



Horváth Tamás

Láthatósági vizsgálat elérhető digitális adatokkal

c. diplomamunka

Infrastruktúra Építőmérnök Msc
Közlekedésépítő Szakirány

Széchenyi István Egyetem
Műszaki Tudományi Kar
Közlekedésépítési Tanszék

Győr, 2013

Tartalomjegyzék

1	Bevezetés.....	4
2	Közúti biztonság és látótávolságok.....	6
2.1	Közúti biztonság	6
2.2	Látótávolság fogalmak	7
2.2.1	Megállási látótávolság.....	8
2.2.2	Előzési látótávolság	8
3	Digitális domborzat és az elérhető adatok.....	10
3.1	Digitális domborzatmodellezés	10
3.2	Digitális modellek Magyarországon.....	12
3.2.1	FÖMI DDM.....	13
3.2.2	Katonai DDM	13
3.2.3	Google SRTM.....	14
3.2.4	ASTER GDEM.....	14
4	GPS technika	16
4.1	A GPS mérésről általában	16
4.2	Folyamatos kinematikus mérés.....	17
4.3	Félkinematikus GPS mérés.....	17
5	AutoCAD Civil 3D	19
6	Modellezés GIS adatok felhasználásával	21
6.1	ESRI ArcGIS 10	21
6.2	Előkészítő lépések.....	22
6.3	Modellezési lehetőségek	23
6.4	Mintaszakasz bemutatása	25
7	Légvonalas láthatósági vizsgálat	27
7.1	Folyamatábra a Model Builderben.....	27
7.2	A légvonalas vizsgálat folyamatábrája.....	28

7.2.1	Adatok beolvasása	29
7.2.2	A domborzatmodellről a látóvonalakig	30
7.2.3	Láthatósági vonalak	33
8	Láthatósági vizsgálat az útpálya mentén	36
8.1	Bevezető lépések	36
8.2	Python programozás	39
8.2.1	A Python programozásról általában	39
8.2.2	FOR ciklus a vizsgálatnál	41
8.2.3	Láthatósági animáció	44
8.2.4	Adatok kivitele Excel-táblázatba	46
9	Excel vizsgálat	47
9.1	A Microsoft Excel határai	48
9.2	Excel programozás	49
9.2.1	Kettős keresés	50
9.2.2	Maximális látótávolság meghatározása	54
9.2.3	Dinamikus diagram készítése	57
9.3	AutoCAD hossz-szelvény Excel adatokból	60
9.4	A vonalvezetés és a látótávolságok összehasonlítása	60
10	Tapasztalatok, jövőbeni tervek	66
10.1	Eredmények értékelése	66
10.2	Jövőbeni vizsgálati lehetőségek	67
10.2.1	Szemmagasság változtatásának lehetősége	67
10.2.2	3D objektumok figyelembe vétele	67
11	Összegzés	70
12	Irodalomjegyzék	72
13	Ábrajegyzék	75
14	Táblázatjegyzék	76
15	Mellékletek	77

1 Bevezetés

Az emberi agy csaknem harmada a látást szolgálja, ezáltal a vezetés során a legfontosabb érzékszervünk a szemünk. Az úthálózat fejlesztések során az egyik legfontosabb kritérium követelmény, hogy minél jobb belátást, láthatóságot tudjunk biztosítani a közlekedők számára. Nem csupán egy, még a tervezés folyamatában lévő útnál fontos a láthatóság követelménye, hanem egy már használatban lévő útszakasz esetén is. A megfelelő látótávolságok kialakítása ennek okán elengedhetetlen. Nem ideális látási viszonyok esetén egy-egy balesetnek a bekövetkezése nagyobb valószínűséggel történik majd meg, hiszen nincsenek biztosítva az előzésre-, vagy a megállásra vonatkozó szabályok.

Az útszakaszok tervezése során nagy gondot kell fordítanunk a megfelelő magassági-, illetve vízszintes vonalvezetésre. Ezek az adatok alapjaiban határozzák meg egy út látási viszonyait. Az utóbbi években, Európában és a világ számos más táján felvetődött a kérdés, hogyan lehetne a balesetek számát egyre inkább csökkenteni. Az adatok jelenleg csökkenő tendenciát mutatnak [*KSH 2010; ORFK, 2011; ORFK, 2012*]. E tendencia szoros összefüggésben áll azzal a törekvéssel - hogy felismerve a balesetek számának alakulását az évek során - igyekszünk egyre inkább biztonságosabbá tenni a hazai utakat. A biztonságosabbá tételt meglévő útszakaszoknál természetesen a felülvizsgálat és a későbbi korrekció jelenti.

Dolgozatomban egy, jelenleg már meglévő főút kiválasztott szakaszát vizsgálom a beláthatóság kérdéskörében. A vizsgálat fő tárgyát képezi, hogy mennyire és milyen minőségben használhatók meglévő digitális adatok ilyenfajta értékelésre. Magyarországon és egyúttal az egész világon többféle, különböző pontosságú és eltérő lefedettséget biztosító információ érhető el. A FÖMI¹ DDM-től² kezdve a Google Earth program digitális domborzatmodelljéig bezárólag. Attól függően, hogy mekkora helyszínre terjed ki a vizsgálat, vagy milyen pontossági igényekre van szükség, az alkalmazott modellek mások és mások lehetnek. A hangsúly nem minden esetben az elérhető adatok pontosságán van. Az egyik legfőbb kritérium lehet akár az ár is. Vizsgálatom középpontjában éppen ezért a felhasználók számára ingyenesen letölthető Google domborzat áll.

¹ Földmérési és Távérzékelési Intézet

² Digitális Domborzat Modell

A kérdés pontosan az, hogy ezek az adatok mennyire használhatók például egy előzési látótávolság értékeléséhez. A látótávolságok elemzésénél többféle módszert alkalmazok a mintaterületen, melyek alapján értékelem az adatok pontosságát. Az elemzések modellezése mindes esetben GIS (földrajzi) adatok felhasználásával történik. Annak érdekében, hogy ne kelljen műveletek sokaságát egymás után többször is elvégezni külön-külön, az automatizálás lehetősége tűnt a legjobb megoldásnak. Az automatizálása így szintén a kutatás tárgyát képezi.

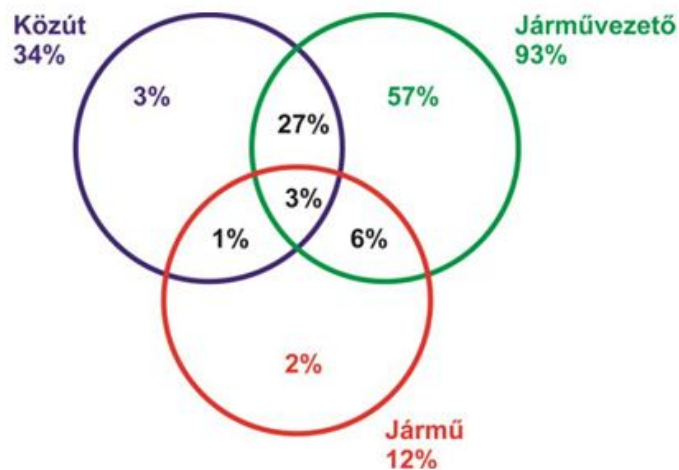
2 Közúti biztonság és látótávolságok

A bevezetőben szó volt általánosságban a közúti biztonság fontosságáról, e fejezetben szeretném ezt egy kicsit részletesebben kifejtetni.

Az 1800-as évek végén, amikor megjelentek az első gépkocsik, a feltalálók még nem gondolkodtak tömeges gyártásban. A gyártósorok megjelenésével azonban az autó a nagyközönség számára is elérhetővé vált. Az infrastruktúra fejlődésével a világ technikailag egyre magasabb fokozatokba lépett. A fejlődés egyre gyorsabb volta vonta maga után, hogy a gépjárművek mind tökéletesebbek, és erősebbek lettek. E tökéletesedés komoly veszélyt jelentett azonban a többi, sérülékenyebb közlekedők számára. Úgyszólván a gépkocsipark fejlődésével összhangban az autók által okozott balesetek is egyre súlyosabbak és gyakoribbak lettek.

2.1 Közúti biztonság

A világ számos országa felismerte, hogy a baleseti statisztikák az 1980-as, '90-es évektől kezdődően riasztó képet festettek, ezért egyre inkább a biztonságosabb megoldások kerültek előtérbe. Kidolgozásuk érdekében felül kellett vizsgálni a már meglévő útszakaszokat, és meghatározni a balesetek kiváltó okait. A magyarázatot a szakirodalom három különböző, egymással szoros kapcsolatban álló tényezőhöz kapcsolja. E tényezőket hivatott bemutatni a következő ábra.



2-1. ábra Kiváltó okok részvétele a balesetekben [Koren, 2013]

Ahogy az ábráról is jól látható, a balesetek bekövetkezésében legnagyobb szerepet a járművek vezetői játsszák. Az utak biztonságosabbá tételét szolgáló célkitűzések éppen ezért elsősorban politikailag bevezetett szigorításokat jelentettek és jelentenek manapság

is. E szigorítások általános esetben az autóvezetőkre vonatkoznak. Megemlíthetnénk például a zéró toleranciát, vagy a pontrendszer teljes körű szigorítását. A szankciók az egyes vétségekkel szemben is sokkal komolyabbak lettek. Persze nem minden baleset okozója kizárólag a járművezető figyelmetlenségére, így ennek okán, az ábrán nem különálló, hanem egymással metsződő helyzetű diagramokat találunk. *[Koren, 2012]*

Látható, hogy a balesetek harmada az út műszaki adatai és állapota miatt következik be. Megfelelő útkonstrukcióval és felülvizsgálattal ezek azonban csökkenthetők. Az európai országok többsége nem csupán a járművezetők felelősségre vonásával és szigorúbb szabályokkal kívánja megoldani a halálozási ráta csökkenését, hanem megfelelő műszaki elvek bevezetésével is. Műszakilag tehát az utak magassági-, és vízszintes nyomvonalvezetését úgy kell megoldani, hogy az minél inkább a biztonságos közlekedést szolgálja.

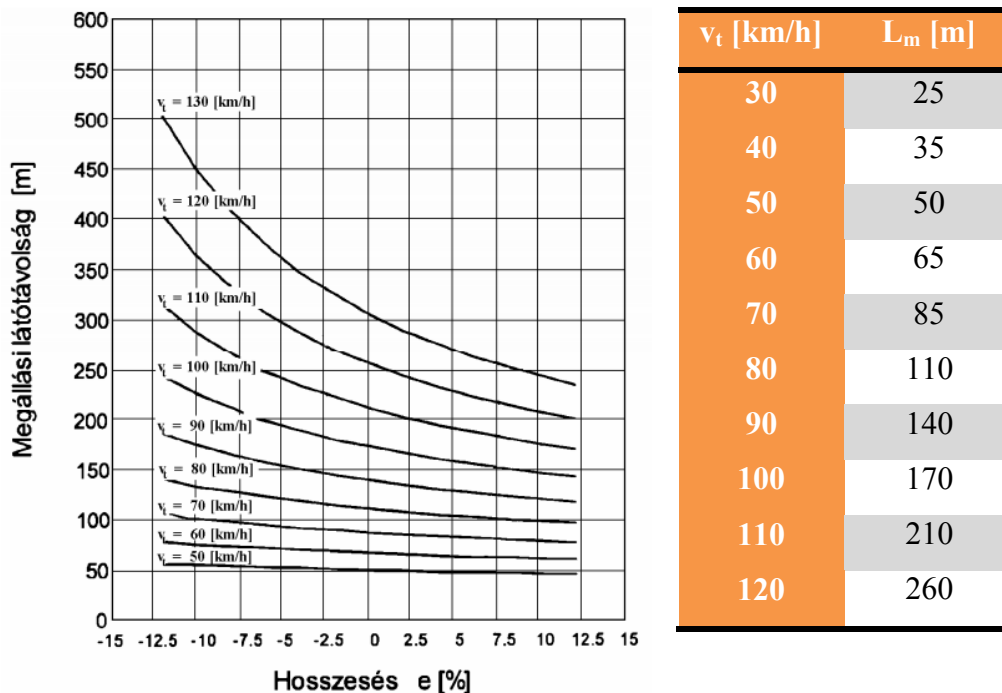
A vonalvezetés általános követelményeihez tartozik, hogy az útnak egyértelmű képet kell nyújtania. A járművezetőknek a haladás során folyamatosan érzékelniük kell a körülöttük lévő világot. Ha egy tájba nem illő úton haladnak, akkor nagyobb mértékben vonhatja el a figyelmüket – a többi közlekedőről – az út környezete. További fontos szempontot jelent a koncentráció szinten tartására az út esztétikai kialakítása is, mely, ha nem megfelelő, szintén megterhelő lehet. Tehát az egyes szakaszoknak nem csupán egyértelműnek kell lenniük, de esztétikailag is illeszkedniük kell a tájba. *[MAÚT, 2008]*

2.2 Látótávolság fogalmak

A legmegfelelőbb biztonsági megoldások érdekében az úton nem csak az autósok, de minden közlekedő számára ideális láthatósági viszonyokat kell teremtenünk. Ideális távolságok esetén, a résztvevők képesek időben észrevenni az úton közlekedő másik járművet, kerékpárost, vagy gyalogost. Ahhoz, hogy minden szempontból korrekt belátásunk legyen az útszakaszon a következő látótávolságoknak kell teljesülnie.

2.2.1 Megállási látótávolság

A megállási látótávolságnak a balesetek elkerülésében rendkívül nagy szerepe van. Az út vonalvezetésének mind vízszintes-, mind pedig magassági értelemben olyannak kell lennie, hogy a vezető még biztonságosan, féktávolságon belül meg tudjon állni. A *Közutak tervezése ÚT 2-1.201*-es műszaki előírás, a megállási látótávolságot 1,0 m-es szemmagasság, és 0,1 m-es akadály esetére fogalmazza meg. A következő ábrán a különböző sebességekre vonatkozó látótávolság értékek figyelhetők meg.



