű adatbázisba. Az adatbázis GSM adatainak felhasználásával, kell módszert találni a fizikailag közel eső pontok keresésére. Mivel az adatbázis már jelenlegi formájában is sok rekordot tartalmaz, ha megtalálunk egy jól használható és paraméterezhető „távolság” fogalmat leíró rendszert, annak segítségével pozíciót meghatározni is lehet de alkalmas az adatbázis tömörítésére is. Feladata megközelítése során kihasználjuk a GSM terjedésről tanultakat, az analízis ismeretek alapján megpróbálunk veszteséges tömöríteni valamilyen ismert módszerrel, és a megkívánt eredmény felől közelítve előre meghatározott mérési pontthalmazok között teszünk különbséget. A saját módszereink és mások eredményeinek összevetéséből határozzuk meg a tovább követendő irányt.

Az én feladatom, a rendszer kiépítése így az egészcsapat tudott Online dolgozni, friss adatokkal, amelyeket természetesen mindenki módosíthat és láthatja a mások munkáját. Itt az adatbázis átírása és annak elérhetővé tétele volt a cél.

Továbbá egyéni munkám az adatbázis hibáinak keresése, mivel a vezeték nélküli méréseket nagyon sok zavaró tényező befolyásol és ezeket a lehető legkisebb mértékig vissza kell fogni mivel ezek nagyon, megzavarhatják, illetve lelassíthatják a későbbi munkánkat.

Rendszer:

Az egész rendszer a 420-as teremben fut egy számítógépen, mely egésznap üzemel. A gépen Linux operációs rendszer van, melyen fut a PostgreSQL, és a Postgis továbbá még számos program melyet még más diákok is használnak. A gépnek nyilvános IP címe is van

(<http://mobilhely.itk.ppke.hu>) így bármikor ellehet érni kívülről. Nekünk az 5432-es port kellett, hogy az adatbázis elérhető legyen, de még a következő portok is nyitva vannak rajta egyéb programok használatához: 80,3306,22022.

Előzmények:

Az előző évek során, számos adat halmozódott fel az egyetem adatbázisában. A korábbi mérések MySQL adatbázist használtak, melyhez tartozott egy PHP-ban írt feltöltő felület. A jobb eredmény érdekében nulladik célnak kitűztük, hogy MySQL helyett, PostgreSQL rendszert használjunk melynek a legfőbb előnye, hogy van hozzá egy Postgis addon. A Postgis kiválóan alkalmas földrajzi adatbázisok kezelésre mivel számos jól használható beépített függvénye van.

Első lépések:

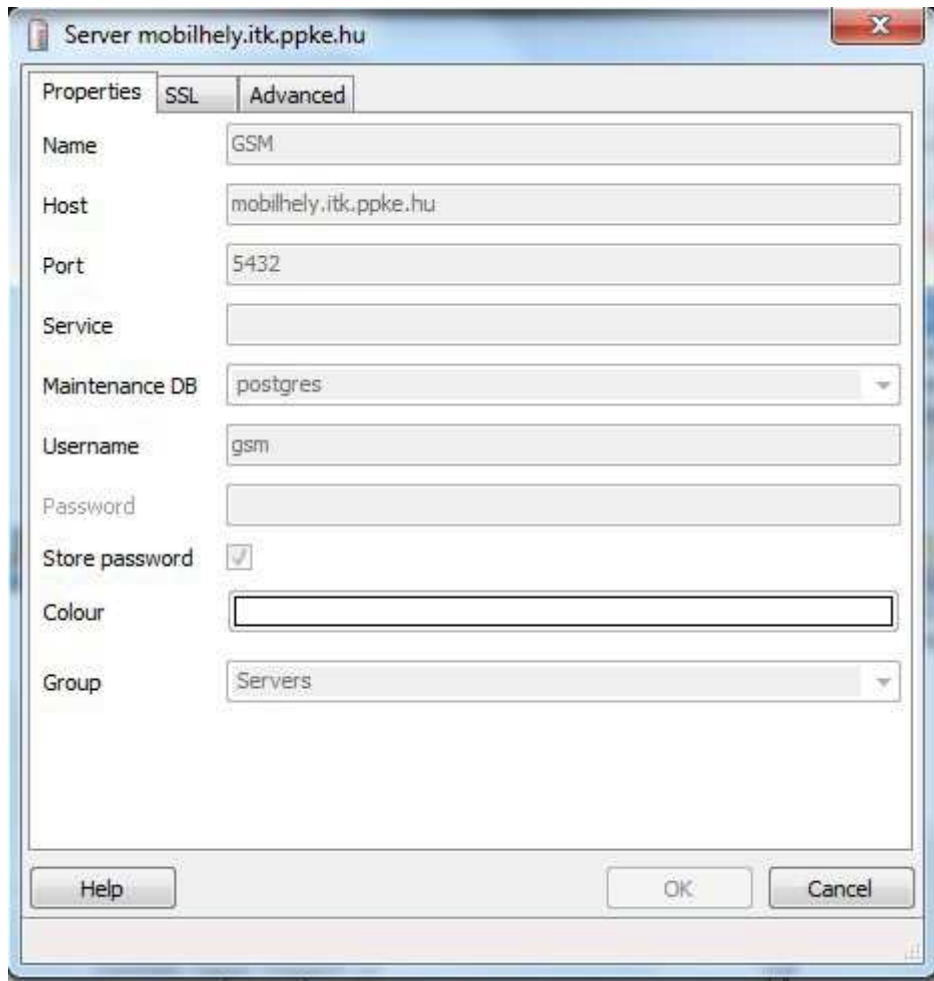
Mindenek előtt, hogy mindenki tudjon dinamikusan haladni, amíg a szerver fel nem épül, készítettem egy SQL fájlt melyet, ha betöltenek , akkor bármikor el tudják kezdeni a munkát OFFline módban. Melynek azon kívül, hogy bármikor lehet használni, még sokkal gyorsabb is így tesztelésre alkalmasabb.

