



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

1

Alba Regia Műszaki Kar
Geoinformatikai Intézet

SZAKDOLGOZAT

Mobiltelefonos helymeghatározás pontosítása szenzorfúzióval

OE-AMK

Hallgató neve: Kovács Benedek Marcell
Székesfehérvár, 2024 Hallgató törzskönyvi száma: T000729/FI12904/A-G



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

2

Alba Regia Műszaki Kar
Geoinformatikai Intézet

SZAKDOLGOZAT FELADATLAP

Hallgató neve: Kovács Benedek Marcell

Szaktervezési száma: SZD2306221136158886

Törzskönyvi száma: T000729/FI12904/A-G

Neptun kódja:

Szak: földmérő és földrendező mérnök

Specializáció: Geoinformatika specializáció

A dolgozat címe: Mobiltelefonos helymeghatározás pontosítása szenzorfüzűvel

A dolgozat címe angolul: Fusion of smartphone sensors to enhance positioning accuracy

A feladat részletezése:

Ismertesse a Kálmán szűrés elméletét!

Mutassa be a mobiltelefonoknak a pozíció meghatározásához felhasználható szenzorait!

Fejlesszen a Kálmán-szűrést megvalósító algoritmust!

Tesztelje az algoritmust valódi mért adatokon!

Elemezze a kapott eredményeket!

Intézményi konzulens neve: Molnár Gábor Péter

A kiadott téma elévülési határideje: 2024. 12. 15.

Beadási határidő: 2024. 05. 15.

A szakdolgozat: Nem titkos.

Kiadva: Székesfehérvár, 2024. 04. 08.



Intézetigazgató

A dolgozatot beadásra alkalmasnak találom: 2024. 05. 15.

belső konzulens

külső konzulens

OE-AMK

Hallgató neve: Kovács Benedek Marcell
Székesfehérvár, 2024 Hallgató törzskönyvi száma: T000729/FI12904/A-G



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

HALLGATÓI NYILATKOZAT

3

Alba Regia Műszaki Kar
Geoinformatikai Intézet

Alulírott hallgató kijelentem, hogy a dolgozat saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült dolgozatban található eredményeket az Óbudai Egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja, a titkosításra vonatkozó esetleges megkötések mellett.

Kelt: Székesfehérvár, 2024. 05. 15

Kovács Benedek

hallgató aláírása



TARTALOMJEGYZÉK

1.	Bevezetés	6
2.	Az adattárolás és a telefon szenzorai	8
2.1	Az adattárolás	8
2.1.1	Utófeldolgozás esetén	9
2.1.2	Valós idejű kiértékelés	10
2.2	Felhasznált szenzorok	12
2.2.1	Műholdas helymeghatározó egység	12
2.2.2	Gyorsulásmérő (Accelerometer)	14
2.2.3	GPS-ből számított ál-sebességmérő	15
2.2.4	Szögsebességmérő (Gyroscope)	15
2.2.5	Mágneses tér mérő (Magnetometer)	16
3.	Adatok	17
3.1	Az adatnyerés jellemzése	17
3.1.1	A MATLAB, mint fejlesztői környezet	17
3.1.2	A tesztmérés	18
3.2	Előfeldolgozás	19
3.2.1	Kritikus mérési helyzetek szimulálása	19
3.2.2	Egyenes vonalú mozgás vizsgálata	19
3.2.3	GPS-ből számított ál-sebességmérő	21
3.2.4	A telefon órájának anomáliája	21
3.2.5	Magnetométer előfeldolgozása	24
3.2.6	Vetületi rendszer meghatározása	25
3.2.7	Forgatómátrix a két rendszer között	26
4.	Módszerek	30



4.1 A Kálmán-szűrő.....	30
4.1.1 Kálmán Rudolf Emil.....	30
4.1.2 A Kálmán-szűrő előnyei	31
4.1.3 A Kálmán-szűrő alkalmazása	31
4.2 Működésének alapja	31
4.3 A Kálmán-szűrő alkalmazása	32
4.3.1 Bevezetés	32
4.3.2 A jóslás	32
4.3.3 A frissítés	36
5. Eredmények	38
5.1 A valóságű mozgásra	38
5.2 A hibaellipszis	43
6. Diskusszió	44
6.1 Mintavételezés.....	44
6.2 Hasonló munkák.....	44
6.3 További kutatási célok.....	45
6.3.1 GPS bias.....	45
6.3.2 Új szenzor implementálása	46
6.3.3 Felmerülő akadályok mobilapplikáció tervezés során.....	46
7. Összefoglalás	48
8. Summary	49
9. Köszönetnyilvánítás	51
10. Idézett forrásmunkák.....	52
11. Ábrajegyzék	54



1. Bevezetés

Mára természetessé vált a mobiltelefonok használata a mindennapi teendők elvégzéséhez. A folyton fejlődő iparág egyre több funkcióval látja el a zsebekben elférő kis számítógépeket. Ezt a fejlődést szemlélteti például, hogy a piac legújabb termékei már nem egy közeli meteorológiai állomásról kérnek le az időjárást leíró adatokat, hanem azt a beépített légnyomás, hő- és páratartalom mérőikből képesek meghatározni, vagy akár a jövőre vonatkozó következtetéseket becsülni. A technológia által nyújtott új lehetőségek hatékony kihasználása azonban a felhasználókon múlik.

Tanulmányaim során megismertem a telefonok által is használt műholdas helymeghatározás módszerét, annak előnyeit és hátrányait. Kutatási témám egy olyan alkalmazás létrehozása, amely él a mobilkészülék által biztosított szenzorok mérésének lehetőségével, ezeket felhasználva szabatosabb helyadat szolgáltatást nyújtva a felhasználó számára.

Bárki, aki a telefonján használt navigációs alkalmazást tapasztalhatta már, azt az esetet, amikor irányunkat vagy helyzetünket pontatlanul határozta meg az adott program. Motivációm ennek a problémának a kiküszöbölését célozta meg, ugyanis diákmunkaként biciklis futárként dolgoztam, ahol elengedhetetlen volt a gyors és megbízható navigáció.

Munkámat a szenzorfüzióra, annak egyik széles körben elterjedt (és magyar vonatkozással is rendelkező) fajtájára, a Kálmán-szűrőre alapoztam. A telefon saját GPS-ét, a gyorsulásmérőt, a szögsebességmérőt és a magnetométert felhasználva megvalósult egy utófeldolgozáson alapuló térbeli helyadatokat számító alkalmazás. A MATLAB fejlesztői környezetben gyorsan és hatékonyan tudtam elkészíteni a programot. Működését próbák ellenőriztem, ahol fontos szempont volt az alkalmazás kritikus helyzetekben való helytállása. Témaköröm megismerése során számos hibaforrást és azok kiküszöbölését ismertem meg a mérés és főként a feldolgozás során. Ahhoz, hogy ezeket most

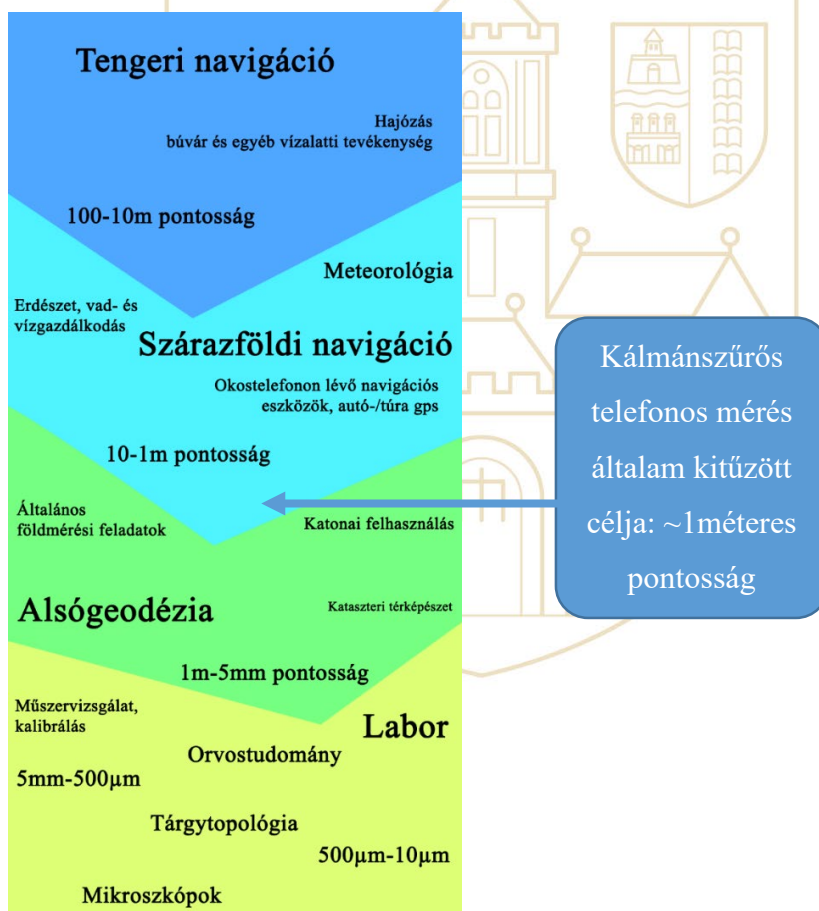


bemutathassam az olvasónak, széleskörű elméleti ismeretszerzésre volt szükségem.

Földmérő hallgatóként megtanultam, hogy a pontosabb adatok szélesebb spektrumú felhasználást tesznek lehetővé. Elért eredményeim lehetővé teszik számomra a jövőben, egy telefonon elérhető valós időben (nem utófeldolgozáson) alapuló applikáció fejlesztését, amellyel olyan feladatok elvégzése is lehetővé válik, amelyek most még csak egy mobiltelefonnál nagyságrendekkel költségesebb műszerek és szolgáltatások használatával lehetségesek.

1.1 Helymeghatározási feladatok csoportosítása pontosság szerint

Az előzőekben említett célkitűzésemet a következő skálán helyeztem el, amely bizonyos feladat helymeghatározáshoz köthető, mérési pontosságának igénye szerint sorbarendezett példákat tartalmaz.



Ábra 1 Helymeghatározást igénylő feladatok csoportosítása



Az 1. ábra különböző feladatok mérési pontosságának igénye szerint van sorba rendezve fentről lefelé haladva, a több 10 méteres mérésektől eljutva az igen nagy precizitást igénylőkig. Az ábrától jobbra látható jelölés mutatja, hogy kutatásommal a telefont, mint mérési eszközt hol lehetne elhelyezni.

2. Az adattárolás és a telefon szenzorai

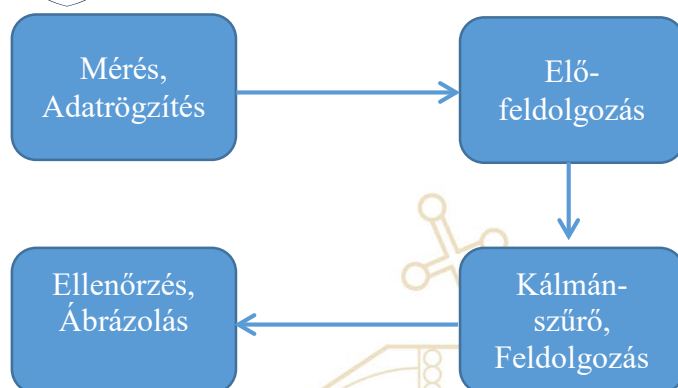
2.1 Az adattárolás

Elsősorban fontos meghatározni a mérés és a feldolgozás időbeli kapcsolatát. Alapvetően kettő lehetőség áll rendelkezésre: az utófeldolgozáson alapuló és a valós idejű feldolgozás.

Az utófeldolgozás sokszor bővebb, részletesebb adatokhoz vezet, lehetőség van a nyers adatok különböző szűrésére, mielőtt felhasználnánk azokat, a mérésekből könnyebben lehet származtatni további adatokat. Mindez annak köszönhető, hogy a mérés és a feldolgozás egyaránt erőforrást igényel, ez a kapacitás (főleg az egyszerűbb telefonok esetében) igen véges, azonban a két munkarész időben elkülöníthető, ezzel hatékonyabbá téve a feladatok elvégzését.

A valós idejű feldolgozás során egyszerre történik a mérés és a feldolgozás, így a technika fő előnye, hogy a felhasználó azonnal fontos információkhoz juthat.

Dolgozatom főleg az utófeldolgozás bemutatását részletezi, azonban munkám folytatásaként szükségesnek érzem a valós idejű módszer kidolgozását is, hiszen a különböző típusú feladatok más feltételeket követelnek meg, és a mobiltulajdonos feladatává válik az éppen alkalmasabb módszer kiválasztása.



Ábra 2 Az utófeldolgozás 4 lépése

Az ábrán az utófeldolgozás 4 fontos lépése látható és a köztük lévő logikai kapcsolat. Ezek közül az elsőt (mérés és adatrögzítés) a telefonnal és a rá telepített alkalmazással, a következő lépéseket pedig egy számítógép segítségével végezhetjük el.

2.1.1 Utófeldolgozás esetén

A telefon szenzorai által mért adatokat egy SensorData nevű alkalmazás segítségével tudtam felhasználni, ami egy egyszerű szöveges dokumentumba (.txt formátumban) mentette le a mérési idő során keletkezett adatokat. Ezek az állományok egyszerűen beolvashatók bármilyen programnyelvben, így ideális volt számomra a választott telefonos applikáció.

