



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem**

Villamosmérnöki és Informatikai Kar

Hálózati Rendszerek és Szolgáltatások Tanszék

Szeile Aliz

**TÁVOLI ÉGITESTEKEN  
ALKALMAZOTT  
SZENZORHÁLÓZATOK  
ENERGIAHATÉKONYSÁGÁNAK  
VIZSGÁLATA**

KONZULENSEK

Dr. Bacsárdi László

Dr. Huszák Árpád

BUDAPEST, 2014

## Összefoglaló

Napjaink tudományos életében igen jelentős szerepet tölt be az űrkutatás és az űrkutatásban használt műszerek, mérőeszközök fejlesztése. Erre nagyon jó példa a pár éve egyetemünk által létrehozott és felbocsátott Masat 1 műhold, vagy az egyre újabb és újabb világméretű fejlesztések. Azonban ahhoz, hogy egy drágább űreszközt, vagy későbbiekben esetlegesen emberi megfigyelést engedhessenek egy távoli égitest felszínére, az adott területet alaposan fel kell térképezni a biztonságos és hatékony eljárás érdekében. Erre a feltérképezésre több megoldás is született a korábbi évtizedekben. A munkámban én egy újszerű és költséghatékony megoldással foglalkozom, aminek hosszabb ideje szentelek nagyobb figyelmet. Ezért ebben a dolgozatban a legújabb nézőpontot és annak eredményeit részletezem. Egy távoli égitest előzetes felszíni feltérképezése során ismeretlen terepviszonyokkal kell dolgoznunk, így nem érdemes drága és bonyolult műszereket használnunk. Ezért a témám egy mozgó szenzorhálózat rendszerén alapul. A szenzorok olyan egyszerű és olcsó érzékelő eszközök, amelyek alkalmasak lehetnek akár talajvizsgálatra, akár hőmérséklet-mérésre, víz keresésére, illetve légköri viszonyok felmérésére is, és kedvező áruk miatt nagy számban alkalmazhatók. Mivel feltételezésem szerint a szenzorok képesek a helyváltoztatásra, fontos szempont a szenzorok helymeghatározó képessége, hogy tudjuk, honnan is származnak pontosan az adatok. Ez bonyolítja ugyan a vizsgálatokat, ellenben nagyobb feltérképezési területet biztosít. Eddigi munkáim során különböző pozicionáló algoritmusokat dolgoztam ki és szimuláltam le, amelyek bizonyos hatékonyságot eredményeztek. Ugyanakkor hiába hatékony egy algoritmus pozicionáló képessége, ha a gyakorlatban irreálisan nagy energiafogyasztással jár. A mostani munkámban ezért az energiahatékonyság vizsgálata állt a középpontban. Az volt a célom, hogy egy valós képet kapjak arról, mennyire alkalmazhatók a kifejlesztett algoritmusok a gyakorlatban. Mindezek mellett egy távoli égitest felszínén több olyan légköri jelenség is megfigyelhető, amelyek befolyásoló hatással vannak a szenzorhálózat működésére. Ilyen például a porvihar, amely hatására a szenzorok közötti kommunikáció lehetetlenné válik, vagy a dűnék és kráterek, amelyek a szenzorok mozgását módosítják. Ezeket a körülményeket figyelembe vettem a szimuláció során, így a kapott eredményeimet is ezek a tényezők befolyásolták. Vizsgálataimhoz eszközként egy C++ szimulációs programot fejlesztettem ki, amelybe beleépítettem egy, az energiafogyasztásnak megfelelő mintát. Ennek segítségével az algoritmusok hatékonysági eredményeit össze tudom vetni a szenzorhálózat energiafogyasztásával.

## **Abstract**

Nowadays, space exploration plays an important role to understand the behavior of our planet and universe. Engineers develop different scientific equipment and devices which are used in the space research and are used for space exploration. Most of these devices are expensive one, but there are many cheap devices as well. One of the good examples is the Masat-1 cubesat which was developed at our university. There are several solutions to discover and continuously investigate a selected area on a surface of a Solar System body, which are based on expensive devices and sometimes on human monitoring. In my work, I present an innovative and cost-effective solution. To discover the unknown terrain conditions, we should not use expensive and complex devices. Instead of this, I decided to use a mobile sensor network. Sensors are simple and cheap devices, which can be capable to soil investigation, temperature measurement, water search or assessment of atmospheric conditions. The sensors can be applied in large numbers. The sensors are able to move, so positioning capability of sensors is an important aspect as well. I have developed and simulated different positioning algorithms during my earlier works. However, the positioning ability of an algorithm is not effective, if it includes unrealistic large energy consumption. This is why the examination of energy efficiency has been in focus in my current work. Different atmospheric phenomenon are observable in surface of a distant planet, which have influencing affect to behavior of sensor network. For example, a dust storm makes impossible the sensor communications, or a dune modifies the movement of a sensor. These different circumstances were taken into account in my work. I developed a C++ simulation program as a tool and placed my model of energy consumption into this program.

# Tartalomjegyzék

1. Bevezetés.....	5
2. Úrkutatás, szenzorhálózatok és energiahatékonyság .....	7
2.1 Úrkutatás .....	7
2.2 Szenzorok, szenzorhálózatok általános bemutatása.....	9
2.2.1 A szenzorok, szenzorhálózatok csoportosítása .....	9
3. A vizsgált szenzorhálózat modellje.....	14
3.1 Rendszerterv.....	14
3.2 A feltételezett szenzorhálózat alapvető felépítése .....	15
3.2.1 Nagyobb teljesítményű szenzorok a hálózatban.....	15
3.2.2 Szenzorok és szuperszenzorok mozgása .....	15
3.2.3 Jelterjedési modellek.....	16
3.2.4 Szenzorok pozicionálása .....	19
3.2.5 Szenzorok állapotdiagramja .....	24
3.2.6 Szenzorok energiafogyasztása.....	26
3.2.7 Szenzorok energia bevétele .....	27
3.3 Szimulációs program .....	29
3.3.1 Térképek alkalmazása .....	29
3.3.2 Paraméterek kezelése.....	30
4. Szimulációs eredmények.....	32
4.1 Akkumulátor kapacitás és az energiaszint kapcsolata .....	32
4.2 Porvihar, mint befolyásoló tényező .....	38
4.3 Konklúzió .....	41
5. Összefoglalás .....	42
Táblázat és ábrajegyzék.....	43
Irodalomjegyzék .....	45

## 1. Bevezetés

Napjainkra jellemző, nagymértékű műszaki fejlődés nem csak a mindennapi életben jelenik meg, hanem számos tudományos területen is. Egyik ilyen nagy fontosságú terület az űrkutatás, ahol az egyre korszerűbb és fejlettebb műszereknek köszönhetően már igen részletes képet kapunk a vizsgált égitestekről. Azonban az újabb és drágább mérőeszközök biztonságos és hatékony használatához előbb bizonyos mértékben fel kell térképezni a vizsgálni kívánt területet. Az égitestek előzetes feltérképezésnek egyik újszerű módszerével, a szenzorhálózatok alkalmazásával foglalkozom dolgozatomban. A szenzorok kisméretű érzékelő eszközök, amelyek számos különböző területen is alkalmazhatóak, mint például orvostudomány, biztonsági rendszerek vagy akár parkolók nagyobb kihasználtsága. Mivel olcsó eszközökről van szó, így nagy számban lehet őket alkalmazni, ezzel növelve a hatékonyságot. Bővebben az űrkutatás fejlődésével és a szenzorhálózatokkal a 2. fejezetben foglalkozom.

Egy távoli bolygó felszínére kihelyezett mozgó szenzorhálózatot különböző mérések és feltérképezés céljára használhatjuk. Dolgozatomban egy marsi környezetben alkalmazható szenzorhálózatot vizsgáltam, beleértve olyan felszíni tényezőket (homokvihar, kráterek, dűnék), amelyek befolyásoló hatással vannak a hálózat eszközeinek helymeghatározására és energiagazdálkodására. A szenzorok feltételezeten képesek különböző mérésekre (sugárzásmérés, atmoszféra mérése és mágneses erőter mérése), képi felvételek készítésére, ezek mellett pedig van köztük pár nagyobb teljesítményű szenzor, amelyek képesek felvenni a kapcsolatot egy földi irányító központtal és a mérőszenzoroktól előzőleg összegyűjtött adatokat továbbítani. Mivel a szenzorok képesek a helyváltoztatásra, így nagyobb terület tudnak megvizsgálni, ugyanakkor nyomon kell őket követni, hogy pontosan tudjuk, melyik mért adat honnan származik. Ezt a problémát oldja meg a szenzorok pozicionáló képessége, amelyre több helymeghatározó algoritmust is javasoltam. Korábbi munkáimban a helymeghatározással és annak két különböző módon optimalizált változatával foglalkoztam [1][2][3][4], melyek a mostani energiagazdálkodási vizsgálataimban is alapul szolgálnak. A szenzorok pozicionálásához kapcsolódó munkáim rövid összefoglalását, a feltételezett szenzorhálózatot, annak működésének részletes modelljét a 3. fejezetben ismertetem.

Mostani témámban a szenzorhálózat energiahatékonysága áll a középpontban. Szimulációs eszközként egy C++ programot fejlesztettem, amely modellezi a szenzorhálózat működését, a marsi környezetet és tartalmaz egy, az energiahatékonyságnak megfelelő mintát. A

szimulációs programot különböző bemenetekre futtattam, a kapott eredményeket pedig táblázatokban, illetve grafikonokon ábrázoltam és elemeztem. Ezek az eredmények részletes képet adnak arról, hogy a különböző bemenetekre, milyen energiahatékonysággal bír a feltételezett szenzorhálózat. A kapott eredményeket, ezek leírását és konklúziókat a 4. fejezetben írom le.

A dolgozat célja az volt, hogy egy komplex szimulációs eszköz segítségével valós képet kapjunk a szenzorhálózat esetleges alkalmazásának várható energiafogyasztásáról, energiahatékonyságáról, mely képes figyelembe venni a speciális környezeti feltételeket is.

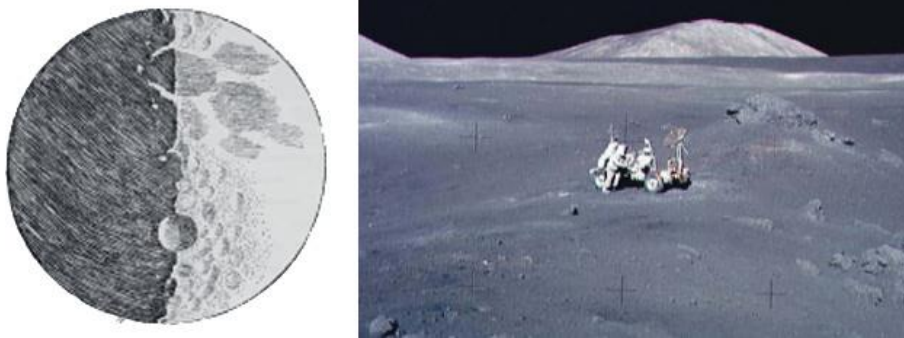
## 2. Űrkutatás, szenzorhálózatok és energiahatékonyság

Ebben a fejezetben röviden bemutatom az űrkutatás fejlődésének nagyobb lépéseit, szenzorhálózatokhoz alkalmazhatóságát a témában, illetve a dolgozat gerincét alakító energiahatékonyság rövid elméleti hátterét.

### 2.1 Űrkutatás

Galileo Galilei kutatásai óta a Naprendszerrel alkotott képünk egyre részletgazdagabb és terjedelmesebb lett. Ma már nem csak távcsövekkel, hanem űrszondákkal és műholdakkal is vizsgálják a világűrt. Az egyre korszerűbb eszközök megjelenésének következtében, főleg az utóbbi 50 évben, ismereteink egyre gazdagabbak lettek, ugyanakkor egyre több a felvetődő kérdés is.

A Nature című folyóirat 2010. július 29-i számában közölt egyik cikke remekül rámutat a Galilei kora óta történt jelentősebb fejlődésekre [5]. A 2.1.a ábra jól mutatja a megfigyelő eszközeink fejlődését. A bal oldali képen a kezdetleges megfigyelési eredményeket láthatjuk, míg a jobb oldali képen a korunkbeli megfigyelések eredményeit. A különbség szembetűnő, nemcsak részletességében fejlődtek, hanem sokkal pontosabb és hitelesebb képeket kaptunk. Míg Galilei korában még csak vázlatrajzok és becslések alapján dolgoztak a tudósok, ma már pontos mérések és részlet gazdag fényképek segítik a munkájukat.



2.1.a ábra

A Holdról alkotott ismereteink régen és napjainkban

A Hold Galilei távcsövén át (balra) és az Apollo program keretében a helyszíni vizsgálatok során (jobbra)

A fejlődés következtében mára már jóval több ismeretünk van az égitestekről, és a rajtuk uralkodó körülményekről. Erre nagyszerű szemléltetés az amerikai űrkutatási hivatal, a NASA honlapjáról származó fotó a 2.1.b. ábrán, amely 2012. szeptember 28-án készült a Mars felszínéről [6]. Jól látható a részletes megjelenítés, ami jellemzi a napjainkban bekövetkező megfigyeléseket.



**2.1.b ábra**

**A Mars felszíne egy 2012. szeptember 28-án készült felvételen**

Nem csak az űreszközök fejlődtek, de azok alkalmazása is sokat változott az évek során. Kezdetben még csak a Föld körül keringő műholdakat használták. Ezután következett a távolabbi égitestek elérése, például a Hold, Vénusz, Jupiter, később a kisbolygók is. Kezdetben még az égitestek melletti elrepülés volt a cél, így rövid ideig ugyan, de lehetett felvételeket készíteni. Az első ilyen elért célpont a Hold volt, 1959. január 2-án. Ezeket követték a becsapódó szondák, amelyek a becsapódás előtti időt használták fel fényképezésre. Később a leszálló, illetve pályára álló szondák következtek. Ezek után a fő cél az volt, hogy felderítsék a terepet az űrhajósok leszállásához. Részletes vizsgálatokat folytattak, ahol lehetett, ott talajmintákat vizsgáltak, vizet kerestek, hőmérsékletet mértek, légköri jelenségeket figyeltek meg. Ezáltal tudták meghatározni, hogy egy nehezebb szonda például képes-e leszállni az adott égiest felszínére. Később, 2010 nyarán hírül adták, hogy felfedeztek egy, a mi Naprendszerünkhöz hasonló bolygórendszert [7]. Mindez azt mutatja, hogy tanulmányozhatunk komplex bolygórendszereket is. Ezek alapján a felszíni vizsgálatok egyre fontosabbá válnak. Előre tekintve ezek a felfedezések maguk után vonják az űreszközök további fejlődését, új módszerek kialakítását a vizsgálatokra, megfigyelésekre. A fejlett űrtevékenységnek köszönhetőek a modern kommunikációs és navigációs rendszerek, emellett



egy új tudományág is született, a planetológia, mely középpontjában bolygókutatás felderítő és felfedező mivolta áll. Ennek eredményeképpen a fizika, kémia és földtudományok ismeretei is gazdagodtak. Nem csak a tőlünk több fényév távolságra lévő bolygókról tudunk meg többet, hanem a saját Földünkről is. A bolygónk alakulásának okait más megvilágításban is láthatjuk.