Első lépésben tehát kitűztem célnak, hogy feltelepítem a rendszert és átirom az adatbázist.

Itt a legelső probléma az volt, hogy senkinek nem volt hozzáférése a géphez. Miután ezt sikerült kiküszöbölni, meg kellett oldani a gép IP címét mivel eddig statikus volt és így onnan nem tudtunk elérni semmit.

Ezek után átkonvertáltam a SQL fájlt és elérhetővé tettük a gépet mindenki számára aki, rendelkezik a megfelelő autentikációs adatokkal.

Rendszergazdától még kértem publikus IP címet továbbá megnyitattuk a PostgreSQL portját mely az 5432.



1. ábra Bejelentkezés az adatbázisba

Ezek után most már mindenki rá tudott csatlakozni az adatbázisra, így mindenki láthatta a naprakész adatbázist. A tavalyi évek során a MySQL adatbázishoz készítettek előző diákok egy jó feltöltő felületet. Amivel nekünk is szükségünk van még további mérések feltöltésére ezért elkezdtek átírni a PHP fájlt nekünk megfelelő formátumra. Ezt a feladatot még nem sikerült teljesen megoldani, mivel egy két apró észrevétlen hiba miatt még nem tökéletesen tölti fel az adatbázist.

Adatbázis felépítése:

Egy két szóban a régi adatbázisról. Az adatbázist az utóbbi pár évben különböző eszközökön való mérések alapján töltötték fel több mint 50 000 adattal. Három eszközzel készítették a méréseket először egy SAMBA 75 GSM/GPRS modem, mely egy notebookra volt kötve. Ez a szerkezet 7 bázis állomást mért GPS koordinátával továbbá ellátott minden mérést egy id-val.

Így kétfajta mérést végeztek mozgó és úgynevezett helyhez kötött mérést (hosszú méréseket). Ez sokban hasonlított az előző mérésekhez is, csak itt nem volt GPS koordináta. Két fő okból, mivel fedett térben nem vehető a GPS jel továbbá, nem sok értelme van több héten át egyhelyben maradó egységnek a GPS koordinátáit is külön bevinni. Az RxLevFull, RxLevSub, RxQual, RxQualFull, RxQualSub, Idle TS paramétereket, melyek elsősorban hívás közben kapnak szerepet.

Míg a későbbiekben, amikor többeközt a mi csapatunk is mért, a Maczák Balázs féle androidos mobil méréssel mértek és mérünk sokat. Ez egy mobil androidos software mely majdnem minden androidos telefonon működik, kisebb hibákkal. Ennek tovább fejlesztése és bugok kijavítása folyamatban van. Legnagyobb hibája a telefon kapacitásának gyengesége mivel egy-egy mérés nagyon sok adatot foglal magában.

SQL adatbázisunk:

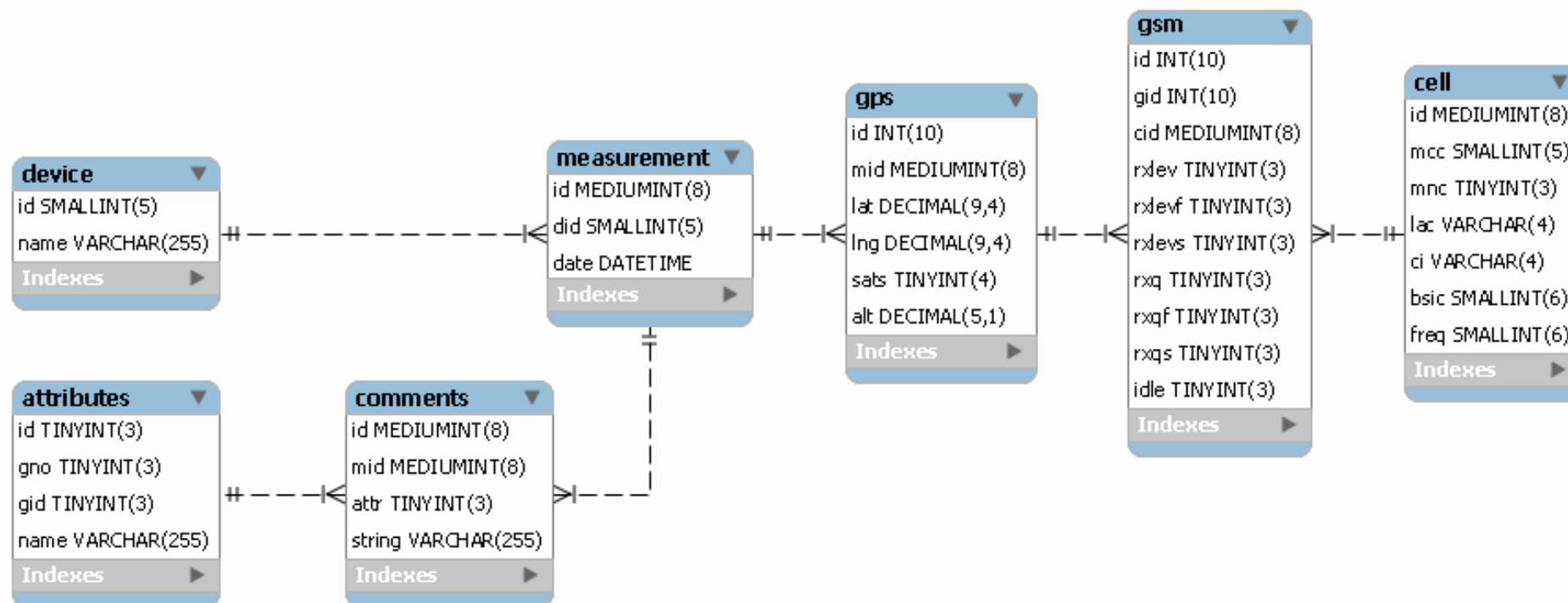
Az adatbázisunk természetesen teljesen ugyan olyan felépítésű, mint a régebbi MySQL adatbázis.