A szöveges állomány elején (fejlécében) a mérés tulajdonságait leíró adatok találhatóak. Ezek a következők: az adott szenzor neve, a mérésre felhasznált telefon gyártója és típusa, a szenzor maximális mérési intervalluma és a mérés felbontása.

Tartalmi felépítésük szempontjából egy táblázat következik, melynek oszlopai tabulátorral, sorai pedig sortöréssel vannak elválasztva. Minden sor egy időpillanatra vonatkozó mérések eredményét tartalmazza. Általánosan mindig az első oszlop tartalmazza az adott időpillanat pontos leírását, ez egy olyan szám, amely az 1970. január elseje óta eltelt ezredmásodperceket számolja, így biztosítva egyéni tulajdonságot minden mérési sornak. Ezzel a módszerrel meghatározott információt



időbélyegeknél nevezzük. Néha azonban anomáliaként két külön méréshez (sorhoz), ugyan azon időpillanat-érték társul, ezt a hibát egy későbbi fejezetben figyelembe veszem és javítom. A többi oszlop a szenzorra jellemző mértékegységben, általában a telefon (lokális) tengelyeinek irányába mért, más-más méréseket tartalmaz.

```
test_accelerometer_t3 - jegyzettömb
Fájl Szerkesztés Megtekintés

Trial: test_accelerometer_t3.txt
Phone/Tablet Model: Xiaomi M2003J15SC

Sensor Information
Name: bmi160_acc ACCELEROMETER
Maximum Range: 78.4532m/s²
Resolution: 0.0012m/s²

Unix Time (ms since Jan 1, 1970) X Y Z
1666688757847 -1.0520000457763672 0.32899999618530273 9.812000274658203
1666688757847 -1.274999976158142 0.15600000321865082 10.187999725341797
1666688757847 -1.152999997138977 -0.003000000026077032 10.21399974822998
1666688757847 -0.875 -0.04899999871850014 10.395999908447266
1666688757847 -0.7649999856948853 -0.006000000052154064 10.748000144958496
1666688757847 -0.7599999904632568 -0.25200000405311584 11.267999649047852
1666688757847 -0.6169999837875366 -0.47699999809265137 11.229000091552734
1666688757847 -0.43700000643730164 -0.5440000295639038 11.279999732971191
1666688757847 -0.33399999141693115 -0.4869999885559082 11.253000259399414
1666688757847 -0.32199999690055847 -0.3580000102519989 11.34000015258789
1666688757847 -0.29600000381469727 -0.2619999945163727 10.98799991607666
1666688757847 -0.21899999678134918 -0.07999999821186066 10.621000289916992

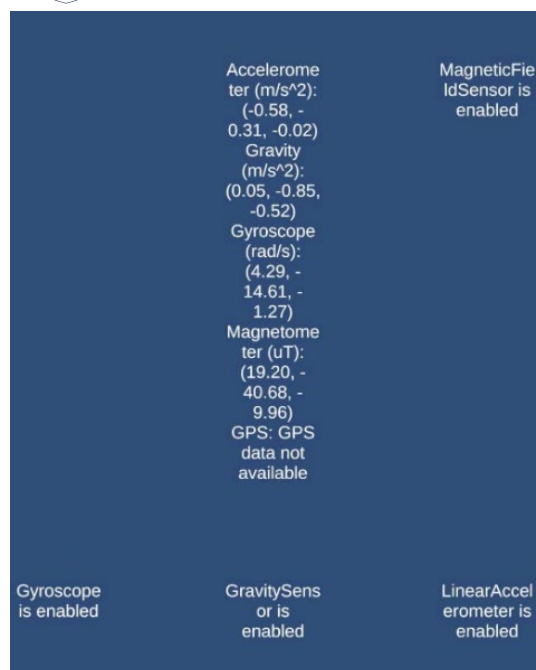
7. sor, 23. oszlop Unix (LF) UTF-8
```

Ábra 3 Rögzített adatállomány

Az ábrán egy adatokat tartalmazó szöveges állomány látható, amelyet a SensorData készített automatikusan.

2.1.2 Valós idejű kiértékelés

A telefonos alkalmazás létrehozása során szükségszerűvé vált, hogy a nyers adatokat ne eltárolt szöveges adatállományokból, hanem közvetlen a programba eljuttatva használjuk fel. Ehhez a Unity alkalmazásfejlesztő platform biztosította az elérést.



Ábra 4 Képernyőkép a készülő alkalmazásból

A képernyőképen a valós idejű mérések megjelenítése látható. Az általam fejlesztett program visszajelzést ad a felhasználónak az adott szenzorok állapotáról, vagyis arról, hogy jelenleg végez-e mérést, illetve használata engedélyezett-e, mint például a helyadat információgyűjtése.

Azért választottam ezt a fejlesztői környezetet, mert korábbi tapasztalataim alapján megbízható és hatékony fejlesztést tesz lehetővé. Az eredetileg videójáték fejlesztésére létrejött motorra mára egyre többen figyelnek fel, hiszen számos funkciója felhasználható más szakterületeken, mint például az animáció-készítés, mechanikai és fizikai szimulációk vagy az informatikai alkalmazások fejlesztése.

Középiskolában már alapszinten elsajátíthattam a platform használatát, így a jelenlegi alkalmazás fejlesztése dinamikusan halad.

A programok, függvények megírását a Microsoft Visual Studio-ban C# nyelven végzem.



```
31 {  
32     GyroText.text = "Gyroscope is enabled";  
33 }  
34 InputSystem.EnableDevice(MagneticFieldSensor.current);  
35 if (MagneticFieldSensor.current.enabled)  
36 {  
37     MagneticFieldSensorText.text = "MagneticFieldSensor is enabled";  
38 }  
39 }  
40  
41 // Start is called before the first frame update  
42 void Start()  
43 {  
44     // Check if location services are enabled  
45     if (!Input.location.isEnabledByUser)  
46     {  
47         Debug.LogWarning("Location services are not enabled by the user.");  
48         return;  
49     }  
50  
51     // Start location service with desired accuracy and update distance  
52     Input.location.Start(10f, 10f);  
53  
54     public Vector3 velocity = new Vector3();  
55     public Vector3 position = new Vector3();  
56  
57  
58     // Update is called once per frame  
59     void Update()  
60     {  
61  
62         // ACCELEROMETER  
63         Vector3 acceleration = LinearAccelerationSensor.current.acceleration.ReadValue();  
64  
65         Vector3 Grav = UnityEngine.InputSystem.GravitySensor.current.gravity.ReadValue();  
66  
67         // VELOCITY  
68     }  
69 }
```

Ábra 5 Saját fejlesztésű programkód részlet az MS Visual Studio-ban

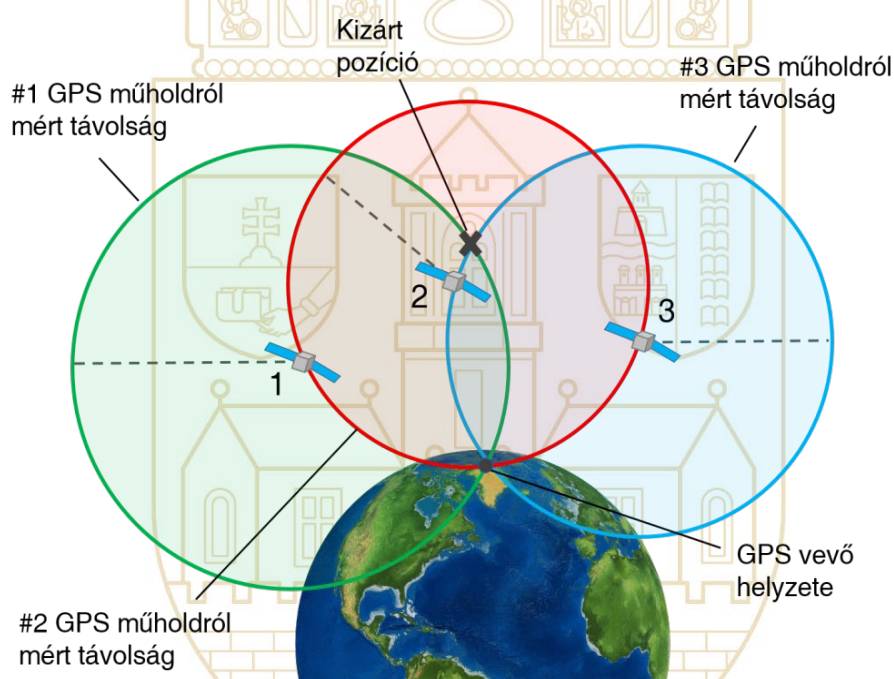
2.2 Felhasznált szenzorok

2.2.1 Műholdas helymeghatározó egység.

Mára számos, az egész bolygót lefedő, helymeghatározásban kulcsszerepet játszó műholdrendszer létezik, ezeket összefoglalóan GNSS (Global Navigation Satellite System), vagyis globális helymeghatározó műholdrendszereknek nevezzük. [1] [2] Ez a technológia a telefonos felhasználók számára általánosan 4.9 méteres pontossággal ad ismereteket a helyzetünkről. [3] A köznyelvben jobban elterjedt a GPS (Global Positioning System) kifejezés, mely annak a műholdas rendszernek a neve, amely a legkorábban nyújtott lehetőséget egyszerű civil felhasználásra, s amely alatt általában szinonimaként ugyanazt értjük, mint a GNSS, így az egyszerűség kedvéért én is így fogok hivatkozni erre a fogalomra a továbbiakban, annak tudatában, hogy egyéb műholdas rendszereket is használnak a telefonok helymeghatározás céljára.

Maga a helymeghatározás a GPS segítségével, a műhold és a vevő közti jel futási idejének mérésre vezethető vissza. A műholdak rádióhullámokat bocsátanak ki, melyeknek terjedési sebessége, a Föld légköre által okozott késleltetése és torzítása ismertek, illetve számba vehetők. A futási idő, és

a rádióhullám terjedési sebessége ismeretében távolság számítható minden műholdra. Általában 8-10 navigációs műhold fogható a nap bármely órájában, a matematikailag szükséges háromhoz képest ez bőven ad lehetőséget fölös mérések rögzítésére, a helymeghatározás kiegyenlítéssel történő pontosítására. Fontos megemlíteni, hogy szükség van a vevő órahibájának ismeretére. Ebből kifolyólag a kiegyenlítés során még egy meghatározandó ismeretlen van, és így a szükséges műholdak száma négyre nő. A műholdak pontos helyzetéből és a vevőkészüléktől való távolságukból a geodéziában térbeli ívmetszésként ismert számítással meghatározhatók a telefon pontos helyzetét leíró koordináták. Ezeket a koordinátákat NMEA szabványos adatformátumként tároljuk.

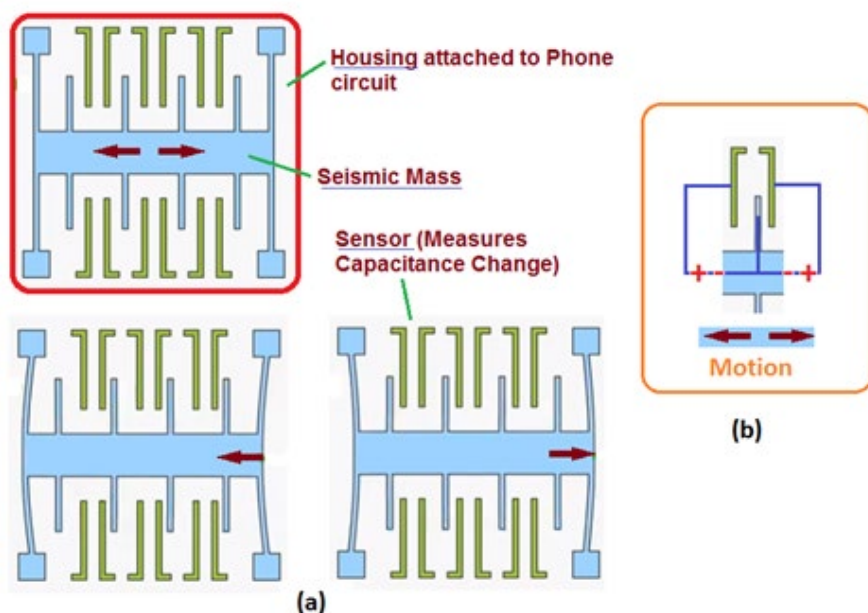


Ábra 6 A műholdas helymeghatározás

Az ábrán látható a GPS rendszer távolságokból meghatározott koordináták számításának szemléltetése [1], vagyis a térbeli ívmetszés.

2.2.2 Gyorsulásmérő (Accelerometer)

A gyorsulásmérő szenzor egy olyan érzékelő, amely a telefon mozgását, annak saját, a telefonhoz kötött, úgynevezett lokális tengelyein megjelenő komponenseit méri. [4] Működésének alapelve, hogy a telefonhoz rögzített ellenállásmérők közt szeizmikusan szabadon mozgó, a tömegvonzás által elmozdított elemek befolyásolják a mért kölcsönhatást. A szenzor ezen értékek egymáshoz viszonyított mértékét feldolgozva képes becsléseket végezni a tengelyek irányába. Ez azonban hibákkal terhelt.