Számos magyar vonatkozású űrtevékenységhez kötődő eszköz is ismert. Az űrkorszak kezdete óta eltelt évtizedekben hozzájárultunk az űrkutatás tudományához, akár műholdátviteli kísérletekről, akár fedélzeti eszközökről, akár olyan sugázmérő készülékről is van szó, mint a Pille. Napjainkban, hazánkban egyre nagyobb teret kap az űreszközök fejlesztése is, erre nagyszerű példa a Budapesti Műszaki és Gazdaságtudományi Egyetemen épített és felbocsájtott Masat-1 műhold is. A szenzorhálózatok megfelelő alkalmazásával előzetes feltérképezi lehetőséget nyújtanak, így segítve a további űrtevékenységeket.

## **2.2 Szenzorok, szenzorhálózatok általános bemutatása**

A szenzorhálózatok alkalmazása széles körben elterjedt, használják őket akár az egészségügy, katasztrófavédelem, tudományos megfigyelések terén is.

A hálózat építőkövei a szenzorok, amelyek fizikai jelenségek mérésére képesek. (pl. hőmérséklet, fény, páratartalom) és a mért adatokat képesek továbbítani egy speciális építőelem, a nyelő felé. A szenzorhálózatok esetében általában nagy darabszámú, olcsó eszközökre gondolunk. Emiatt általában egyszerű eszközökről beszélünk, amelyeknél az erőforrás utánpótlás nehezen kivitelezhető feladat. Például, ha a szenzorhálózat egy távoli égitest felszínén helyezkedik el, akkor az egyes elemek a költséghatékonyság miatt nem rendelkezhetnek nagy napelemmel, illetve nagy generátorral/akkumulátorral. Ezért fontos, hogy a lehető legkevesebb energiafelhasználással használjuk a szenzorok funkcióit. Ezen hatékonyan lehet segíteni, ha a szenzor csak az alapvető funkciókkal rendelkezik, amelyek kevés energiát emésztenek fel (mérések, kommunikáció, stb.). A hálózat egy kiemelt eleme a nyelő, amely a többi szenzorhoz képest nagy energiatartalékok tárolására képes, így biztosítva a mérési adatok továbbküldésének a lehetőségét (pl. képes kommunikálni a műhoddal, ami el tudja juttatni a mért adatot a Földre).

### **2.2.1 A szenzorok, szenzorhálózatok csoportosítása**

A szenzorok egyik fő csoportosítása az alapján történik, hogy mi váltja ki az adatküldést. Ez alapján három nagyobb csoportba sorolhatjuk a szenzorokat [8].

- eseményvezérelt szenzor,
- lekérdezés vezérelt szenzor,
- idővezérelt szenzor.

Az eseményvezérelt szenzor esetében egy adott esemény a katalizátora az adatküldésnek. Ha ez az esemény bekövetkezik, akkor a szenzor elindítja az adatok átküldését. Ilyen lehet például a légköri jelenségek vizsgálatánál egy vihar bekövetkezése. A lekérdezés vezérelt szenzorokból álló hálózatnál ezzel szemben a nyelő intéz egy kérést az adott szenzor felé. Ennek a kérésnek a hatására kezd el adatot küldeni. Ebben az esetben a nyelő dönti el, hogy mikor van szükség arra, hogy a mérési adatok küldése meginduljon. Az idővezérelt szenzorok esetében pedig egy meghatározott időközönként történik meg az adatküldés. Például, ha napi átlaghőmérsékletet akarunk mérni, akkor ez a periódus 24 óra.

Következő csoportosítás a hálózaton belüli kommunikáción alapul. Két nagyobb csoportot különböztethetünk meg egymástól angol szakszóval megnevezve.

- single-hop hálózat
- multi-hop hálózat

A single-hop hálózat esetében a szenzorok csak egy lépcsőben tudnak kommunikálni, azaz közvetlenül a nyelővel tudnak kapcsolatot teremteni. A mérésre szánt szenzorok ilyenkor nem képesek az egymással való kommunikálásra. Ezért probléma lehet, ha a nyelő szenzor nincs megfelelő közelségben ahhoz a szenzorhoz, amelyikkel fel szeretné venni a kapcsolatot. Ezért ebben az esetben meg kell oldani, hogy a nyelő szenzor mindig képes legyen fogadni a kommunikációs kéréseket, különben elveszhetnek a mérési adatok. Erre nagyon jó módszer a nyelők, illetve a mérésre szánt szenzorok mozgatása. A multi-hop hálózatok esetében azonban a szenzorok egymás között is tudnak adatot átküldeni, így az adat több lépésben jut el a nyelőhöz. Itt már nem fontos, hogy a nyelő közvetlen kapcsolatot tudjon létrehozni a mérésre szánt szenzorokkal, hiszen a nélkül is eljuthat hozzá az adat.

Single-hop hálózatok esetén a nyelőtől távolabb levő szenzorok energiafelhasználása a nagyobb, hiszen több energia kell a kommunikációs távolság áthidalásához. Multi-hop hálózatok esetében azonban a nyelő közelében lévő szenzorok használnak fel több energiát, mert ebben az esetben az ő terhelésük jóval nagyobb, mivel gyakorlatilag a hálózat forgalmának nagy része rajtuk halad át. Tehát a nyelő mozgatása ugyan valamivel több energia felhasználást igényel, de megfelelő nyelő/szenzor darabszámok meghatározásával ezen csökkenthetünk. Felvetődik a kérdés, hogy a nyelőt milyen stratégia szerint érdemes

mozgatni. Ez az alapja a következő csoportosításnak. Három módszer alkot egy-egy csoportot.

- véletlenszerű mozgatás,
- jósolható mozgatás,
- vezérelt mozgatás.

A véletlenszerű mozgatás esetében a nyelő nem előre meghatározott módon, azaz véletlenszerűen mozog. Második a jósolható mozgatás. Ebben az esetben a nyelő egy előre kijelölt pályán halad végig. Így a szenzorok mozgásához igazodva tud haladni és a szenzorok adatátküldése hatékonyabb lehet, kialakíthatjuk úgy az útvonalat, hogy a lehető legtöbb szenzor kommunikációs távolságát érintse. Harmadik módszer a vezérelt mozgatás, amely nagyban hasonlít az előzőhöz. A különbség annyi, hogy itt a hálózat aktuális állapotát is figyelembe kell venni az útvonal meghatározásakor, vagyis a nyelő útvonala dinamikusan változhat. Azonban ezt a három csoportot könnyen lehet finomítani további alcsoportosításokkal, alkalmazott protokollokkal.

A véletlen mozgatás során MULE (Mobile Ubiquitous LAN Extension) eszköz használata javasolt olyan hálózatok esetében, amelyeknél az adatok továbbításának nem kell valósidejűnek lennie [9]. Ezen felül a szenzorok olyan ritkán vannak telepítve, hogy nem tudják egymáson keresztül eljuttatni az adatokat a nyelőhöz. Elsőre ezt két fajta módon is ki lehetne küszöbölni. Vagy több bázisállomást használunk, vagy nagyobb teljesítményű szenzorokat telepítünk. Mindkét módszer elég költséges, de ezért jó, ha a MULE eszközök használatát választjuk. Ebben az esetben a nyelót magával hordozza egy mobil eszköz, ami véletlenszerű bolyongással mozog. Ha egy olyan szenzor közelébe ér, amelyik adatot akar átadni, akkor fogadja ezt az adatot, majd az átadás után tovább halad. Az összegyűjtött eredményeket egy fix pontban elhelyezett hozzáférési pontnak adja át, amint a közelébe ér. A SENMA (SEnsor Networks with Mobile Agents) architektúra is véletlenszerű mozgást végző nyelót alkalmaz [10]. A különbség az előző módszertől az, hogy itt sűrűn telepített szenzorhálózatot feltételezünk. A nyelő nem a felszínen, hanem a felszín felett mozog véletlenszerű mozgást követve. Így több oldalról csökken az energiafelhasználás. Egyrészt minden szenzor közvetlenül a nyelóval kommunikál, így az adatnak nem kell több csomópontot érintenie. Másrészt a jel csillapítása kisebb függőleges irányban, mintha a felszínen mozogna a nyelő és a csillapítás vízszintesen történne. Ezért a szenzoroknak elég kisebb erősségű jelet sugározniuk, ami jelentős energia megtakarítást eredményez.

A jósolható mozgásnál a jövőbeli viselkedés kiszámítható, hiszen a nyelő egy meghatározott pályán mozog. Chakrabarti [11] szerint egy a szenzorhálózat minden elemén érintőlegesen periodikusan végighaladó nyelő alkalmazását javasolja. A nyelő úgy halad, hogy minden szenzor hatósugarát érintse, így mindegyik szenzornak lehetősége adódik az adatai átküldésére. Azonban egy szenzornak csak az a meghatározott időintervallum áll rendelkezésére, ami alatt a nyelő áthalad a hatósugarán belül. Ha ez az időintervallum túl rövid, akkor az adatok egy része elveszhet. Luo és társa szerint egy sűrűn telepített, erősen összefüggő, kör alakú szenzorhálózatot feltételezve hatékony megoldást kapunk [12]. Elemzik, hogy egy ilyen hálózatban mekkora a nyelő körüli szenzorok terhelése. Az eredményekből adódóan a nyelő körüli szenzorok terhelése nagyobb, így a nyelőnek mozognia kell, és ezáltal a terhelés kiegyenlítődhet. A mozgás útvonala periodikus és szimmetrikus a kör középpontra, ilyen, ha a nyelő a hálózat szélén mozog körbe. Egy finomított módszer erre, ha a nyelő a kör középpontjához közelebbi körpályán mozog, a csomag viszont a kör szélén lévő körpályán. A csomagot a nyelő akkor kapja meg, ha a nyelő és a csomag egy magasságba ér, így a legrövidebb útvonalon keresztül haladhat át a csomag, ezzel csökkentve az energiafelhasználást.

A vezérlő algoritmussal történő mozgás a legintelligensebb az eddig felsoroltak közül. Itt ugyanis a hálózat aktuális állapotát figyelembe véve alakítja a nyelő az útvonalat. Az algoritmusok általában két paraméter módosításával valósíthatók meg. Egyik paraméter a mozgás sebessége fix útvonal esetén, másik esetben pedig a mozgás irányát és a mozgás sebességét is paraméterként kezelik. Az első esetre példa az AIMMS (Autonomous Intelligent Mobile Micro-server) protokoll [13][14]. Itt egy véletlenszerűen telepített, multi-hop hálózatot feltételezhetünk. Egy szerver eszköz mozog körbe egy meghatározott útvonalon és gyűjti be közben az adatokat. Az útvonal mellett elhelyezkedő szenzorok közvetlenül a szervernek küldik át a mérések eredményeit, a távolabbi szenzorok pedig a többi csomóponton keresztül juttatják el az adatukat a szerverhez. A szerver a megtett útja során összegyűjtött adatokat egy meghatározott fix ponton lévő nyelőnek továbbítja. A szerver  $T$  idő alatt érhet maximum végig a meghatározott útvonalán, valóságban pedig  $t$  idő alatt ér körbe. Ha  $t < T$ , tehát gyorsabban végigér az útvonalon, mint a maximális megengedett idő erre, akkor lehetősége van bizonyos csomópontoknál lelassítani. Az előzőekben már felvetődött az a probléma, hogy ha csak meghatározott időintervallum áll rendelkezésre az adatok átküldéséhez, abban az esetben, ha ez az intervallum túl kicsi, adatvesztés léphet fel. Ez a módszer ezt próbálja meg kiküszöbölni. Ha a szerver úgy véli, akkor lelassít, máshol

felgyorsíthat. Lelassítani rossz jelterjedési viszonyok, illetve közeg hozzáférési nehézségek miatt kellhet.

Ahogy az előzőekben is említettem, a szenzorok energiafogyasztása, energia hatékony alkalmazása fontos szempont a megfelelő működés eléréséhez [15]. Mivel a szenzorok egyszerű eszközök, ezért az energiakapacitásuk véges. Nagyon sok esetben a lemerült energiaforrások cseréje időigényes, vagy egyáltalán nem megoldható, tehát valamilyen energiagazdálkodási eljárás szükséges az alkalmazásukhoz. A szakirodalomban többféle módszer is található erre a problémára, megfelelő energiamedzsmenttel, energiafogyasztások csökkentésével, vagy akár külső energiaellátással is megoldható [16][17][18]. A hagyományos energiátároló eszközök (elem, akkumulátor) stabil, megbízható és jól becsülhető működést eredményeznek, viszont élettartalmuk véges, ezáltal korlátozzák a működést is. Azonban léteznek újratölthető tárolók is, amelyeket egy külső energiaforrás ismételten fel tud tölteni. Környezetünkben több olyan természeti, gépi, vagy akár emberi jelenség található, amelyek képesek az energia leadásra, ilyen például a napsugárzás, levegőáramlás, különböző gépek rezgése, mozgása, emberi testhőmérséklet.