Összesen hét tábla, van melyben a legfontosabb adatok a GPS és a GSM táblákban vannak. Minden táblában meg található az ID, mellyel jól össze van kapcsolva, így könnyen meg lehet találni az adatokat.

Táblák felépítése:

A következő pár sorban a táblák felépítését vázolom fel továbbá a fontosabb mezőik jelentését. Így későbbi eligazodás is könnyebbé válik. A következő oldalon látható a táblák felépítése és kapcsolati rajza.

Adatbázis felépítése, táblák kapcsolata



2. ábra Adatbázis felépítése

Legfontosabb táblák:

Ezen táblák már az átírt PostgreSQL táblák.

GPS:

```
5 CREATE TABLE gps
6 {
7     id serial NOT NULL,
8     mid integer NOT NULL,
9     lat numeric(9,4) DEFAULT NULL::numeric,
10    lng numeric(9,4) DEFAULT NULL::numeric,
11    sats smallint,
12    alt numeric(5,1) DEFAULT NULL::numeric,
13    CONSTRAINT gps_pkey PRIMARY KEY (id ),
14    CONSTRAINT gps_mid_check CHECK (mid >= 0)
15 }
```

3. ábra GPS tábla

id: saját ID – minden GPS mérési bejegyzéshez

mid: measure ID – melyik mérőműszerrel készült

lat: latitude, azaz É-D hosszúsági fok

lng: longitude, azaz K-NY szélességi fok

sats: a GPS vevő által látott műholdak száma

alt: altitude, azaz tengerszint feletti magasság

Továbbá megkötés a kulcsokra, hogy muszáj megfelelő értéket felvenniük.

GSM:

```
5 CREATE TABLE gsm
6 (
7     id serial NOT NULL,
8     gid integer NOT NULL,
9     cid integer NOT NULL,
10    rxlev smallint,
11    rxlevf smallint,
12    rxlevs smallint,
13    rxq smallint,
14    rxqf smallint,
15    rxqs smallint,
16    idle smallint,
17    CONSTRAINT gsm_pkey PRIMARY KEY (id ),
18    CONSTRAINT gsm_cid_check CHECK (cid >= 0),
19    CONSTRAINT gsm_gid_check CHECK (gid >= 0),
20    CONSTRAINT gsm_idle_check CHECK (idle >= 0),
21    CONSTRAINT gsm_rxlev_check CHECK (rxlev >= 0),
22    CONSTRAINT gsm_rxlevf_check CHECK (rxlevf >= 0),
23    CONSTRAINT gsm_rxlevs_check CHECK (rxlevs >= 0),
24    CONSTRAINT gsm_rxq_check CHECK (rxq >= 0),
25    CONSTRAINT gsm_rxqf_check CHECK (rxqf >= 0),
26    CONSTRAINT gsm_rxqs_check CHECK (rxqs >= 0)
27 )
```

4. ábra GSM tábla

id: egyedi azonosító minden bejegyzéshez

gid: GPS ID, kapcsolódási pont a GPS táblához

cid: Cell table ID, kapcsolódási pont a CELL táblához (nem a cell_id!!!)

rxlev: rssi, a vett rádió jel erőssége (0..63 között)

A többi attribútum a jelen helyzetben nem fontos, valamint azok mindig nullák.

CELL:

```
5 CREATE TABLE cell
6 (
7     id serial NOT NULL,
8     mcc smallint,
9     mnc smallint,
10    lac character varying(4) DEFAULT NULL::character varying,
11    ci character varying(4) DEFAULT NULL::character varying,
12    bsic smallint,
13    freq smallint,
14    CONSTRAINT cell_pkey PRIMARY KEY (id ),
15    CONSTRAINT cell_mcc_check CHECK (mcc >= 0),
16    CONSTRAINT cell_mnc_check CHECK (mnc >= 0)
17 )
```

5. ábra CELL tábla

id: minden bejegyzéshez egyedi azonosító

mcc: Mobile Country Code: mobil országhód, melynek hossza 3 számjegy, és egyértelműen meghatározza a mobil előfizető hálózata szerinti országot. A mobil országhódokat az ITU jelöli ki. Magyarország mobil országhódja: 216. A mi esetünkben jelenleg csak a magyarországi adatok kellenek így a többit kiszedtük onnan is a Budapest és környéke jelenleg. (lsd rendszerintegritások).

mnc: Mobile Network Code: mobil hálózati kód, melynek hossza két számjegy. Az MCC az MCC-vel együtt egyértelműen meghatározza a mobil rádiótelefon szolgáltatást igénybe vevő végberendezés vagy előfizető honos hálózatát. Az MNC az MCC-vel együtt, a mobil szolgáltatást nyújtó hálózatokkal jelzéstechnikailag kompatibilis szolgáltatás nyújtása céljából egyértelműen azonosíthat helyhez kötött telefonhálózatot vagy hálózat csoportot is. A mobil hálózati kódot a hatóság jelöli ki. A kijelölés feltételeit külön jogszabály tartalmazza.

lac: Location Area Code: A 4 számjegyből álló azonosító, ami egy nagyobb terület azonosítására szolgál.

ci: Cell Identifier: 4 hexadecimalális számjegyből álló azonosító ami azonosítja a cellát

bsic: Base station identity code: a mobil készüléket segíti a különböző szomszédos bázisállomások megkülönböztetésében. A BSIC egy úgy nevezett „színkód” ami azt jelenti, hogy a szomszédos cellák más képzeletbeli színnel vannak jelölve, és nem lehet egymás mellett két azonos színű cella.

freq: az a frekvencia amin az adó torony sugároz

További táblák melyek fontosabbak még:

Comments:

```
CREATE TABLE comments|
(
  id serial NOT NULL,
  mid integer NOT NULL,
  attr smallint NOT NULL,
  string character varying(255) DEFAULT NULL::character varying,
  CONSTRAINT comments_pkey PRIMARY KEY (id ),
  CONSTRAINT comments_attr_check CHECK (attr >= 0),
  CONSTRAINT comments_mid_check CHECK (mid >= 0)
)
WITH (
  OIDS=FALSE
```

6. ábra Comments tábla

id : egyedi azonosító

mid: összekapcsoló pont a measurement táblához

string: maga a comment. Itt főleg időjárási adatok, erre azért volt szükség mivel számottevő különbségek voltak pl esős kódos illetve tiszta időben mért jelerősségek között.