Ábra 7 MEMS szenzor, gyorsulásmérő a telefonban

A legszembeötlőbb gond a Föld gravitációs vonzása, amely gravitációs gyorsulásként, a bolygó középpontja felé mutató irányba hat minden földi tárgyra átlagosan 9.81m/s^2 gyorsulással, (ami a 45° földrajzi szélességre, tengerszinten érvényes). Telefonunk ezt a gyorsulást is érzékeli, s ez alapján észleli például, hogy a készülékünk 'álló' (portrait) vagy 'fekvő' (landscape) pozícióban van-e. Amennyiben térbeli tájékozás során helytelenül határozzuk meg a tömegvonzási erő vektorát, tehát nem egyezik meg a mért érték és a modellezett globális Z tengely, ez megjelenik az erőt leíró vektorkomponensek változásában, így hibával



terhelve mérésünket. Erre kiemelten oda kell figyelni, hiszen akár kismértékű 'ferdeség', a mérési idő múlásával arányosan torzítja a mérést.

A helyi G (nehézségi gyorsulás vektor) pontos meghatározására az utófeldolgozás megoldást nyújt a mérések azon szakaszának kiátlagolásával, mikor még egy helyben állt a telefon. Ehhez természetesen szükséges az is, hogy mérésünk megkezdésekor tudatosan 3-4 másodpercig ne végezzünk nagymértékű mozgást, elfordítást a telefonnal.

A másik akadály, hogy az okostelefonokban dinamikus gyorsulásmérők vannak. Amint már korábban is említettem, a telefon koordinátarendszere nem egyezik meg a végleges modellezési koordinátarendszerrel. Azonban egy számítás előtti, illetve számítási időpillanatonként újraszámolt forgatómátrixsal a lokális elfordulást figyelembe véve átszámíthatók a gyorsulások a globális tengelyek irányába. Amennyiben jó a tájékozásunk, a már korábban megismert gravitációs gyorsulás vektor egyszerűen kivonható méréseinkből, az elmozdulás pedig a gyorsulás kétszeres időszerinti deriváltjaként mérhetővé válik.

2.2.3 GPS-ből számított ál-sebességmérő

A mozgás meghatározására közvetett módon a sebesség is alkalmas. A gyorsulásmérőtől függetlenül a GPS mérésekből is szimulálható egy 'hamis' sebességmérő szenzor, amely összevethető a számítás során. Fontos megjegyezni, hogy a módszer csak abban az esetben alkalmazható, ha ismertek az előző mérési időpillanat helyadatai, és az azóta eltelt idő mértéke. A későbbiekben részletesen tárgyalom e szenzor jellegzetességeit és alkalmazását.

2.2.4 Szögsebességmérő (Gyroscope)

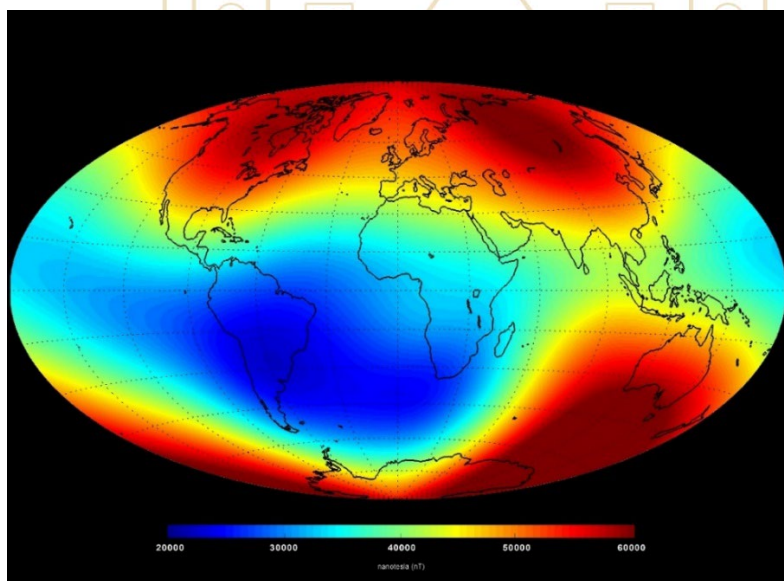
Az ember az elfordulást, a tájolást nehezen érzékeli. Ehhez azonban segítséget nyújt a telefon szögsebességmérője. A magyar elnevezése a giroszkópnak onnan ered, hogy a már említett készülék azt méri, hogy három lokális tengelyéhez képest milyen sebességgel fordul el. Mérési mértékegysége radián/másodperc. A telefonban lévő giroszkópok, melyek

a MEMS (micro-electromechanical systems) mikroelektromechanikai rendszer részei, méretükből és költségükből fakadóan nem tartalmaznak folyamatosan vákuumban pörgő részeket, így a korábban szintén így nevezett eszközökkel ellentétben nem alkalmasak közvetlen tájékozásra, ezt a szerepet a gyorsulásmérő és a magnetométer ötvözése tölti be. Azonban jelentős hatással vannak a tájékozást leíró kvaterniók változására, ezzel a telefon-mozgás irányának meghatározásában.

2.2.5 Mágneses tér mérő (Magnetometer)

A következő szenzor a Föld mágneses mezőjét jellemző mennyiségnek, a mágneses indukciónak (télerősségének) az értékét nanoTeslában adja meg a készülék három tengelyére vonatkozóan. A három komponensből egy vektor számítható, mely a Föld mágneses északi pólusa felé mutat. A magnetométer az orientáció, tájolás szempontjából szükséges, hiszen megállapítható, hogy a telefonhoz képes milyen térbeli irányban található a mágneses észak.

Jelentős problémát okoz azonban, hogy a legtöbb tárgy rendelkezik mágneses mezővel, amelyek kis mértékben, de sokszor és sok irányban, tehát kiszámíthatatlanul torzítják a szenzor pontosságát. Ezt mérési zajnak hívjuk és a számítás során figyelembe is vesszük.



Ábra 8 A Föld mágneses terének erősségét jellemző mennyiség nagysága a Föld felszínén



3. Adatok

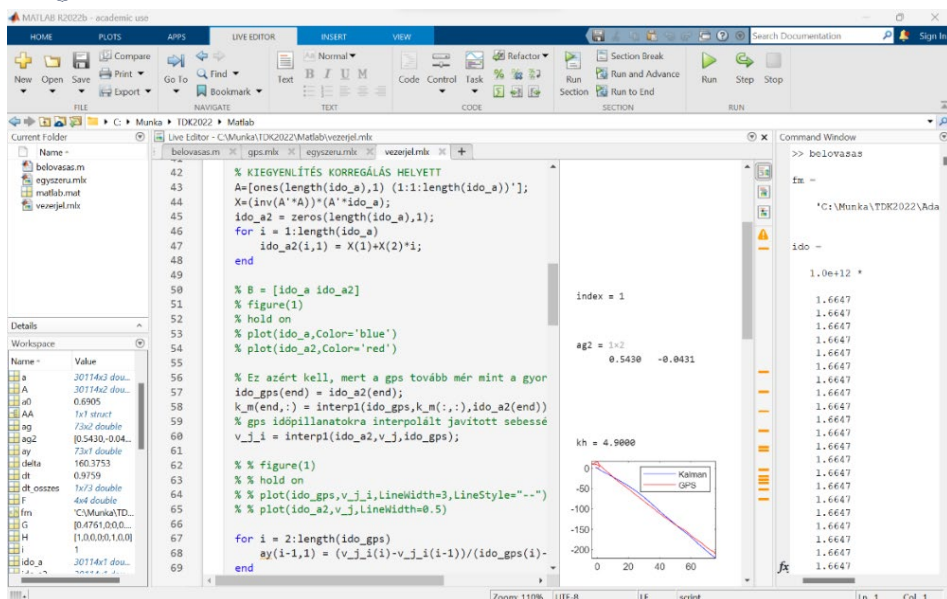
3.1 Az adatnyerés jellemzése

Minden mérésemet ugyanazon telefon használatával végeztem el. A mobil típusa Xiaomi Redmi Note 9, ami teljesen alkalmas volt az adatszerzéshez, hiszen a felhasznált szenzorokkal együtt összesen 11 különböző érzékelővel rendelkezik. Ezen műszerek mérési tartománya, adott idő alatt érzékelhető mérések száma (mérések frekvenciája) és a mérések középhibája is megtalálható a gyártó honlapján, illetve a már említett SensorData nevű telefonos applikációban.

A SensorData által eltárolt .text formátumú adatokat áthelyeztem számítógépemre. Az utófeldolgozásra ideális környezetnek bizonyult a MATLAB, így itt végeztem a mérésen kívül minden egyéb munkarészt. A számítógépes platform tette számomra lehetővé, hogy a telefonos alkalmazás elméleti és gyakorlati kezdeti fázisait megvalósítsam.

3.1.1 A MATLAB, mint fejlesztői környezet

A MathWorks által fejlesztett program egy olyan numerikus számításokra specializált logikával rendelkezik, ami a mátrixműveletek gyors kiszámítását teszi lehetővé. Ezen kívül a felhasználókat helyezi előtérbe azáltal, hogy képes függvények és adatok ábrázolására, algoritmusok implementációjára és felhasználói interfész egyéni igények szerinti kialakítására. A program az M programozási nyelvet használja, amit nagyrészt felhasználóbarátnak ítélt meg, mivel interaktív sűgóval és egyéni függvények készítésének lehetőségével is rendelkezik. Logikailag a Python programozási nyelvhez hasonlít, aminek alapja a C nyelv, így az egyetemen tanult programozási tudásomat is felhasználhattam, viszonylag gyorsan tudtam haladni a gyakorlati megvalósítással. A szoftver használatához szükséges hozzáférési jogosultságot az Óbudai Egyetem biztosította.



Ábra 9 A MATLAB felhasználói környezete

Az ábrán látható az említett számítógépes felület, amiben megvalósítottam a számolás gyakorlati részét.

3.1.2 A tesztelés

A természetben történő folyamatok igen részletesek és összetettek, ezért azokat megpróbáljuk törvényszerűségekre foglalni a modellezésük és működésük megértése végett. Szükséges volt az említett, leírhatóság szempontjából komplex mozgások leegyszerűsítése, ezzel könnyebbé téve a matematikai és fizikai folyamatok összefüggéseinek megértését.

A mérést kerékpáron, a székesfehérvári Hosszúsútatóren végeztem. Két ívet is terveztem útvonalamba, ezzel bonyolítva az adatgyűjtést. A tengerszint feletti magasságváltozást nullának tekintettem, mert a terep valóban közel sík volt. Azonban fontos megjegyezni, hogy a kód már képes térbeli koordinátákat fogadni, és eredményül adni egyaránt.

A kezdeti tájékozáshoz szükséges volt pár másodpercig egy helyben állni, majd egy előre megtervezett útvonalon haladtam.

A telefont úgy helyeztem el a kerékpáron, hogy annak egyik helyi tengelye (nevezetesen az y) megegyezzen a mozgásom irányával, a képernyő síkja pedig párhuzamos legyen a felszínnel.

3.2 Előfeldolgozás

3.2.1 Kritikus mérési helyzetek szimulálása

A program alkalmazhatóságának jó ismérve a kritikus helyzetek vizsgálata. Ilyen mérési helyzetnek számít, ha a sűrű növényzet, vagy egyéb takaró tárgy megszakítja a GPS jelek fogadását. Ehhez külön mérést nem kellett végezni, elég volt csupán a tárolt adatokból kivenni adott mérési sorokat.



Ábra 10 GPS jelvesztés szimulálása egyenes szakaszon

A későbbiekben ilyen típusú szimulált mérésekből is fogok következtetéseket levonni.

3.2.2 Egyenes vonalú mozgás vizsgálata

A program fejlesztésének korai szakaszában még nem tudtam kezelni az elfordulásokat, ezért egyenes vonalú mozgásokat vizsgáltam. A telefon dőlését úgy határoztam meg, hogy a globális koordinátarendszerbe forgatott mért gyorsulásokból számolt sebességek jól visszaadják a GPS mérésekből levezethető sebességeket.

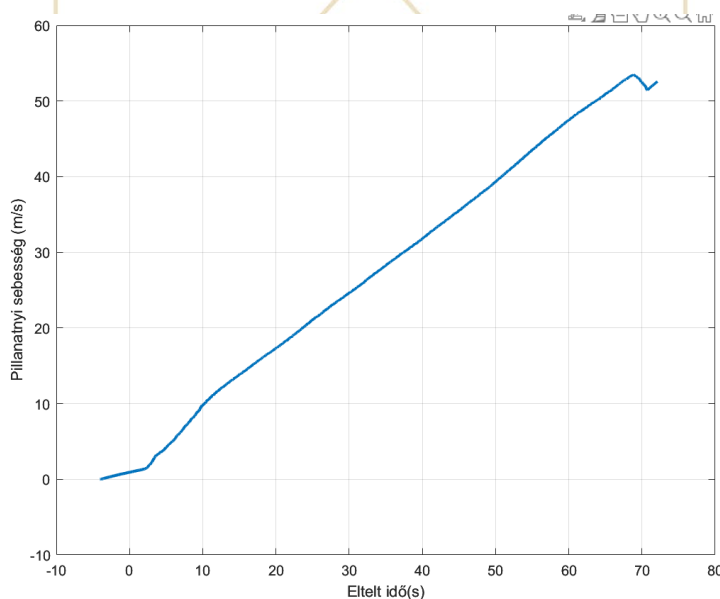


A nyers adatokat az *importdata()* függvénnyel olvastam be a program rövidtávú memóriájába, majd vektorokba és mátrixokba mentettem el a beolvasott fájl oszlopait. A gyorsulásmérő nyers adatai többféle hibával terheltek, ide tartozik az úgynevezett 'bias', magyarul elferdülés, elcsúszás is. A hiba eredete, hogy a mérőkészülék nem lett tökéletesen vízszintes irányba állítva, így a mért gyorsulás valamennyivel több a valóságnál. Az ebből számított pillanatnyi sebesség az idő múlásával egyenletesen torzul. A pillanatnyi sebesség a következőképpen számolható:

$$\bar{v}_i = \bar{v}_{i-1} + \bar{a}_{i-1} \cdot (\bar{t}_i - \bar{t}_{i-1})$$

ahol a kezdő sebességet (v_0) nullának tekintjük.