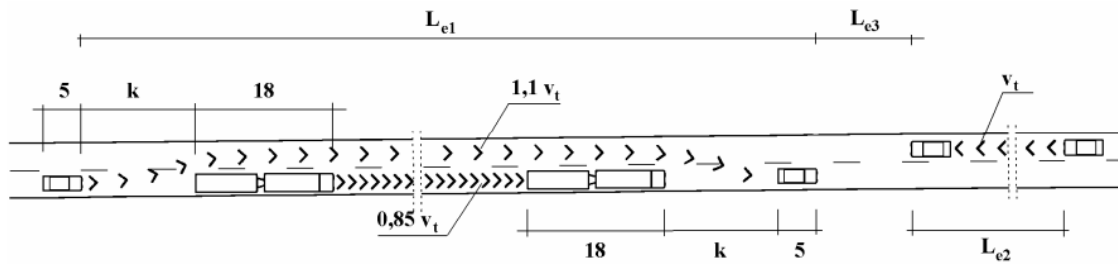
2-2. ábra Megállási látótávolságok a sebesség és hosszsesés függvényében [MAÚT, 2008; Schuchmann 2010]

2.2.2 Előzési látótávolság

Minden útszakaszon meg van határozva a maximális sebesség, melynek megválasztásával egyik pontból a másikba még biztonságosan eljuthatunk. Ideális esetben az utakon a közlekedők egymást követve folytonos és egyenletes sebességgel haladnának. A sebesség, határokon belüli megválasztása szervesen kapcsolódik a sofőr személyéhez, így ilyen kedvező szcenárió csak ritkán alakul ki. A különböző járművek változó sebességgel közlekednek, éppen ezért helyenként előzési szakaszokat kell biztosítanunk.

Egy előzési manőver véghezvitele sokkal nagyobb koncentrációt igényel, mint a nyugodt tempójú haladás, így ezeken a szakaszokon hosszabb látótávolságok szükségesek. Előzési látótávolságnak azt a hosszat nevezzük, melynél 1,0 m-es szemmagasság mellett az ellenirányú forgalmi sávban közlekedő 1,0 m magas járművet

még képesek vagyunk érzékelni. [MAÚT, 2008]A következő ábra részletesen ismerteti az előzési látótávolság meghatározásához alkalmazott módszertant. [Schuchmann, 2010]



2-3. ábra Előzési látótávolság adatai [Schuchmann, 2010]

- k – távolság az előzés előtt és után [m]
- t – előzési idő [s]
- v_t – tervezési sebesség [km/h]
- Le_1 – az előzési út hossza [m]
- Le_2 – az előzési idő alatt szemből érkező jármű útja [m]
- Le_3 – biztonsági távolság az előző és a szemből érkező jármű között [m]

A BME által alkalmazott számítási mód és a szabvány szerinti értékek némely esetben jelentősen eltérnek. A teljesség igénye nélkül az egyes látótávolságok BME-s és a *Közutak tervezése ÚT 2-1.201*-es szabvány szerint értelmezett hosszát a következő táblázat tartalmazza. A dolgozat megírásakor a szabvány szerinti távolságok kerültek alkalmazásra.

2-1. táblázat Az előzési látótávolság értékei [Schuchmann, 2010]

v_t [km/h]	Le^{BME} [m]	$Le^{szabvány}$ [m]
30	470	300
40	480	330
50	495	360
60	510	400
70	530	440
80	555	500
90	585	560
100	615	640
120	705	800

3 Digitális domborzat és az elérhető adatok

Láthatósági vizsgálatok végzéséhez a szabványban előírt értékeket valamilyen módon össze kell hasonlítanunk az általunk számolt értékekkel, ehhez feltétlenül szükségünk van 3 dimenziós információkra. A 3D adatok nem csupán a megjelenítés, de a vizsgálat végrehajtása érdekében is elengedhetetlenül fontosak. Ahhoz, hogy meg tudjuk állapítani, egy útszakaszon mennyire takarnak be különböző objektumok, mennyire lógnak bele az egyes tereptárgyak a látóvonalunkba, vagy az út úrszelvényébe, szükségünk van egy domborzatmodell létrehozására. A következő alfejezetekben bemutatom mit is nevezünk domborzatmodellnek.

3.1 Digitális domborzatmodellezés

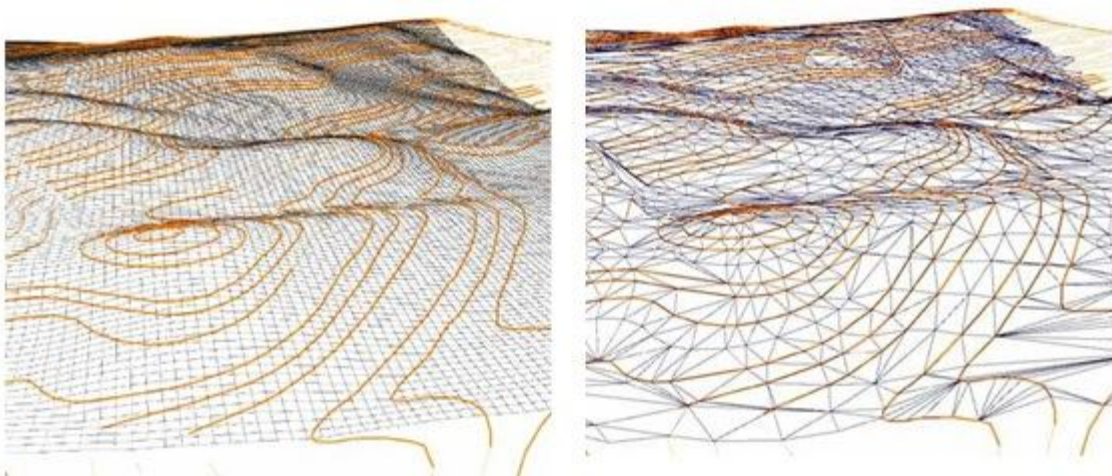
A számítástechnikával segített domborzatmodellezés az utóbbi néhány évtizedben rendkívül sokat fejlődött, ez a fejlődés a jelenleg is tart. Térhódítása annak köszönhető, hogy a mérnöki munkák világában egyre nagyobb igények merülnek fel a 3 dimenziós ábrázolásra. Manapság egy munka végrehajtásának szigorú követelménye a pontos magassági felmérés és ábrázolás. A mai világban már léteznek e modellezési eljárások lebonyolítására szolgáló célprogramok, melyekben az adatok beolvasását követően pár gombnyomás után egy digitális modell áll a rendelkezésünkre. A digitális módszerekkel előállított modelleket hívjuk összefoglaló néven Digitális Domborzat Modellnek (DDM). Egy digitális modell sohasem fogja teljes mértékben valósághűen tükrözni a terep alakulatait. Attól függően, hogy milyen pontos adatokat használunk, vagy milyen sűrűséggel mérjük fel a területet a virtuális domborzat célszerűen változni fog. Minél jobban igazodunk a terep ki-, és bemélyedéseihez, annál jobban fog igazodni a modell is a valósághoz. [Márkus, 2006]

Minden esetben, amikor egy digitális domborzat előállítása a célunk, ki kell választanunk úgynevezett támpontokat. A támpontok olyan pontok, melyeknek van vízszintes koordinátájuk, valamint a magasságuk valamilyen magassági rendszerben szintén meghatározott. A különböző eljárások persze különbözőképpen számolnak magassági modelleket, de abban az egyben megegyeznek: mindegyiknek szüksége van a támpontok meglétére. Ha ezek mérésből, vagy lekért adatokból már rendelkezésünkre állnak, akkor indulhat a számolás, már csak ki kell választanunk az eljárást. Az egyes interpolációs eljárások a támpontok segítségével különböző módon egészítik ki a

magasságokkal nem rendelkező helyeket. A kiegészítések elvégzése után rendelkezésünkre áll digitális formában a terep célszerűen egyszerűsített mása.

Az alapján, hogy a támpontok miként oszlanak meg a terepen többféle modellről beszélhetünk:

- a pontok szabályos rácsháló mentén helyezkednek el;
- strukturális felépítéskor figyelembe veszik a domborzati jelleget;
- teljesen véletlen módon helyezkednek el a pontok, nem tükrözik a terep alakulatait.



3-1. ábra Egy terület GRID-, (bal oldalt) és TIN-modellje (jobb oldalt) [Szikszai, 2003]

A 3-1. ábrán egy terület GRID-, és TIN-modelljét láthatjuk. Mivel a TIN-modell szabálytalan háromszög hálózat együttese, ezért sokkal jobban képes illeszkedni a terephez, mint a négyszöghálós elrendezés.

Egy terület felmérése során elsősorban a strukturált modelleket szokás alkalmazni. Ha van elegendő műszerünk és időnk, akkor nagy pontszámú szabályos modellt is használhatunk. A szabályos modellek (GRID³) hátránya, hogy nem alkalmazkodnak teljes mértékben a felmért terep valós alakjához, azonban mind a mérés, mind pedig a számítás és a későbbiekben szükséges interpolálás egyszerűbben hajtható végre velük.

A legelterjedtebb alkalmazási lehetőség az úgynevezett TIN⁴-modell, mely során a támpontokat szabálytalan háromszöghálózatban helyezzük el a terepen. A pontokat megpróbáljuk úgy kiválasztani, hogy azok minél jobban igazodjanak a terep

³ Szabályos négyszöghálózat

⁴ Triangulated Irregular Network – Szabálytalan Háromszög Hálózat

alakulataihoz, ezáltal a később előállított DDM is sokkal jobban fogja tükrözni a valós állapotokat. A TIN egyik leggyakoribb megoldása az egymással, oldalukkal kapcsolódó, a támpontokat csúcspontként kezelő, háromszögekből álló hálózati kialakítás, az ún. Delaunay háromszögelés. Itt igyekszünk a lehető legkisebb háromszögeket kialakítani a támpontokra illeszkedően. A dolgozat végrehajtása során a számítások minden esetben TIN-modell felhasználásával történtek. [Horváth, 2011a; Horváth, 2011b]

A következőkben szeretném részletesebben bemutatni a Magyarországon elérhető domborzatmodelleket.

3.2 Digitális modellek Magyarországon

A mai világban nem mindig van szükségünk, vagy nincs is időnk saját felmérésre, ez azonban nem jelenti azt, hogy nem juthatunk domborzati információkhoz. Minden országban rendelkezésre állnak már digitális adatok a felhasználók különböző igényeihez alkalmazkodva. Egy láthatósági vizsgálat esetében így könnyebb dolgunk van. Annak érdekében, hogy minél jobb eredményt kapjunk, szükségünk van az említett digitális domborzatra. Ez a domborzatmodell kiterjedhet egyszerűen csak az út felületére, vagy akár az út felületétől bizonyos távolságra is. A kiterjedésnek akkor van jelentősége, ha a vizsgálat nem csupán az előzésre vagy a megállásra korlátozódik, hanem a környezet ismeretére is. Első lépésként el kell döntenünk, hogy milyen pontosságú vizsgálat készítése a cél, és ehhez milyen adatok jöhetnek szóba. Magyarországon – ahogy arról a bevezetőben is volt már szó – többféle adat érhető el, melyek mindegyikének más és más pontossági jellemzői vannak. Az adatok természetesen nem csak pontossági jellemzőikben, hanem árukban is rendkívül eltérnek egymástól. Éppen ezért nem minden esetben a pontosságé a döntő szerep.

Ha meghatároztuk, hogy milyen információt szeretnénk kinyerni a meglévő 3D-s adatokból, akkor annak alapján már fel tudunk állítani egy követelményrendszert magunknak, hogy mely források jöhetnek szóba. Fontos azt is eldöntenünk, hogy mekkora a vizsgált terület, ezt megfelelő módon le kell határolnunk. A lehatárolásnak persze csak akkor van nagyobb szerepe, ha megszabott összegű információt kívánunk igénybe venni.

3.2.1 FÖMI DDM

Magyarországon két nagy intézmény létezik, melyek digitális adatokat szolgáltatnak a felhasználók számára. Ezen intézmények közül az egyik a Földmérési és Távérzékelési Intézet (FÖMI). A FÖMI a magyarországi földmérési munkák koordinálásáért felel elsősorban, de a koordinációs feladatok mellett adatokat is szolgáltat az ország teljes területéről.

Nagy szerepet játszik a 3D domborzatmodellek előállításában, hiszen az ő tulajdonában áll a jelenlegi legpontosabb magyar modell a DDM-5. A DDM-5 az EOTR⁵ M=1:10000-es méretarányú topográfiai térképekből levezetett digitális domborzatmodell. A DDM nevében az 5 jelző arra utal, hogy 5x5 m-es terepi felbontással készült és a modell magassági értékei valós magassági értékeket - Balti alapszintre vonatkozó, méter egységben, két tizedes jegyre adott magasságot - jelölnek. A domborzatmodell készítése során az egyes területeket fotogrammetriai úton javították és az előállított TIN hálózaton belül lineáris interpolációt végeztek.

A FÖMI azonban nem csak a DDM-5 modellel szolgál a felhasználók számára, hanem egyéb, olcsóbb, de pontatlanabb modell tulajdonosa is. A dolgozatban annak érdekében, hogy választ kapjak a Google adatok pontosságára a FÖMI DDM-5 került felhasználásra.

3.2.2 Katonai DDM

A Magyar Katonai Térképészet a másik nagy szereppel rendelkező intézmény a hazai gyakorlatban, manapság HM Térképészeti Nonprofit Kft. néven ismerik a köztudatban. Mint minden országban, Magyarországon is a kétféle (civil és a katonai) térképezés külön utakon jár. A katonaság már a kezdetektől (1919-től) a topográfiai térképezés gazdája.