Measurement:

```
4
5 CREATE TABLE measurement
6 (
7   id serial NOT NULL,
8   did smallint NOT NULL,
9   date timestamp without time zone,
10  CONSTRAINT measurement_pkey PRIMARY KEY (id ),
11  CONSTRAINT measurement_did_check CHECK (did >= 0)
12 )
```

7. ábra Measurement tábla

id: egyedi azonosító

did: kapcsoló pont a a device táblához

date: a mérési dátum időpontja

Device:

```
4
5 CREATE TABLE device
6 (
7     id serial NOT NULL,
8     name character varying(255) DEFAULT NULL::character varying,
9     CONSTRAINT device_pkey PRIMARY KEY (id )
10 )
```

8. ábra Device tábla

id: egyedi azonosító

name: a mérőműszer neve.

Nagy szükség van a mérőműszereket és a velük való méréseket összekötni, hiszen az egyes
hiszen ha felfedezünk egy hibát egy műszerben akkor egyszerűen tudjuk lokalizálni az
azonosítójával az adatbázisban.

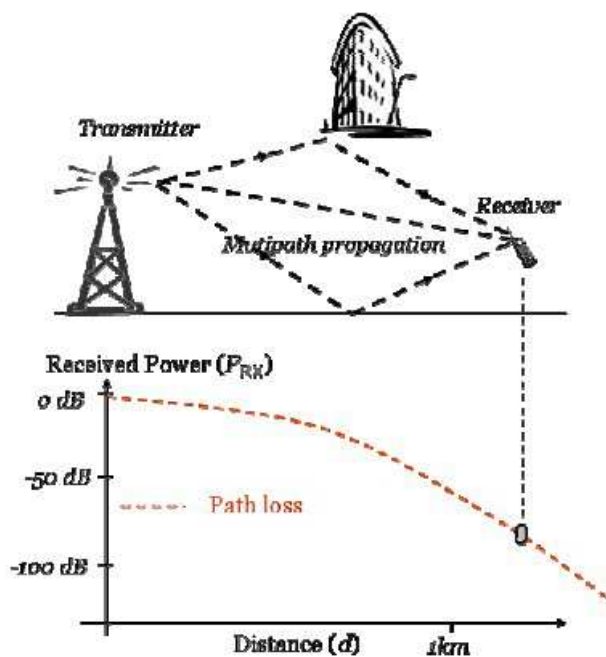
Adatbázishibák, zavaró tényezők keresése, javítása

A következőkben bemutatom feladatom második részét, amelyben lehetséges hibákat
kutattam, majd azokat igyekeztem kiszedni az adatbázisból.

Az első részben a vezetékek nélküli csatorna legfőbb ellenségeit mutatom be. Majd ezek után a
rendszerintegritások keretében az adatbázisból kiszűrt adatokkal foglalkozom.

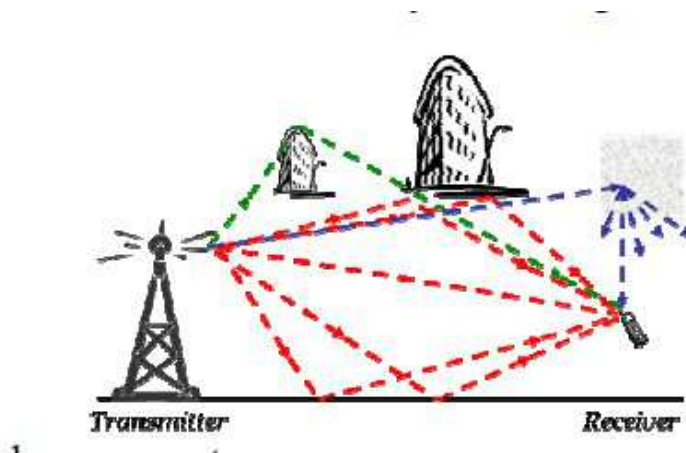
Több utas terjedés és Fading

Látható, hogy nagyon sok különböző mérés típusunk van, melyből a mi későbbi
munkánkhoz csak olyanok kellenek, melyeket tudunk használni és megfelel elvárásainknak.
Továbbá mint minden mérőműszer ezeknek is vannak mérési hibái. Tehát szükségünk van
GPS koordinátákra további értékelhető jelerősség adatok melyeket nem, vagy csak
elfogadható mértékben zavarunk a fading és a több utas terjedéses zavaró jelenségek.