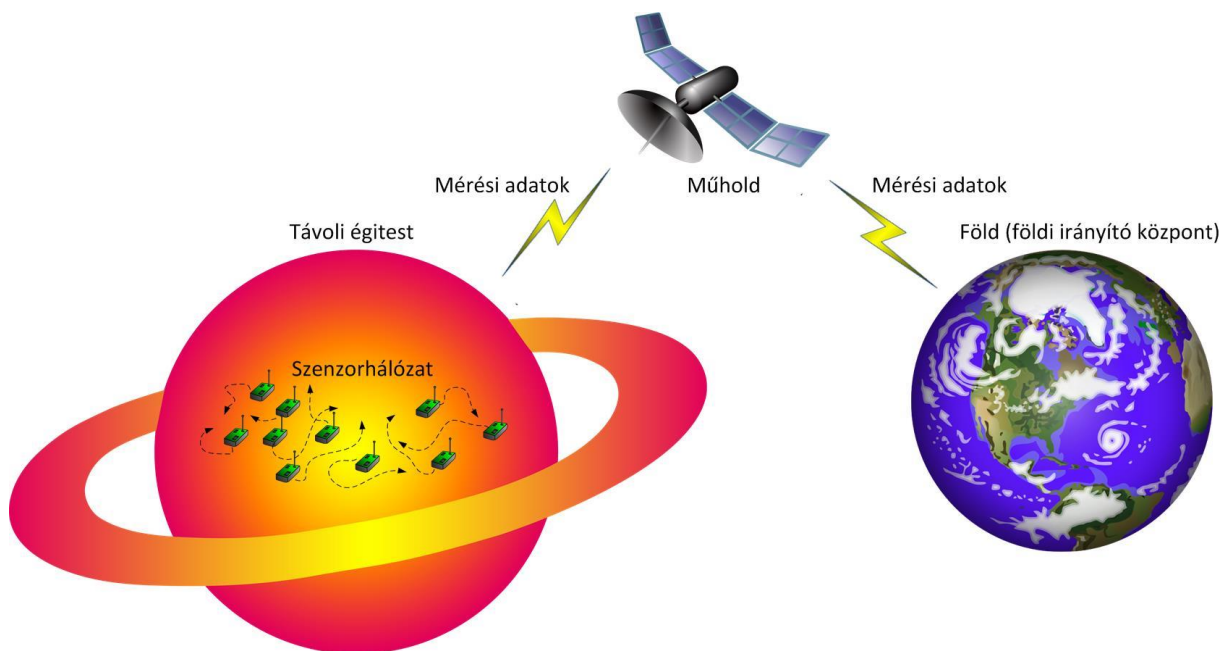
### 3. A vizsgált szenzorhálózat modellje

Ebben a fejezetben bemutatom az általam feltételezett rendszert, szenzorhálózatot, annak működését, fontosabb jellemzőit, végül pedig a fejlesztett szimulációs program alapvetőbb és legfontosabb részeit.

#### 3.1 Rendszerterv

Az általam feltételezett rendszer 4 fő entitással rendelkezik, egy földi irányító központtal, egy távoli égitesttel, egy kommunikációs keringő egységgel a távoli égitest körül (műhold vagy űrszonda) és egy a távoli égitesten elhelyezett szenzorhálózattal.

A földi irányító központ a kommunikációs keringő egységgel áll kapcsolatban, amely a távoli égitest felszínéről kapott adatokat – mihamarabb tudja – továbbítja az irányító központ számára. A kommunikációs egység lehet műhold (bolygó körül) vagy űrszonda (pl. kisbolygó körül), és jelen rendszerben elsősorban összekötő szerepe van a távoli égitest és a Föld között. (Ezzel párhuzamosan természetesen más feladatot is elláthat, pl. távérzékelés.) Csak egy ilyen egység van jelen a rendszerben a költséghatékonyság miatt. A kihelyezett szenzorhálózat egységei végzik a megfelelő méréseket és gyűjtik össze az adatokat, amelyeket periódusonként továbbítanak a keringő egység felé. A rendszer elemeinek elhelyezkedését a 3.1 ábra szemlélteti.



3.1 ábra

A feltételezett rendszer szemléltetése

A munkámban a fent leírt rendszer speciális esetével foglalkoztam, amelyben a távoli égitest szerepét a Mars tölti be, így konkrét tényezők jellemezhetik a vizsgálni kívánt területet. A felszíni egyenlőtlenségeket szemléltetik a kráterek és dűnék, ezek befolyásolják a szenzorok haladását. Emellett az égitest felszínét két fajta réteg boríthatja, a kavicsos és a homokos réteg. Mindeket réteg más-más mértékben lassítja a szenzorok mozgását. A Marson jelenlévő jelenségek közé tartozik a porvihar, amely ideje alatt a szenzorok között megszakad a kommunikáció.

A szenzorhálózat háromféle mérést képes elvégezni, sugárzásmérést, atmoszféra mérését és mágneses erőtér mérését, ezen kívül pedig képi felvételeket készít a belátható területről. A mért adatokat meghatározott periódusok szerint küldi át a műholdnak.

A szimuláció kialakítása a speciális rendszer alapján valósult meg, így a továbbiakban ez tekinthető alapesetnek.

## **3.2 A feltételezett szenzorhálózat alapvető felépítése**

### **3.2.1 Nagyobb teljesítményű szenzorok a hálózatban**

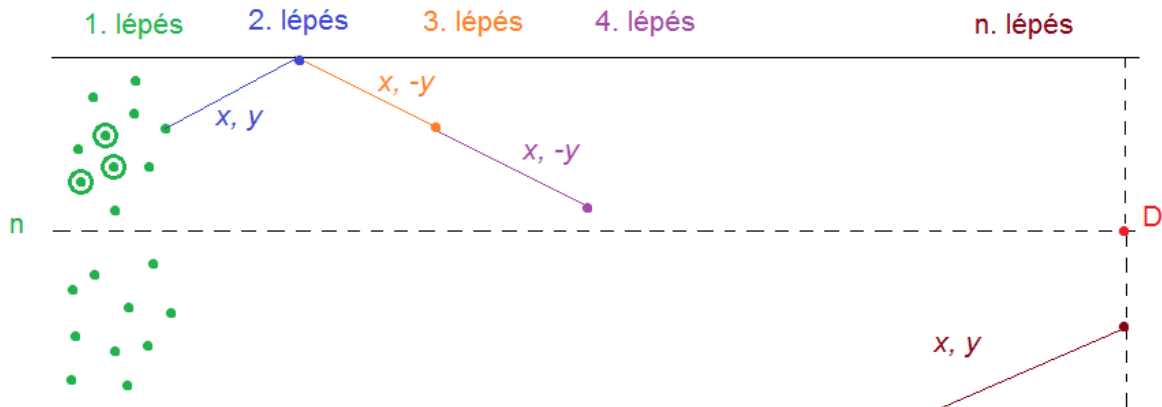
A szenzorhálózatot alakító szenzorok két csoportba oszthatók. Az első csoportban bonyolultabb, nagyobb teljesítményű nyelő szenzorok találhatóak, amelyek fő feladata a második csoportban lévő szenzorok által mért adatok összegyűjtése és ezek továbbítása a keringő kommunikációs egység felé. Ezekre a szenzorokra a továbbiakban szuperszenzorokként hivatkozok. A második csoport egyszerűbb felépítésű szenzorokból áll, amelyek a méréseket végzik.

A szuperszenzorokra azért van szükség, mert így nem szükséges minden szenzornak felvenni a kapcsolatot a műhoddal, elég, ha csak a többi szenzorral képesek kommunikálni. Így a hálózat energiafogyasztása jóval kisebb lesz, mint ellenkező esetben.

### **3.2.2 Szenzorok és szuperszenzorok mozgása**

Ahhoz, hogy nagyobb területet lehessen feltérképezni a szenzorhálózatok segítségével, szükséges, hogy a szenzorok képesek legyenek a helyváltoztatásra. A 3.1. fejezetben megadott speciális rendszer esetében mind a mérő szenzorok, mind pedig a szuperszenzorok tudnak mozogni. A helyváltoztatás módját a 2.2.1 alfejezetben ismertetett véletlenszerű mozgás és előre meghatározott pályán való mozgás kombinációja adja. Eszerint a szenzorok véletlenszerű irányba kezdenek el mozogni, majd pedig ezt az irányt megtartva haladnak tovább egy előre meghatározott pont síkja felé. Ahhoz, hogy a szenzorok együtt maradjanak

(egy bolyban haladjanak tovább) bevezettem egy alsó és felső biztonsági sávhatárt, amely a szenzorok koordinátáinak ad alsó és felső korlátot. Ha egy szenzor eléri ezt a sávhatárt, akkor irányt változtat a 3.2.2 ábra szerint és ebbe az irányba halad tovább. A zöld pontok jelölik a szenzorokat, a körrel jelzett pontok a szuperszenzorokat. A képen egy kiragadott szenzor mozgását követhetjük nyomon,  $x$  és  $y$  jelölik az elmozdulás irányát, a  $D$  pont pedig az elérendő célt.



3.2.2 ábra

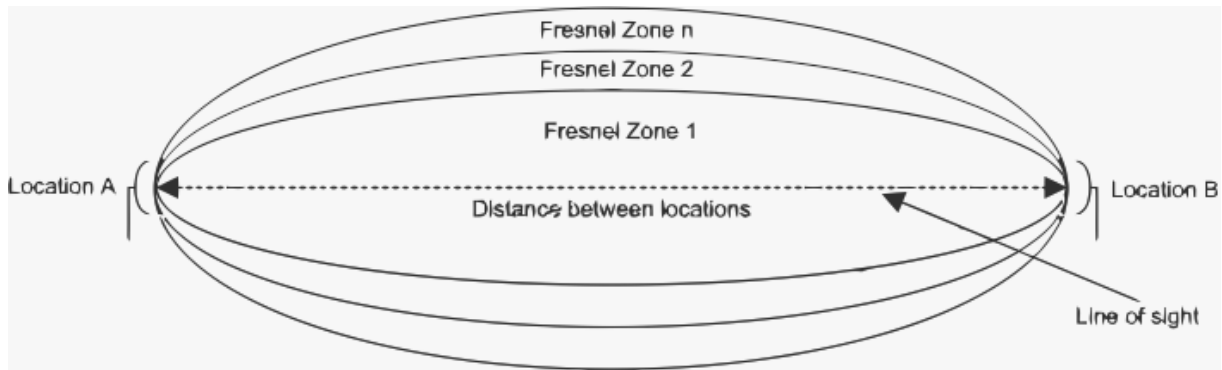
A szenzorok mozgásának szemléltetése

### 3.2.3 Jelterjedési modellek

A szenzorok közötti kommunikációban nagy szerepet játszik a jel terjedése, csillapodása. Mivel a Mars felszíne nem sík terep, hanem kráterek, dűnék, sziklás emelkedők felelnek a felszíni egyenlőtlenségekért, ezért az érzékelők az átküldött jeleket más és más vevőtéljesítményekkel tudják érzékelni. A jelterjedés alkalmazására a Deygout módszert és a szabadtéri jelterjedési modellt vettem alapul a szimulációm során [19][20].

A Deygout módszerben a vizsgálni kívánt terepmetszet egy intervallumot határoz meg. Az átvitel szakaszcsillapítását nagymértékben befolyásolja, ha a Fresnel-zóna [21] takarásba kerül. Kiemelt szerepe van az első Fresnel zónának, mert az energia körülbelül 90%-a ezen a nyomvonalon halad keresztül. A Fresnel-zóna számítása az alábbi 3.2.3.a ábrán és (1) képleten látható.





3.2.3.a ábra  
Fresnel-zóna

$$r = 17.3 * \sqrt{\frac{d * (D - d)}{f * D}}, \quad (1)$$

ahol az alábbi jelöléseket használtam

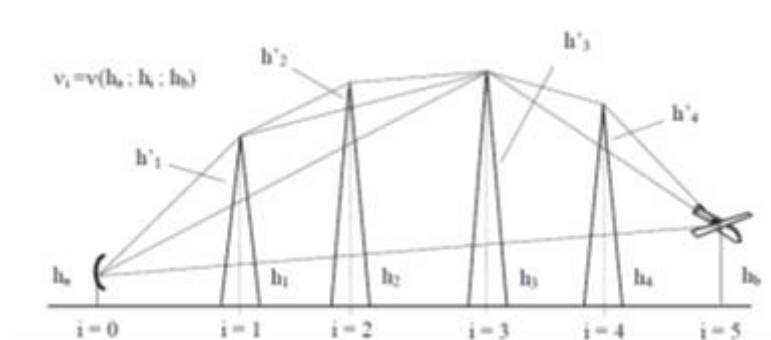
$r$  = Fresnel-zóna sugara (méterben)

$f$  = frekvencia (GHz-ben)

$d$  = vizsgált távolság (kilométerben)

$D$  = teljes távolság (kilométerben).

Ezek alapján tehát az I. Fresnel-zóna gyújtópontjai az intervallum határain lesznek. Ezek után az intervallum összes csúcsát figyelembe véve meg kell határozni a maximális fadinget okozó csúcs helyét, azaz ahol a legnagyobb a benyúlás mértéke. A maximális fadinget okozó késélnél az intervallum két részre válik szét. Majd rekurzívan ezekre a részintervallumokra is meghatározzuk a hozzá tartozó maximális fading helyét, ami szintén tovább osztja az intervallumot. Ez a rekurzív eljárás addig folytatódik, amíg a részintervallumokban késél található. Az így kiszámított maximális fadingek összege adja ki az eredő diffrakciós fadinget. Ez az eljárás a 3.2.3.b ábrán kísérhető figyelemmel a (2) és (3) képlettel kiegészítve.



3.2.3.b ábra  
Deygout-modell

$$v = h * \sqrt{\frac{2}{\lambda}} * \left(\frac{1}{d_1} + \frac{1}{d_2}\right) \quad (2)$$

$$L(v_0)[dB] = \begin{cases} 20 * \lg(1) & v_0 \in (-\infty; -0.8) \\ 20 * \lg(0.5 * 0.62 * v_0) & v_0 \in [-0.8; 0) \\ 20 * \lg(0.5 * 10^{-0.95 * v_0}) & v_0 \in [0; 1) \\ 20 * \lg(0.4 - \sqrt{0.1184 - (0.38 - 0.1 * v_0)^2}) & v_0 \in [1; 2.4) \\ 20 * \lg\left(\frac{0.225}{v_0}\right) & v_0 \in [2.4; \infty) \end{cases} \quad (3)$$

A második a szabadtéri jelterjedési modell, amelynek alapja, hogy nincs fading jelenség, azaz a jeleknek nem kell áthatolni semmilyen akadályon. Ebben az esetben a vett jel erősségét a (4) képlet írja le.

$$Pr = Pt + Gt + Gr + \frac{\lambda^2}{4 * \pi * d}, \quad (4)$$

ahol az alábbi jelöléseket használtam:

*Pr*: vett jel teljesítménye (W)

*Pt*: adóteljesítmény (W)

*Gt*: adóantenna nyeresége (viszonyszám)

*Gr*: adóantenna nyeresége (viszonyszám)

$\lambda$ : hullámhossz (m)

*d*: távolság (m).

Az egyenletből kivehető, hogy a vételi teljesítmény egyenes arányban áll az antennanyereségekkel, illetve fordított arányban van a távolság négyzetével.