Ez elvégzendő minden komponens minden időpillanatra. A következő, ábrán a telefon y irányú pillanatnyi sebességét ábrázolom az eltelt idő szerint, amely jól szemlélteti a jelenséget.



Ábra 11 A bias-t tartalmazó számított pillanatnyi sebesség

A bias, vagyis az elferdülés javítása a következő egyenlettel számítható:

$$a_0 = \frac{v_n - v_1}{t_n - t_1}$$

ahol n a mérések száma



A mérést több időintervallumra osztva, több bias korrekciós együtthatót számolva pontosabb javítást érhetünk el, azonban az egyszerűség kedvéért én ugyanazzal az értékkel javítottam az összes gyorsuláserőteket. Mivel a bias minden tengely mérését más mértékben befolyásolja, így más-más javítási együtthatót kell számítani a különböző irányokba. A hibával terhelt sebesség a következő képlettel javítható, az immáron kiszámított a_0 -val.

$$\bar{v}_i = \bar{v}_{i-1} + (\bar{a}_{i-1} - a_0) \cdot (\bar{t}_i - \bar{t}_{i-1})$$

A sebességekhez a hozzájuk tartozó javításokat kell alkalmazni.

3.2.3 GPS-ből számított ál-sebességmérő

Amennyiben szeretnénk alkalmazni az ál-sebességmérőt, ügyelni kell, hogy megfelelő időbélyegeket kapjon minden mérés, különben az 'elcsúszott' adathalmaz jelentős hibával terhelheti a Kálmán-szűrő frissítés részét.

Lehetőség van a `writematrix()` paranccsal a SensorData-hoz hasonló adatállomány létrehozására. Ezt a függvényt használtam az ál-sebességmérő eredményeinek tárolására, a későbbi felhasználás érdekében.

Az ál-sebességmérő alkalmazásának bemutatására a későbbiekben kerül sor.

3.2.4 A telefon órájának anomáliája

A következő akadályt az okozta, hogy a telefon órája, mely minden szenzor mérésének első oszlopában lévő idő meghatározásáért felel, véletlenszerű időközönként rossz, hamis időpillanatot határoz meg. Ennek oka feltételezhetően az lehet, hogy a telefon processzora teljesítményének határán volt, hiszen 2-3 ezredmásodpercenként számtalan adatot kellett kezelnie, tárolnia a mérés során. Habár a hiba elhanyagolható volna, ez mégis gondot okoz számunkra. A következőkben a hiba figyelembevételének okát fejtem ki. A mobil GPS vevője közel



másodpercenként képes mérni, a 2-3 ezredmásodpercenként adatot tároló egyéb szenzorokhoz képest. Hogy a Kálmán-szűrőt alkalmazni lehessen, egységes időközöket kell alkalmazni a mérésekre.

Időközökre értelmezett mérések

Az első teendő az általános időközök megválasztása. A közös időbélyeglista létrehozásához két adathalmazt fésültem össze. A GPS időpillanatait tekintettem ideálisnak, mivel a többi szenzor könnyebben és pontosabban alkalmazkodik a ritkásabb mérésekkel rendelkezőhöz, mint fordítva. Az összes időpillanatot felhasználtam, amelyekben a GPS mért. Ezek közül ki kellett válogatni azon időbélyegeket, amikor már az összes többi szenzor is mért, mivel a program nincs felkészítve olyan kritikus helyzetre, amiben valamelyik szenzor (a GPS-en kívül) ne rendelkezne méréssel. Feltételezzük, hogy ilyen kritikus pillanat csak a mérés megkezdése és befejezése környékén léphet fel. A másik komponense az időbélyeglistának a dinamikusan állítható mintavételezési időpillanatok. Ezzel a funkcióval lehetővé válik másodperc alatti koordinátameghatározás, vagyis a GPS-nél gyakoribb mérés. Ha ezt a *Freq* változót 1 értéken hagyjuk, akkor a kód, csak a GPS időbélyegeit használja.

A szenzorok közös időbélyegénkénti értékét interpolálással becsültem meg. Ehhez az *interp1()* függvényt használtam a következő szintaktikával:

$$\bar{v}_{\text{közös időnkénti}} = \text{interp1}(\bar{t}_{\text{gyorsulásmérő}}, \bar{v}, t_{\text{közös}})$$

A program nyelvén ez azt határozza meg, hogy az anomáliamentes gyorsulásmérő időpillanataiban értelmezzük a bias-mentes pillanatnyi sebességeket, majd a közös időpillanatokhoz becsüljük a hozzájuk tartozó sebességértéket. A kapott pillanatnyi sebességek vektorának elemei segítségével megbecsüljük azt a vektort, amely azokat a gyorsulásokat tartalmazza, melyeket mérnie kellett volna egy olyan gyorsulásmérőnek, mely kizárólag a közös-időpillanatokban mér.



$$\bar{a}_{i-1} = \frac{(\bar{v}_i - \bar{v}_{i-1})}{(\bar{t}_i - \bar{t}_{i-1})}$$

A kapott $\bar{a}$ gyorsulás-vektor egy háromszlopú gyorsulás-mátrix.

Időbélyegkiegyenlítés

Az említett *interp1()* parancsnak a feltétele, hogy minden nyers mérési időbélyegnek egyedi értéket kell felvennie. Ez azonban nem minden esetben teljesült.

A problémát úgy szüntettem meg, hogy az időpillanatok értékét és indexét pontok koordinátájaként értelmeztem. Ezen pontokra kiegyenítő számítással egy, a pontokra legjobban 'rásimuló' egyenes illeszthető, melynek két paramétere (a_0 – y tengely metszete és a_1 - egyenes meredeksége) alapján írható fel. A paraméterek megismeréséhez szükség van az $\bar{A}$ alakmátrixra, és az $\bar{l}$ tisztatag vektorra. Az alakmátrix oszlopai az egyenes egyenletének paraméterek szerinti deriváltjai, sorai pedig az adott mérés értékeinek (x - időmérés indexe, y – időmérés értéke) behelyettesítésével tölthetők fel. A tisztatagot jelen esetünkben a mért idők értékei fogják feltölteni.

$$\bar{A} = \begin{bmatrix} 1 & i \\ 1 & i+1 \\ \vdots & \vdots \\ 1 & n \end{bmatrix}; \bar{l} = \begin{bmatrix} t \\ t_{i+1} \\ \vdots \\ t_n \end{bmatrix}$$

Az egyenes paramétereit egy $\bar{x}$ vektorban kapjuk meg következő képpen:

$$\bar{x} = (\bar{A}^T \cdot \bar{A})^{-1} \cdot (\bar{A}^T \cdot \bar{l})$$

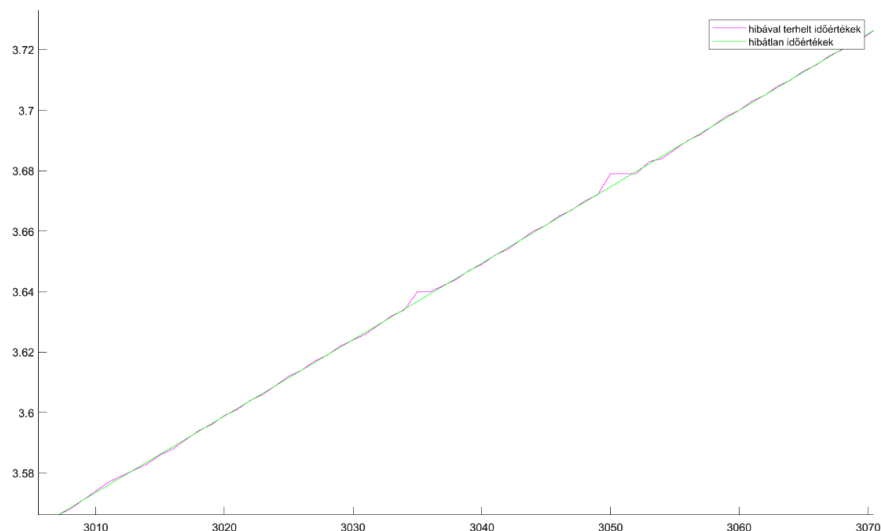
Ezt követően minden addigi gyorsulásmérő-időpillanatra kiszámoltam, hogy a fentebbi paraméterekkel rendelkező egyenes az adott helyen milyen értéket venne fel. Mivel az egyenes a tengelyeken ferdén helyezkedik el, így biztosan nem lesz két azonos értékkel rendelkező időpillanat.

$$\bar{s}_i = \bar{x}(1) + \bar{x}(2) \cdot i$$

A következő, 12. ábrán magenta színnel az telefon órája által mért eredeti időpillanatok, zölddel a kiegyenlítéssel számolt egyenesen elhelyezkedő



időpillanatok láthatóak. Az óra időanomáliái jól megfigyelhetők azokon a helyeken, ahol az egyenestől eltérnek.

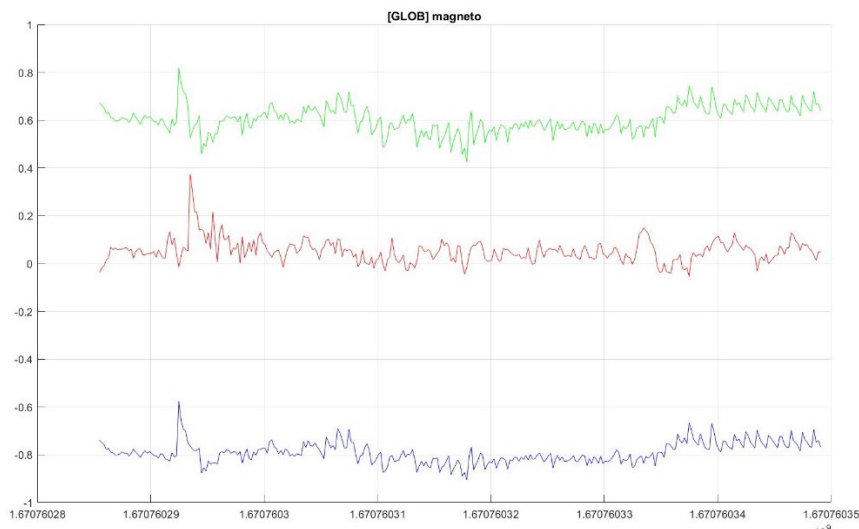


Ábra 12 Időanomáliakiegyenlítés

3.2.5 Magnetométer előfeldolgozása

A magnetométer a kezdeti, és minden mintavételezési időpontbéli térbeli tájékozáshoz elengedhetetlen információkat biztosít számunkra. Ahhoz, hogy felhasználhassuk ezeket a célokra, előfeldolgozás szükséges.

Először is a magnetométer méréseit tartalmazó vektort lenormáltam, mivel csak az irányára volt szükségem, a hosszára nem. Ezt követően a közös mintavételezési időpontokra interpoláltam, majd a hasonló módon előfeldolgozott gyorsulásmérő vektorával egy tájékozási kvaterniót számítottam. Az imént említett lépést egy külön függvényben egyesítettem, melyre a programomban hivatkozni tudok, így átláthatóbbá téve a struktúráját a kódnak.



Ábra 13 Tájékozási kvaterniók ellenőrzése

Az ábrán láthatóak a tájékozási kvaterniókkal visszaforgatott lokális magnetométer vektorkomponensei. Ideális esetben 3 változatlan, tehát x tengellyel párhuzamos egyenest kéne kapni, ez a számítási képesség azonban csak elméletben létezik, az elért eredményeket elfogadhatónak tekintettem.

3.2.6 Vetületi rendszer meghatározása

A GPS által mért koordináták ellipszoidi rendszerben vannak, tehát minden pontot egy szélesség, egy hosszúság és egy ellipszoid feletti magasság jellemez. Ez nekünk számítás szempontjából nem ideális, ezért érdemes másik vonatkozási rendszerbe transzformálni.

A legegyszerűbb módszer egy vetület nélküli rendszerbe való átszámítás, mellyel koordinátáink méterben lesznek meghatározva, és egy úgynevezett Északias-Keleties koordinátpárral lesznek definiálva.

$$É_{méter} = GPS_{szélesség} \cdot 111000$$

$$K_{méter} = (GPS_{hosszúság} \cdot 111000) \cdot \cos(GPS_{szélesség})$$

A módszer előnye, hogy a modellezéshez semmiféle transzformációra nincs szükség, hátránya pedig pont ebből következik, mivel nem kapunk



szabályos vetületi koordinátákat, tehát például nagyobb távolságok megtétele esetén geometriai ellentmondásokba ütközhetünk.