9. ábra Vételi jelszint változása

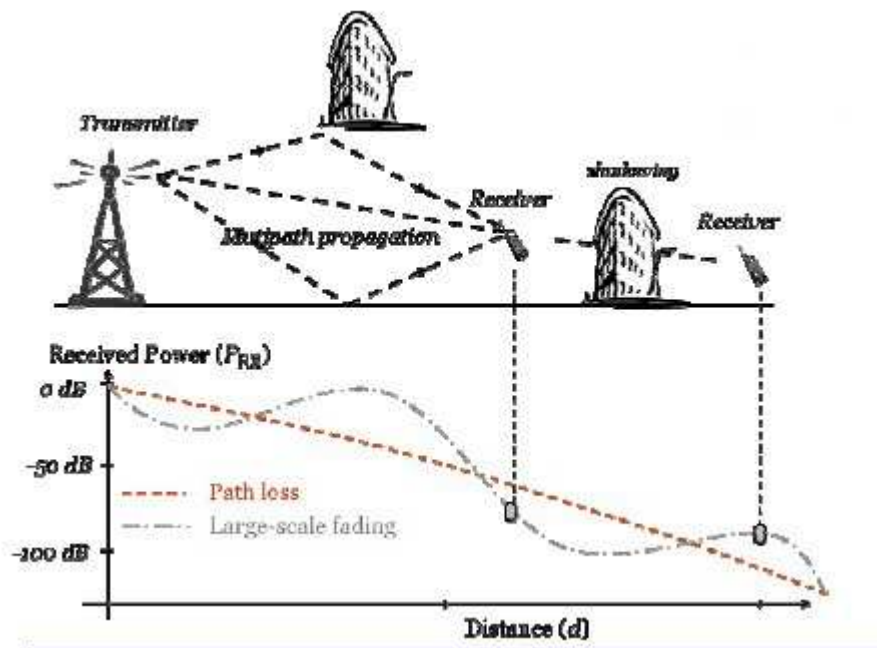
Látható, hogy a jelszint a távolsággal nagyjából arányos, ha kevés a zavaró tényező, de a következő ábrákon viszont látszódik, hogy a valóságban nem ilyen szép a jel terjedési képe.



10. ábra Zavaró tényezők vezeték nélküli hálózatoknál

A három legfontosabb zavaró tényező a következő három:

- Reflections / reflektáció
- Scattering / szóródás
- Diffraction / elhajlás



11. ábra Vételi jelszint változása távolság arányában

A fenti ábra pedig a fading jelenség mutatja be a távolság arányában. Megfigyelhető, hogy ugyan a távolsággal csökken átlagban, de semmi egyéb információt nem tudunk róla és ez sajnos minden mérés során előfordul.

Ezért foglalkoztam sokat a második feladatommal melyben rendszer integritásokat kerestem, többek közt az ilyen hibák és anomáliák csökkentése érdekében!

Rendszerintegritások:

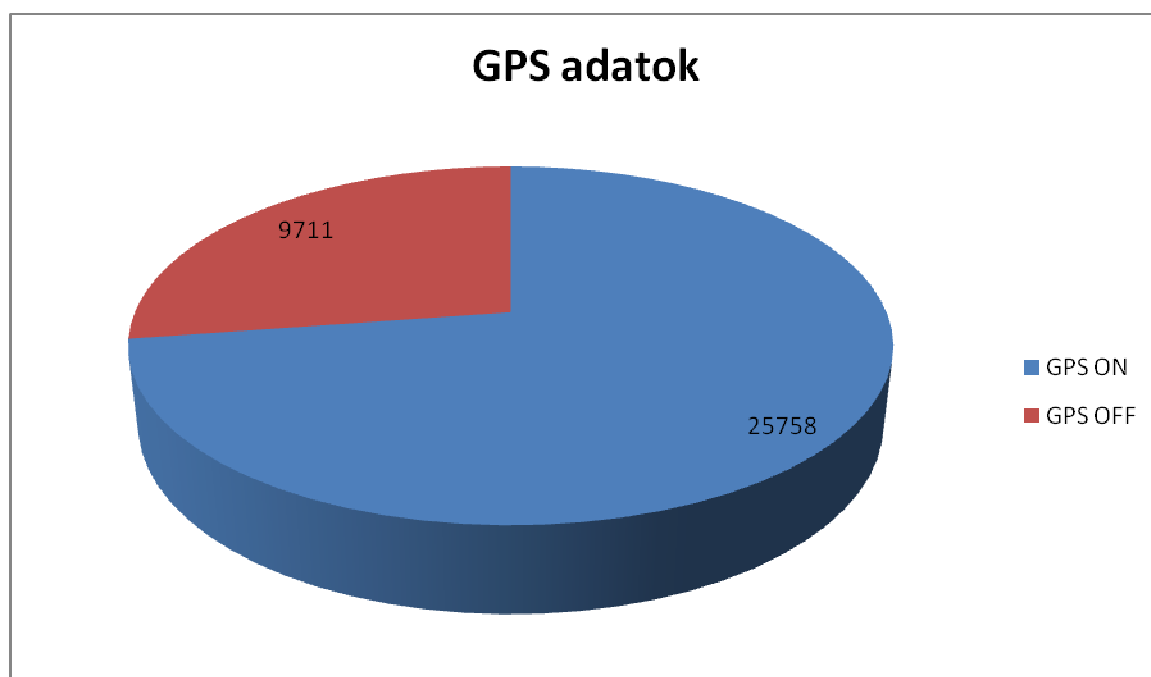
Egy rendszer akkor jó, ha minél gyorsabban eléri el a kívánt célt, továbbá nem tartalmaz, illetve nem ad fel hibás adatokat. Előzőekben láthattuk, hogy a vezeték nélküli méréseknél számos zavaró tényezővel állunk szemben. Ezen károk adatbázisunk minőségének romlásával jár. Ezeket az adatokat jelentősen csökkenteni kell.

Esetünkben egy helymeghatározása nem tarthat sokáig, hiszen ha a turista utazik a buszon és az adatok túl sok ideig utaznak, majd a rendszer lassúsága miatt a feldolgozás is lassú mire visszaér a felhasználóhoz az eredménnyel valószínűleg már elavult lesz. Ezért szükség van a rendszert mindig javítani, gyorsítani.

Először is rendszer integritásokat keresünk, mellyel a rendszer hibáit vagy a rendszerben fölösleges adatokat keressük, melyeket javítunk illetve fölösleges adatoktól mentes új view-okat hozunk létre.