Amikor egy szenzor adatot küld át egy másíknak, akkor a jel egy valós terepviszonyokkal rendelkező területen halad keresztül. A terepviszonyok befolyásolják a jel csillapítását, ezért ehhez a folyamathoz a Deygout-modellt használtam fel. Ezután a vevő szenzor megkapja a jelet, de nem ismeri a befolyásoló terepviszonyokat, ezért ő a szabadtéri jelterjedési modell szerint számolja ki a küldő szenzortól való távolságát. Mivel a jel valóságban megtett Deygout-módszer szerint kiszámolt távolsága eltér a szabadtéri modell szerint kiszámolt távolságtól, így a szenzor valamelyest arrébb helyezi magát a valós helyzetétől. Ekkor történik az a mérési eltérés, ami a későbbiekben hibatorlódáshoz vezet.

Hibatorlódás esetében a valós és számolt koordináták közötti eltérés kezdetben még minimális, ugyanakkor többlépcsős helymeghatározásnál már olyan nagymértékű lehet, hogy adatvesztés léphet fel. Ez abból adódik, hogy a szenzor úgy érzékeli, hogy már nincs olyan közelségben a többi érzékelőhöz, hogy továbbküldje a mért adatait (ellenben a valóságban még beleszne a hatótávolságba) és így az adatátvitel nem valósul meg. Másik jelentős probléma, amikor a szenzor úgy érzékeli, hogy még nem lépte át a biztonsági sávhatárt és nem változtat irányt, a valóságban azonban már túllépte, ennek következtében leszakadhat a bolytól.

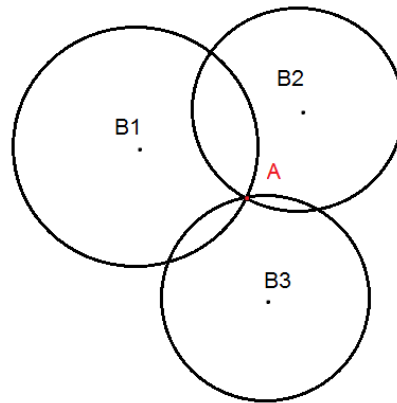
### **3.2.4 Szenzorok pozicionálása**

Mint ahogy már említettem, a 3.1 fejezetben megadott speciális rendszerben kifejezetten fontos a mérési helyek pontos ismerete. Korábbi munkáimban már részletesen foglalkoztam a pozicionálás témakörével, így most egy áttekintő összefoglalást adok erről.

#### **3.2.4.1 Többugrásos pozicionálás és hibatorlódás**

Mivel GPS-jellegű rendszer nem áll rendelkezésre, ezért a pozíció meghatározáshoz a háromszögelés módszerét vettem alapul. A pontok egymáshoz viszonyított helyzete alapján való helymeghatározásra ez az egyik legpontosabb módszer [22]. A háromszögelés valójában nem áll másból, mint háromszögek oldalainak és szögeinek meghatározásából. Ezért bármilyen sok pont esetén is, ha csak a pontok nem esnek egy egyenesbe, akkor meghatározhatóak a nekünk kellő háromszögek. Így ha a háromszögek alkotóelemeit ismerjük, akkor a pontok egymáshoz viszonyított helyzetét is ismerjük.

Az első lépés az úgynevezett bázismérés, ami azt jelenti, hogy ha egy oldalhosszt már megmértük, akkor az alapul szolgál a többi oldalhossz megismerésének is. Ezért ezt a kezdő oldalhosszra, mintegy bázisra támaszkodik a többi oldal hosszának meghatározása. Ezután következhet a szögmérés. A szögek mérése sokkal kisebb hibával jár, így ha lehetséges, akkor inkább ezzel a módszerrel érdemes mindent meghatározni, ezáltal helymeghatározás pontossága a lehető legjobb eredményt adja. Esetünkben a pozicionálás lényege, hogy egy szenzor (legyen ez  $A$ ) helyzetét három másik szenzor (legyenek ezek  $B1$ ,  $B2$ ,  $B3$ ) segítségével állapítja meg, ezt szemlélteti a 3.2.4.1 ábra.



**3.2.4.1 ábra**

**Háromszögelés szemléltetése.**

A *B1*, *B2* és *B3* szenzorok hatósugarai egy-egy kört alkotnak, amely körön belül a szenzor érzékeli tud egy másik érzékelőt, így az adatküldés köztük lehetségessé válik. A *B1*, *B2* és *B3* szenzorok és hatósugarai által meghatározott körök egy területet metszenek ki, amelyen belül megbecsülhető az *A* szenzor helyzete. Ezek alapján a *B1*, *B2* és *B3* szenzorok referenciapontok az *A* szenzor számára. A helymeghatározás számítása után az *A* szenzor is egy olyan referenciaponttá válik, amely segítségével egy következő *C* szenzor elhelyezkedése kiszámítható, ez a többugrásos helymeghatározás. Azonban ha már az *A* szenzor helyzetének meghatározásában jelen volt egy adott hiba, akkor ezt a hibát „tovább görgeti” és a *C* szenzor helymeghatározásában is benne lesz valamekkora súlyozással. Tehát a hiba torlódik és halmozódik is, ami azt jelenti, hogy többugrásos helymeghatározás esetén növekszik is.

Az adott szenzorhálózatban a szuperszenzorok állandóan referenciapontnak minősülnek, mivel folyamatosan ismerik a pontos helyzetüket a műholddal való kommunikáció és egy speciális beépített iránytű segítségével.

Mivel térben elhelyezett szenzorhálózatról van szó, így nem lenne elég három szenzor koordinátája a pozicionáláshoz, viszont a szimulációs program olyan térképekkel dolgozik, amelyben minden pixelhez egy magasságérték van rendelve, amely elegendő plusz információ a helymeghatározáshoz.

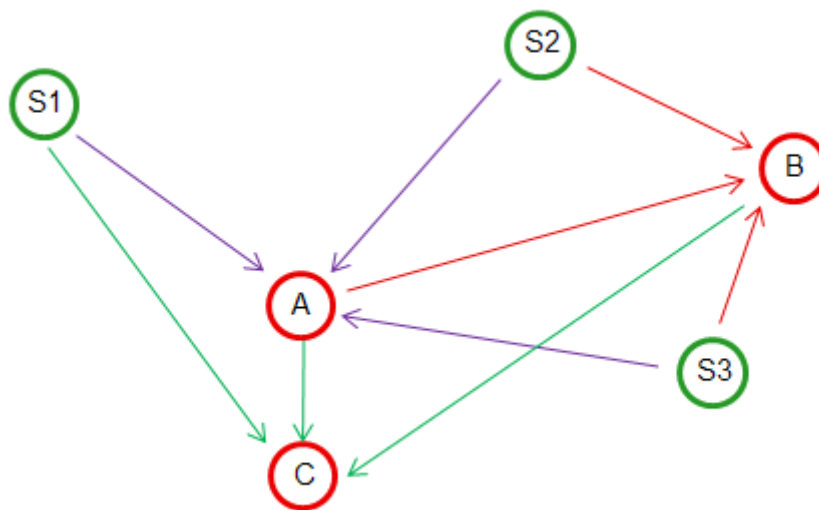
**3.2.4.2 Pozícióbecslési algoritmusok**

Három olyan algoritmust alakítottam ki, amelyek a referencia pontok közül különböző módon választanak ki szenzorokat a számításokhoz, ezekre a szenzorokra a továbbiakban szülő szenzorokként hivatkozok. Ez azt eredményezi, hogy különböző pontossággal dolgoznak.

### Optimalizálatlan alap algoritmus (Basic Position Algorithm without optimization - BPA)

A kezdeti algoritmusom esetében még nincs optimalizáló eljárás. A kiszámítandó helyzetű szenzor egy egyszerű, bár kevésbé hatékony módszerrel választja ki a potenciális referencia szenzorok közül a három szülőjét. Az algoritmus elkezd vizsgálni a bolyban szereplő szenzorokat, és ha referencia pont, akkor beveszi a szülők közé. Az első három találat után a vizsgálat megáll és adottak lesznek a használni kívánt referencia szenzorok. Tehát a bolyból az első három referencia szenzort választja ki.

A módszer azért nem hatékony megoldás, mert ha a lehetséges referencia szenzorok közül három olyat választ ki az algoritmus, akik már torlódott hibát hordoznak magukkal, akkor ő hibájukat az adott szenzorunk is hordozni fogja. Tehát a számítással kapott eredmény valamilyen mértékben biztosan eltér majd a valós helyzetétől. Ezzel szemben, ha választhatná szülőkné a három alap szuperszenzort, vagy kisebb hibával rendelkező szenzort, akkor jóval kisebb eltérés adódna a kiszámolt értékhez. Az algoritmusra egy példa a 3.2.4.2.a ábrán látható, ahol  $S1$ ,  $S2$  és  $S3$  jelölik a szuperszenzorokat,  $A$ ,  $B$  és  $C$  pedig a kiszámítandó helyzetű szenzorokat.



3.2.4.2.a ábra

A BPA algoritmus működésének szemléltetése

A példa kezdeti referencia szenzorjai:  $S1$ ,  $S2$ ,  $S3$

Első lépés:  $S1$ ,  $S2$ ,  $S3 \rightarrow A$

Első lépés utáni referencia szenzorok:  $A$ ,  $S1$ ,  $S2$ ,  $S3$

Második lépés:  $A$ ,  $S2$ ,  $S3 \rightarrow B$  ( $S1$  túl messze van)

Második lépés utáni referencia szenzorok:  $A$ ,  $B$ ,  $S1$ ,  $S2$ ,  $S3$

Harmadik lépés:  $A, B, S1 \rightarrow C$

Harmadik lépés utáni referencia szenzorok:  $A, B, C, S1, S2, S3$

### Számítási hop számot figyelembe vevő algoritmus (Positioning Algorithm based on Hop Number - PAHN)

Ebben az esetben a szenzorok tárolnak egy olyan hop értéket, ami arra utal, hogy hány lépcsős pozicionálás eredménye a kiszámolt koordinátájuk. Minél későbbi lépcsőben kerül sor egy szenzor pozíciójának becslésére, a hibatorlódás annál jelentősebb. A cél ezért az, hogy olyan szenzorokat jelöljünk ki referenciaként, melyeknek koordinátáit a lehető legkevesebb lépcsőben számíthatók ki. A szuperszenzorok hop száma alaptól az egész vizsgálat során 0. A szülők hopszáma átlagolódik, és ehhez hozzáadódik egy egységnyi érték, ezzel jelezve, hogy plusz egy számítás történt.

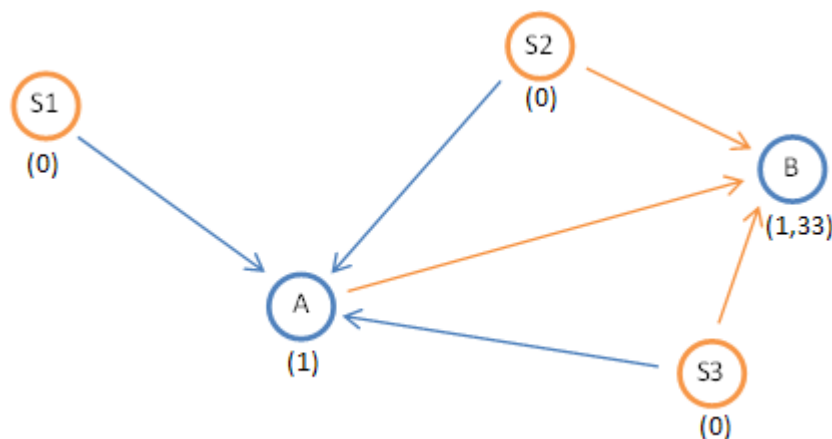
Példaként vegyünk egy  $A$  szenzort, aki nem referencia pont és a szülői legyenek a szuperszenzorok,  $S1, S2, S3$ . Ekkor  $A$  hopszáma (5) alapján számítható ki:

$$(0+1 +0+1 +0+1)/3=1 \quad (5)$$

Ha  $B$  szintén nem referencia szenzor szülői közé beletartozik ezután  $A$  szenzor és két szuperszenzor, akkor az  $B$  hopszáma (6) alapján alakul.

$$(1+1 +0+1 +0+1)/3=1,33 \quad (6)$$

A bemutatott példa a 3.2.4.2.b ábrán látható.



3.2.4.2.b ábra

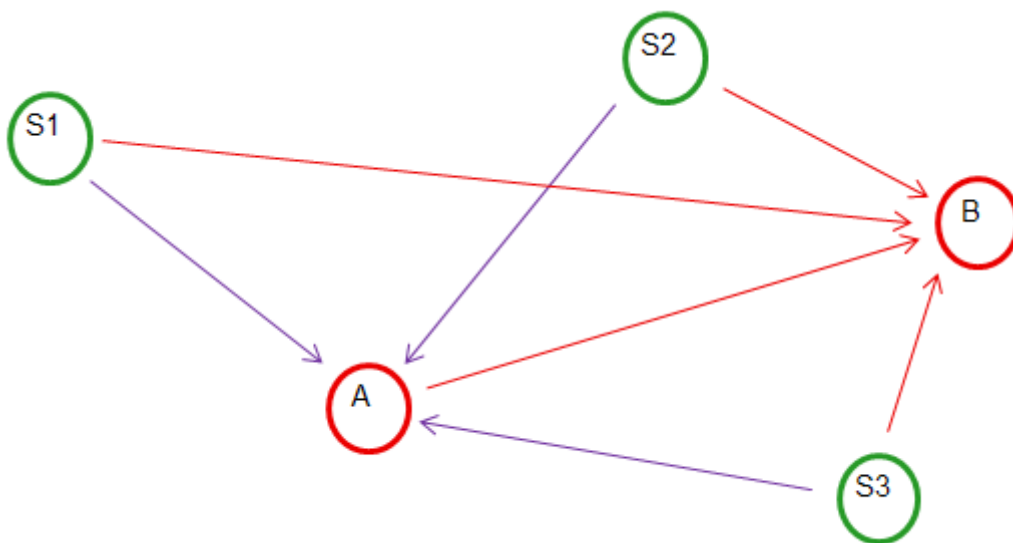
A PAHN algoritmus működésének szemléltetése

Ebből kifolyólag a hopszám azt is mutatja, hogy mennyire pontos számítást tudunk végezni a pozicionálás során. Minél kisebb egy szenzor hopszáma, annál pontosabb számítási

eredményt kapunk. Pont ezért egy szenzor az alapján választ magának szülőket, hogy minél kisebb legyen a hopszáma. Így mindig a lehető legpontosabb koordinátákkal tud majd számolni.