Az ő kezelésében áll a DDM-10 nevű domborzatmodell, mely Magyarország minden kerek tíz méteres pontjának magassági adatait tartalmazza. A modell alapját a Gaus-Krüger rendszerű topográfiai térképek jelentik. E topográfiai térképekből került levezetésre a modell 10x10 m-es terepi felbontással. Rendszerét tekintve, a felhasználó igényeinek megfelelően többféle vetületi és szelvényrendszerben is kérhető, így például a mérnöki gyakorlatban használt EOVR⁶-ban is, EOTR szelvényezéssel. A katonai modell

⁵ Egységes Országos Térkép Rendszer

⁶ Egységes Országos Vetületi Rendszer

legnagyobb hibája, hogy a magassági értékeket egész számokkal (méter élesen) ábrázolja, ennek következtében a megjelenítés során a szintvonalak és maga a modell is lépcsőzetesek lesznek.

A lépcsőzetesség miatt e modell nem került felhasználásra, hiszen az út felülete is lépcsőzetessé vált volna, ami a kiértékelésben nem vezetne megfelelő eredményre.

3.2.3 Google SRTM

A rohamos technikai fejlődésnek köszönhetően megjelentek a piacon olyan cégek is, melyek meghatározott felbontás mellett kínálják saját adataikat a nagyközönség számára. Az egyik ilyen multinacionális vállalat, a mindenki számára ismert Google Inc. A Google fejlesztésében álló program a Google Earth (Google Föld) segítségével manapság a felhasználók ingyen tölthetik le a program domborzati adatait.

A virtuális földgömb vetületi rendszere a WGS84 világ-koordinátarendszer, felszínét az 1996-ban elindított amerikai térképezési program az SRTM⁷ biztosítja. A modell mérnöki szempontból cm-es, vagy dm-es pontosság elérésére nem alkalmas. A dolgozat fő vezérvonalát azonban épp az képezi, hogy vajon attól függetlenül, hogy az említett pontossági követelmények kielégítésére nem alkalmas a modell, egy láthatósági vizsgálat lefolytatására alkalmas lehet-e.

A legnagyobb előnye az ingyenesség, ezért rendkívül sokan alkalmazzák akár eredeti, akár javított formában. A javított forma azt jelenti, hogy a letöltött terepet a felhasználó saját egyéb forrásból elérhető adatokkal pontosítja. Ez a fajta pontosítás a dolgozatomnak is tárgyát képezi, ezért a későbbiekben több szót fogok róla ejteni.

3.2.4 ASTER GDEM

Ingyenesen elérhető digitális adatokat nem csupán a Google kínál a világ számára. Egy másik, hasonló felbontással jellemezhető modell a japán TERRA műholdra telepített berendezés, az ASTER domborzata is. Átlagosan 30 m-es terepi felbontással készült, és 2011 novembere óta a szabadfelhasználás számára is elérhető. Itt lényegesen nagyobb terület került ábrázolásra, mint az SRTM esetében.

⁷ Shuttle Radar Topography Mission

A műhold által készített terepmodell első változatát még 2009-ben publikálták. A 2009-es adatok azonban azóta több, mint 200 ezer sztereoképpár⁸ segítségével kerültek javításra, így érve el a nagyobb pontosságot. A modellről csak említés szintjén kívántam beszélni, hiszen a dolgozatban felhasználásra nem került. [*Úrvilág, 2011*]

⁸ Sztereoképnek nevezzük, azokat a képeket, melyek bizonyos %-ban – vertikálisan és horizontálisan egyaránt – fedik egymást, ezáltal egy 3D képet tudunk a segítségükkel előállítani

4 GPS technika

Láthatósági elemzések esetén a domborzat ki-, és beugrásai képezik az egyik legfontosabb információforrást. Az előző fejezetben bemutattam, hogy manapság az egész világon elérhetők különböző pontosságú digitális adatok. Ezek azonban nem minden esetben elegendők a vizsgálat elvégzésére. Előfordulhat olyan eset, amikor a költségvetésbe nem fér bele az elemzéshez szükséges adatok megvétele, vagy az ingyenes felhasználású adatok a területről nem nyújtanak megfelelő pontosságot. Ezekben az esetekben kétféle megoldást alkalmazhatunk, mindkettőnél a meglévő modell javítása a cél. Az első, amikor rendelkezésünkre áll a vizsgált útszakasz 3D terve – lehet ez tervezési-, vagy megvalósulási térkép egyaránt –, a fő az, hogy valamilyen többletinformációt nyújtson. Ilyenkor könnyebb dolgunk van, hiszen a díjtalan domborzatmodellt további saját munkát nem igénylő információkkal vagyunk képesek korrigálni.

Mi van akkor, amikor nem áll rendelkezésünkre az út 3D vonalvezetése? Erre is létezik manapság technikai megoldás. Ilyen esetekben szintén többféle lehetőséggel élhetünk a GPS-es domborzatfelméréstől kezdve a légifényképezéseken át egészen a lézerszkennerekig bezárólag. A GPS-el történő méréseknek többféle változata lehetséges. Természetesen az egyes opciók közötti választás nagyban függ a feladat lényegétől. Általános esetben a GPS méréseket statikus és kinematikus meghatározási módszer szerint szokás csoportosítani. A statikus mérések, lényegüket tekintve alappont meghatározás céljait szolgálják elsősorban, így ezekről a továbbiakban nem kívánok beszélni.

A domborzatfelmérés szempontjából fontos üzemmód a kinematikus GPS mérés, mely a következőkben részletesebb bemutatásra kerül.

4.1 A GPS mérésről általában

A GPS technika fejlődésével lehetővé vált többféle megoldást kidolgozása a munkák gyorsítása érdekében. Valamennyi megoldásnál egy álló, és egy mozgó vevő vesz részt a pontmeghatározás folyamatában. Az álló vevőt szokás bázisnak-, a mozgó vevőt pedig rovernek nevezni. A munkákat több módon is végrehajthatjuk, attól függően, hogy egy vevő pár, vagy csak egyetlen vevő áll a rendelkezésünkre.

A korszerű technikának köszönhetően, ma már Magyarországon is kiépült a bázisállomások rendszere. E rendszer a két vevővel való mérés egy korszerűbb változata, melynek során elegendő egyetlen GPS-t vennünk (a rovert), a bázis által szolgáltatott adatokat pedig a bázisállomások rendszeréből nyílik lehetőségünk letölteni. Természetesen egy bázisállomás információinak igénybevétele megfelelő összeg ellenében történhet.

Olyan esetekben, amikor bázis és rover műszerünk is van, nem feltétlenül fontos a fizetős szolgáltatás használata. Ekkor persze saját magunknak kell gondoskodni olyan pontról, melynek 3 koordinátája megfelelő pontossággal ismert. Sajnálatos módon ilyenkor nem csak a pont meglétére, de a műszer őrzésére is időt kell fordítanunk. Bármelyik lehetőséget is választjuk a domborzat pontosítására a feladatmegoldás hasonlóképpen alakul, így a továbbiakban, általánosságban mutatom be külön-külön a folyamatos kinematikus-, és a félkinematikus mérést.

4.2 Folyamatos kinematikus mérés

A folyamatos kinematikus mérésnél az elmondottak alapján egy mozgó-, és egy álló vevő kerül alkalmazásra. A többi mérésfajtaival ellentétben itt nem a felkeresett pontok koordinátái képezik az érdeklődéstárgyat, hanem a bejárt útszakasz. Tehát, ha van egy meglévő útszakaszunk, és ezt a lehető leggyorsabban szeretnénk felmérni, akkor célszerű ezt a megoldást előtérbe helyezni. Mivel nem a felkeresett pontok, hanem az útvonal a kiértékelés tárgya, így a lehető leggyorsabb munkavégzés miatt praktikus egy járműre van szükségünk.

A GPS mérések sajátos velejárója, hogy célszerű a mérés időpontját előre megtervezni, hiszen a meghatározás pontosságát akár kritikusan is befolyásolhatja az aktuális műhold konstelláció. Folyamatos kinematikus módszernél nem csak a mérés időpontját, de a bejárni kívánt útvonalat is ajánlatos előre terveznünk, mert ilyenkor nem kell felesleges köröket tennünk ahhoz, hogy mindent meg tudjunk határozni.

Minden GPS vevő (rover) esetén fontos, hogy beállítsuk a rögzítés időközét, melyet a legegyszerűbben a haladási sebesség függvényében tudunk megválasztani.

4.3 Félkinematikus GPS mérés

Félkinematikus módszernél a folyamattal ellentétben nem a bejárt útvonal a fontos, hanem a felkeresett pontok. Ezzel a megoldással előnyösebben tudunk mérni

mikro domborzatos területeken, ahol a domborzat rendkívül változó jelleget mutat. Értelmszerűen ennek a lehetőségnek nagyobb az időigénye és ugyanúgy előre kell terveznünk, mint a folyamatos mérésnél, azonban az említett területeken jobban tudunk a domborzathoz igazodni.

Mint minden GPS mérés esetében, itt is a mérés indulása előtt inicializálnunk kell a műszert. Az inicializálás során határozza meg a rover saját közelítő helyzetét a bázishoz, vagy bázisállomáshoz képest. E folyamat a beállításoktól és műszerfajtától függően akár több percig is eltarthat. Végeztével azonban már – persze a takarások (fák, épületek) folyamatos figyelésével – indulhat a mérés.

5 AutoCAD Civil 3D

A bevezetőben említettem, hogy a dolgozat vezérfonalát a GIS adatok és egy térinformatikai program képezi, azonban a tisztánlátás érdekében szeretnék megemlíteni egy további lehetőséget is. Ez az AutoCAD Civil 3D.

Az Autodesk programcsalád Civil 3D nevű tagja egy önállóan is működő program, mely az egyszerű tervezési funkcióktól kezdve a mérnöki munkálatok számtalan területén alkalmazható. Hasonlóan a sima AutoCAD-hez, a Civil 3D-t is kifejezetten mérnökök számára fejlesztették. Itt kapott helyet az általános programcsomagból kimaradt úttervezési funkciói. Az alkalmazás telepítésével együtt a csomag az általánosan használt AutoCAD-t is felrakja. Az elindítást követően az egyes, előre kialakított kinézetekből a munka jellegének megfelelően választhatunk, így a Civil 3D telepítésével nem csak a tervezési funkciók, de az általános programcsomagban is megtalálható modulok szintén elérhetők. Az általános programcsomag és a Civil 3D közötti leglényegesebb különbség talán a domborzati és földmérési adatok kezelésében rejlik. Ezek azok, melyek lehetővé teszik a magassági-, és vízszintes adatok ismeretében az út-, vasút és egyéb infrastruktúraelem tervezését. Mint minden, a domborzati adatokat kezelő alkalmazásnál, ennél is többféle, különböző beállítással találkozhatunk.

Az AutoCAD esetén azért, hogy egy láthatósági vizsgálatot véghez tudjunk vinni, ez elejétől a végéig fel kell építenünk az elemezni kívánt mintaszakaszt. Az építés első lépésében definiálnunk kell a domborzati adatokat. A felületeink itt is TIN-, vagy GRID elrendezésűek lehetnek, ezt természetesen, vagy önkényesen, vagy a feladat jellege alapján határozzuk meg. Egy felület létrehozásánál többféle objektum képezheti a domborzati információkat, így egyaránt dolgozhatunk pontokkal, vonalakkal és poligonokkal.

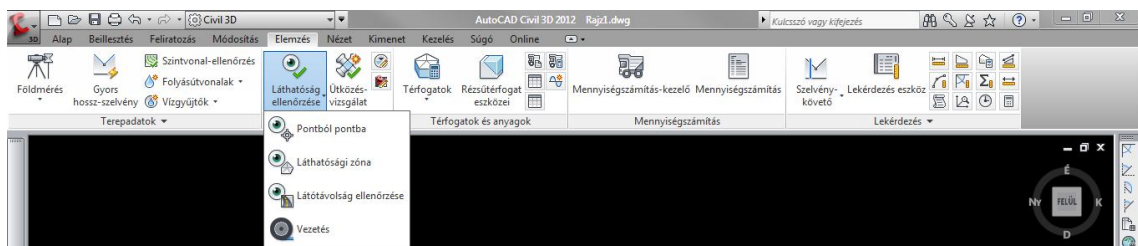
Ha egy, a tervezés folyamatában lévő útszakaszt vizsgálunk a láthatóság kérdéskörében, akkor egyszerűbb dolgunk van, hiszen ilyenkor a tervezéssel együtt, ugyanazon adatokból a láthatósági vizsgálatot is el tudjuk végezni. Az előny abban áll, hogy ekkor nem kell lépésről lépésre utólag újra felépítenünk a domborzatot és az útszakaszt is. Ha azonban egy már régebben átadott olyan szakasz képezi a vizsgálat tárgyát, melynek valószínűsíthető, hogy nincsenek meg a kitűzési adatai, akkor komplikáltabb a helyzet. Ilyenkor ugyanis nincs tudomásunk arról, hogy a valóságban hol kezdődik egy átmeneti ív, vagy egy magassági lekerekítő ív, így csak különböző

mérésekkel tudjuk közelíteni a valóságos helyzetet. Természetesen a főutak vízszintes-, és magassági vonalvezetéséről is kérhetők adatok, hasonlóan a domborzati információkhoz, ezek azonban nem biztos, hogy minden esetben helytállóak. Tévedésnél pedig rossz eredményekből, helytelen következtetéseket vonhatunk le.

Eltekintve attól, hogy a vizsgálat tárgyát milyen útszakasz képezi, általánosságban írható az út kialakítás folyamata. A folyamat során első lépésben fel kell venni az útnyomvonalát. A nyomvonal kialakítását követően már képezhetjük a hossz-szelvényt, ahol a tervezési-, vagy meglévő útnál a már meglévő kritériumoknak megfelelően kell előállítanunk a vonalvezetés műszaki paramétereit (magassági lekerekítés értéke, helye, átmeneti ív adatai ...stb.).

A hossz-szelvényt követően a kereszt-szelvények megszerkesztéséhez fel kell vennünk egy, vagy több minta kereszt-szelvényt. Mintakereszt-szelvények képezésénél különféle előre gyártott műtárgy közül választhatunk, melyeket a feladat követelményei szerint testre is szabhatunk. Csak a magassági-, és vízszintes vonalvezetés ismeretében tudunk következtetéseket levonni a látótávolságok értékeiről.