Szükségét éreztem a későbbi fejlesztésekre való tekintettel egy vetületi rendszer bevezetésének, mellyel az imént említett hátrány (nagy megtett távolságok esetén) is elhanyagolható. A hazánkban is alkalmazott EOVS (Egységes Országos Vetület) több szempontból is előnyös volt számomra. Koordinátarendszere métert használ, és teljes Magyarország területe az első szögnegyedben helyezkedik, ezek a tények a számolást és az ábrázolást egyaránt ideálissá tették. Emellett a telefon által adott WGS84 (forgásellipszoid) ellipszoidi koordinátáinak átszámítását konzulensem által korábban elkészített, 'proj' nevű programban megadott kódsor elhanyagolható mértékű hibával képes azonnal átváltani. [5]

```
cs2cs -r -f "%.4f" +proj=latlong +datum=WGS84 +to +proj=somerc +lat_0=
47.14439372222222 +lon_0=19.04857177777778 +k_0=0.99993 +x_0=650000 +y_0=
200000 +ellps=GRS67 +nadgrids=C:\Munka\TDK2024\proj\etrs2eov_notowgs.gsb
+geoidgrids=C:\Munka\TDK2024\proj\geoid_eht2014.gtx +units=m +no_defs
```

Ábra 14 Az átváltásért felelős wgs2eov.bat nevű kód

3.2.7 Forgatómátrix a két rendszer között

Síkbeli forgatás

A kutatás kezdeti szakaszában egy egyszerűsített modellt alkalmaztam, ahol az okostelefon helyzetét vízszintesnek tekintettem, az eszköz lokális és a modellezés globális koordinátarendszere közti átszámítást egy egyszerű forgatómátrix biztosítja, legyen a jele: $\bar{M}$. Felírásához ismernünk kell a δ elforgatási szöveget. Ezt a GPS mérések méteres mértékegységben lévő értékeinek utolsó tagjaival, és az $\text{atan2}()$ függvénnyel számítható ki, a következő kódsorral, ahol n a mérések száma:

$$\delta = \text{atan2}(\text{GPS_X}(n), \text{GPS_Y}(n))$$

Az eredményt radiánban kapjuk meg, így a fent megkapott δ -t még meg kell szorozni $\frac{180^\circ}{\pi}$ -vel, ahhoz hogy fokban kapjuk meg az eredményt. A forgatómátrix a következőképp fog kinézni:



$$\bar{M} = \begin{bmatrix} \cos \delta & -\sin \delta \\ \sin \delta & \cos \delta \end{bmatrix}$$

Az utolsó lépés, hogy elforgassuk a telefon helyi méréseit, ezen esetben a GPS mérései és az y irányban mért gyorsulást. GPS mérésenként, a két mért koordinátánkat oszlopvektorként értelmezve, a forgatómátrixszal baloldaltól megszorozzuk.

$$GPS_g(1,2) = (\bar{M} \cdot [GPS_l(1) \quad GPS_l(2)]^T)^T$$

A gyorsulásmérő 'helyére forgatása' hasonló az előbb felírt képlethez, azonban mivel nullának tekintetem az x-irányú gyorsulást, így az egyenlet jobb oldala nullával lesz egyenlő. Amennyiben figyelembe akarjuk venni a haladási irányunkra merőleges gyorsulást, azt a 0 helyén kell megtenni. Ne feledjük, az $\bar{a}_y$ alatt nem az eredeti gyorsulás mérő értékei értendők, hanem a GPS-időpillanatokra becsültek.

$$\bar{a}_g = (\bar{M} \cdot [0 \quad \bar{a}_y]^T)^T$$

Térbeli forgatás

Ha nem egyenes vonalú mozgást is vizsgálni szeretnénk, szükség van egy kezdeti- és minden időpillanatban újabb $\bar{M}$ mátrix felírására, amely ezúttal nem egy egyszerű síkbeli forgatási mátrix lesz, hanem egy 4 oszlopból és sorból álló kvaterniók forgatómátrix. [6] Minden új $\bar{M}$ mátrix felírásához szükség van egy $\bar{R}$ mátrixra is, mely a forgatómátrixot transzformálja. Az $\bar{M}$ és az $\bar{R}$ mátrix is 4-4 együtthatóval fejezhető ki, melyből az $\bar{M}$ komponenseit az állapotvektorban tároljuk.

A kvaterniók alkalmazásának előnye más térbeli forgatást leíró módszerekkel szemben abban rejlik, hogy mindössze 4 számmal kifejezhető (egy klasszikus térbeli forgatási mátrix 9 eleméhez képest), illetve nem áll fent a gimbal lock jelenség veszélye a roll-pitch-yaw forgatási szögeket alkalmazó módszerhez képest.

A kezdeti tájékozáshoz tehát definiálnunk kell a telefon lokális tengelyei és az általunk alkalmazni kívánt globális vonatkoztatási rendszer



tengelyeinek kapcsolatát. Amennyiben egy helyben állunk, a telefon rendszerében mérhetjük a helyi függőlegest a gyorsulásmérő segítségével, ami kvázi megegyezik a globális vetületünk z tengelyével. Emellett szintén ismertek a lokális rendszerünkben mért mágneses északi pólus irányába mutató vektor térbeli komponensei, melynek a vízszintes (helyi függőlegesre merőleges sík) vetületére vagyunk kíváncsiak. A két mért vektor által kifeszített síkra merőleges harmadik tengely már meghatározható vektoriális szorzatként. A felsorolt lépésekkel tehát definiáltuk a globális koordinárendszer tengelyeit a telefon lokális rendszerében. Ez számunkra már elegendő adat ahhoz, hogy meghatározzuk a két rendszer közti elforgatás tengelyét és mértékét (kvaternió) két lépésben.

Először kiegyenlítéssel iterációs módszert alkalmazva meghatározzuk a három tengely körüli elforgatásokat. Ehhez kitalálunk egy kezdőértéket, megvizsgáljuk, hogy ha megszorozzuk azzal a forgatási mátrixszal a lokális méréseinket, mekkora eltérés mutatkozik a mért értékekhez képest, majd a különbséget figyelembe véve ismét módosítjuk a kezdőértékeket, vagyis a tengelyek körüli elforgatást. A tapasztalatok azt mutatják, hogy 3 iteráció elegendően pontos, és a számítási erőforrása még nem számottevő.



```
20 for it=1:3 % iteraciok szama
21     par0=par00;
22     for i=0:3 % parc.deriv.-ok szama: 0-tol parameterek szama
23         par=par0;
24         if i~=0
25             par(i)=par(i)+eps;
26         end
27
28     % egyes parameterek kiszedese a 'par' vektorbol
29     fi=par(1);
30     om=par(2);
31     ka=par(3);
32
33     % kozvetito egyenletek a parameterek aktualis ertekeivel
34
35     % --- kozvetito egyenletek ---
36     forg_fi=[1 0 0
37             0 cos(fi) -sin(fi)
38             0 sin(fi) cos(fi)];
39     forg_om=[cos(om) 0 sin(om);
40             0 1 0
41             -sin(om) 0 cos(om)];
42     forg_ka=[cos(ka) -sin(ka) 0
43             sin(ka) cos(ka) 0
44             0 0 1];
45     R=forg_fi*forg_om*forg_ka;
46     lok_gy_SZ=R*glob_gy;
47     lok_m_SZ=R*glob_m;
48     % eredmenyuk: a 'szamitott ertekek'
49
```

Ábra 15 Tájékozási forgatómátrix

Az itt látható programkódrészlet felel a két tengely közti forgatás definiálásáért.

Ezt követően pedig a kapott forgatásokat egyetlen forgatómátrixba felírjuk, majd ezt kvaterniókká transzformáljuk. [7]



4. Módszerek

4.1 A Kálmán-szűrő

Az első probléma, ami felmerült, hogy bár minden felhasznált szenzor valamilyen szinten kapcsolódik a helymeghatározáshoz, mindegyik különböző jellegű és mértékegységében is különböző tulajdonságokat határoz meg. Ezzel a problémával, és ennek megoldásával, az úgynevezett sensor fusion-el a Kálmán-szűrő foglalkozik, amely az egész dolgozatom központi eleme. A következő fejezetben Kálmán Rudolf Emil rendszerének kialakulásáról, előnyeiről és működéséről, egy későbbi fejezetben pedig a feladatomba való implementációjáról fogok beszélni.

4.1.1 Kálmán Rudolf Emil

Elsősorban a szűrő szellemi atyját, Kálmán Rudolf Emilt életét és munkásságát ismertetem. Az 1930-ban született, 2016-ban elhunyt magyar-amerikai villamosmérnök és matematikus a műszaki tudományok terén végzett számos kutatást. Kifejezetten érdekelte, hogy a földi tárgyak, folyamatok és mozgások állapotait hogyan lehet meghatározni, megbecsülni. Számos tudományos publikációja ehhez köthető, mint például a sokdimenziós állapottermodellek axiomatikus elmélete, ezen túl továbbfejlesztette a lineáris dinamikus rendszerek elméletét. A matematika területén a statisztikával foglalkozott, ahol többek közt megalkotta a nemlineáris diszkrét rendszerek állapotterére alkalmazható matematikai eljárást, a Kálmán-szűrő elvét.

Az Óbudai Egyetem őt választotta legelső díszdoktorává 2010. június 30-án. Később az Óbudai Egyetemet fenntartó alapítványt is ő róla nevezték el.



Ábra 16 Kálmán Rudolf Emil (1930-2016)

4.1.2 A Kálmán-szűrő előnyei

A metódus nagyszerűsége több tulajdonságában is megnyilvánul. A mért, hibával terhelt adatokból kiszűrhetővé, figyelembe vehetővé válik a zaj, a bizonytalanság és a pontatlanság. Hiányos információk esetében is rekurzív paraméterbecsléssel megállapítható az adott rendszer időállapota. Működésének részletes leírására később visszatérünk.

4.1.3 A Kálmán-szűrő alkalmazása

Kezdetben az Apollo-programban alkalmazták az eljárást, az űrhajók, és a Hold-szondák navigációs rendszerének optimalizálására. Ezeken túl a repülésirányításhoz alkalmazták a '60-as évektől. Később továbbgondolva rendszermodellezéshez, és az algebrai rendszerelmélet megalkotásához használták fel a szűrőt. A Kálmán-szűrő túlnőtte névadóját, hiszen a mai napig alkalmazzák az algoritmust repülőgépeknél, űrhajóknál, robotrepülőgépeknél és egyéb jelfeldolgozó rendszerekben.

4.2 Működésének alapja

Működésének alapja, hogy az érzékelők által kifejezett adatokat a telefon állapotát leíró, egymással függő vagy független kapcsolatban álló állapotkomponenseknek tekintjük. Minden időpillanatban meghatározunk egy jóslást arról, hogy szerintünk mi lesz a mobil következő állapota, melyet az előző időpillanat-beli állapotából becsüljük. Ezt követően a szenzorok éppen mért adatainak figyelembevételével megjavítjuk becslésünket. A szenzorok pontosságára is tekintettel vagyunk, ezt dinamikusán változtathatjuk, s megadva a módszer magától határozza meg az adott



állapotkomponensekre adott javítások súlyát. Így a kevésbé pontos szenzorok (pl.: GPS) kisebb mértékben befolyásolják a meghatározott helyzetet. [8]

A beolvasott nyers adatokat elsősorban fel kell dolgozni, a program belső adataiként el kell tárolni. A következő fejezetben bemutatom, mely szenzorokat használtam fel, azok mérési eredményeit milyen módon mentettem el.

4.3 A Kálmán-szűrő alkalmazása

4.3.1 Bevezetés

A Kálmán-szűrő két fő szegmensből épül fel: jóslás (predict) és frissítés (update). A következő részben erre a két egységre bontva vezetem végig a szűrő elméletét, a programkódba fejtett példákkal végigkísérve. Az imént említett kettő lépést egy úgynevezett állapotvektorra $\bar{x}_{k|k}$, és annak kovariancia mátrixára $\bar{P}_{k|k}$ értelmezzük. Az állapotvektor egy oszlopvektor, amelynek alkotóelemei leírják a vizsgált tárgy állapotának számunkra lényeges tulajdonságait. Ilyen komponensek jelen esetünkben a telefon térbeli orientációját leíró – már korábban említett – tájékozási kvaterniók; x, y és z irányú globális (EOV) koordinátái; ugyanezen tengelyek irányába mutató pillanatnyi sebesség és magnetométer méréseinek vektorkomponensei; illetve a korrekciós komponens a gravitáció hatásának kiszűrésére.

Amennyiben térben vizsgáljuk a mozgást, z irányú koordináta, illetve pillanatnyi sebesség is megjelenik. Ha nem egyenes vonalú mozgást is vizsgálni szeretnénk, szükség van a már említett tájékozási kvaterniókra.

4.3.2 A jóslás

A szűrőt minden közös időpillanatban használnunk kell, ezért érdemes szintaktikailag egy ciklusba foglalni a számítást. A jóslás lényegében azt jelenti, hogy a telefon előző időpillanat-béli $\bar{x}_{k-1|k-1}$ állapotát és $\bar{P}_{k-1|k-1}$ kovarianciamátrixát állandónak tekintjük, és az alapján becsüljük meg, mi



is volna a következő pillanatban az $\hat{x}_{k|k-1}$ állapotvektor, illetve a hozzá tartozó $\hat{P}_{k|k-1}$ kovarianciamátrix. Ehhez szükség van több nevezetes mátrixra is.