### Átlagolást alkalmazó algoritmus (Extended Basic Positioning Algorithm - EBPA)

Ebben az esetben nem csak egy lehetőséget vizsgál a módszer, hanem minden lehetséges hármas csoportot kiválaszt, akik szülők lehetnek és ezek alapján kiszámolja mindegyikre a koordinátákat. Végül pedig ezen értékek átlagát veszi. A 3.2.4.2.c ábra szemléltet egy példát az algoritmusra, itt szintén  $S1$ ,  $S2$  és  $S3$  jelöli a superszenzorokat,  $A$ ,  $B$  és  $C$  pedig azokat, amelyek pozíciójára kíváncsiak vagyunk.



3.2.4.2.c ábra

Az EBPA algoritmus működésének szemléltetése

A szenzorok helyzetét az alábbi (7) képlet alapján számoljuk ki.

$$A = \frac{\sum_{i=1}^n A_i}{n} \quad (7)$$

A példa kezdeti referencia szenzorjai:  $S1$ ,  $S2$ ,  $S3$

Első lépés:  $S1$ ,  $S2$ ,  $S3 \rightarrow A$

Első lépés utáni referencia szenzorok:  $A$ ,  $S1$ ,  $S2$ ,  $S3$

Második lépés:

a.  $S1$ ,  $S2$ ,  $S3 \rightarrow B$

b.  $A$ ,  $S1$ ,  $S2 \rightarrow B$

c.  $A$ ,  $S1$ ,  $S3 \rightarrow B$

d.  $A, S2, S3 \rightarrow B$

Vesszük  $B$ -re a kiszámolt koordináták átlagát, ez adja  $B$  koordinátáit.

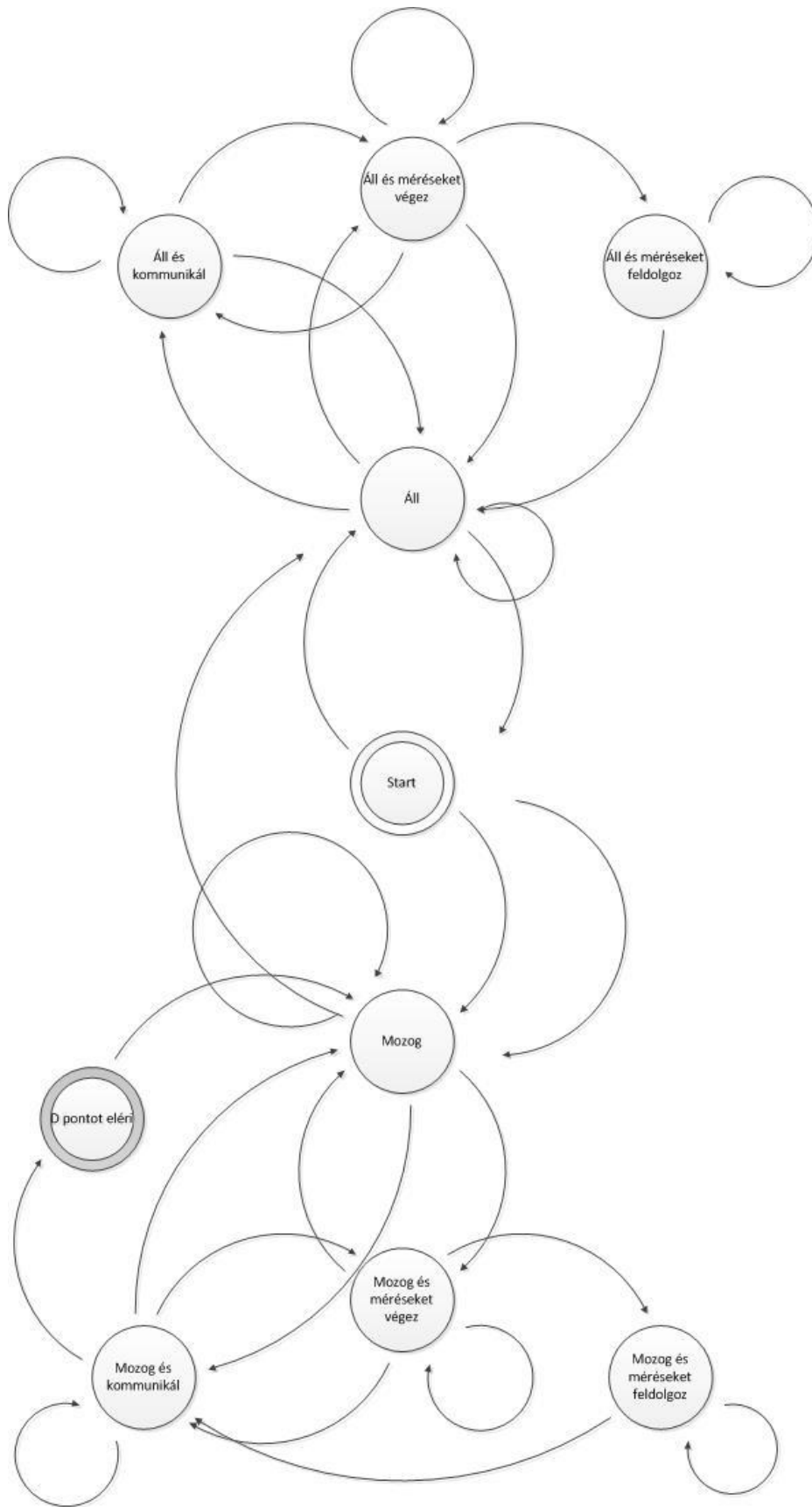
Második lépés utáni referencia szenzorok:  $A, B, S1, S2, S3$

Több szimulációs vizsgálatot végeztem és összesítve a PAHN algoritmus bizonyult a leghatékonyabbnak pozicionálás terén.

### **3.2.5 Szenzorok állapotdiagramja**

A szenzorok viselkedéséről összeállítottam egy általános állapotdiagramot, amelyet alapján történt a szenzorok energiafogyasztásának kiszámítása, és amelynek egy speciális változatát beleépítettem a szimulációs programba is mintaként. A diagram a 3.2.5 ábrán látható.





**3.2.5 ábra**  
**Szenzorok általános állapotdiagramja**

A diagram két fő részre osztható a kiinduló „Start” állapot után, egy mozgó és egy álló részre. A „Mozog” állapotban a szenzor a mozgáson kívül más cselekvést nem végez, ebből az állapotból a szenzor vagy „Áll” állapotba kerül, vagy tovább mozog és kommunikáló, illetve mérő helyzetbe juthat. A kommunikációs rész magába foglalja mind az adatok továbbküldését a szuperszenzoroknak, mind pedig a pozicionáláshoz szükséges helyzetjelentést is. A mérések állapotában végzi a szenzor a 3.1 fejezetben, a speciális rendszertervben felsorolt méréseket, illetve a felvétel készítését. Innen az érzékelő a kommunikáló és a szimpla mozgó állapot mellett átjuthat egy adatfeldolgozó állapotba, ahol a mért adatokról dönti el, hogy érvényesek, felhasználhatók-e.

Az álló rész szintén tartalmaz a szimpla „Áll” állapot mellett kommunikáló, mérő és feldolgozó állapotokat is, amelyek funkciója megegyezik a mozgó részben leírtakkal, azzal az eltéréssel, hogy itt a szenzorok közben nem végeznek mozgást. Ez úgy valósulhat meg, hogy például mozgásra már nem elegendő az energiatartaléka, de méréseket még képes végezni, vagy megakadt egy emelkedőnél és nem tud tovább haladni. A kiinduló és végpont kivételével minden állapot lehet hurokállapot is, azaz több lépésen keresztül csak saját magába megy át. A végpont csak a mozgó részből érhető el, ami egyértelmű, hiszen csak mozgással juthatnak el a szenzorok a megadott  $D$  célpontba.

A szimulációs program ennek a diagramnak egy speciális változatát tartalmazza, amelyben a „Mozog és kommunikál”, „Mozog és méréseket végez”, „Mozog és méréseket feldolgoz”, illetve az „Áll és kommunikál”, „Áll és méréseket végez” és „Áll és méréseket feldolgoz” állapotok össze vannak vonva egy-egy külön mozgó és álló állapottá. Ez az egyszerűsítés azért volt szükséges, mert a program diszkrét időszelvényeket vizsgál, és egy időszelvénybe esik bele a három mozgó és a három álló állapot. Tehát gyakorlatilag a program szempontjából a kommunikáció, a mérések elvégzése és a feldolgozás együtt történik. A szimulációs program működését bővebben a 3.3 fejezetben ismertetem.

### **3.2.6 Szenzorok energiafogyasztása**

A szenzorok energiafogyasztása a 3.2.5 fejezetben bemutatott állapotdiagram alapján történik. Azaz a mozgás, a kommunikáció, a mérések és azok feldolgozása, ami energiaveszteséggel jár. Ezek a cselekvések csak akkor kerülhetnek kivitelezésre, ha a szenzor energiaszintje elé magas. Így például ha a szenzor energiatartaléka nem elég a mozgathoz, de a mérések elvégzéséhez még igen, akkor álló állapotban fogja azokat megvalósítani.

Kiindulásképp meg kellett határozni, hogy mekkora legyen egy szenzor akkumulátor kapacitása. Ehhez alapul vettem egy távirányítású kisautó akkumulátorának méretét. A távirányítású kisautó ugyan bonyolultabb eszköz egy szenzornál, de alapfunkciói, a mozgás és a távirányítóval való kommunikáció, jó kiindulási alapot adnak az akkumulátor kapacitás becslésére. Az általam alapul vett kisautó [23] akkumulátor mérete 9,6 [V] és 2 [Ah].

A jelenlegi, úrben használatos szenzorok energiaigénye nem ismert előttem, így a felvétel készítésének energiaigényét egy [24] típusú digitális fényképezőgép alapján becsültem meg. Ennek a típusú gépnek a maximális akkumulátor kapacitása 3,7 [Wh]. Egy töltéssel 200-250 kép készíthető, ezek alapján 1 kép készítése ~0,0185 [Wh], azaz 66,6 [J]. Mivel a szimulációs programom 10 [s]-os időszelletekkel dolgozik, így a realitás szempontja miatt egy időszak alatt 5 kép készítését vettem alapnak, ami 333 [J] energiafogyasztással jár. A további mérések és a mozgatás energiafogyasztásának becslése is ezen érték alapján történt. A kommunikáció a mérésekhez és a mozgatáshoz képest elhanyagolható energiamennyiséget igényel, így ezt a mérések fogyasztás becslésébe vontam bele. Az alábbi táblázat mutatja a 10 [s]-os időszak alatt szükséges energiamennyiségeket.

Felvétel készítése	333 [J]
Sugárzásmérés	333 [J]
Atmoszféra mérése	
Mágneses erőter mérése	
Kommunikáció	333 [J]
Mozgás	

3.2.6 táblázat

Energiaszükségletek paraméterei 10[ s] alatt egy szenzor esetében

### 3.2.7 Szenzorok energia bevétele

Ahhoz, hogy a szenzorok hosszú távon használhatóak legyenek, szükséges egy a 2.2 fejezetben említett külső energiaforrás, ami az akkumulátor lemerülése esetén újratöltheti azt és így képesség teszi a szenzort a további működésre. Napelemmel felszerelt szenzorok esetén ez a küldő energiaforrás maga a Nap. A Naptól jövő energia bevétel kiszámításához a következő 8. képletet alkalmaztam [25].

$$W[J] = t * \mu * En * A \quad (8)$$

ahol

$\mu$  – hatásfok, amelynek értéke 6-14% között mozoghat [25].

$En$  – napsugárzás energiája, amely értékét (600 [W/m<sup>2</sup>]) [26] Sík Andrástól, az ELTE (Eötvös Loránd Tudományegyetem) adjunktusától kaptam.

$A$  – a napelem felülete. Ennek értéke  $\sim 10\text{cm}^2$ , amely egy általános szenzorok valós paraméter lehet.

$t$  – használati idő. Ez a szimulációs program esetében az előző alfejezetben leírtak alapján  $10[\text{s}]$ .

A könnyebb átláthatóság kedvéért az alábbi 3.2.7 táblázat tartalmazza a fenti értékeket.

Hatásfok	6-14%
Napsugárzás energiája	600 [W/m <sup>2</sup> ]
Napelem felülete	$\sim 10 [\text{cm}^2]$
Használati idő	10 [s]

**3.2.7 táblázat**  
**Energia bevétel paraméterei egy szenzor esetén**

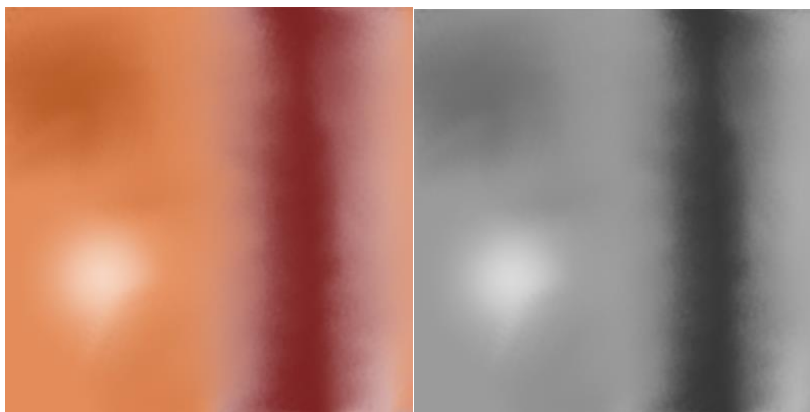
### 3.3 Szimulációs program

Azért, hogy a 3.2 fejezetben bemutatott szenzorhálózatot megfelelően tudjam vizsgálni, egy szimulációs programot fejlesztettem C++ nyelven. A szimulációs programot úgy írtam, hogy minél inkább közelítse egy valós szenzorhálózat működését. Ebben a fejezetben ismertetem nagyvonalakban a program fontosabb részeit és működését.

#### 3.3.1 Térképek alkalmazása

A szimulációm során a vizsgálandó terület adatait térképekből nyertem ki. Két térképet használtam, egy domborzati térképet, és egy talajtérképet. Ezek a térképek határozzák meg a vizsgálandó terület felszíni egyenlőtlenségeit és talajtípusát, amik a szenzorok mozgását befolyásolják.