Az említett vonalvezetési információk ismerete után már használható az „Elemzés” fülnél található „Láthatóság ellenőrzése” funkció. Itt szintén többféle lehetőség közül választhatunk, azonban egy, az út nyomvonala mentén történő vizsgálathoz a legcélravezetőbb a „Látótávolság ellenőrzése” utasítás. A láthatósági vizsgálat és az egyes domborzati adatokkal való munkában részletesebb segítséget kaphatunk a *Mastering AutoCAD Civil 3D 2012* című felhasználói kézikönyvből. [Graham and Holland, 2012]



5-1. ábra Láthatósági vizsgálat az AutoCAD Civil 3D-nél [a szerző munkája]

Az AutoCAD Civil 3D részletesebb ismertetését mellőzve, a továbbiakban szeretném bemutatni a láthatósági elemzés elvégzésének lehetőségeit az ArcGIS nevű térinformatikai programban.

6 Modellezés GIS adatok felhasználásával

Az AutoCAD Civil 3D alternatíváit ismerve úgy döntöttem, hogy a láthatósági vizsgálatot megpróbálom GIS⁹ adatok, illetve egy térinformatikai program segítségével megvalósítani. A GIS adatok felhasználása azért tűnt sokkal előnyösebbnek, mint az AutoCAD kínálta lehetőségek, mert ebben az esetben a felhasználó képes különböző attribútum adatokkal dolgozni, melyek minden egyes réteg esetén egy úgynevezett attribútum táblában vannak eltárolva. A spanyol TRIVIUM program bár képes láthatósági vizsgálatok elvégzésére [*Castro et al, 2009*], de nem ötvözhető a különböző elemzések. Egy térinformatikai programban azonban több, más jellegű tanulmányt is végrehajthatunk ugyanazokkal az információkkal.

6.1 ESRI ArcGIS 10

Az ESRI cég által fejlesztett programcsalád az ArcGIS 10 képezte a dolgozat alapját, hiszen e programban került megvalósításra az egész láthatósági vizsgálat. Az ArcGIS szoftvercsalád vektor alapú térinformatikai programok együttesét jelenti, melyben a 2D-s információk kezelésétől a 3D megjelenítésen át egészen egy SRTM alapú földmodellig bezárólag minden megtalálható.

A programcsalád legfontosabb, központi részét képező alkalmazása az ArcMAP. Elsődleges funkciója a megjelenítés mellett a különböző lekérdezések, leválogatások, elemzések elvégezhetősége. A felsorolt funkciókon kívül természetesen támogatja a modellezési lehetőségeket is, mely nagyban meghatározó volt a dolgozati feladat megfogalmazása során. Az ArcGIS további részeit képező programok szervesen kapcsolódnak az ArcMAP-hez. Míg utóbbival történnek a számítási és modellezési megoldások, addig az ArcCatalog programrész adatbázis építési funkciókkal van teletűzdelve. Mint minden nagyobb programcsalád esetében, természetesen itt is van 3D megjelenítés és a vizuális animáció, ez az ArcScene programrészben történik. Fontos eszköz lehet, ha például egy víztározó kiöntését kívánjuk modellezni animációs támogatással. De abban az esetben is hasznos, ha egy láthatósági vizsgálatot tűzünk ki célul.

A legfontosabb, amit az ArcGIS működéséről tudni kell, hogy vektor alapú térinformatikai program, így az AutoCAD-hez hasonlóan itt is megvan a

⁹ Geographic Information System – Földrajzi Információs Rendszer

rétegmegfeleltetés. A rétegek itt azonban nem különböző típusú vonalakat jelölnek. Definiálásuk során a felhasználónak el kell döntenie, hogy a létrehozott objektum pont, vonal, vagy felület besorolást kapjon, ennek alapján kerülnek osztályozásra az egyes elemek. Ahhoz, hogy az ArcGIS-ben végre tudjak hajtani egy láthatósági elemzést fontos volt, hogy pár előkészítő lépéssel éljek, melyek meghatározták a munkám alapjait. Az előkészítő lépésekről bővebben a következő fejezet részben esik szó.

6.2 Előkészítő lépések

Egy láthatósági vizsgálat elvégzéséhez nem ugorhatunk csak egyből az információk tengerébe. Lehetőség szerint végig kell gondolni, hogyan is hajtsuk végre a vizsgálatot, és ehhez milyen adatok definiálására, milyen rétegek létrehozására van elsődlegesen szükség. Mivel a dolgozatban arra a kérdésre keresem a választ, hogy mennyire használhatók ingyenesen elérhető Google adatok egy elemzésre, ezért először meg kellett vizsgálnom, miképp egyesíthető egy ilyen adatsor az ArcGIS által támogatott formátumokkal.

Egy Google domborzat letöltésére többféle lehetőség is adódik. Az egyik, hogy ha rendelkezünk AutoCAD Civil 3D licenccel, akkor közvetlenül a programba importálhatók az adatok. Sajnos ez a megoldás nem minden esetben a legcélravezetőbb, mert ezt csak az AutoCAD Civil 3D 2012-es programmal tudjuk megtenni. További hibái e megoldásnak akkor jelentkeznek, amikor egy nagyobb területéről szeretnénk információt gyűjteni. A dolgozat készítése során megpróbáltam egészben lekérni a zalaegerszegi 761-es elkerülő utat és annak tágabb környezetét. Ekkor több hiba is jelentkezett, mely során a letöltött állomány nem csak vízszintes x,y irányú torzulást szenvedett, de egy elcsavarodás is fellépett. Természetesen nem ezek az egyetlen problémák, melyekkel találkozunk, ha ezt a megoldást választjuk, hiszen az AutoCAD nem képes a pontos helyre letölteni a domborzatot, ezért ezeket utólag saját magunknak kell illeszteniünk pontpárok megfeleltetésével.

A másik megoldási lehetőség, hogy egy ingyenes forráskódú szoftvert a Sketch' Up-ot használjuk. A Sketch' Up, korábban – a Google Földhöz hasonlóan – szintén a cég fejlesztésében állt, azonban 2012-ben a Trimble vállalat felvásárolta. A szerkeszthetőség sokszor határt szab a programok kezelésének, ám ez a Sketch' Up-nál nem jelent problémát, ugyanis kezelőfelületei úgy lettek kialakítva, hogy az egyszerű felhasználók számára is érthetőek, használhatóak legyenek. Eredetileg épületek hobbi szintű

modellezésére lett fejlesztve, és köszönhetően a különböző feltöltési kritériumnak (pl.: az épület georeferáltsága) képesek vagyunk közvetlenül letölteni nem csak a terepet, de ezzel együtt a domborzat részét képező műholdas felvételeket is. További előnyként könyvelhető el, hogy ha egyszerre több, egymáshoz kapcsolódó területet is szeretnénk importálni, úgy nem kell a két, vagy több modell egymáshoz illesztésével bajlódunk, mert a program automatikusan elvégzi helyettünk. Sajnos itt is hátrányként jelentkezik, hogy a modell és a felvételek nem lesznek az eredeti helyükre illesztve, de ez esetben sokkal könnyebb az utólagos helyreigazítás.

A domborzat meglétét követően lehatároltam a kiválasztott mintaszakasz útfelületét. A lehatárolás nagyon egyszerű módon történt, mivel, mint említettem a Sketch' Up-ba a műholdas felvételek is letöltődnek. A felvételek felbontása arra elegendő, hogy közelítőleg körülrajzoljuk az út felületét. A felület megléte után immáron a Civil 3D-ben georeferálhatóvá váltak az adatok pár közös pont ismeretében. A műveletre az oda-, és vissza nyomvonalak meghatározása érdekében volt szükség. A nyomvonal kialakítása után, kerek 20 m-es távolságra egymástól felvettem az oda-, és vissza nyomvonalpontokat. A 20 m-es távolság tűnt a legcélszerűbbnek. Ha a pontokat 5, vagy 10 m-enként vennénk fel, akkor túlságosan sok adat a számítási-, és futásidőt is nagyon megnövelné. Az ennél nagyobb távolságok azonban nem reprezentálnák megfelelően a domborzat alakulását. A pontok felvételét követően persze magassági illesztésre nem került sor, így csak a vízszintes koordináták álltak rendelkezésemre egy Excel-táblázat formájában.

6.3 Modellezési lehetőségek

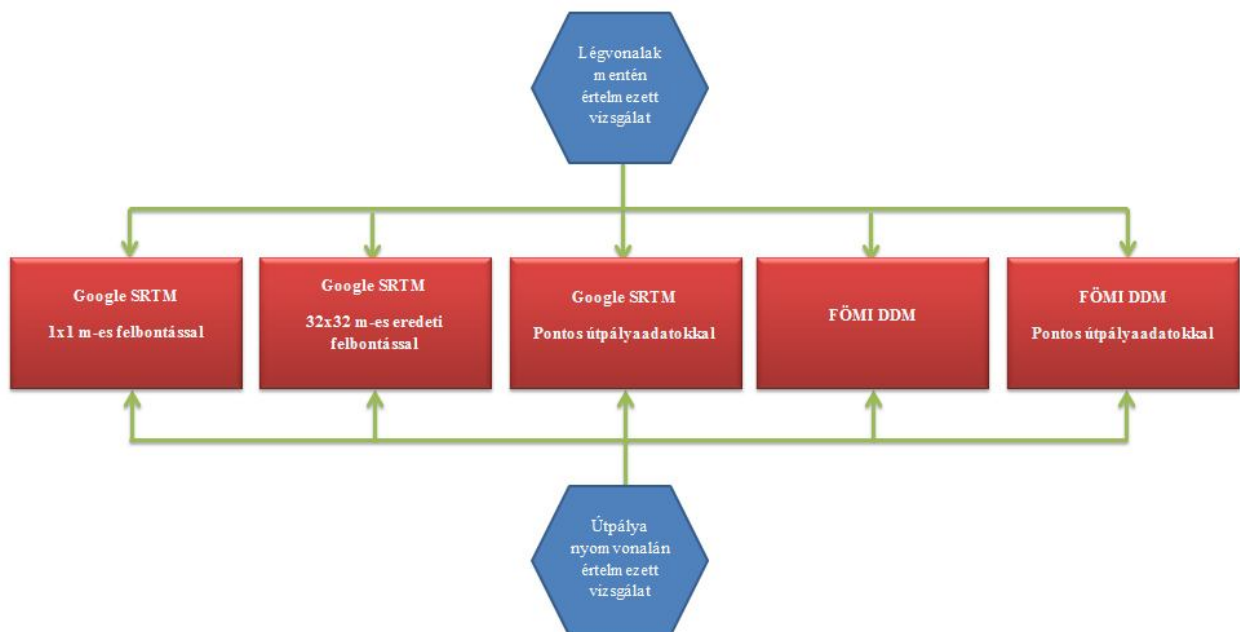
Bár a térinformatikai programok sokféle lehetőséget biztosítanak egy felhasználói elemzésre, azonban egy láthatósági vizsgálat elvégzéséhez számba kell venni a meglévő funkciókat és kikeresni, egymásután rakni őket egy jó eredmény elérése érdekében.

Korábbi tanulmányaim során már volt szerencsém használni az ArcGIS-t, így működése nem volt ismeretlen számomra. A programban egy-egy feladathoz több, előre megírt lekérdezésből választhatunk, de lehetőségünk van arra is, hogy valamilyen szisztéma szerint megadjuk, mely lekérdezéseket milyen adatokkal és milyen sorrendben hajtson végre. Ezt a lehetőséget reprezentálja a Model Builder nevű alkalmazás. [ESRI, 2012a]

A Model Builder úgy képzelhető el, mint egy folyamatábra szerkesztője, ahová a programban található funkciók egytől egyig behívhatók, és egymás után helyezve valamilyen logikai kapcsolattal szerkeszthetők. Ez tökéletes lehetőséget nyújt egy olyan vizsgálatra, mint a láthatóság értékelése.

Az előző alfejezetben elhangzott, hogy az ArcGIS tényleges használata előtt már rendelkezésemre állt az út felület AutoCAD-es állománya, valamint az út nyomvonalpontjainak Excel-táblázatos koordinátái. Ahhoz, hogy ki tudjam számoltatni a programmal a láthatósági értékeket, kétféle megoldási lehetőséget fogalmaztam meg. Ezek bemutatják, milyen jó, vagy rossz adott esetben egy Google adatsor. A megfogalmazott lehetőségek a következők:

- Láthatósági vizsgálat a nyomvonalpontok között légvonalban értelmezve;
- Láthatósági vizsgálat a nyomvonalpontok között az út nyomvonala mentén értelmezve.