Nevezetes mátrixok

Az $\bar{\bar{F}}_k$ (néhol $\bar{\bar{A}}$ -val jelölt) állapot-átmenet mátrixra, amely két egymástól eltérő időbeli állapot közt határozza meg a komponensek egymástól való függését. Ez a mátrix jelen esetben állandó és a következőképpen írható fel:

$$\bar{\bar{F}}_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

ahol Δt a két mérési időpillanat közt eltelt idő másodpercben, a g a gravitációs gyorsulás.

Hasonlóan állandó a $\bar{\bar{G}}_k$ (máshol $\bar{\bar{B}}$ -vel jelölt) vezérlőjel-állapot kollineációs mátrix. Ez a mátrix a vezérlőjelnek tekintett mérések állapotvektorra gyakorolt befolyását határozza meg. Szerkezetileg ez esetben 9 oszlopból és 16 sorból áll:

$$\bar{\bar{G}}_k = \begin{bmatrix} 0[4,3] & \frac{\Delta t}{2} \cdot q[4,3] & 0[4,3] \\ \frac{\Delta t^2}{2} \cdot \bar{q}[3,3] & 0[3,3] & 0[3,3] \\ \Delta t \cdot \bar{q}[3,3] & 0[3,3] & 0[3,3] \\ 0[3,3] & \bar{q}[3,3] & 0[3,3] \\ 0[1,3] & 0[1,3] & 0[1,3] \end{bmatrix}$$



ahol $\bar{q}$ a kvaterniók vektora forgatómátrixba alakítva, a mátrix szorzásnak megfelelő komponense, q a kvaterniók megfelelő komponense a következő alakban:

$$q = \begin{bmatrix} \frac{q_1 \cdot \Delta t}{2} & \frac{q_2 \cdot \Delta t}{2} & \frac{q_3 \cdot \Delta t}{2} \\ -q_0 \cdot \Delta t & -q_3 \cdot \Delta t & q_2 \cdot \Delta t \\ \frac{q_3 \cdot \Delta t}{2} & \frac{-q_0 \cdot \Delta t}{2} & \frac{-q_1 \cdot \Delta t}{2} \\ -q_2 \cdot \Delta t & \frac{q_1 \cdot \Delta t}{2} & \frac{-q_0 \cdot \Delta t}{2} \end{bmatrix}$$

A harmadik mátrixnak, a $\bar{H}_k$ (vagy $\bar{C}$) állapot-mérés mátrix, amelyre a frissítés résznél lesz szükség, azonban az átláthatóság érdekében előrébb vettem ismertetését. Alakját dinamikusan változtathatjuk aszerint, hogy mennyi bejövő adat van az adott pillanatban, például én két lehetséges scenáriót különböztettem meg: amikor van- és amikor nincs GPS mérés (és vele együtt ál-sebességmérés). Minden esetben a ténylegesen összefüggésben és azonos koordináta-rendszerben értelmezendő mennyiségeket kell alkalmazni. A mátrix funkciója az, hogy az előzetes (jósolt) állapotvektorral összeszorozva megkaphatjuk a jósolt mérésünket. Tehát a feltételezésünket, hogy mit kellett volna mérni, hogy a számított jósolt állapotvektort kapjuk. Példa egy egyszerű lineáris esetre, ahol az állapotvektorunk a síkbeli helyzetünk és sebességünk, a mért adat pedig ugyanezen rendszerben mért helyadatunk:

$$\bar{H}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

A kiterjesztett Kálmán-szűrő

Ha teljesen lineáris lenne a rendszer, akkor a jósolt állapotvektort megszorozva H mátrixszal olyan mértékegységű elemeket adna vissza, amelyek összehasonlíthatóak a mért mennyiségekkel (ez a frissítés első lépése). Azonban a feladatunk a valóságban nem lineáris, hanem linearizáltuk.



Ezért a H mátrixot numerikus deriválással határozzuk meg minden időpillanatba iterálással, emiatt nem lineáris függvényt definiál. Ez a kiterjesztett Kálmán-szűrő alapelve.

Számítás

A legelső állapotot a kezdeti tájékozó kvaterniókkal és a legelső mért GPS koordinátákkal definiáljuk. A hozzá tartozó kovarianciamátrixot nullákkal feltöltött mátrixnak tekintjük. Az előzetes (jósolt) állapotvektorhoz szükséges: az előző állapot, az adott időpillanat-béli vezérlőjel (esetünkben a gyorsulás, a szögsebesség és a magnetométer mérései tehát $\bar{u}_k = [\bar{a}_g \quad \bar{\omega} \quad \bar{m}_g]$), a nevezetes mátrixok közül pedig az $\bar{F}_k$ és a $\bar{G}_k$.

A vezérlőjel (vagy gerjesztőjel) az a bemeneti jel vagy információ, amelyet a szűrőbe táplálunk, hogy az segítsen a becslési folyamatban. Ez a bemenet lehet olyan külső adat, amely hatással van az állapotra, vagy olyan vezérlőművelet, amelyet végrehajtunk az állapotra. A Kálmán-szűrő működésének nem feltétele a vezérlőjel és a hozzá tartozó $\bar{G}_k$ mátrix használata, azonban használatával pontosabbá teheti a becsléseit az állapotváltozókról, még akkor is, ha a szenzoradatok zajosak vagy hiányosak.

$$\hat{x}_{k|k-1} = \bar{F}_k \cdot \bar{x}_{k-1|k-1} + \bar{G}_k \cdot \bar{u}_k$$

Az előzetes állapothoz tartozó kovarianciamátrixhoz szükséges ismerni az előző állapot $\bar{P}_{k-1|k-1}$ kovarianciamátrixát, az $\bar{F}_k$ mátrixot, és a mérések bizonytalanságából megkapható $\bar{Q}_k$ kovariancia-, avagy zajmátrixot.

$$\bar{Q}_k = \bar{G}_k \cdot \bar{S}_g \cdot \bar{G}_k^T$$

ahol $\bar{S}_g$ a nyugalmi állapotban és a globális rendszerben lévő $\bar{u}_k$ vezérlőjelek szórását, (vagyis a mérések átlagtól való átlagos eltérését) tartalmazó mátrix. Például:

$$\sigma_x = \frac{\sum |\bar{a}_x|}{n}$$



ahol n a mérések száma. $\bar{\alpha}_x$ -hez hasonlóan a többi szórást is így kapjuk, melyek aztán $\bar{S}_g$ főátlójában kapnak helyet.

A $\bar{Q}_k$ ismeretében most már felírható a $\hat{P}_{k|k-1}$ mátrix:

$$\hat{P}_{k|k-1} = \bar{F}_k \cdot \bar{P}_{k-1|k-1} \cdot \bar{F}_k^T + \bar{Q}_k$$

4.3.3 A frissítés

A frissítés során megvizsgáljuk az adott időpillanatban számított méréseket, ezeket összevetjük a jóslt értékekkel, majd a $\bar{K}_k$ Kálmán-mátrix felírásával, megfelelő súlyozással megjavítjuk az állapotvektort és a kovarianciamátrixát. Ezt követően utóellenőrzést végzünk, majd átlépünk a következő időpillanatba.

A Kálmán-mátrix meghatározása

A korábban már említett $\bar{H}_k$ állapot-mérés mátrixot most felhasználva különbségvektort számítunk ki a mért és a jóslt (számított) koordináták között a következő módon:

$$\tilde{y}_k = z_k - \bar{H}_k \cdot \hat{x}_{k|k-1}$$

ahol z_k a méréseink oszlopvektorba rendezve.

Amennyiben a már említett kritikus pillanatokra fel akarjuk készíteni a rendszerünket, itt egy logikai vizsgálattal elágazást kell készítenünk. Vizsgáljuk meg, hogy az adott időbéllyeghez (k) tartozik-e GPS mérés, majd a válasz alapján eltérő z_k és $\bar{H}_k$ változót kell alkalmaznunk. Ezzel lehetővé tesszük, hogy olyan pillanatokban is végezhesünk jóslást és becslést, amikor nincs rendelkezésre álló GPS mérés:

Ha nincs GPS jel: $z_k = [\bar{m}_g]$

Ha van GPS jel: $z_k = [\overline{GPS} \quad \bar{v}_g \quad \bar{m}_g]$

A $\bar{K}_k$ Kálmán-mátrix felírásához egy $\bar{S}_k$ segédmátrix szükséges, amely a kovarianciák javítását, avagy módosítását határozza meg. Kiszámításához ismernünk kell az $\bar{R}_k$ mátrixot, ami a középhibák várható értékeit, vagyis



a mérések szórását tartalmazza. Esetünkben a sebesség, a magnetométer és a GPS pontosságát használtam a fő átlóban. Az alábbi képlet egy egyszerűsítés, a főátló annyi elemből kell álljon, ahány mért komponens van.

$$\bar{\bar{R}}_k = \begin{bmatrix} \mu_{GPS}^2 & 0 & 0 \\ 0 & \mu_v^2 & 0 \\ 0 & 0 & \mu_m^2 \end{bmatrix}$$

$$\bar{\bar{S}}_k = (\bar{H}_k \cdot \hat{P}_{k|k-1} \cdot \bar{H}_k^T) + \bar{\bar{R}}_k$$

Azt, hogy a mérések milyen súlyokkal befolyásolják az egyes állapotokat, a $\bar{\bar{K}}_k$ Kálmán-mátrix adja meg. Kiszámolásához szükséges a segédmátrix inverze, azonban ezt a program kiszámolja helyettünk az *inv()* parancs segítségével.

$$\bar{\bar{K}}_k = \hat{P}_{k|k-1} \cdot \bar{H}_k^T \cdot \bar{\bar{S}}_k^{-1}$$

Az új állapot

Az utolsó lépés az új állapot és kovarianciájának meghatározása volt. Ehhez a következő képlet nyújt segítséget:

$$\bar{x}_{k|k} = \hat{x}_{k|k-1} + \bar{\bar{K}}_k \cdot \tilde{y}_k$$

$$\bar{P}_{k|k} = (\bar{I} - \bar{\bar{K}}_k \cdot \bar{H}_k) \cdot \hat{P}_{k|k-1}$$

ahol $\bar{I}$ egy annyi oszlopból és sorból álló egységmátrix, ahány komponense van az állapotvektornak.

Az újonnan kapott állapotot visszaellenőrizhetjük, hogy mennyiben térünk el attól az állapottól, amit csak az adott pillanatra értelmezett GPS mérésekből számítottunk volna.

$$\tilde{y}_{k|k} = z_k - \bar{H}_k \cdot \hat{x}_{k|k}$$

Ezt az eltérést a további számításokban azonban nem vesszük figyelembe.



5. Eredmények

5.1 A valósághű mozgásábra

A következőkben szemléltetem, hogy a Kálmán-szűrő módszerével kapott helyadatok hogyan viszonyulnak a nyers GPS mérésekhez, és a ténylegesen bejárt úthoz.

Minden ábrázolást teljes mértékben a MATLAB segítségével készítettem. A program alkalmas nagy mennyiségű adathalmazok megjelenítésére, illetve képes kezelni a .tif formátumú ortofotókat. Így képes voltam felhasználni az iskola által biztosított felvételeket a következő szintaktikával.

Readgeoraster()

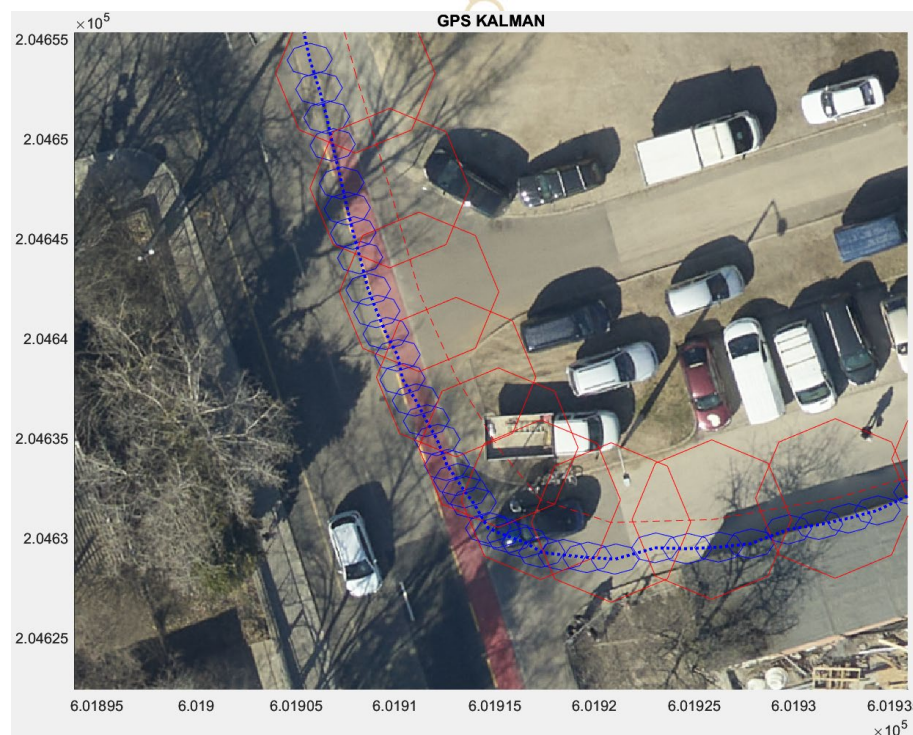
Mapshow()

Ez azért jelentős tulajdonság, mivel így koordinátahelyes nagyfelbontású alaptérképként használható fel. A nagy (5 centiméteres) felbontás lehetővé teszi számomra, a kép interpretálásával, hogy meghatározhassam a ténylegesen bejárt útvonalat. Ezzel ellenőrizhetővé válik Kálmán-szűrő javításának mértéke.