A domborzati térképen a RGB (Red, Green, Blue) színekkel olvasom ki a felszíni magasságot. A fehér, azaz a (255, 255, 255) RGB kódú pixel jelöli a legalacsonyabb pontot, míg a fekete, a (0, 0, 0) RGB színekkel jelöli a legmagasabb pontot. Ez alapján a pixelek színeit átlagoltam, azaz a térképet szürkeárnyalatúra módosítottam, így egy egységes és mérhető skálát tudtam vizsgálni a magasságértékekre. A térkép felbontása pixel/méterben értendő. Így az adott pixelhez tartozó területrész felszíni magassága azonos lesz. Ilyen térképre példa látható az alábbi 3.3.1.a ábrán.



3.3.1.a ábra

Példa a domborzati térképre

A domborzati jellemzők igen nagymértékben számítanak bele a szenzorok haladási sebességébe. Emelkedőn természetesen lassabban, lejtőn pedig gyorsabban tudnak mozogni.

A talajtérkép szintén RGB színekkel tartalmaz. Itt a barna (185, 122, 87) és szürke (127, 127, 127) színek jelölik a különböző talajtípusokat. A barna szín a homokos talajt szemlélteti,

míg a szürke rész egy apró törmelékes, kavicsos rész ad vissza. Példa a talajtérképre az alábbi 3.3.1.b ábrán látható.



**3.3.1.b ábra**  
**Példa a talajtérképre**

A talajtípus azért fontos meghatározni, mert mindegyik fajtája különböző mértékkel lassítja a szenzorokat. Ezt a szimulátorban egy [0; 1] intervallumba eső szorzóval jelöltem.

### 3.3.2 Paraméterek kezelése

A programot úgy írtam, hogy a vizsgálatoktól függően más és más paraméterek összefüggéseit elemezni tudjam. Ezért azokat a paramétereket, amelyeket később vizsgáltam, globálisként vettem fel, így az értékük könnyen állítható, változtatható. Ez azért szükséges, hogy a szimuláció ne csak egy értékre fusson le, hanem egy egész mintasorozatra is alkalmazható legyen. Ezek a globálisként megadott paraméterek jellemzik a szimulációs környezetet, környezeti jellemzőként viselkednek. A jelentősebb paramétereket a 3.3.2 táblázatban foglaltam össze. Fontos megemlítenem, hogy egy szimulációs vizsgálat alatt többször futtattam le a programot és a futások kiértékelését átlagoltam, hogy a véletlenszerű értékek ne tolják el az eredményeket helytelen irányba.

<b>Paraméter neve</b>	<b>Alapértelmezett érték</b>
Szenzorok száma	100 [db]
Terület	400 [m <sup>2</sup> ]
Elmozdulás maximális értéke	4 [m]
Szenzorok hatótávolsága	30 [m]
D pont távolsága	400 [m]
Pixel/méter	0.5 [pixel/m]
Színváltozás	0.1 [m]

**3.3.2 táblázat**

**Jelentősebb paraméterek a szimulációs programban**

A *szenzorok száma* ( $n$ ) adja meg, hogy összesen hány szenzor alkotja a szenzorhálózatot. Ez a szám magában foglalja a superszenzorokat és a méréső szenzorokat is.

A *terület* ( $w$ ) azt a felszíni részt foglalja magában, amelyet a szenzoroknak fel kell térképezni.

Az *elmozdulás maximális értékével* ( $m$ ) azt a számot határoztam meg, amely egy felső korlátja a mozgásvektor értékének. Azaz egy lépésben maximálisan ekkora lépést tehet meg szenzor.

A *szenzorok hatótávolsága* ( $r$ ) az a távolság, amelyen belül egy szenzor még képes egy másikat érzékelni és így képes a vele való kommunikációra is

A *D pont távolsága* ( $d$ ) pedig azt a távot határozza meg, amelyet el kell érnie legalább egy szenzornak a kiinduló helyzettől, ahhoz, hogy a szimuláció véget érjen.

A *Pixel/méter* változó azt adja meg, hogy a beolvasott térképen egy pixel hány métert jelöl.

A *Színváltozás* változónak pedig a térképek beolvasásánál van nagy jelentősége. Ez adja meg, hogy az RGB színek átlagolásánál egy egységnyi értéknövekedés hány méter magassági különbséget jelöl.

## **4. Szimulációs eredmények**

A szimulációs vizsgálatok során olyan befolyásoló tényezőket kerestem, amelyek hatással vannak a szenzorhálózat energiafogyasztására, és amelyek segítségével megállapíthatók különböző alap összefüggések az energiafogyasztással kapcsolatban.

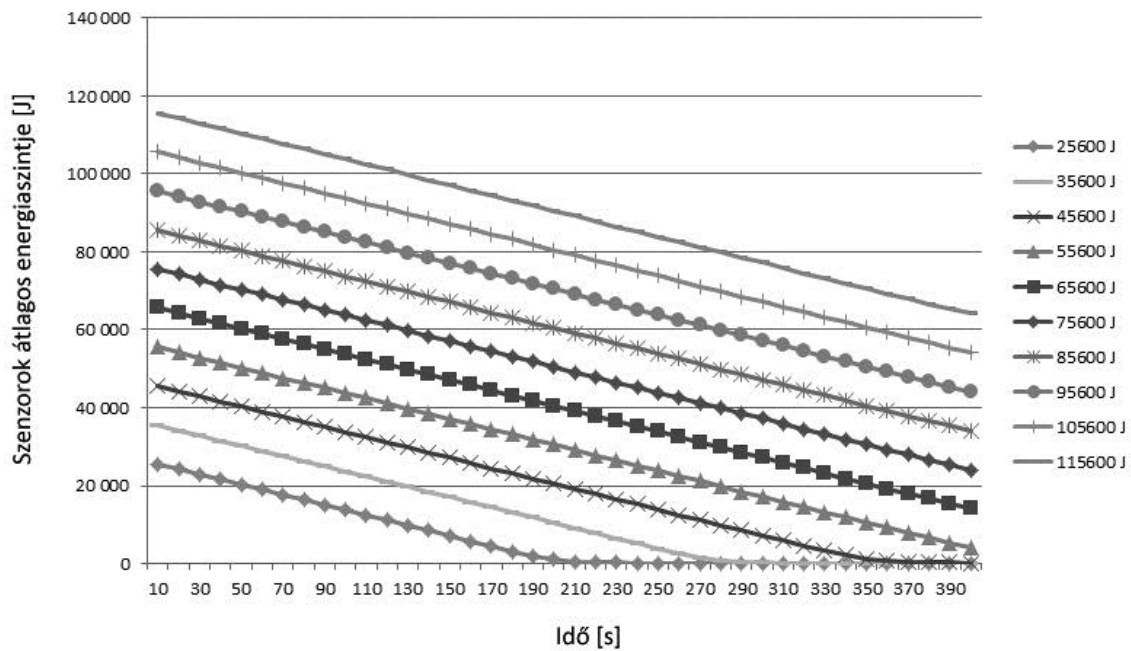
Mivel a szenzorok közötti kommunikáció energiaigénye elenyésző a mérések és a mozgás energiaigényéhez képest, ezért a 3.2.4.2 fejezetben ismertetett három pozicionáló algoritmus között energiahatékonysági szempontból nem volt érdemleges különbség. Az algoritmusok csak a pozicionáláshoz szükséges kommunikáció egy bizonyos részében különböznek, a szimulációs programom ellenben magát a kommunikációt egységesen kezeli. Érdembeli különbség akkor látszódná, ha a kommunikációs folyamatot csomagszintre lebontva vizsgálnám, ami bonyolultságát tekintve nem fér bele ebbe a dolgozatba.

A szimulációim célja, hogy képet kapjunk az energiahatékonyságot befolyásoló paraméterek közötti kapcsolatról, és számszerű becsléseket kapjunk ezen paraméterek hatékony működéshez szükséges arányára.

### **4.1 Akkumulátor kapacitás és az energiaszint kapcsolata**

Elsőként a szenzorok átlagos energiaszintjét vizsgáltam az idő és az akkumulátor méret függvényében. Ebben az esetben a szenzorok maximális akkumulátor kapacitása egy változó értékű paraméter volt, amely értéke 25 600 [J]-tól egészen 115 600 [J]-ig terjedt. A kapott eredmények grafikonját a 4.1.a ábra szemlélteti.



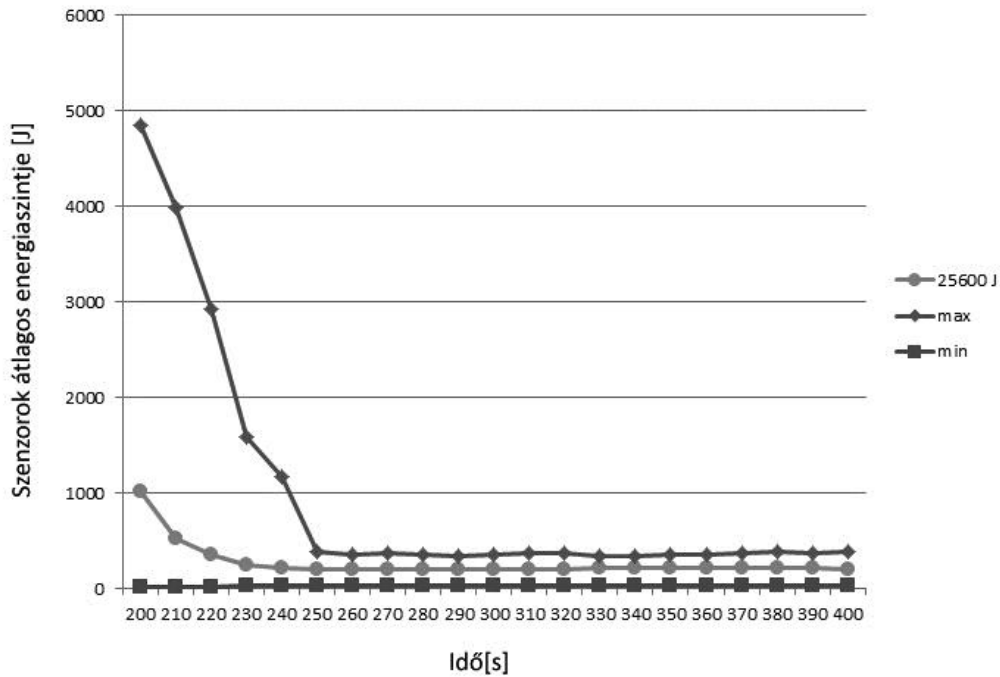


4.1.a ábra

A szenzorok energiaszint változása az idő és a maximális akkumulátor kapacitás függvényében

Az eredményekből megfigyelhető az a triviális összefüggés, hogy minél nagyobb az akkumulátor mérete, annál később merülnek le a szenzorok energiaszintjei. Ez az eredmény egybeesik a vártakkal, a nagyobb akkumulátorral tovább üzemeltethető egyszeri töltéssel a szenzorhálózat, mint kisebb akkumulátorral. De fel kell tennünk azt a kérdést, hogy vajon megéri-e befektetni nagyobb méretű akkumulátorokra 100 darab szenzor esetében, ha rendelkezésre áll minden érzékelőhöz egy napelemes töltőegység. Erre a kérdésre a következő mérési eredményekből kapunk választ többféle szempontból is megvizsgálva.

Kiválasztottam a vizsgált akkumulátor méretek közül a legalacsonyabbat, ez érték mellett vizsgált energiaszint változás látható az idő függvényében a 4.1.b ábrán. Ezen a grafikonon az az érdekes, hogy határozottan látszik, a szenzorok energiaszintje átlagosan nem éri el a nullát, ~200 [J] körül mozog. Látható még a grafikonon a szórás maximális és minimális szintje is.

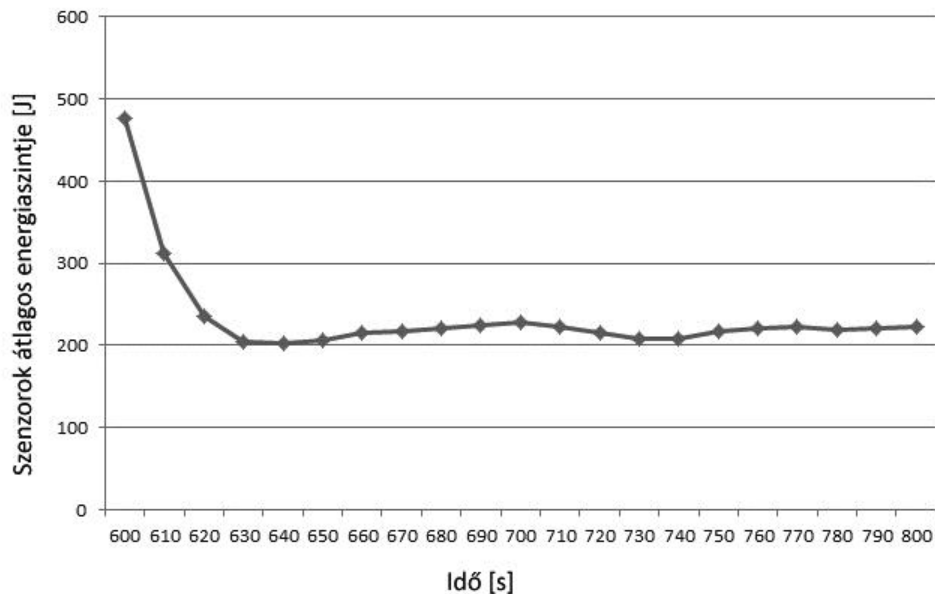


**4.1.b ábra**

**Szenzorok átlagos energiaszintjének változása az idő függvényében 25 600 [J] akkumulátor kapacitás esetén**

A 4.1.a és a 4.1.b ábrán is megfigyelhető, hogy egy bizonyos idő után a szenzorok energiaszintje látszólag megmarad egy adott szinten (~200 [J]), kisebb-nagyobb eltérésekkel. Ez azért lehetséges, mert a szenzorok energiaszintje már nem éri el a 3.2.6 táblázatban megadott mérésekhez, illetve mozgáshoz szükséges energiaszintet. Valójában az energiaérték nem stagnál ezen ponton túl, hanem növekszik a napelemes töltés segítségével, csak ez olyan lassan történik, hogy ez a grafikonon már nem észlelhető.

A töltés által növekedett energiamennyiséget szemlélteti a következő 4.1.c grafikon, amelyen látható egy lassú, de észlelhető energiaszint növekedés.