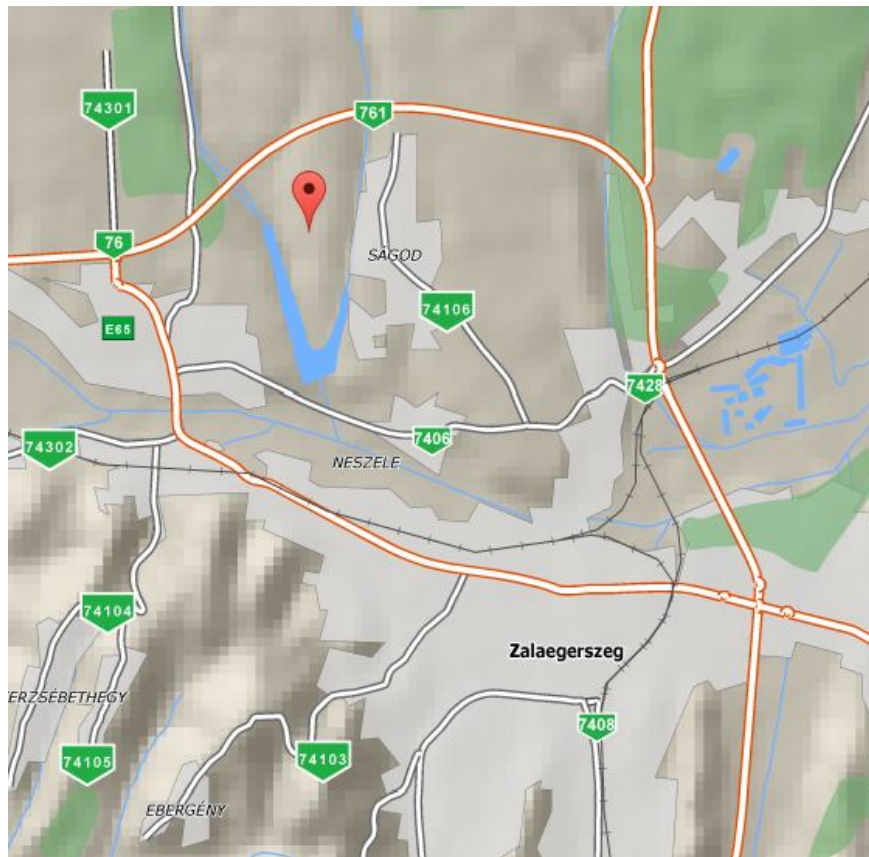
6-1. ábra Láthatósági vizsgálat megoldásai [a szerző munkája]

Az egyes eredmények értékelésére, a kétféle vizsgálati lehetőségből több megoldást dolgoztam ki, melyek a 6-1. ábra ismerteti. Az egyes alternatívák mind a megállási, mind pedig az előzési látótávolság számolásánál is alkalmazásra kerültek. A lehető legjobb értékrend felállítása érdekében az ingyenes-, és a FÖMI-s domborzat is javításra került az

elkerülő megvalósulási térképével. A szabad felhasználású információkat továbbá eredeti és lineáris interpolációval ellátott pontosítással is figyelembe vettem. A különböző módszerek végrehajtásáról és értékeléséről a következőkben lesz szó.

6.4 Mintaszakasz bemutatása

A dolgozatban vizsgálat mintaútszakasz a szervesen kapcsolódik Zalaegerszeg életéhez. Zalaegerszeg a Nyugat-Dunántúlon a Zala-folyó két partján terül el. Nyugat fekvése révén közel egyenlőre távolságra helyezkedik el a horvát, a szlovén és az osztrák határtól. A városon a határok felé haladó forgalom egészen 2006-ig végig kellett, hogy haladjon, ezáltal nagymértékben szennyezve a városi légkört. Bár a város levegőjét továbbra is a közlekedés befolyásolja a legnagyobb mértékben, azonban 2006 óta valamelyest sikerült a problémát orvosolni az északi elkerülő megépítésével.

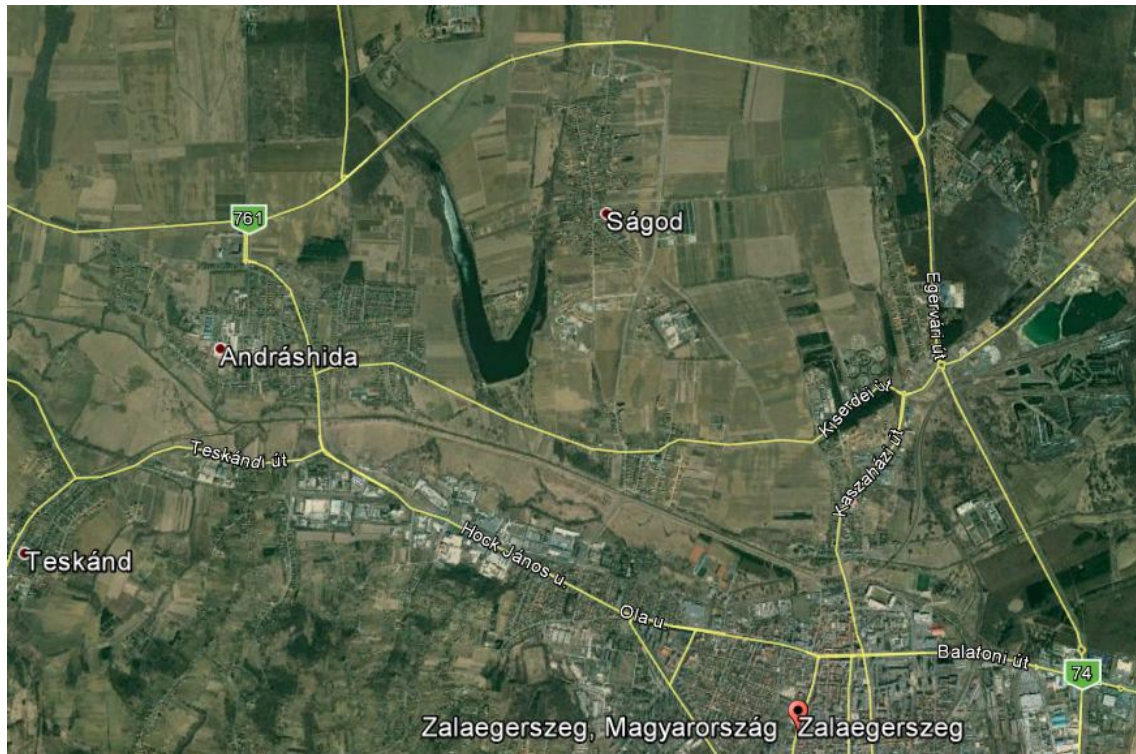


6-2. ábra A főbb utak helyzete Zalaegerszeg körül [KKK, 2013]

A 761. számú elkerülő út nem csak a légszennyezés okozta káros hatásokat hivatott javítani a korábban túlterhelt városon átmenő forgalom csökkentése is a célok között szerepelt. Az elkerülő megépítésével összhangban a tranzitforgalom hatása a városi utakra jelentős mértékben redukálódott. Az elkerülő a 74-es főutat köti össze a 76-os főúttal, ez

alatt több, Zalaegerszeg körül elterülő kisebb faluból induló alsóbbrendű utat keresztezve. A 2006-os átadást követően, valószínűleg a megváltozott forgalmi rendből kifolyólag több, súlyos baleset is bekövetkezett a 7404-es Nagykutas – Zalaegerszeg és a 74106-os Kispáli – Zalaegerszeg közötti utak, elkerülővel alkotott kereszteződéseiben. A problémák nagyobb forgalmi jelek kihelyezésével mérséklődtek, azonban a két említett kereszteződés közé eső szakaszt, vonalvezetése miatt továbbra is nehéz belátni.

A fentebb említett okokból kifolyólag helyiként szerettem volna megbizonyosodni arról, hogy a bevezetőben ismertetett látótávolságok még a beláthatóságukban korlátozottabb helyeken is teljesülnek. A kiválasztott mintaszakaszom ezért a Zalaegerszeget északról elkerülő 761. számú főút 55+620 és 61+000 szelvénye közé eső szakasza.



6-3. ábra Zalaegerszeg a Google Földben [Google Earth]

7 Légvonalas láthatósági vizsgálat

Egy beláthatósági elemzés során fontos, hogy több szempontból kezdjünk neki a kutatásnak, az elemzés ugyanis nem biztos, hogy első körben megfelelő eredményt fog adni. Első nekifutásra egy légvonalakban történő vizsgálat jutott az eszembe. A lényege, hogy az előkészítő lépések című fejezetben meghatározott út nyomvonalpontokat beolvassuk az ArcGIS-be. Ezek között a program meghatározza, hogy a domborzati viszonyok függvényében melyek azok, amik teljes egészében összeláthatók. A légvonalak létrehozásának lépéseit a továbbiakban ismertetem.

7.1 Folyamatábra a Model Builderben

Az ArcGIS különböző kiegészítő, úgynevezett „toolbox”¹⁰-okkal rendelkezik, melyekben a korábban említett utasítások, mint a lekérdezések, elemzések és hasonló műveletek megtalálhatók. A program alapfunkciói között sajnálatos módon nem kapott helyet a 3 dimenziós adatokkal való dolgozás. Ez egy külön kiegészítő, melynek a neve „3D Analyst Tool”. [ESRI, 2012b]

A 3D Analyst Tool-ban van majdnem minden olyan lekérdezési és értékelési megoldást, mely nevéhez híven a 3D-s adatokkal dolgozik. Itt találunk egy műveletet, mellyel meghatározott rétegek között légvonalakat tudunk létrehozni. [ESRI, 2012c] Ahhoz azonban, hogy ezek a légvonalak létrehozhatók legyenek meg kell első lépésben határoznunk, miként történjen meg a már meglévő adatok beolvasása. Ha tudjuk, hogy nem csak egy mintaterület lesz a vizsgálat célja, akkor már célszerű a legegyszerűbb funkciókat is – mint a beolvasás – a Model Builderben elkészíteni.

A légvonalas láthatóság létrehozásához sok műveleten kell végigmennie a programnak, míg a keresett adatok megfelelőek lesznek az eredmények értékeléséhez. Minden egyes részek futása az adatok mennyiségétől függően változó időtartamú, így előfordulhat, hogy a keresett légvonalak lehatárolása akár órákig is eltarthat. Az időtartam persze attól is függ, hogy milyen és mennyi műveletet helyezünk el a folyamatábrában és azok hogyan kapcsolódnak egymáshoz. A folyamatábra egyes lépései muszáj, hogy megfelelő logika szerint kövessék egymást, hiszen ha ez a logika nem strukturáltan van felépítve, akkor téves eredmények keletkeznek. Fontos megemlítenem, hogy a létrehozási folyamatban minden esetben keletkeznek ideiglenes állományok, melyek mindössze

¹⁰ Az ESRI „Toolbox”-nak nevezi a különböző funkciók végrehajtására irányuló egyes eszköztárakat az ArcGIS programban

addig szükségesek, amíg a következő funkció végrehajtódik. A végrehajtást követően persze már a továbbiakban nincs szükség ezekre az adatokra, így akár törölhetők is.

A folyamatábra egy idő után akarva akaratlanul is átláthatatlanná válhat, erre találták ki az automatikus rendezés gombot, mellyel a Model Builder automatikusan önmaga rendezi el az egyes, egymás után meghatározott funkciókat. Az automatikus elrendezésre szükséges is volt, mivel a légvonalas láthatóság folyamatábrája közel 50 egymás után következő műveletsort tartalmaz.

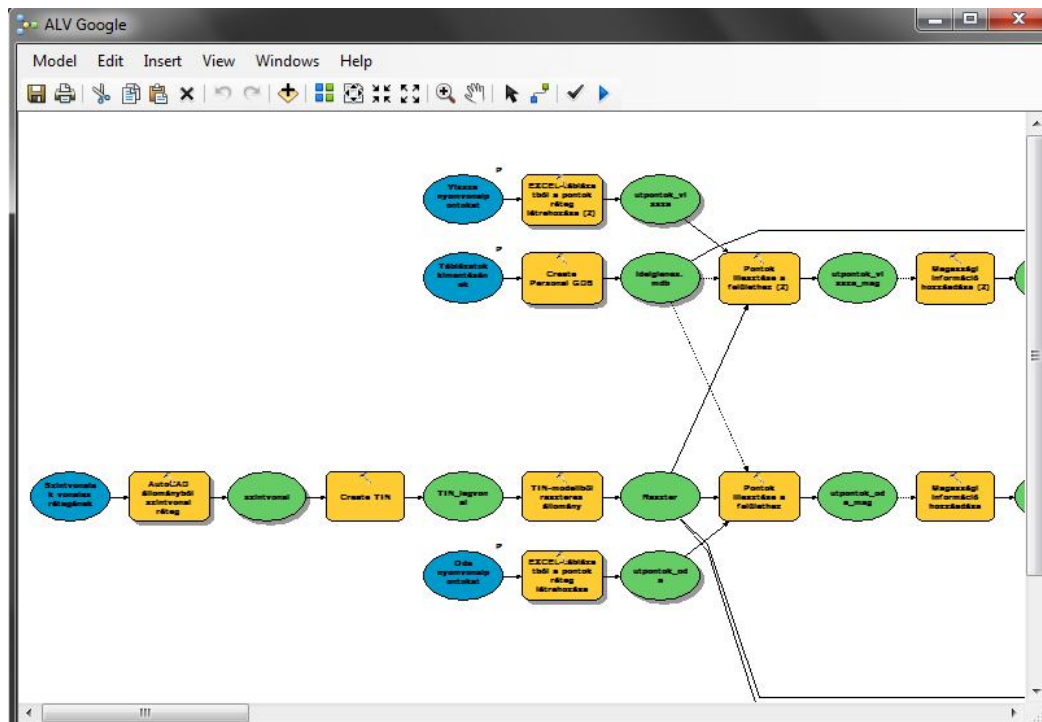
Egy új modell építésénél nem csupán a kiindulást kell meghatározni, de a végkimenetelt is, hogy a modell folyamatábrája a legkevesebb szerkesztést igényelje. Fontos tudnunk, hogy melyek azok a funkciók, amiknél valamilyen külső adatforrásból importálunk, továbbá melyek azok, amelyek a tényleges végkimenetelt tartalmazzák. Az ilyen kritikus funkciók megjelölését szolgálja a paraméterezés. Egy paraméter sok minden lehet:

- Bemenő adat;
- Kimenő adat;
- Egy művelet saját követelménye (pl.: TIN-modell esetén az adatok koordinátarendszere).

A paraméterek beállítását gondosan kell elvégezni, hogy minden megfeleljen az ArcGIS követelményeinek. Ha nem megfelelő dolgot veszünk paraméternek, akkor egy hibaüzenettel akár le is állhat a futás. Ha azonban mindent a kívánt módon állítunk be, akkor a folyamatábra elkészülte után csak annyi a teendőnk, hogy elindítsuk a modellt, és a program automatikusan egymás után végrehajtja a folyamatsort.

7.2 A légvonalas vizsgálat folyamatábrája

Az előző alfejezetben említettem, hogy a légvonalas vizsgálat folyamatábrája hozzávetőleg 50 összekapcsolódó műveletet tartalmaz. E folyamatábra egy részletét mutatja a *7-1. ábra*, teljes terjedelmében az *1. számú melléklet* tartalmazza. Az egyes lépések között vannak ismétlődések is, így feleslegesnek tartom leírni a teljes végrehajtási folyamatot. A jobb megértés érdekében azonban a fontosabb funkciókat célszerű elemezni, hogy a vizsgálat logikája megfelelően követhető legyen.