GPS hibák:

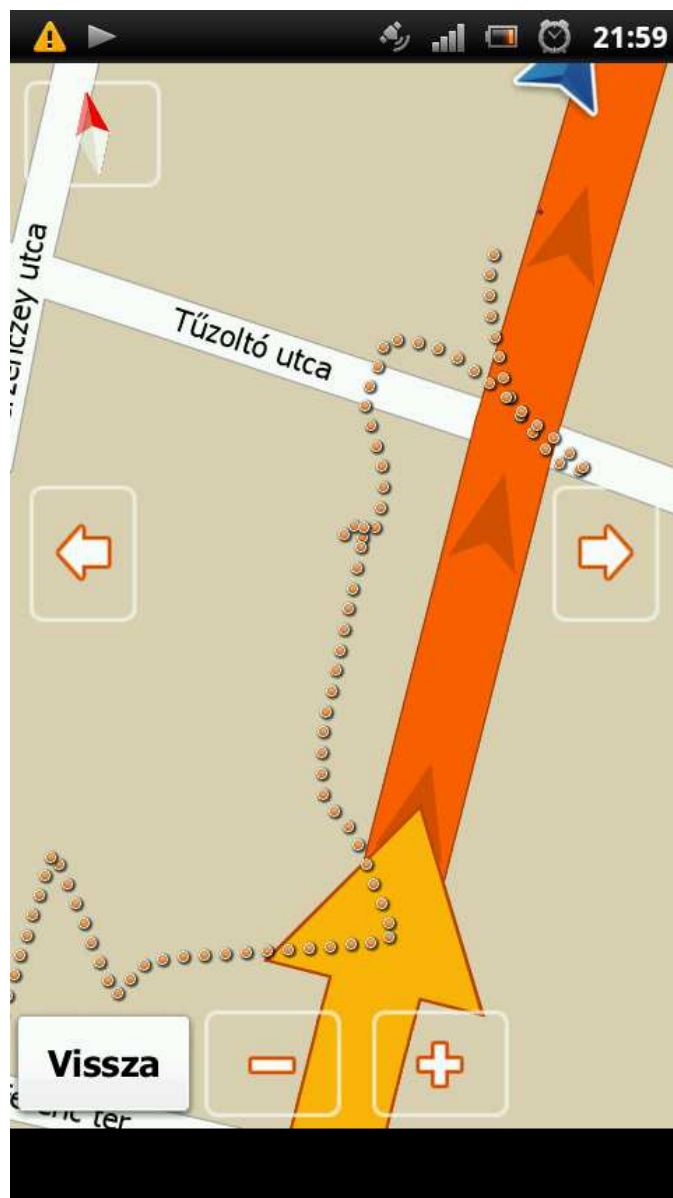
Az először is azokat emeljük ki, amely méréseknél **nincsen GPS**-jel, mivel ezekre nem tudunk dolgozni ebből összesen **9711**.



Az adatbázisba 1994 cella id van, és összesen 1991 van, felhasználva GPS-hez tehát van **3** olyan, amit nem használunk sehol fölösleges adat! Ezek valószínűleg törölt adatok után maradtak meg.

Továbbá a mérőműszerek hibáiból kifolyólag számos pontatlansági hibát fedeztem fel. Ilyenek például irreális helyzetek és az útvonal pontatlan meghatározása. Ez várható volt, hiszen az útvonaltervező programok is használnak útra igazítást továbbá útvonal egyenesítést.

Egy-két példa ilyen mérésekre térképpel, útvonallal, és a GPS szerinti útvonal:





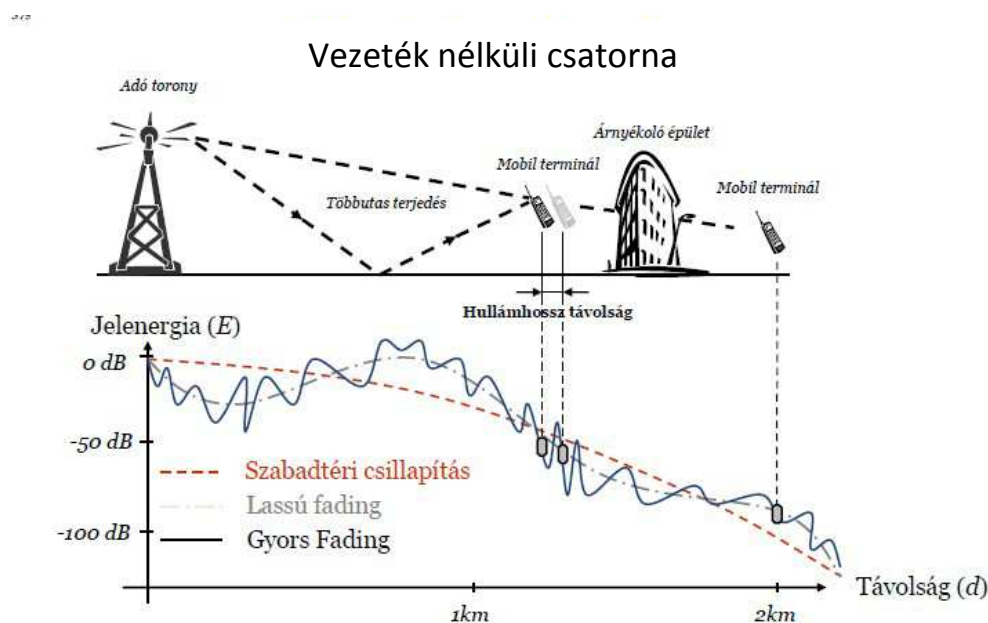
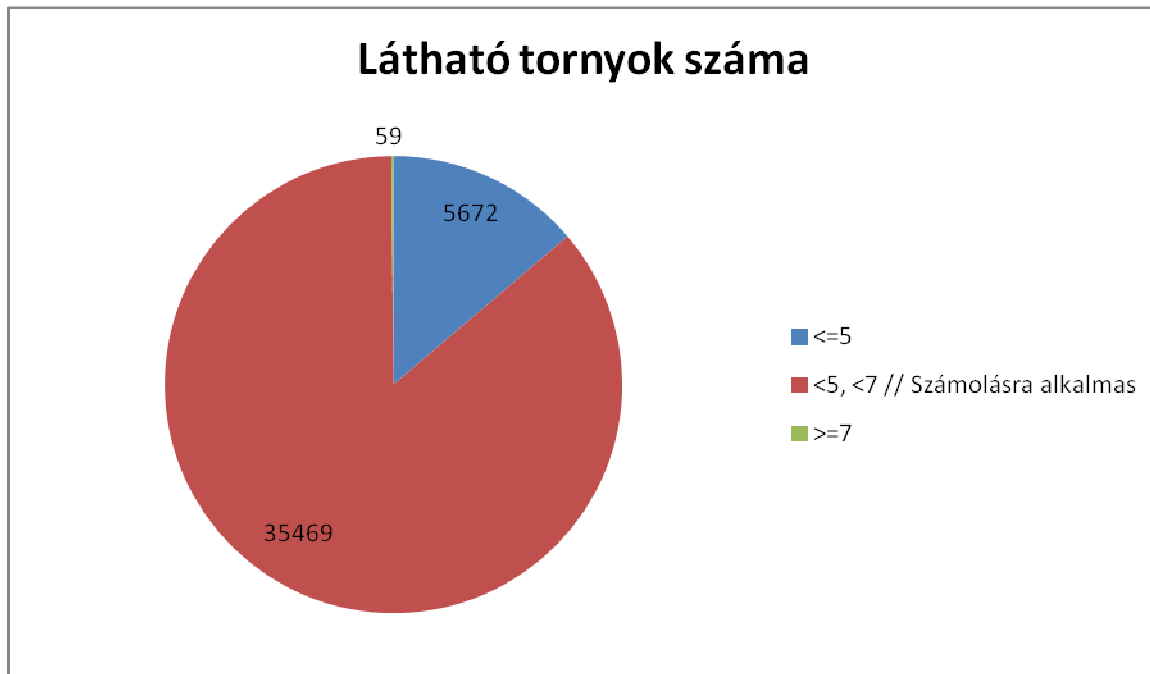
Továbbá ha elmegy a GPS jel, a rendszer továbbra is halad az eddig irányba addig amíg nem fogja be újra a jelet akkor jönnek a nyílegyenes irányok jobb esetben a NULL GPS jelű mérések.

Torony láthatósági zavarok

Továbbá vannak olyan mérések, ahol feltehetően mérési hibák miatt vagy egyéb zavarok miatt túl sok torony tartozik egy méréshez (7+), ezt azért probléma, mert a mérőműszer maximum hetet tud látni, ebből összesen 59 mérésünk van. Kizártuk azokat az eseteket, ahol azért van több torony, mert egyhelyben végeztünk a mérés így esetleges toronyváltozások miatt lett több torony látható.

A másik véglet a tornyok láthatóságába, ha kevés tornyot látunk, mert így a meghatározás pontossága jelentősen csökken. Ha öt, hat vagy hét tornyot látunk, azt fogadjuk el.

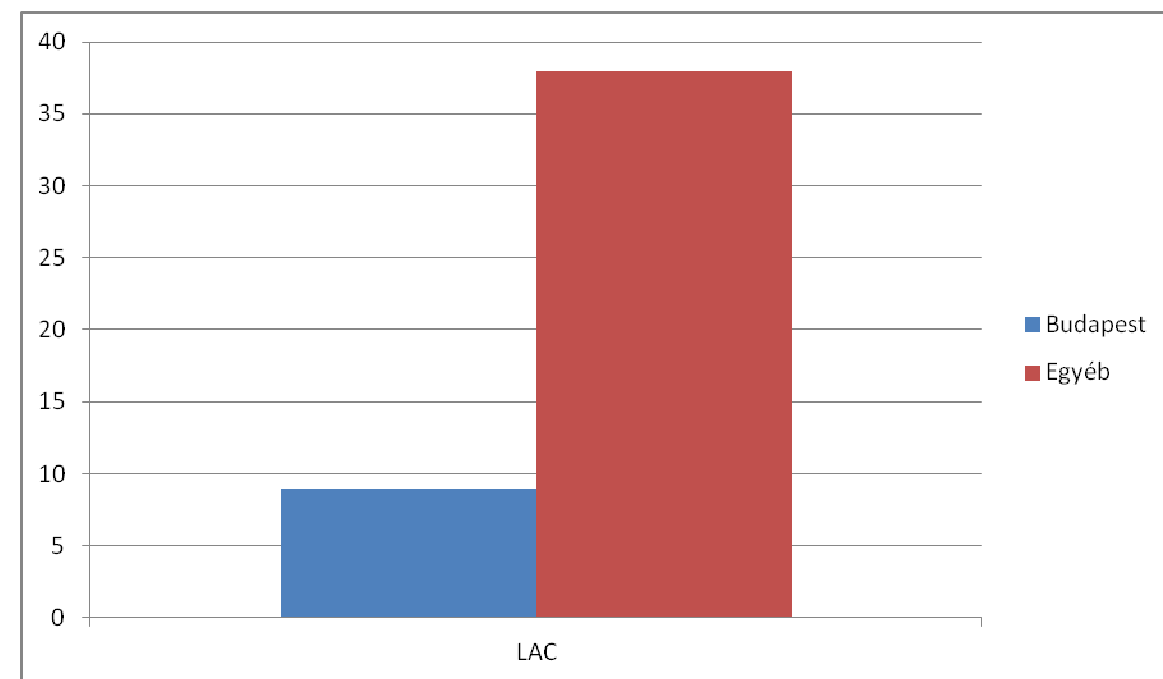
Összesen **35469** ilyen mérésünk van mely ≤ 5 és ≤ 7 , továbbá kiszűrhetjük azokat a cellákat mely ötnél kevesebb és hétnél több tornyot lát, előbbiből **5672** van, míg az utóbbiból az előzőekben olvashatjuk, hogy 59 ez összesen **5731**