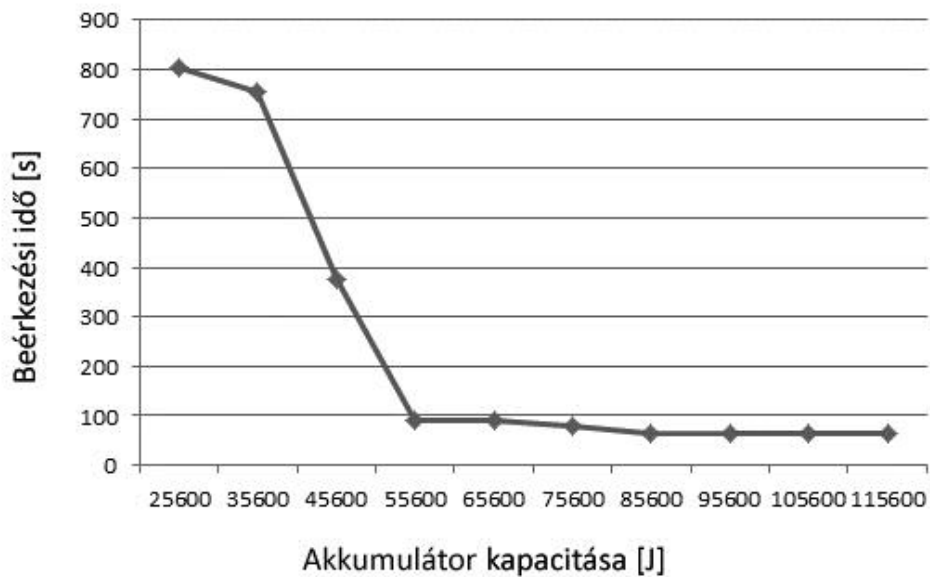


**4.1.c ábra**

**A szenzorok átlagos energiaszintjének változása az idő függvényében 75 600 [J] akkumulátor kapacitással**

Ezek alapján megállapítható akkumulátor méretenként egy szakaszidő, amely azt mutatja meg, hogy egy teljesen feltöltött akkumulátorral mennyi idő alatt merülnek le a szenzorok olyan energiaszintre, ahol már nem képesek se mozgásra, se pedig mérések elvégzésére. Ez leolvasható a 4.1.c ábráról, értéke  $\sim 610$  [s]. Ezzel párhuzamosan egy másik szakaszidő (2. szakaszidő) is meghatározható, amely azt jelöli, hogy mennyi idő kell ebből a lemerült állapotból ahhoz, hogy újra mozgó, ill. mérőképes legyen egy szenzor. Ez egyszerű arányossággal kiszámolható. 10 [s] alatt átlagosan 4,5 [J] energia bevétel termelődik, akkor a szükséges 999 [J] (mérésekhez és mozgáshoz szükséges energia) 2220 [s] alatt lesz elérhető. E két szakaszidő együttesen kiad egy mérési periódust, amely meghatározza, hogy a szenzorboly adott paraméterek mellett milyen időközönként kerül újra megfelelő energiaszintű állapotba. Ahhoz, hogy rövidebb idő alatt termelődjön vissza a szükséges energia, vagy az energiafogyasztásokat kell csökkenteni, vagy pedig a napelem befogadó képességét kell növelni. Utóbbi elérhető akár a napelem méretének növelésével is. Az átláthatóság kedvéért a továbbiakban maradok a kiinduló értékeknél.

Ezek mellett vizsgáltam azt is, hogy különböző akkumulátor méretek mellett, mennyi idő alatt jár le a szimuláció, azaz mennyi idő kell ahhoz, hogy egy szenzor beérjen a célpontba. Ennek eredményét a 4.1.d ábra és a 4.1.a táblázat szemlélteti.



4.1.d ábra

A beérkezési idő változása az akkumulátor kapacitásának függvényében

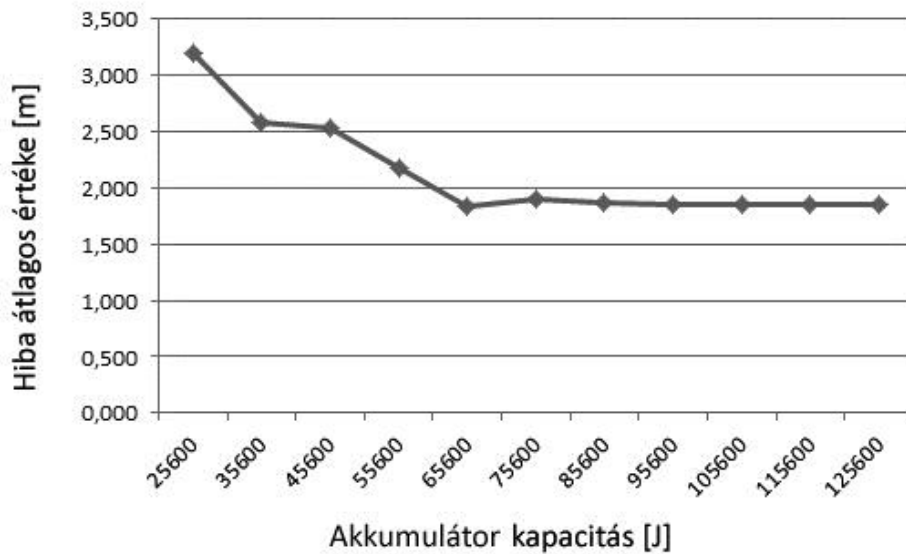
25600[J]	35600[J]	45600[J]	55600[J]	65600[J]	75600[J]	85600[J]	95600[J]	105600[J]	115600[J]
804[s]	753[s]	377[s]	91[s]	89[s]	80[s]	65[s]	65[s]	65[s]	65[s]

4.1.a táblázat

Beérkezési idő az akkumulátor kapacitásának függvényében

Láthatjuk, hogy egy adott akkumulátor méretig (~55 600 [J]) nagy lépéskben csökken a beérkezési idő értéke, majd ezen érték után még megfigyelhető egy kismértékű csökkenő tendencia, de ~85 600 [J] után már stagnálnak az értékek. Ez azért lehetséges, mert ekkora akkumulátor kapacitás felett már a szenzorok a futás során nem érik el azt az értéket, ahol már mozgás- és mérésképtelenek, hanem még ez idő előtt beérnek a célba. Tehát megállapítható, hogy ebből a szempontból nem érdemes 85 600 [J] érténél nagyobb akkumulátor kapacitást választanunk, mert 85 600 [J] értékkel is ugyanazt a hatékonyságot érnük el.

Felső becslést már kaptunk az akkumulátor kapacitásra, de mi az az érték, aminél mindenképpen érdemes nagyobb választani bizonyos szempontból? Ennek a kérdésnek a megválaszolására végeztem el azt a szimulációs vizsgálatot, ahol a pozicionálásból adódó hibaérték változását követtem végig az akkumulátor kapacitás függvényében. Hibaérték alatt a pozicionáló algoritmus által kiszámolt (becsült) és a valós koordináták közötti eltérést értem. Ennek eredményét a 4.1.e grafikon szemlélteti.

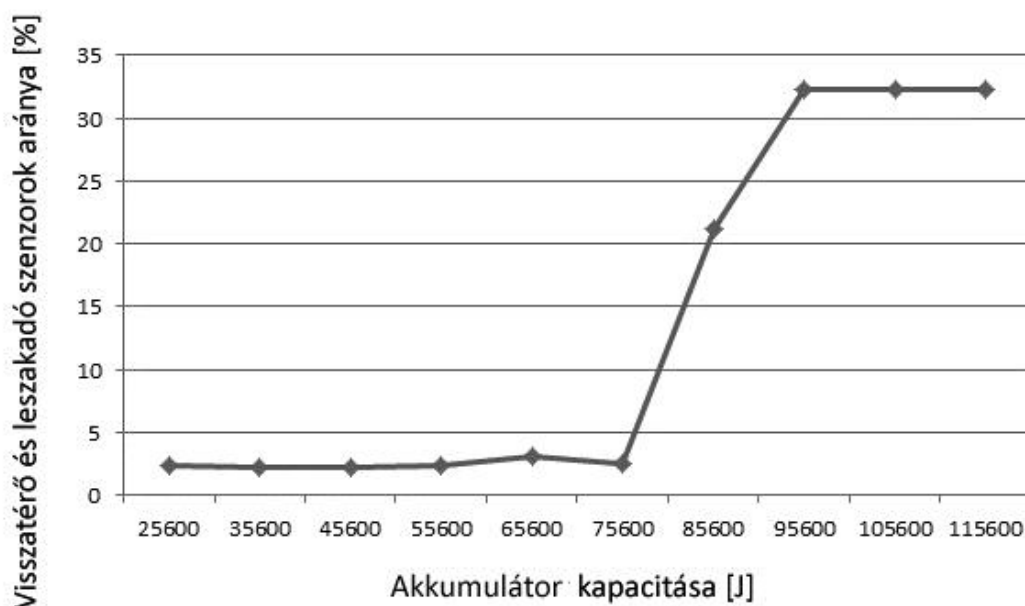


4.1.e ábra

Hiba átlagos értékének változása az akkumulátor kapacitás függvényében

Megfigyelhető, hogy 65 600[J] feletti kapacitás értékekre egy viszonylag állandó hibaérték jellemző, kisebb értékekre azonban nagyobb a hiba mértéke. Ennek oka az, hogy kisebb akkumulátor kapacitás esetén hamarabb merülnek le a szenzorok olyan energiaszintre, amely már nem elég a kommunikációra sem, így nagyobb a lehetősége, hogy a kevesebb hibával rendelkező szenzorok nem állnak rendelkezésre. Ez tehát azt mutatja, hogy ebből a szempontból vizsgálva érdemes 65 600 [J] feletti akkumulátor kapacitás értéket választani a megadott paraméterek mellett.

Utolsó akkumulátor kapacitással kapcsolatos vizsgálatom a visszatérő és a leszakadó szenzorok arányszámának változása az akkumulátor kapacitás függvényében. Visszatérő szenzoroknak nevezem azokat, amelyek egy ideig lemerült állapotban voltak úgy, hogy közben referencia pontokként sem voltak használhatók a túl alacsony energiaszint miatt, majd később feltöltődtek olyan szintre, hogy képesek legyenek részt venni a pozicionálásban. Ennek eredményét a 4.1.f grafikon szemlélteti.



4.1.f ábra

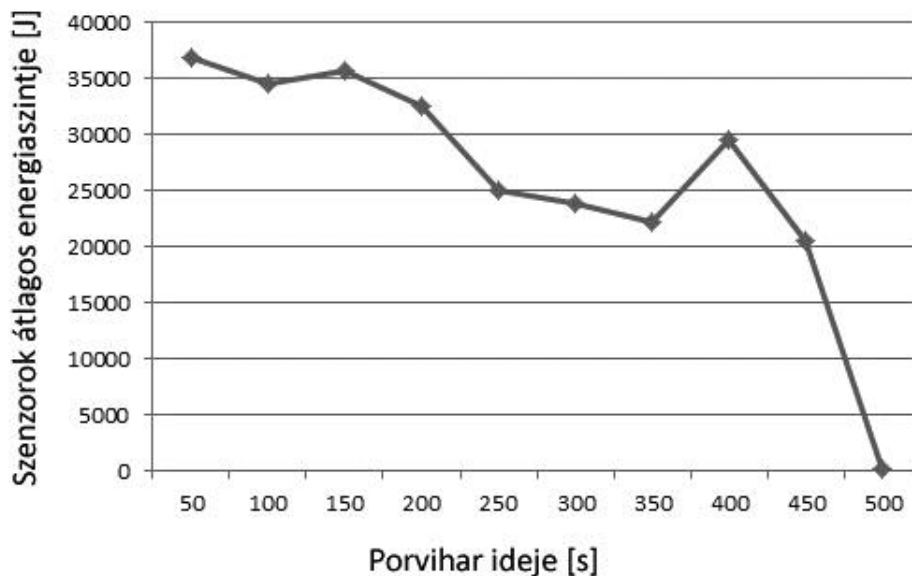
Visszatérő és leszakadó szenzorok arányának változása az akkumulátor kapacitás függvényében

Megfigyelhető egy határérték (~ 75 600 [J]), amelynél kisebb értékekre kisebb ez az arányszám. Azaz a visszatérő szenzorok száma / leszakadó szenzorok száma kicsi. Tehát a leszakadó szenzorok száma a visszatérőkhöz képes nagy ezekben az esetekben. Az ennél magasabb értékekre ellenben jóval nagyobb ez az arányszám. Mivel nagyobb az akkumulátor méret, ezért a pozicionálás is pontosabb, kevesebb lehetőség adódik a szenzoroknak leszakadni a bolytól. ~95 600 [J]-től felfelé pedig egy stagnálás figyelhető meg. Így ebből a vizsgálati szempontból egy alsó (~ 75 600 [J]) és egy felső (~95 600 [J]) becsült korlát is megadható az akkumulátor méret kapacitására.)

## 4.2 Porvihar, mint befolyásoló tényező

A következő nagyobb vizsgálati téma a porvihar befolyásoló hatása a szenzorhálózat energiafogyasztására. Porvihar ideje alatt a szenzorok nem képesek kommunikálni egymással, így a pozicionálás is lehetetlenné válik erre az időre tekintve. Azonban a mozgásuk megmarad, így könnyebben eltávolodhatnak egymástól, átléphetik a biztonsági sávhatárt is észrevétlenül. A porvihar másik jelentős hatása, hogy nem működik a napelemes töltés, így ez közvetlenül érinti a hálózat energiahatékonyságát.

Első megfigyelésem ebben a témában az akkumulátor átlagos energiaszintje a homokvihar idejének függvényében. Az eredményeket a 4.2.a grafikon szemlélteti.