7-1. ábra A légvonalas vizsgálat folyamatábrájának részletes [a szerző munkája]

7.2.1 Adatok beolvasása

Az előkészítő munkálatok című fejezetben szó volt arról, hogy a vizsgálatához rendelkezésemre álltak a mintaterület szintvonalai, az út felülete, valamint a nyomvonalpontok és azok koordinátái oda-, és vissza irányban. Ezeket az adatokat különböző formátumuk révén, más-más módon kell beolvasni a programba.

Egy Excel-táblázat beolvasására a legmegfelelőbb megoldás a „Make XY Event Layer” funkció. Lényege, hogy megadjuk, melyik az a táblázatos adat – jelen esetben egy Excel-táblázat – amelyik az útpontok koordinátáit tartalmazza. Az Excel-táblázatban fontos, hogy benne legyenek a pont vízszintes koordinátái, és valamilyen azonosítója, mint a pontszám. A vízszintes koordináták mellett azonban én eltároltam a magassági értékeket is (első lépésben ez mindegyik pontra nulla), valamint a további azonosításhoz szükséges szelvényszámot. A szelvényszám az útnyomvonalán az adott pont elhelyezkedésének útvonalbeli adata. Importálás során meg kell adnunk, mi alapján helyezze el a program a pontokat, tehát melyik koordináta melyik oszlophoz tartozik. A geodéziai koordináták esetében a vízszintes tengely az y, a függőleges pedig az x, a programban viszont matematikai koordinátarendszer van, így az x és y koordinátákat felcserélve kell beolvasatni, máskülönben egy 90°-kal elforgatott nyomvonalat kapunk.

Az AutoCAD-állomány betöltése már sokkal egyszerűbb, hiszen az ArcGIS képes kezelni a dwg és a dxf kiterjesztésű fájlokat. A Model Builderben a szintvonalak importálását így megtehetjük a „Feature To Line” modul segítségével. Itt nincs más dolgunk, mint megadni a bemeneti és kimeneti adatokat. Mind a pontok, mind pedig a vonalak beolvasásánál lehetőség van egy további beállításra, melyre a későbbiekben néhány funkciónak szüksége lehet. Ez a koordinátarendszer meghatározása. Az ArcGIS rengeteg koordinátarendszer ismer, így többek között a Magyarországon alkalmazott EOVT és EOMA¹¹-t is.

A beolvasáshoz szervesen kapcsolódik a különválogatás, mellyel rendszerezzük a felesleges és nélkülözhetetlen információs elemeket. A különválogatás nem olyan nehéz, hiszen az ArcCatalog programrész adatbázis építéssel van ellátva, így itt létre tudunk hozni úgynevezett „Personal GeoDatabase”-eket. A „Personal GeoDatabase” egy adatbázis elem, arra szolgál, hogy az egyes műveletek végeredményeit ne kelljen különböző helyeken eltárolnunk, hanem egy helyre tudjuk koncentrálni őket. Hasonló elven működik, mint egy mappa, csak egy fokkal zártabb rendszert képez. Az adatok különválogatását így két adatbázis segítségével végeztem el. Az egyik tartalmazta a fontos, a másik pedig az ideiglenes információkat, melyek a folyamatára utolsó lépésében törlésre kerülnek.

7.2.2 A domborzatmodellről a látóvonalakig

A beolvasást követően rendelkezésemre álltak a kiindulási alapok. Korábban volt már szó a domborzatmodellekről, és arról, hogy egy-egy ilyen domborzat szervesen kapcsolódik a láthatósági vizsgálatokhoz. Miután a szintvonalak már az ArcGIS-ben is kirajzolásra kerültek, előállíthatóvá vált a virtuális modell. Egy 3D adatokkal dolgozó modell a már említett „3D Analyst Tool” segítségével állítható elő [ESRI, 2012b]. Itt találunk egy „Creat TIN” funkciót, ami a TIN-modell létrehozását támogatja. A modul beállítási lehetőségei között itt is szerepel a koordinátarendszer meghatározása, továbbá egy kiegészítő lehetőséggel is élhetünk, mellyel a létrehozott háromszögek előnyösebben illeszkednek a terep alakulatihoz. Ez a Delaunay háromszögelés.

A láthatósági vonalak létrehozásához nem egyszerűen csak egy terepet követelünk meg, hanem azt is, hogy ez egy raszter legyen. A raszter előállításához a kialakított TIN-modellt át kell konvertálnunk egy képi felületté. A konverzió során fontos, hogy

¹¹ Egységes Országos Magassági Alappont hálózat

megfelelően járjunk el, mert a program automatikusan felajánl nekünk egy cellaméretet, ami alapján kialakításra kerül a kép. A felajánlott cellaméret általában elegendő, hiszen a program ezt a TIN-modell alapján méretezi, azonban ha nagyon sűrűn változó mikro domborzatos területet modellezünk, akkor a művelet valószínűleg nem lesz célravezető az automatikus cellamérettel. Ilyen esetben saját tapasztalatok alapján kell megválasztani azt az értéket, melyet a legjobban igazodónak ítélünk meg. Egy raszterképnél a modell magassági információi a cellákban kerülnek eltárolásra, ezáltal a folyamat végeredményeként egy GeoTiff állományt kapunk. A GeoTiff egy olyan képformátum, melyben a cellák már rendelkeznek vízszintes és magassági koordinátákkal.

A GeoTiff megléte után már illeszthetjük az út nyomvonalpontjait vagy a raszteres, vagy a vektoros felületünkhöz. Mindegy, hogy melyiket használjuk, mert egyiket a másikból számolja a program, így mindkettő ugyanazon értékeket tartalmazza, tehát a végeredmény teljesen ugyan az lesz.

Ha a pontok magassági adatai is rendelkezésünkre állnak, akkor megnézhetjük közöttük a légvonalakat. Ezeket a légvonalakat tekintsük egyszerűen látóvonalaknak, hiszen ezeknél még nincsen meghatározva, hogy metszik-e a terepet vagy sem, vagyis, hogy az adott kiindulási pontunkból látszik-e – légvonalban – az azt követő. A látóvonalak létrehozásáért a „Construct Sight Lines” funkció a felelős. [ESRI, 2012c] Megnyitásakor meg kell határozni, mik azok a pontok, vonalak, vagy felületek, amelyekből meg szeretnénk nézni, hogy látszódnak-e egymásból. Annak érdekében, hogy az oda és vissza pontok között is az út haladási irányának megfelelően létrehozzam a látóvonalakat, bemeneti rétegnek külön-külön adtam meg az oda-, és vissza pontokat. Az előzési látótávolságoknál mindkét esetben a cél objektum és a kiinduló objektum is ugyan az volt, vagyis ha az oda pontokat nézzük, akkor a vonalak kiindultak az első pontból, és összekötötték azt minden további ponttal. Ezt követően a művelet lépett a következőre, egészen az utolsóig. Ennek a megoldásnak persze hátránya, hogy ha az első pontot összekötjük minden további ponttal, és ezt követően a következő ponttal is ugyanezt megcsináljuk, akkor minden egyes vonal duplán lesz tárolva, tehát az adatbázisunk kétszer akkora lesz, mint szükséges lenne. Továbbá, mivel légvonalakról van szó, például egy 6 km-es kanyargós útszakasz esetén valószínűsíthető, hogy az első vizsgált pontból nem fogjuk látni az utolsót. Az előállított látóvonalakat tehát javítani kell.

A megállási látótávolság számolása egy kicsit eltért az előzésitől. Előzés esetén mind a kiinduló-, mind pedig a záró pont 1 m magasan haladt a terep felett. Megállási vizsgálatnál azonban a kiindulópont, tereptől számolt 1 m-es magasságából a végpontot 10 cm-re már észlelnünk kellett. Ezen okokból kifolyólag megállásnál ugyanazon nyomvonalpontokból két különböző magassággal ellátott réteget alakítottam ki. A látóvonalak így az egyes pontok eltérő magasságú összekötésével álltak elő. Javításra, az információs halmaz szűkítésére viszont itt is szükség volt.

A javítást több lépésben végeztem el, mert egyetlen funkció általában csak egyetlen műveletet hajt végre. A lépések a következők voltak:

- 0 hosszúságú vonalak törlése;
- 600 m-nél rövidebb vonalak leválogatása;
- a redundancia csökkentése (kizárólag előzési látótávolságnál).

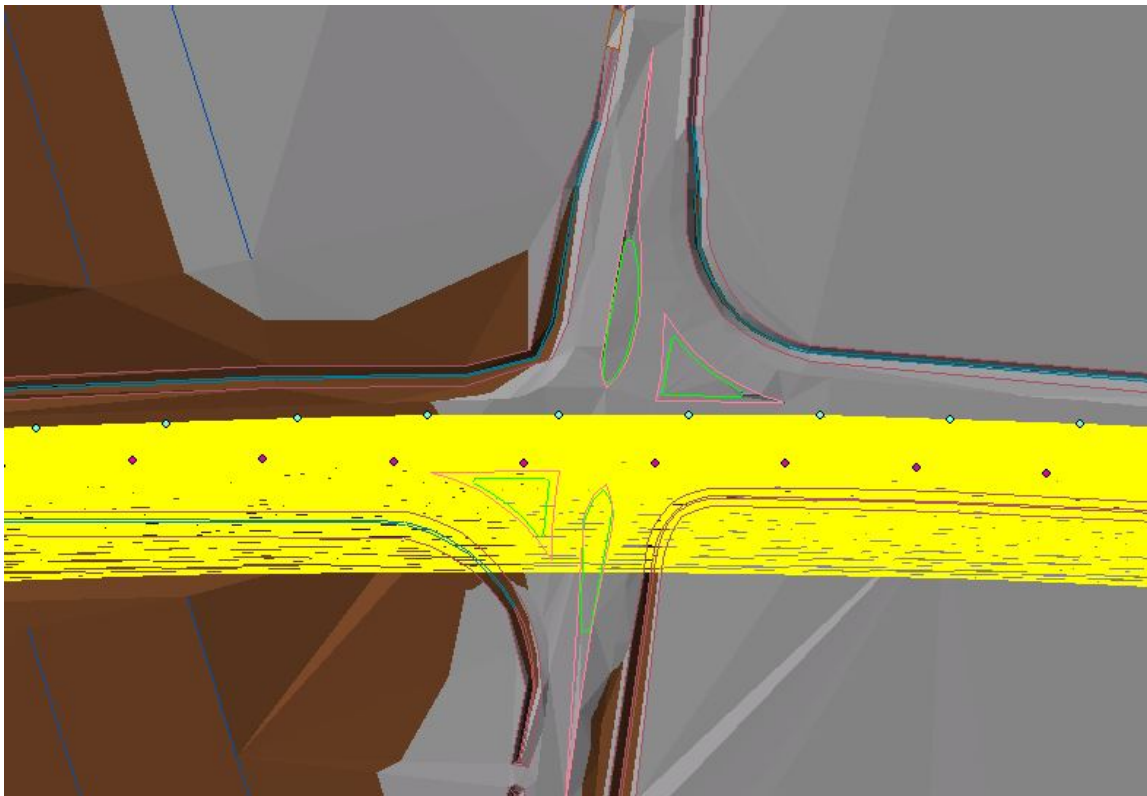
Az elvégzett javításoknak köszönhetően az adatok mennyisége lecsökkent, így a felesleggel már nem kellett foglalkoznom. A látóvonalak létrehozása során a program alapértelmezetten ad a vonalaknak attribútumokat. Mivel a vonalak-, és a pontok attribútumai nem egyeznek, nehéz eldönteni szemrevételezés nélkül, hogy az egyenes mely pontból indul és melyikben végződik. Az eldöntést és a későbbi elemzést segítő, fontos volt, hogy valamilyen módon összekapcsoljam az eltérő rétegek adattábláit. Adattábla kapcsolatot többféle módon is lehetőségünk van kialakítani, azonban mindegyik kapcsolat másképp reagál. Ahhoz, hogy különböző típusú rétegeket, mint a pont és a vonal össze tudjunk kötni a „Spacial Join” és az „Add Join” funkciókra együttesen van szükségünk. A „Special Join” kapcsolatot hoz létre az ugyanolyan típusú adatok között. Az összekötés több módon is történhet, így

- Tartalmazza;
- Metszi;
- Beleesik;
- Legközelebbi;

relációk alapján. Hogy egy „Special Join” típusú műveletet végrehajtsunk, a látóvonalakból elő kell állítani egy, vagy több pont típusú objektumot. Erre szintén találunk lehetséges megoldást, mellyel ki tudjuk külön rétegbe írni a vonalak kezdő-, és végpontját. A kezdő-, és végpontokat külön rétegekben helyeztem el, hiszen az egyikhez

a kezdőpont szelvényszámát, a másikhoz pedig a végpont szelvényszámát kapcsoltam. Az adattáblák összeillesztése minden esetben a „One to Many” összeillesztési paranccsal történt, mert több vonal is indult és végződött ugyanabban a pontban, így egy-egy nyompontra a vonalpontok adattábláiban több rekord is megfelelt. A kezdő-, és végpontokat a kapcsolat létrehozását követően persze össze kellett illeszteni ismét egy réteggé, ezután következhetett csak az „Add Join” funkció.

„Add Join” -al olyan táblázatokat tudunk összeilleszteni, melyekben van legalább egy közös oszlop, mint a látóvonalak és a látóvonalakból számolt pontok esetén az egyenesek azonosítói. Ha már megvannak a látóvonalak attribútum tábláiban a kezdő-, és végszelvény számok, akkor következhet a láthatósági vonalak képzése.