Természetesen más ellenőrzési módok is léteznek, például olyan mérés végzése, amely során szimultán rögzítem a telefon ismertett szenzorjainak méréseit, és e közben egy RTK vevőt is magammal szállítok. Ebben az esetben az említett műszer – ellentétben a telefon GNSS vevőjével - korrekciókat kap, amellyel a tényleges helyzetem nagy pontossággal ismertté válik, ezen információ birtokában hasznos

következtetéseket vonhatók le. Azért választottam mégis az ortofotós ellenőrzési módot, mivel a kutatás során ez állt rendelkezésemre.

A jövőben szeretném az RTK ellenőrzési módszert is implementálni, ugyanis a kettő ötvöztetésével meghatározható az ortofotó esetleges hibája is.



Ábra 17 A mért- és számított bejárt út az ortofotón

Ahogy azt az ábra is mutatja, külön színnel ábrázoltam a GPS által mért (piros) és a Kálmán-szűrő által számított (kék) pillanatnyi helyadatokat. A projekt egyik célja annak vizsgálata volt, hogy elérhető-e pontosabb (valóságnak jobban megfelelő) helymeghatározás. Ennek ellenőrzésére nyújt lehetőséget a valóságban bejárt úttól való eltérés vizsgálata. Mivel a tényleges útvonal koordinátáit nem ismerem (RTK mérés hiánya miatt), így az ortofotón jól elkülöníthető objektumokat kellett kihasználni. Erre tökéletesen megfelelt a bicikli- és forgalmi sáv elválasztására felfestett



sárga színű vonal. A mérés során tehát részben ezen haladtam, és érdemben ez a szakasz használható ellenőrzésre.

Az ábrán egy bonyolultnak mondható mozgás, vagyis egy kanyarodás látható. Ha csupán a GPS mérését jelenítenénk meg és vizsgálnánk, úgy tűnne, mintha nem az úton, hanem a járdaszigeten áthaladva közlekedtem volna. A valóságban azonban azt megkerülve az aszfaltburkolaton végeztem a mozgást, így tehát a kék szaggatott vonallal ábrázolt trajektória pontosabbnak bizonyult ezen a helyen.



Ábra 18 Egyenes szakasz vizsgálata

Hosszabb egyenes szakasz során, amilyen a Hosszúsétatér is, enyhe 'imbolygás' figyelhető meg, véleményem szerint ez a szenzorok zajának hiányos kiszűrésének, és a mérés során enyhén jobbra-balra dőlő biciklikormánynak tudható be. Ezeknek mértéke azonban figyelmen kívül hagyható, mivel az említett hiba így is minimalizálva van a szűrő által. A zaj mindig is jelen lesz, mivel ez a 'telefonos' szenzorok általános jellemzője.



Ábra 19 Egy helyben állás vizsgálata

Az előző két ábra folytatásaként látható a bejárt út vége, ahol vizsgálni tudtam, mit tesz a szűrő egy helyben állás esetén. Amíg a piros színnel jelölt GPS jól láthatóan ugyanazon koordinátákat adja, a Kálmán-szűrő eredményei egy 3-4 méteres területen folyamatosan 'vándorolnak'. Következtetésem szerint oka a gyorsulás hibás frissítése. Ennek alapvetően két forrása lehet:

Elsőként a gyorsulásmérőt befolyásoló hatások figyelmen kívül hagyása, mint például a szenzor bias pontatlan meghatározása, vagy a pontatlan tájékozás, ami részben a környezet tulajdonságából származik (tehát előre meghatározhatatlan, ezáltal kiszűrhetetlen), részben a feldolgozás során a telefontípusonként változó szenzorok megbízhatóságának súlyozásából (ami kalibrációs szándékú tesztmérések során csökkenthetővé válik).

A másik tényező a GPS és a belőle származtatott ál-sebességmérő súlyozásának hibája. Habár alapvetően korrekciók hiányában pontatlan, mégis arra a következtetésre jutottam, hogy a műholdas mérésekből a sebességre származtatott adatok csökkenthetik a jelenség által okozott



hibát. Ennek alátámasztására a jövőben további ellenőrző méréseket kell végezzenek.

Korábban már említettem, hogy a szűrő egyik jelentős előnye, hogy azon esetekben is képes helyadatokat szolgáltatni, amikor a műholdas helymeghatározás lehetősége korlátozott vagy teljesen megszűnik. Ezekben a kritikus helyzetekben a szűrő kizárólag a szenzorok mérésére és az előző állapotvektorra hagyatkozik.



Ábra 20 Kanyarodó mozgás trajektóriája GPS jelvesztés esetén

Az ábrán látható, korábban is bemutatott kanyarban modelleztem az egyik legelőnytelenebb esetet. Nem csupán 5 másodpercen át tartó műholdas jelvesztés, de egy kanyar is terheli a mérést. A számítási időköz fél másodpercre volt állítva, így egy igen részletes képet kapunk a bejárt útról. A valóságban a hibaellipsziseket éppen érintő szaggatott sárga vonalon haladtam, így látható, hogy úgy tűnik mintha 'kisodródnék' a kanyarban, ez a tájékozás és a giroszkóp pontatlanságának tudható be. Az eredményt a szenzorok pontosságához viszonyítva elfogadhatónak találtam.



5.2 A hibaellipszis

Az x állapotvektor P kovarianciamátrixát ismerve kifejezhető egy L mátrix, amellyel az adott pillanat Kálmán-szűrővel meghatározott koordinátákra illesztett kör az adott számítás hibaellipszisévé torzítható.

Statisztikailag 66,3%-os megbízhatósággal állíthatjuk, hogy a valódi helyzetünk ezen hibaellipszisen belül helyezkedett el. A paraméterek, például a fiktív (ál-) és tényleges mérések szórásainak módosításával hatással lehetünk az ellipszis területére és irányára. Minnél kisebb az ellipszis, annál biztosabbak vagyunk a tényleges helyzetünkben.

A 3.2.1 fejezetben bemutatott ábrán, ahol hosszabb egyenes szakaszon kieső GPS jelet szimuláltam jól látható a hibaellipszisek területének idő múlásával arányos növekedése. Amennyiben huzamosabb ideig nem szerezne a szűrő GPS jelet, a hibaellipszisek területe tovább nőne, ez a hibaterjedés következménye.

A korábbi ábrákon a számítási kapacitás optimalizálása érdekében a hibaellipsziseket generalizáltam nyolcszögekre, ezzel az eszmei értékükön nem csökkentve. Az eredeti GPS mérések köré is felszerkesztettem a 4.9 méter átmérőjű köröket, melyek általánosan megfelelő mérési körülmények között biztosan magukban foglalják a tényleges pozíciónkat.



6. Diszkusszió

6.1 Mintavételezés

A kapott eredmények vizsgálatánál látszik, hogy a mintavételezés gyakorisága befolyásolja a kimenetet.

A korábban már említett mintavételezések között eltelt idő kedvünk szerint dinamikusan változtatható. Ezzel finomabb mozgások (például egy hirtelen irányváltás) is modellezhetővé válnak, amelyeket addig a GPS korlátai nem engedtek meg, de a szenzorok igen jól felhasználhatóvá váltak ilyen feladatokhoz.

Fontos megemlíteni, hogy amennyiben csökkentjük a számítási időközöt, azzal arányosan csökken a szenzorokból származó apró zajok kiszűrésének mértéke. Nagyobb időköz esetén több mérést használhatunk fel az előfeldolgozás során, ezzel növelve a zajjal terhelt mérés javíthatóságát, azonban ez a kis mértékű tényleges változások 'kiátlagolásához' is vezet. A feladattól függően tehát a felhasználó feladatává válik a prioritás megválasztása.



Ábra 21 Mintavételezési időköz előnyei

6.2 Hasonló munkák

Napjainkban számos megoldás születik a helymeghatározás egyre pontosabb és hatékonyabb megoldására. [9] Ezen megoldások azonban gyakran valamilyen ideális rendszerben, környezetben képesek csak



működni, ezért tartom kiemelkedőnek az alkalmazkodni képes mobiltelefonba épített szenzorok „rendszerét”, amivel én is dolgoztam.

A Kálmán-szűrő, habár egy rendkívül hasznos módszer a helymeghatározás pontosságának javítására, amennyiben nem veszünk figyelembe minden mérést és számítást terhelő hibát, addig csekély javulást vehetünk észre az egyszerű egy-szenzoros (GPS) mérésekhez képest.

Személyes véleményem szerint a dolgozatom kutatási témáját nem tekintem lezártnak. Ahogy azt már korábban említettem, a Kálmán-szűrő tovább bővíthető határok nélkül, ezzel pontosabb, és akár más célra is felhasználható eredményeket elérve. Én személy szerint izgalmasnak tartom ezt a tudományterületet és remélem, tudok a jövőben is ezzel foglalkozni.

6.3 További kutatási célok

Következő lépésként fontos lenne a jelenlegi rendszer minél sokrétűbb, változatosabb mérési körülményeknek ’kitenni’, ezekből újabb következtetéseket levonni, melyek új megoldásokhoz vezethetnek. Ilyen mérés lenne például szintkülönbség jelentkezésével járó mozgás vizsgálata. Továbbá olyan mérés, amely során a magnetométer mérése nagymértékű, környezetből adódó torzulásnak van kitéve.

6.3.1 GPS bias

A GPS-ből származó méréseket nem teljesen nyersen kapjuk meg, azok előfeldolgozott, szűrőkön átvitt koordináták. A műholdas helymeghatározás által kapott adatokra általánosan jellemző egyfajta ’kisodródás’, amely azt jelenti, hogy a kanyarokban nem rögtön követi a bejárt utat, és a gyors irányváltozásokat is csak idővel (pár másodpercet követően) követi le. Ezt GPS biasnak hívjuk. Ennek figyelembevételére léteznek megoldások, azonban ismereteimet a témakörben még mélyítenem kell. [10] Megoldáshoz vezethet olyan telefonos alkalmazás használata, amely a ténylegesen nyers műhold adatokat rögzíti.



6.3.2 Új szenzor implementálása

A GNSS vevő a magasságot csak nagy hibával tudja meghatározni a vevő és a műholdak geometriájából adódóan. Erre nyújthat megoldást a légnyomásmérő, vagyis a barométer.

Ezzel a szenzorral a tengerszint feletti abszolút magasságot lehet meghatározni, amennyiben ismerjük az adott magasságokhoz tartozó átlagos légnyomást. A jelenlegi próbaterületen végzett mérések főként a vízszintes koordináták javításának vizsgálatára lettek kihegyezve, mivel ezek ellenőrzésére és mérésére volt lehetőségem. Egy barométerrel rendelkező telefon méréseinek feldolgozása és annak szimultán RTK méréssel történő ellenőrzése érdekes következtetésekhez vezethet. Jelenleg azonban ezek a feltételek nem adottak számomra.

6.3.3 Felmerülő akadályok mobilapplikáció tervezés során.

Az említett mobilalkalmazás, habár kezdeti fázisban van még, de bizakodó vagyok fejlődésének mértékével kapcsolatban. Első lépésként az elméleti háttérrel kell megteremteni, amelyből a legfontosabb tényező a valós idejű feldolgozás módszerének megtervezése.

Felmerülő kérdés a Kálmán-szűrő használatához szükséges előfeldolgozási lépések helyettesítésének, vagy módosításának megoldása. A programot mindenképp fel kell készíteni a mérések megbízhatóságának helyes meghatározására, ugyanis feltételezésem szerint az adathalmaz, amit felhasznál, pontatlanabb tartalommal fog bírni.

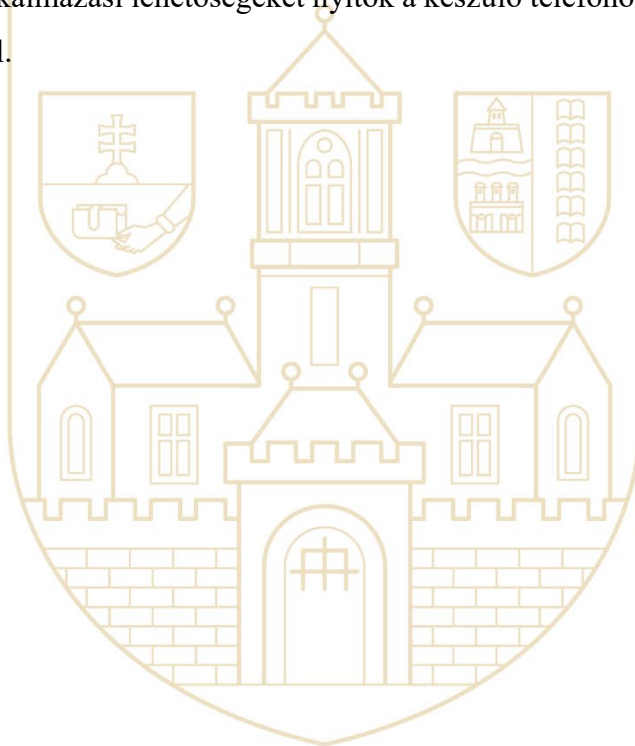
További következtetésekhez vezethet egy esetleges kombinált módszer vizsgálata. A valós idejű, az utófeldolgozáson alapuló, illetve a kettőt közösen alkalmazó metódus összehasonlítása különböző mérési helyzetekben, alkalmas lesz később a számítás pontosságának jellemzésére, ezzel a felhasználókat segítve a tudatos módszerválasztásban.