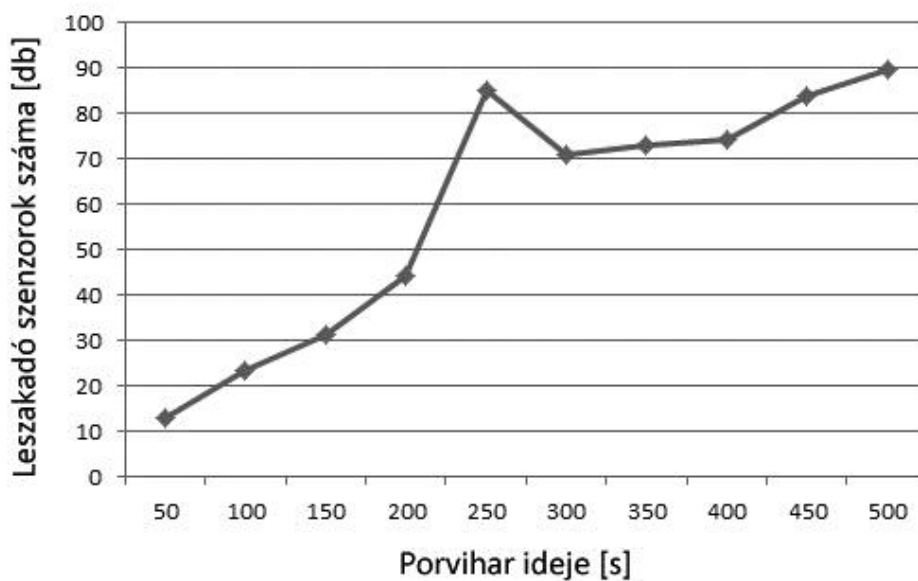


4.2.a ábra

Szenzorok átlagos energiaszintje a porvihar idejének függvényében

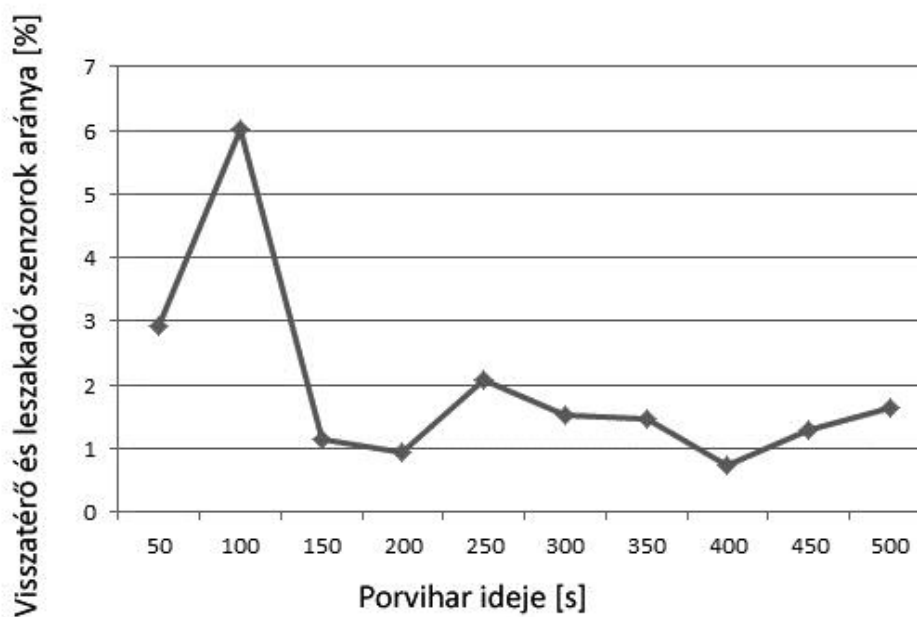
A kilengésektől eltekintve egy csökkentő tendencia figyelhető meg az átlagos energiaszint és a homokvihar idejének kapcsolatában. Minél hosszabb ideig tart a homokvihar, annál kisebb lesz az átlagos energiaszint. Ez az eredmény egybeesik a várttal, ez bizonyítja, hogy hiába kicsi a napelem töltési képessége a fogyasztáshoz képes, azért hiánya jelentős energiavesztéssel jár.

Következő két szimulációm szorosan összefügg. Az első a lezakadó szenzorok számának változását vizsgálja a homokvihar hosszának függvényében - az eredmény a 4.2.b ábrán látható. A második pedig a visszatérő és lezakadó szenzorok arányának változását mutatja szintén a homokvihar hosszának függvényében - ennek eredményét a 4.2.c ábra szemlélteti.



4.2.b ábra

Leszakadó szenzorok számának változása a porvihar idejének függvényében



4.2.c ábra

Visszatérő és leszakadó szenzorok arányának változása a porvihar idejének függvényében

Megfigyelhető, hogy a várakozásnak megfelelően minél hosszabb ideig tart a homokvihar, annál több szenzor szakad le a bolytól. Ezt a megfigyelést összevetve a 4.2.c grafikonnal, látható, hogy a leszakadó szenzorok számának növekedése nem esik pontosan egybe az



arányszám csökkenésével. Ebből arra lehet következtetni, hogy a porvihar idejének növekedése nem csak a leszakadó szenzorok számát növeli, de a visszatérő szenzorok számát is befolyásolja. Nyilvánvalóan homokvihar alatt nem lehet visszatérő szenzor, hiszen töltés sincs ekkor a rendszerben, csak a porvihar ideje előtt és után. Ez a megfigyelés megerősíti azt a tényt, hogy ha hosszabb idejű porviharral kell számolnunk a 3.1-es fejezetben megadott rendszer esetében, akkor az akkumulátor kapacitás értékének megválasztásába bele kell kalkulálnunk az energia bevétel jelentős kiesését.

### **4.3 Konklúzió**

A bemutatott szimulációs vizsgálatok eredményeképpen kaptunk közelítő alsó és felső becslést a szenzorok akkumulátor kapacitás értékére. Több szempontból vizsgálva ezek az értékek eltérhetnek, de a becslés szerint  $\sim 65\,600$  [J] és  $\sim 95\,600$  [J] közötti érték az, ami az adott paramétereknek megfelelően hatékony működést produkál. Azonban mivel a marsi környezetre jellemző porvihar több irányból is befolyásolja a hálózatunk működését, így figyelembe kell venni, hogy az energia bevétel egy időre kiesik a rendszerből. Ezért érdemesebb a kapacitás érték felső becslét közelítő mennyiséget választani.

## 5. Összefoglalás

A távoli égitestek feltérképezése egyre nagyobb szerepet kap nemcsak a nemzetközi tekintetben, hanem hazánkban is. A gyors fejlődés maga után vonja az űreszközök fejlesztését is, ami egyre részletgazdagabb és pontosabb eredményeket produkál. Ezek inspiráltak arra, hogy kifejtssem a szenzorhálózatok alkalmazhatóságát egy ilyen feladaton belül. Kialakítottam egy olyan (elméleti) rendszert, amely egy földi irányító központból, egy keringő kommunikációs egységből (műhold) és egy távoli égitesten elhelyezett mozgó szenzorhálózatból áll. Koncentráltan a marsi környezettel foglalkoztam, figyelembe vettem a szenzorhálózat működését befolyásoló, a Marsra jellemző tényezőket (jellemző felszíni formák, mint a dűne és a kráter, különböző talajtípusok, porvihar), majd a rendszert egy szimulációs programba építettem. Munkám központi eleme a szenzorhálózat energiahatékonyságának, energiafogyasztásának vizsgálata volt. A különböző energiafogyasztások (mérések, kommunikáció és mozgás) és energia bevételek (napelemes töltés) különböző arányait és változásait vizsgáltam a szimulációk során különböző paraméterekre nézve és a szimulációk eredményeképpen becslést tudtam adni a szenzorok szükséges akkumulátor szintjének méretére a hatékony működéshez.

Munkámban igyekeztem arra törekedni, hogy egy adott aktuális problémát minél részletesebb és hatékonyabb vizsgálatokkal közelítsek meg. Eredményeim megmutatták, a fontosabb összefüggéseket az adott rendszeren belül és a különböző paraméterekhez viszonyított akkumulátor kapacitások becsült hatékonyságát.

## Táblázat és ábrajegyzék

- 2.1.a ábra: A Holdről alkotott ismereteink régen és napjainkban
- 2.1.b ábra: A Mars felszíne egy 2012. szeptember 28-án készült felvételen
- 3.1 ábra: A feltételezett rendszer szemléltetése
- 3.2.2 ábra: A szenzorok mozgásának szemléltetése
- 3.2.3.a ábra: Fresnel-zóna
- 3.2.3.b ábra: Deygout-modell
- 3.2.4.1 ábra: Háromszögelés szemléltetése
- 3.2.4.2.a ábra: A BPA algoritmus szemléltetése
- 3.2.4.2.b ábra: A PAHN algoritmus szemléltetése
- 3.2.4.2.c ábra: Az EBPA algoritmus szemléltetése
- 3.2.5 ábra: Szenzorok általános állapotdiagramja
- 3.2.6 táblázat: Energiaszükségletek paraméterei 10 [ s ] alatt egy szenzor esetében
- 3.2.7 táblázat: Energia bevétel paraméterei egy szenzor esetén
- 3.3.1.a ábra: Példa a domborzati térképre
- 3.3.1.b ábra: Példa a talajtérképre
- 3.3.2 táblázat: Jelentősebb paraméterek a szimulációs programban
- 4.1.a ábra: A szenzorok energiaszint változása az idő és a maximális akkumulátor kapacitás függvényében
- 4.1.b ábra: Szenzorok átlagos energiaszintjének változása az idő függvényében 25 600 [J] akkumulátor kapacitás esetén
- 4.1.c ábra: A szenzorok átlagos energiaszintjének változása az idő függvényében 75 600 [J] akkumulátor kapacitással
- 4.1.d ábra: A beérkezési idő változása az akkumulátor kapacitásának függvényében
- 4.1.a táblázat: Beérkezési idő az akkumulátor kapacitásának függvényében
- 4.1.e ábra: Hiba átlagos értékének változása az akkumulátor kapacitás függvényében
- 4.1.f ábra: Visszatérő és leszakadó szenzorok arányának változása az akkumulátor kapacitás függvényében

4.2.a ábra: Szenzorok átlagos energiaszintje a porvihar idejének függvényében

4.2.b ábra: Leszakadó szenzorok számának változása a porvihar idejének függvényében

4.2.c ábra: Visszatérő és leszakadó szenzorok arányának változása a porvihar idejének függvényében

## Irodalomjegyzék

- [1] Szeile Aliz, Bacsárdi László, Huszák Árpád, "Pozicionálás szenzorhálózatokkal egy naprendszerbeli égitesten". In: Dr Frey Sándor (szerk.) Űrtan évkönyv 2012. Budapest: Magyar Asztronautikai Társaság, pp. 53-60.
- [2] Aliz Szeile, Laszlo Bacsardi, Arpad Huszak, "Analyzing Sensor Based Positioning on the Surface of a Distant Planet", In: Proc. of 64th International Astronautical Congress. Beijing, China, 2013, pp. 1-7. Paper IAC-13-B2.2.12.
- [3] Aliz Szeile, Arpad Huszak, Laszlo Bacsardi, "Advanced Sensor Based Positioning and Monitoring System", In: Proc. of Global Space Application Conference (GLAC). Paris, France, 2014., pp. 1-6. Paper GLAC-14,S4A,4x20428.
- [4] Bacsárdi L, Huszák Á, Szeile A, Haidegger T (2014) Novel Methods for Multi-hop Sensor Positioning on the Surface of a Solar System Body. J Aeronaut Aerospace Eng 3: 131. doi:10.4172/2168-9792.1000131
- [5] Joseph A. Burns: „The four hundred years of planetary science since Galileo and Kepler”, Nature, 2010. július 29.
- [6] „Rocknest” From Sol 52 Location, NASA/JPL-Caltech/MSSS, <http://www.nasa.gov>, (Utolsó látogatás: 2014. október 18.)
- [7] European Southern Observatory: 2010. augusztus 24-ei cikke, <http://www.eso.org>, (Utolsó látogatás: 2014. október 18.)
- [8] Vincze Zoltán, Vida Rolland, „Mobil eszközök alkalmazása szenzorhálózatokban”, Híradástechnika LXI/12, pp. 13-17.
- [9] R. C. Shah, S. Roy, S. Jain, and W. Brunette: „Data MULEs: Modeling a three-tier architecture for sparse sensor networks”. In Proc., IEEE Workshop on Sensor Network Protocols and Applications (SNPA), pp. 30-41, Anchorage, Alaska, USA, May 2003.
- [10] L. Tong, Q. Zhao, and S. Adireddy: „Sensor networks with mobile agents”, in MILCOM 2003 - IEEE Military Communications Conference, Vol. 22, No. 1, pp. 688–693, Boston, MA, Oct. 2003.
- [11] A. Chakrabarti, A. Sabharwal, and B. Aazhang: "Using predictable observer mobility for power efficient design of sensor networks", in Proceedings of IPSN'03, Second International Workshop on Information Processing in Sensor Networks, pp. 129–145, Palo Alto, CA, USA, Apr. 2003.
- [12] J. Luo and J.-P. Hubaux: „Joint mobility and routing for lifetime elongation in wireless sensor networks”, in Proceedings of IEEE INFOCOM'05, Mar. 2005.
- [13] A. Kansal, M. Rahimi, W.J. Kaiser, M.B. Srivastava, G.J. Pottie, and D. Estrin: „Controlled mobility for sustainable wireless networks”, in IEEE Sensor and Ad Hoc Communications and Networks (SECON), Santa Clara, CA, Oct. 2004.

- [14] A. Kansal, A. Somasundara, D. Jea, M.B. Srivastava, and D. Estrin: „Intelligent fluid infrastructure for embedded networks”, in ACM MobiSys’04, pp. 111–124, Boston, MA, USA, June 2004.
- [15] Bandyopadhyay, S.; Coyle, E.J.: „An energy efficient hierarchical clustering algorithm for wireless sensor networks”, INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, 2003
- [16] Seung Jun Baek; de Veciana, G.; Su, X.: „Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation”, Selected Areas in Communications, IEEE Journal on, 2004
- [17] Duarte-Melo, E.J.; Mingyan Liu: „Analysis of energy consumption and lifetime of heterogeneous wireless sensor networks”, Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE, 2002
- [18] Chou, J.; Petrovic, D.; Kannan Ramachandran: „A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks”, INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, 2003
- [19] G. Wang, G. Cao, and T. Porta: „Movement-assisted sensor deployment”, in Proceedings of. IEEE INFOCOM’04, Hong Kong, 2004.
- [20] F. Aurenhammer: „Voronoi Diagrams — A Survey of a Fundamental Geometric Data Structure”, ACM Computing Surveys (CSUR), 1991
- [21] Z. Butler and D. Rus: „Event-based Motion Control for Mobile Sensor Networks”, IEEE Pervasive Computing, Vol. 2, No. 4, pp. 34–42, Oct.-Nov. 2003.
- [22] Zafer, Sahinoglu, Sinan Gezici, Ismail Güvenc: „Ultra-wideband Positioning Systems”, Cambridge University Press, 2008
- [23] <http://www.tenergy.com>, *Utolsó látogatás: 2014.10.22*
- [24] <http://www.canon.hu>, *Utolsó látogatás: 2014.10.19*
- [25] Kovács Csaba: „Napelemek működése és alkalmazása”, Óbudai Egyetem, Kandó Kálmán Villamosmérnöki Kar, Megújuló energiaforrás Kutató Hely, 2008.
- [26] Sík András, Kereszturi Ákos, Hargitai Henrik: „A víz és a jég szerepe a Mars felszínfejlődésében”, Földrajzi Közlemények CXXIX (I.IV.) kötet, 2005