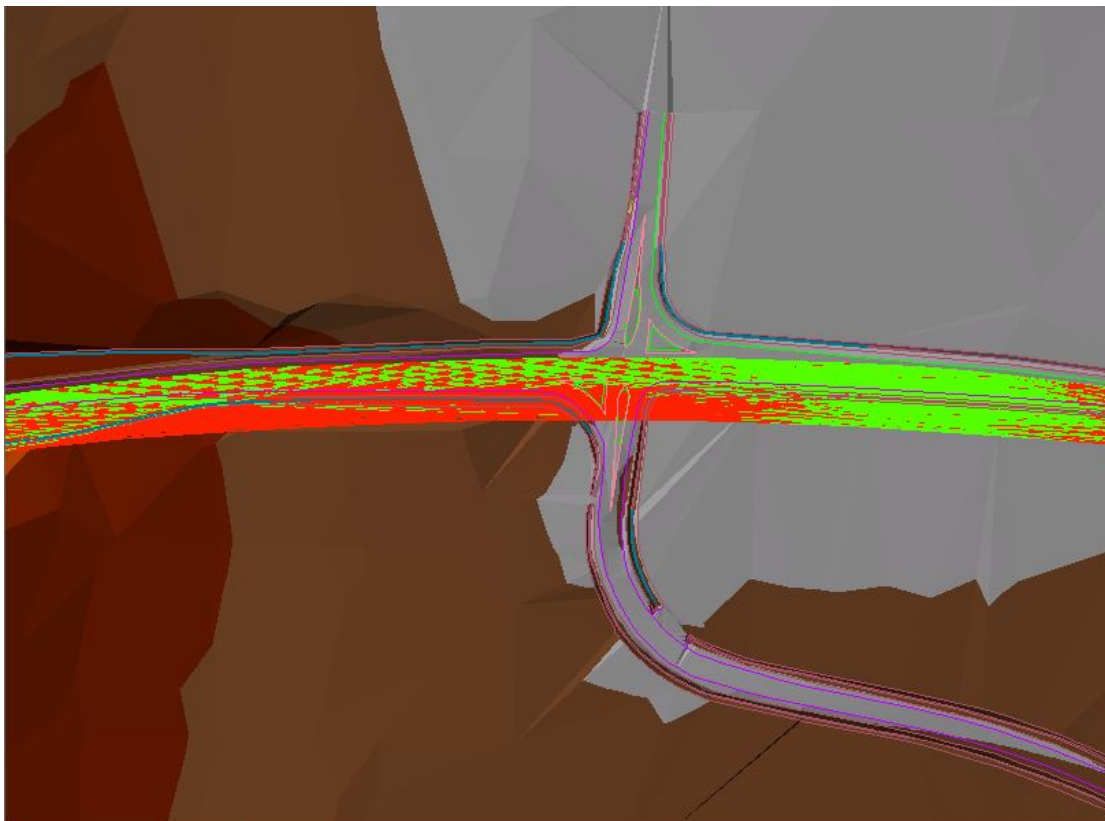


7-2. ábra Javításokkal ellátott és leválogatott látóvonalak [a szerző munkája]

7.2.3 Láthatósági vonalak

A látóvonalakból láthatósági vonalak előállításához szükségünk van egy raszter felületre. A raszter felület megléténél már csak annyi a dolgunk, hogy meg adjuk a programnak, melyek a bemenő adatok. Jelen esetben a látóvonalak, valamint a felület képi információi. A program ezek alapján a látóvonalakból létrehozza a láthatósági vonalakat. Sok különbség a látóvonal és a láthatósági vonal között nincsen, azonban egy jelentős dologban mégis eltérnek. Akkor, ha két pont között a légvonalas összelátás nem

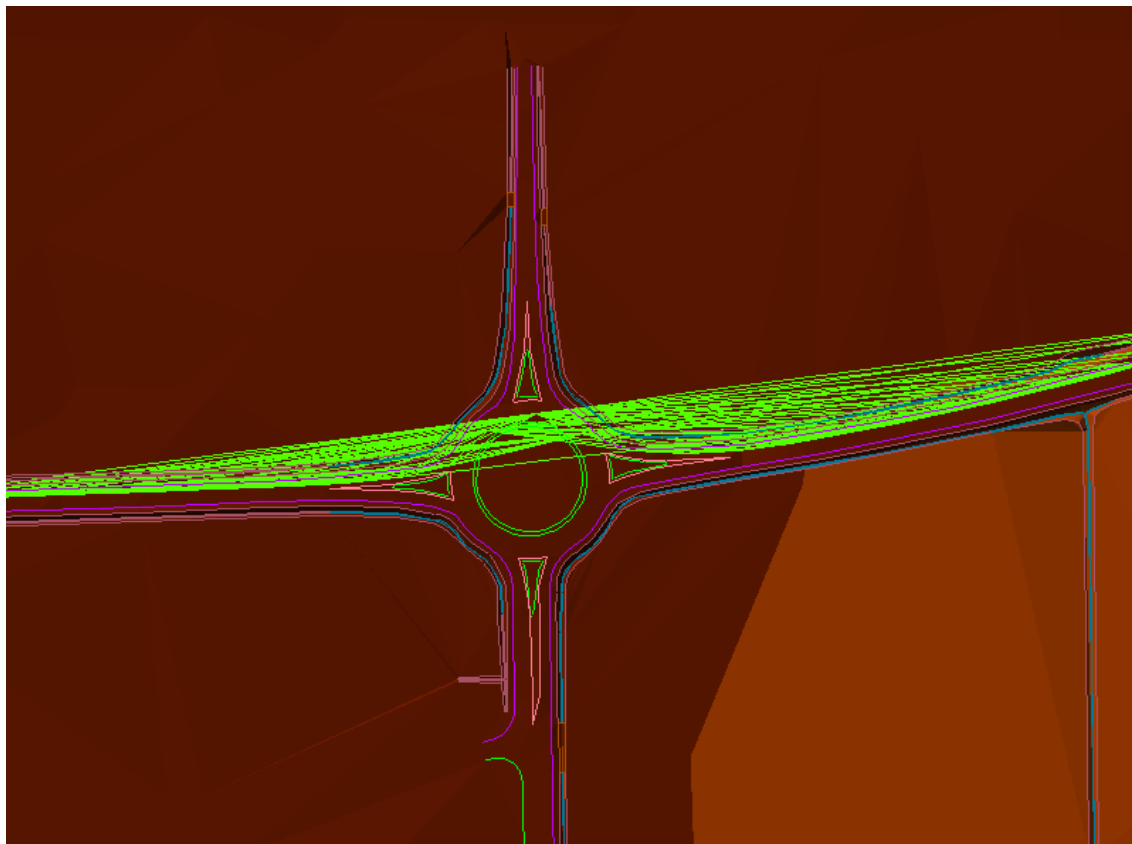
teljes hosszban van biztosítva, tehát a látóvonalnak van olyan része, mely belemetsz a terepbe, akkor a láthatósági vonal megtörik és ennek alapján kerül osztályzásra. Ez az osztályzást azt jelenti, hogy a „Line of Sight” funkció a bemenő egyeneseket vagy egészben hagyja, vagy több egymáshoz kapcsolódó darabra töri, attól függően, hogy a teljes felülete látszik, vagy sem. Ha két pont teljes egészében összelátható, akkor az eredmény egy teljes hosszában zöld vonal, egyébként annyi darabra törik, ahányszor belemetsz a terep ki-, és beugrásaiba. A láthatósági vonalakat a jobb szemrevételezés érdekében a program egy „VisCode”, azaz láthatósági kód nevű oszlop alapján 0, vagy 1 osztályba sorolja. A 0 értékek az adott pontból nem látszanak, és piros színezést kapnak, az 1-esek pedig a láthatók, ezáltal zöld színt vesznek fel.



7-4. ábra Láthatóság alapján tördelt egyenesek [*a szerző munkája*]

A kérdéses adatok végkifejletéig azonban még mindig nem jutottunk el, hiszen azok a vonalak, melyek több részből állnak, azaz van olyan részük, ami nem látszik, azok a további elemzésekhez nem szükségesek. A vonalakat éppen ezért valamilyen rendszer szerint összegeznem kellett. Erre megfelelőnek bizonyult a „Frequency” eszköz. Működésének lényege, hogy képes valamely oszlop szerint csoportosítani az adattáblában

lévő értékeket, és ebből a csoportosításból már leválogathatók a ténylegesen keresett információk.



7-5. ábra Teljes egészében látható egyenesek *[a szerző munkája]*

Utolsó lépésként az adatok további értékeléséhez a teljes egészében zöld láthatósági vonalak attribútumai egy Excel-táblázatban kerültek kiírásra.

A légvonalashoz hasonlóan az útvonal menti elemzésnél is meg kell tennünk bizonyos előkészítő lépéseket, melyek a program megfelelő strukturális felépítését és az egyértelmű lefutást szolgálják.

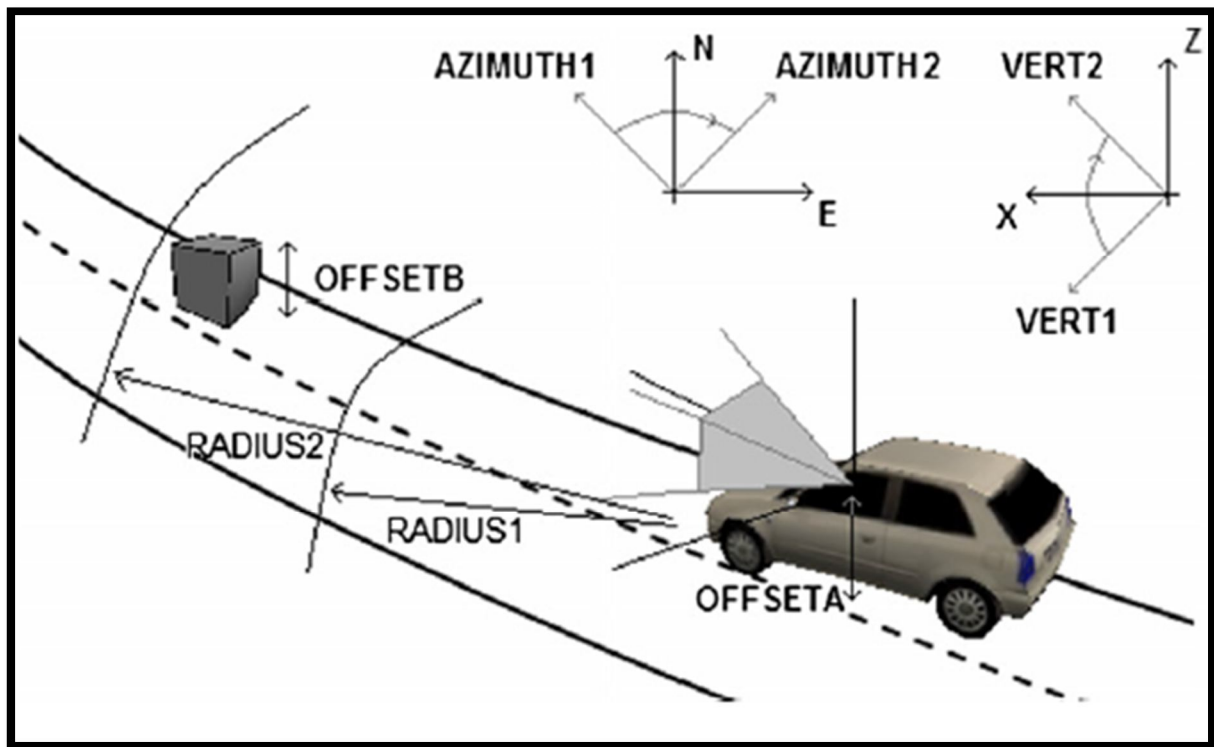
A legelső lépés itt is az importálás, mely teljesen ugyanazon elvek szerint történik, mint a korábban leírtak, azonban ezt követően a folyamatára persze megváltozik. Az útvonal menti vizsgálat folyamatábrájának részletét mutatja a *8-1. ábra*, a teljes utasítássorozatot a *2. melléklet* tartalmazza. A további lépésekben az adatok elsősorban nem leválogatásra kerülnek, hanem az adattáblákat egészítem ki újabb oszlopokkal, a későbbi funkciók korrekt végrehajtása miatt. Az ArcGIS „3D Analyst Tool” tartalmaz egy funkciót takarási ábrák számolásához, ez a „Viewshed” [ESRI, 2012c]. Lényege, hogy egy-egy kiválasztott pontban a program a terephez viszonyítva megnézi mi látható, és mi nem az. A láthatóság végeredményét az alkalmazás egy raszter adatsorban tárolja, mely „GRIDCODE” alapján kerül látható és nem látható részek szerinti osztályozásra. A „GRIDCODE” hasonló tulajdonságokkal rendelkezik, mint a légvonalas vizsgálatnál említett „VisCode”. Ahhoz, hogy egy ilyen elemzést végig tudjunk futtatni, nem kell mást megadnunk, mint a terepmodellt tartalmazó raszter állományt, valamint egy pontot, amelyre számolunk. Ha azonban különböző kikötésekkel kívánunk élni, melyeket a későbbiekben állítani is szeretnénk, akkor a beimportált pont adattáblájába többféle új oszlopot írhatunk, melyek a következő adatokat tartalmazzák [ESRI, 2012e; Castro et al., 2009]:

- SPOT: a magassági adat definiálása;
- OFFSETA: a kiinduló pont látómagassága, jelen esetben a sofőr szemmagassága, ez alapértelmezésként 1 m;
- OFFSETB: a szemlélési végpont magassága, megállásnál az útfelületen található 0,1 m magas tereptárgy, vagy akadály, előzésnél pedig a szemből érkező 1m magas jármű;
- AZIMUTH1: A kezdő irányszög az óra számlapját nézve az óramutatónak megfelelő haladási irányban;
- AZIMUTH2: A végső irányszög értéke;
- VERT1: a horizont alapján számolt magassági szög felfelé;
- VERT2: a horizont alapján számolt magassági szög lefelé;
- RADIUS1: a belső kör sugara, jelen esetben 0;
- RADIUS2: a külső kör sugara, itt a vizsgált látótávolság értéke.

Alapértelmezésben a program a következő táblázat második sorában található értékek szerint számol, melyek a megállási látótávolságnál a 3. az előzésinél pedig a 4. sor szerint alakulnak. A felsorolt állítási lehetőségeket szemlélteti a 8-2. ábra.