Emellett el kell mélyítenem tudásomat a telefonos alkalmazások fejlesztésében. Amennyiben rendelkezni fogok a tudással, szeretném azt



felhasználni és a földmérésben hasznosítani különböző célú applikációk létrehozásával. Egy ilyen ötlet például az igen költséges műszereket és azoknak üzemeltetését kiváltó erdészeti és mezőgazdasági kitűző program. Természetesen olyan felhasználási módoknak tervezek, amelyek a jogszabályok betartását nem akadályozzák, így biztosítva a megbízhatóságot és a szaktudás meglétének igényét. Egy másik ötlet a korábban említett futárok navigációját segítő alkalmazás helymeghatározásának új módszertani alapja. Ez azonban sokkal nagyobb szabású és együttműködést igénylő projektötlet.

Minden nap több órát használjuk okostelefonjainkat és még több területen képesek lennénk alkalmazni ezt a kisméretű számítógépet. Remélem, munkám teret nyit és segíti a témakörben érdekelteket, tevékenykedőket; illetve új alkalmazási lehetőségeket nyitok a készülő telefonos alkalmazás létrejöttével.





7. Összefoglalás

Kutatásom céljával egy alkalmazás létrehozását tűztem ki, amely képes felhasználni a mobiltelefonokba épített szenzorokat, és azok méréseit felhasználni a helymeghatározáshoz használt GPS szolgáltatás pontosítására. Kijelenthetem, hogy ez a cél megvalósult, azonban nem olyan módon, ahogyan azt először elképzeltem.

A jelenleg elkészült, utófeldolgozáson alapuló módszerrel meg vagyok elégedve, hiszen olyan biztos alapot ad a kutatásom folytatásához, amivel hatékonyan és részletekig hatolóan tudok foglalkozni a különböző felmerülő kérdésekkel.

Az eredményekből levont következtetések is biztatók számomra, és igazolják a módszer hasznosságát. A kezdeti célok mellett rájöttem, hogy nem csak a műholdas helymeghatározás pontosítására, de a jelvesztés problémájára is megoldást nyújthatok. Az összegyűjtött tapasztalatok alapján érdemes több célt és mérföldkövet kitűzni a projektnek, hiszen szinte biztosan lesznek olyan akadályok, amelyek lassítják vagy meggátolják a fejlesztést. Ezekben az esetekben érdemes élni a lehetőségek tárházával.

Az elméleti háttér mellett fontosnak tartom, hogy gyakorlati oldalról is megteremthessem a szükséges és azon túli kapacitást, ezzel a projekt minőségének növelésére törekedve. Ilyen például a már említett RTK vevő vagy további légifelvételek beszerzése a szabatos ellenőrzéshez. Kitűzött cél egy vagy később akár több telefonon végzett mérések összehasonlítása is. A hardware oldalon túl, a software részről pedig fejlesztői környezet szempontjából kényelmesebb, de költségesebb alkalmazások használata.



8. Summary

My thesis explores the enhancement of positioning data accuracy in smartphones using sensor fusion and the application of the Kalman filter. By utilizing various sensors such as the accelerometer, gyroscope and magnetometer along with the GNSS receiver, I aimed to improve the calculated location and trajectory modeling.

The field measurements conducted in Székesfehérvár provided valuable insights into the performance of the developed method. Utilizing MATLAB for data processing, I addressed challenges like varying sensor frequencies and data quality differences. The Kalman filter, known for its versatility in aerospace and navigation systems, was used to integrate sensor data and enhance accuracy.

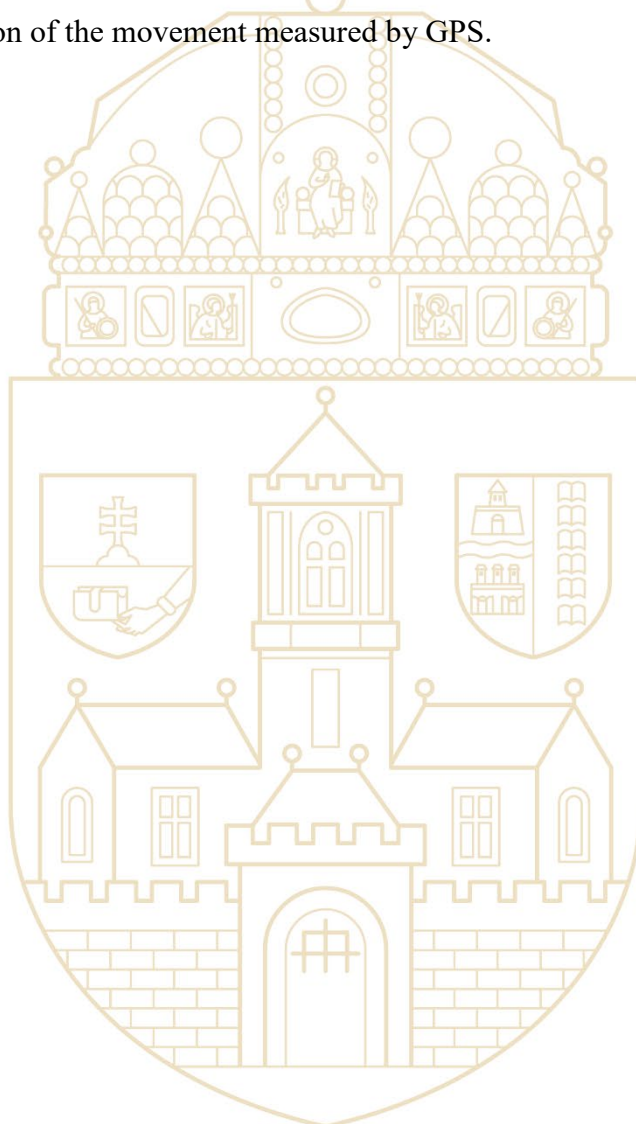
The Kalman filter operates in two phases: prediction and update, where the state vector and covariance matrix are estimated and corrected based on measured and predicted states. I considered 16 parameters, including orientation quaternion, position, acceleration, and gravitational vector, to describe the phone's state comprehensively.

The outcomes were visualized using MATLAB, showcasing the effectiveness of the Kalman filter in improving positioning accuracy. Figures demonstrated the comparison between GPS coordinates and those calculated using the filter, highlighting its ability to model movement accurately.

Nowadays there are many solutions for more and more accurate and efficient positioning. However, these solutions are often only able to operate in some ideal system or environment, and in other cases self-adaptable instruments, like geodesy GNSS receivers are very expensive, which is why I consider the "system" of sensors built into the adaptable mobile phone to be outstanding.



Currently, the method works with post-processing, which means that the results are not obtained in real time. The possible direction of continuation is the implementation of the program on the phone, where the measurements coming from the sensors are processed by an application in real time. This would make it possible to have frequent location data between GPS measurements, which are more accurate than simple extrapolation of the movement measured by GPS.





9. Köszönetnyilvánítás

Magát a kutatás témáját még első éves hallgatóként ismertem meg. A kapcsolódó tudományágak ismeretét egyetemi éveim alatt szereztem meg, és így voltam képes bonyolítani a modellt. Azzal, hogy az egyetemen tanult elméleti tantárgyakat gyakorlatba tudtam helyezni, hozzásegítettek a könnyebb megértéshez. A projektemmel hallgatótársaimat is segíteni tudtam témaválasztásban, vagy már meglévő ötletek kidolgozásában. Munkámat számos tudományos konferencián publikálhattam, az ott elhangzó kérdéseket és tanácsokat pedig hasznosítottam. Köszönöm konzulensemnek, Dr. Molnár Gábor Péternek, hogy végig segített ezen az úton.

Dolgozatom elkészítéséhez sok támogatást kaptam családtagjaimtól, közeli ismerőseimtől, egyetemi szaktársaimtól és tanáraitól. Kiemelt köszönet illeti: Lengyel Ágoston, Pusztai Henrik Antal, Rehák Dániel.

A Kálmán-szűrőről és a szenzor fúzióról legtöbbet a MathWorks weboldalán, és az interneten megtalálható ingyenes videósorozatukból tanultam, ahol ezekben a tágabb témákban is elmélyedhettem. [11] [12]



10. Idézett forrásmunkák

- [1] T. Simon, Magyarország és a világűr - Szerepünk a világ űrtevékenységében, MCC Press Kft., 2021.
- [2] „ELTE Informatikai Kar, Térképtudományi és Geoinformatikai Intézet honlapja,” [Online]. Available: <http://lazarus.elte.hu/~climbela/gps3.htm>. [Hozzáférés dátuma: 9 október 2022].
- [3] F. v. D.-P. Enge, "The World's first GPS MOOC and Worldwide Laboratory using Smartphones," in *28th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2015)*, Tampa Convention Center, Florida, 2015.
- [4] G. Paresh, "Techulator," 25 március 2020. [Online]. Available: <https://www.techulator.com/resources/8930-How-does-smart-phone-accelerometer-work.aspx>. [Accessed 20 július 2022].
- [5] G. Molnár és G. Timár, „Az EOv-koordináták nagypontosságú közelítése Hotine-féle ferdetengelyű Mercator-vetülettel,” ELTE Geofizikai Tanszék, Űrkutató Csoport, 2002.
- [6] B. K. P. Horn, "Closed-form solution of absolute orientation using unit," Department of Electrical Engineering, University of Hawaii at Manoa, Honolulu, Hawaii, 1986.
- [7] S. Soheil és T. Federico, „Accurate Computation of Quaternions from Rotation Matrices,” Institut de Rob`otica i Inform`atica Industrial, Barcelona, Spain, 2016.
- [8] A. Becker, "Kalman Filter," 2022. [Online]. Available: <https://www.kalmanfilter.net/kalmanmulti.html>.



- [9] M. Landy-Gyebnár, „National Geographic - Tudomány - Helyzetmeghatározó, műhold nélkül,” 26 November 2022. [Online]. Available:
<https://ng.24.hu/tudomany/2022/11/26/helyzetmeghatarozo-muhold-nelkul/>.
- [10] R. Zanetti és B. Robert H., „Kalman Filters with Uncompensated Biases”.
- [11] MATLAB®&Simulink®, „MathWorks - Understanding Kalman Filters,” Simulink®, [Online]. Available:
https://www.mathworks.com/videos/series/understanding-kalman-filters.html?s_eid=PSM_15028.
- [12] MathWorks®, „MathWorks - Estimating Orientation Using Inertial Sensor Fusion and MPU-9250,” [Online]. Available:
https://www.mathworks.com/help/fusion/ug/Estimating-Orientation-Using-Inertial-Sensor-Fusion-and-MPU-9250.html?s_eid=PSM_15028.
- [13] „Számítástechnikai és Automatizálási Kutatóintézet (SZTAKI),” 3 május 2018. [Online]. Available:
<https://www.sztaki.hu/tudomany/hirek/kovetkezo-felevtol-elinditjuk-rudolf-e-kalman-distinguished-lecturer-program-ot>.
- [14] Team, Earth Science Communications, "NASA/Global Climate Change," 23 augusztus 2021. [Online]. Available:
https://climate.nasa.gov/internal_resources/2414/. [Accessed 10 október 2022].



11. Ábrajegyzék

Ábra 1 Helymeghatározást igénylő feladatok csoportosítása	7
Ábra 2 Az utófeldolgozás 4 lépése.....	9
Ábra 3 Rögzített adatállomány.....	10
Ábra 4 Képernyőkép a készülő alkalmazásból	11
Ábra 5 Saját fejlesztésű programkód részlet az MS Visual Studio-ban..	12
Ábra 6 A műholdas helymeghatározás.....	13
Ábra 7 MEMS szenzor, gyorsulásmérő a telefonban.....	14
Ábra 8 A Föld mágneses terének erősségét jellemző mennyiség nagysága a Föld felszínén	16
Ábra 9 A MATLAB felhasználói környezete	18
Ábra 10 GPS jelvesztés szimulálása egyenes szakaszon	19
Ábra 11 A bias-t tartalmazó számított pillanatnyi sebesség	20
Ábra 12 Időanomáliakiegyenlítés.....	24
Ábra 13 Tájékozási kvaterniók ellenőrzése	25
Ábra 14 Az átváltásért felelős wgs2eov.bat nevű kód	26
Ábra 15 Tájékozási forgatómátrix.....	29
Ábra 16 Kálmán Rudolf Emil (1930-2016)	31
Ábra 17 A mért- és számított bejárt út az ortofotón.....	39
Ábra 18 Egyenes szakasz vizsgálata	40
Ábra 19 Egy helyben állás vizsgálata	41
Ábra 20 Kanyarodó mozgás trajektóriája GPS jelvesztés esetén.....	42
Ábra 21 Mintavételezési időköz előnyei.....	44