Mint minden vezeték nélküli rendszernél itt is előjön a legveszélyesebb zavaró tényező a fading. Először is az alacsony **RX level**-ket szűrjük ki mely **0 közeli**, nagyjából kisebb, mint 3, ebből **595** adat van, míg a túl nagy **RX-level** > 63 mely szintén a fading miatt történhet ilyen, mivel a mérőműszer elve nem tud akkora számot tárolni. Ilyenből **453** adatunk van.

Tehát rossz RX level miatt a használhatatlan adatok száma: **1048**.

Nagyon hasznos adat a **Local Area Cell_ID**, amely adattal eleve egy kis területre szűkítjük a számításainkat. Méréseink alapján ilyenből **47** egyedi van, amelyek közül 9 Budapesti. Mivel csak Budapesten számolunk egyelőre.



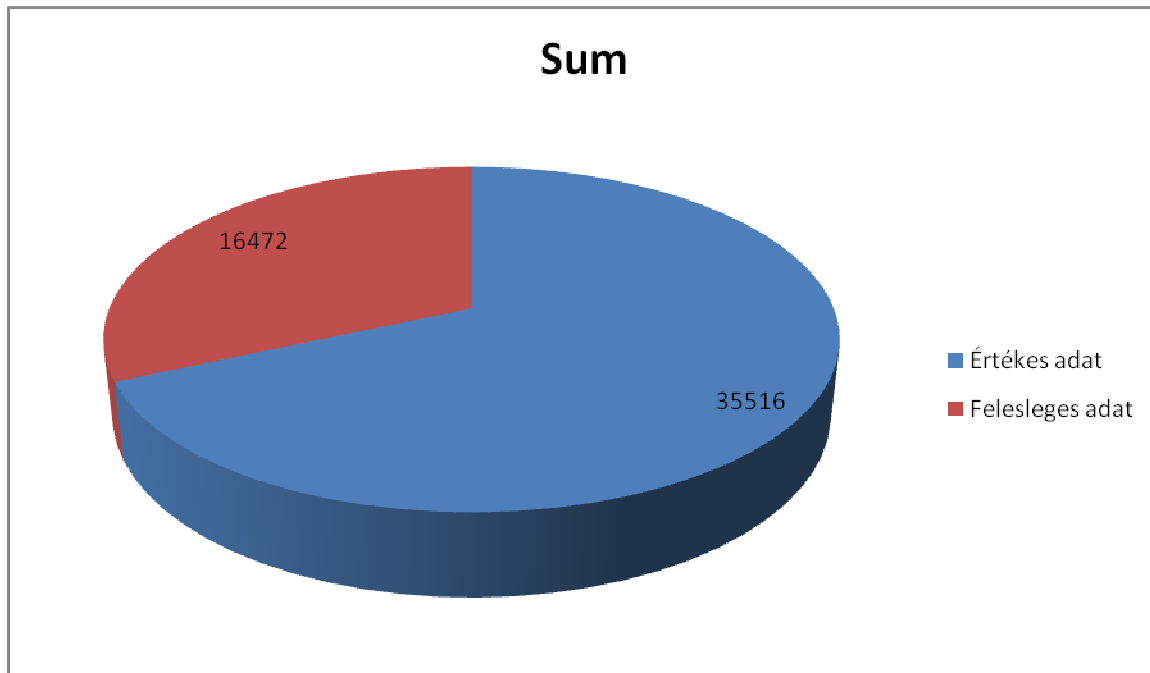
Összegzés:

Összefoglalva $9711+59+5672+595+453 = \mathbf{16472}$ adatunk van melyre nincs szükségünk, tehát ezek nélkül, létre kell hozni egy view-ot, így csökkentve az adatokat növelve a sebességet.

Nagyon fontos adatból pedig **35469+47** adatunk van.

Ezek az a számok már a Réti Dániel által sorokban leredukált, míg oszlopokban kibővített adatok, mivel egy mérési pont annyi sorból állt ahány tornyot látott (id-vel összekapcsolva). Ezért diák társam megoldotta, hogy ezen mérések egy sorban legyenek és a látható tornyok külön oszlopokba kerüljenek!

Így az adatok ~26%-át sikerült kiszűrni, a minőség romlása nélkül. Így jelentős sebesség növekedést és stabilitást értünk el.



Konklúzió:

Az grafikonon kiválóan látszik, hogy egy rendszer sem tökéletes, és nincs olyan mérőműszer, ami mindig megfelelő adatot ad. A rendszer integritás során nagyjából 26%-át az adatoknak sikerült úgy kivenni, hogy a tovább számolásaink során semmilyen hátrányt ne szenvedjünk, hiszen ezeket az adatokat nem tudtuk használni. Ezzel a mennyiséggel jelentősen megnöveltük a sebességet, így amikor a felhasználó lefogja, majd kérni a helyzetét a feldolgozás hamarabb fog megtörténni, ez alatt ő kevesebbet fog várni rá így kevesebb esélye van elmozdulni a helyzetéről vagy a buszon, melyen utazik nem lesz képes átmenni reményeink szerint egy másik megállóba addig.

Források:

- Oláh András Mobil hálózatok diái (Fading, és több utas terjedés)
- Vinis Ádám szakdolgozata (Adatbázis)
- Kelemen Mihály szakdolgozata (Adatbázis)