8-1. táblázat Viewshed beállítások [a szerző munkája]

	OffsetA [m]	OffsetB [m]	Azimuth1 [°]	Azimuth2 [°]	Vert1 [°]	Vert2 [°]	Radius1 [m]	Radius2 [m]
Alap	1	0	0	360	90	-90	0	Végtelen
Megállás	1	0,1	Irányszög -20°	Irányszög +20°	45	-45	0	600
Előzés	1	1	Irányszög -20°	Irányszög +20°	45	-45	0	600



8-2. ábra Beállítási lehetőségek szemléltetése [Castro et al., 2009]

Az azimut értékének meghatározásánál található egy irányszög nevű adat, melyből az AZIMUTH 1 és 2 oszlopokat számoltam. Az irányszög értékének meghatározását még az Excel-táblázatban végeztem el. Itt a kiválasztott pontról a haladási irány szerint következő pontra mutató geodéziai szögértéket jelenti. Ha figyelembe vesszük tehát, hogy az első azimut számolása 20° levonásával, a másodiké pedig 20° hozzáadásával történt, akkor megállapítható, hogy a két érték között egy 600 m sugarú 40° szélességű körívk kerül megvizsgálásra.

Az előbb látott adatoszlopok meghatározását követően természetesen valamilyen módon ki kellett számoltatnom a pontokból kiinduló takarási ábrákat. A takarási ábrák számolásánál fontos tudnunk, hogy mivel végeredményként egy raszter adatsort hoz létre a program, így ha egynél több megfigyelési álláspontot (Observer Point) adunk be és ezeknél a látható és nem látható részek kisebb nagyobb mértékben fedik egymást, akkor nem egyszerűen 0 és 1 értékeket keletkeznek. Két pont esetén a 0 és 1 értékeken kívül találni fogunk az adatsorban egy 2 értéket is, mely a közösen látható területet jelenti. Ezen okokból kifolyólag nem célravezető egy láthatósági vizsgálat esetén több pont bevitele egyszerre. A műveletet azonban mégis meg lehet oldani több pontra is, ekkor természetesen a program futásideje exponenciálisan nőni fog. A több pontra való takarási ábra számolásához már szükség van egy kis programozási tudásra.

8.2 Python programozás

Mint minden alkalmazáshoz, az ArcGIS-hez is tartozik egy fejlesztői környezet, ez a Python. A Python egy programozási nyelv, mely segítségével a programban alapról is helyet foglaló funkciókat egészíthetjük ki saját igényeink szerint. Az eszköztárakban található összes funkció a Model Builderbe importálva kiírhatóvá válik egy *.py nevű kiterjesztéssé, mely a Python programozás formátuma.

8.2.1 A Python programozásról általában

Fontos, hogy az ArcGIS-ből kiírt python programok visszafelé esetben változtatás nélkül nem fognak lefutni. A zavartalan működés érdekében kisebb módosításokat kell eszközölnünk. Példaként bemutatnám egy kiválasztó műveltet kiírt és javított python fájlját.

1) Kiírt fájl

Import arcpy module

import arcpy

Local variables:

Utpontok_ODA = Utpontok ODA

utpontok_oda_magassag_Select2 = kivlaszt

Process: Select

arcpy.Select_analysis(Utpontok_ODA, kivlaszt, "\"PSZ\" =1001")

Ha megnézzük ezt a fájl, akkor láthatjuk, hogy az első sor egy #-el indul. A # jelölés szolgál python programozásnál a megjegyzések hozzáfűzésére. A következő

sorban található egy import szócska. Mivel a python nem csupán az ArcGIS, hanem további más alkalmazás fejlesztői környezetének is alapja, így valamilyen módon meg kell határozni, hogy melyikben szeretnénk használni az általunk megírt scriptet. Az ArcGIS esetében e meghatározó szó az „arcpy”, ez minden egyes alkalmazott modul előtt beírandó, ezáltal fogja elvégezni a program a kért műveletet. A második részletben találjuk a használt változókat, ezek a művelethez szükséges rétegek. Lényegét tekintve a változtatás a rétegnevekre koncentrálódik. További módosítást jelent a tárolási útvonal hozzáadása. A következő bekezdésben látható az átalakított fájl.

2) Átalakított fájl

Import arcpy module

import arcpy

arcpy.env.workspace = "E:\\Modell\\Ideiglenes"

Local variables:

Utpontok_ODA = "Utpontok ODA.shp"

utpontok_oda_magassag_Select2 = "kivalaszt.shp"

Process: Select

arcpy.Select_analysis(Utpontok_ODA, utpontok_oda_magassag_Select2, "\"PSZ\" =1001")

Az elmondottak alapján megváltoztatott fájlt behívva, a kért művelet már hibaüzenet nélkül lefut.

Az előző fejezet részben említett takarási ábra létrehozásánál azért volt szükség python programozásra, mert egyfajta ciklikusságot kellett megoldanom, hiszen bár a „Viewshed” képes egyszerre több pontra láthatóságot számolni ez mégsem célorientált. A ciklikus futás megoldható a többi programnyelvben is megtalálható „for” utasítással. A „for” egy elől tesztelő ciklus, aminél meg kell határoznunk mely értékektől, mely értékekig fusson a program. Ez egy pár plusz percbe telhet, hiszen nem minden esetben tudjuk, hogy mennyi aktuális ponttal rendelkezünk az adott mintaszakaszon.

Egy ciklus futása mindösszesen néhány percre tart, ez azonban akár több órát is jelenthet egy több km hosszú szakasz esetén. Mivel a ciklus nem csak egy műveletet tartalmaz, ezért ez a több percre való futás teljesen reális. Jelen esetben az ismétlődés körülbelül 25 egymás utáni funkciót tartalmaz. A leírt javításokat természetesen itt is el kell végezni minden egyes funkció esetében.

Minden programozási nyelv másként reagál az egyes függvények argumentumaira. A pythonnál alapelvnek számít, hogy ha egy függvényt elkezdünk, akkor a tartalmi részleteket egy tabulátorral beljebb kell kezdenünk, különben a futás egy hibaüzenetet követően leáll.

A programozás elkészültével be kell olvastatnunk a fájlunkat, ez az ArcCatalog programrésszel tehető meg. Itt a különböző rétegdefiniálások mellett programozási scripteket és saját – a korábbiakban említett – „Toolbox”-okat is létre lehet hozni. Egy script készítéséhez nem kell mást tennünk, mint rákattintani a „New Script” fülre. Egy felugró ablakban a fájl többféle beállítási lehetőségét tudjuk változtatni. Itt adjuk meg, hol található az előzőleg megírt *.py kiterjesztés, továbbá a kimeneti és bemeneti adatokat állíthatjuk be. Mind a ki-, mind pedig a bemenet esetén különböző típusú információk használhatók, a könyvtárak helyétől kezdve a vektoros adatokon keresztül egészen a raszterekig bezárólag.

A vizsgálat során alkalmazott ciklust a továbbiakban részletesebben kívánom ismertetni, a teljes programozási fájlt a *3. számú melléklet* tartalmazza.

8.2.2 FOR ciklus a vizsgálatnál

A ciklus első lépésében lényeges, hogy valamilyen attribútum alapján kiválasztásra kerüljön először az első, majd – a benne foglalt utasítássorozat teljesülését követően – az azt követő pontok. Ez az attribútum a pontszám. Ez tűnt a legalkalmasabbnak, hiszen egyrészt egy egyedi ismertető, másrészt az egyszerűség kedvéért a pontszámokat oda irányban 1001-től, vissza irányban pedig 2001-től kezdődően kezdtem el számozni. A kiválasztás ismételtetése tehát a pontszám mező alapján történt.

Az első lépés a kiválasztó funkció, ahol minden esetben az útpontok rétegből leválasztásra kerül az aktuális pont. A program ez alapján egészen addig ismétli a cikluson belüli folyamatokat, amíg el nem jut az utolsó, a programozásban szereplő pontszámig.

A keresett pont leválogatását követően már csak ezzel az eggyel dolgozik tovább a többi művelet. Miután a leválogatás megtörtént, meg kellett határozni egy 40° szélességű 600 m sugarú, megadott irányba mutató körcikket, mely később a raszteres adatok vágását segítette elő. Ilyen művelet elvégzéséhez nem létezik alapfunkció, azonban az interneten található egy kiegészítő, mely hasonló elven végez szintén láthatósági

vizsgálatokat. A kiegészítőnél a programozók írtak egy python programot, mely az említett területet képes meghatározni. Ez egyfajta könnyebbséget jelent egy saját python program írásával szemben. Természetesen e programozási fájl is beolvasható az általunk írt adatok mellé, így könnyedén tudjuk alkalmazni a ciklus futásánál.

A műveletsorozat következő lépése már a korábban említett takarási ábra készítése, egyetlen pontra vonatkoztatva. A takarásnál, bár megadtuk előre a pontokból számolandó irányszögeket és távolságokat, ennek ellenére a program automatikusan a teljes területre állítja elő az adatokat. Az értékek megadásának a futási idő szempontjából van jelentősége. Ha az alapértékekkel dolgozunk, egyéni helyett, akkor a láthatósági ábra a teljes területre kerül kiértékelésre, tehát az egész domborzatról képet kapunk. Saját adatok használata során a látható részeket csak arra a szűk sávra határozza meg, melyet a bevitt információkkal lehatároltunk, a felület többi részét a pontból nem láthatónak tekinti.

Mivel a láthatósági ábra mindkét esetben a teljes területre kiszámolásra kerül, így ezt természetesen el kell vágnunk, hiszen a továbbiakban feleslegessé váló adatokkal való munka csak lassítaná a programot. A raszter vágását legegyszerűbben egy maszkolási művelettel oldhatjuk meg. Segítségével bizonyos területre vonatkozó képi információk nyerhetők ki az eredeti fájlból. E maszkolási művelet miatt volt szükség a felvázolt internetes kiegészítőre.

A kivágás végrehajtásával persze a továbbiakban is csak egy képi adatsorunk vannak, melyből vektoros adatokat kell létrehoznunk, hogy meghatározzuk mi az a távolság, ameddig egy-egy megfigyelőállásból ellátunk. Korábban volt már szó vektor raszter konverzióról, itt a fordítottjára van szükségünk. A két átalakítás hasonló elven működik, azonban, míg az előbbinél pontokról, vonalokról és felületekről térünk át képekre, addig az utóbbinál egy meglévő rasztert szeretnénk felületekké alakítani. Egy képi adatból egy vektoros létrehozásánál a program minden egyes képpontot megvizsgál, hogy mely vektoros elemhez tartozik. Ennek alapján létrehoz egy új réteget, melyben az összetartozó pontok egy felületet képeznek. Az átalakítás egyszerűen megoldható a „Raster to Polygon” paranccsal. Nem kell mást tenni, mint megadni a bemenő képi adatot és az átalakításhoz használt adattáblát.

Az átalakítás után már rendelkezésre áll egy vektoros adathalmaz a számolt – az adott pontból látható – minden egyes részletről. A létrehozott állomány azonban még nem tartalmazza sem a kiinduló-, sem pedig a róla látható pontok szelvényszámát. Ezen

halmazok összekapcsolásához ismét egy „Join” művelet szükséges. Korábban szóba került, hogy „Join” funkciókkal tudunk a különböző táblázatok között kapcsolatot létesíteni. A kapcsolat létesítés előfeltétele egy közös oszlop megléte a kiinduló és a kapcsolandó táblák mindegyikében. Csakhogy a konverzió lefutását követően még nem áll rendelkezésre ilyen, ezt először létre kell hoznunk. A kapcsolat műveletét jelen esetben a pontszám segítségével oldottam meg, hiszen ahogy a fejezet elején is ismertettem a ciklusfüggvény alapját is ez képezte. Mihelyst megvan mindkét adattáblában a közös oszlop, a kapcsolat kialakítható. E funkcióval így képesek vagyunk a kezdőpont szelvényszámát az attribútumokhoz adni.

Mivel a konverzió során létrehozott formátum egy felület típusú objektum, így egy felülethez több végpont szelvényszámát is hozzá kell rendelni. Első lépésben leválogattam azon adatokat a – konverzió során létrehozottakból –, melyek láthatósági kódja 1. Erre a későbbieket tekintve, a számolás Excel-táblázatba mentése miatt volt szükség. A leválogatást követően persze már egy újabb – most azonban „Special Join” – összekötés segítségével a végpontok szelvényszámát is hozzákapcsoltam az attribútum táblához. Fontos megjegyezni, hogy, amikor felület típusú objektumhoz próbáljuk egy pont típusú adatait hozzárendelni, akkor több lehetséges végkimenetel következhet be, attól függően, hogy hogyan végezzük el a kapcsolat kialakítását. A kialakításról bár korábban volt már szó, itt bővebben ismertetném az egyes lehetséges eseteket egy felület és pont objektumok között.

Az első lehetséges megoldás, amikor megadjuk, hogy a felület által tartalmazott összes pontot kapcsolja az adott elemhez. Ilyenkor a program csak a legkisebb azonosítójú pont kapcsolását végzi el, ha nem pipáljuk ki a „Keep all Join” lehetőséget. A „Keep all Join” hivatott arra, hogy több információt is egy másikhoz csatoljunk. E funkció használatát jelenti a második eset. Egy térinformatikai program sem képes azonban mindenre, így ahhoz, hogy megoldja az egy adathoz több kapcsolási gondot, létrehozza az eredeti objektum másolatát. A másolat annyiszor kerül tárolásra, ahány pontot a felület tartalmaz. Tehát ha az első pontból előrelátunk a 9.-ig, akkor a kapcsolat kialakításánál a másolás kilencszer történik meg. A replikák létrehozása persze adatnövekedéssel jár, azonban számomra mégis különösen előnyös volt a további eredmények kiértékelését tekintve.

8.2.3 Láthatósági animáció

Bár a láthatósági animáció létrehozása is a „For” ciklus részét képezi, mégis, eltérő volta miatt, külön alfejezetet érdemel.

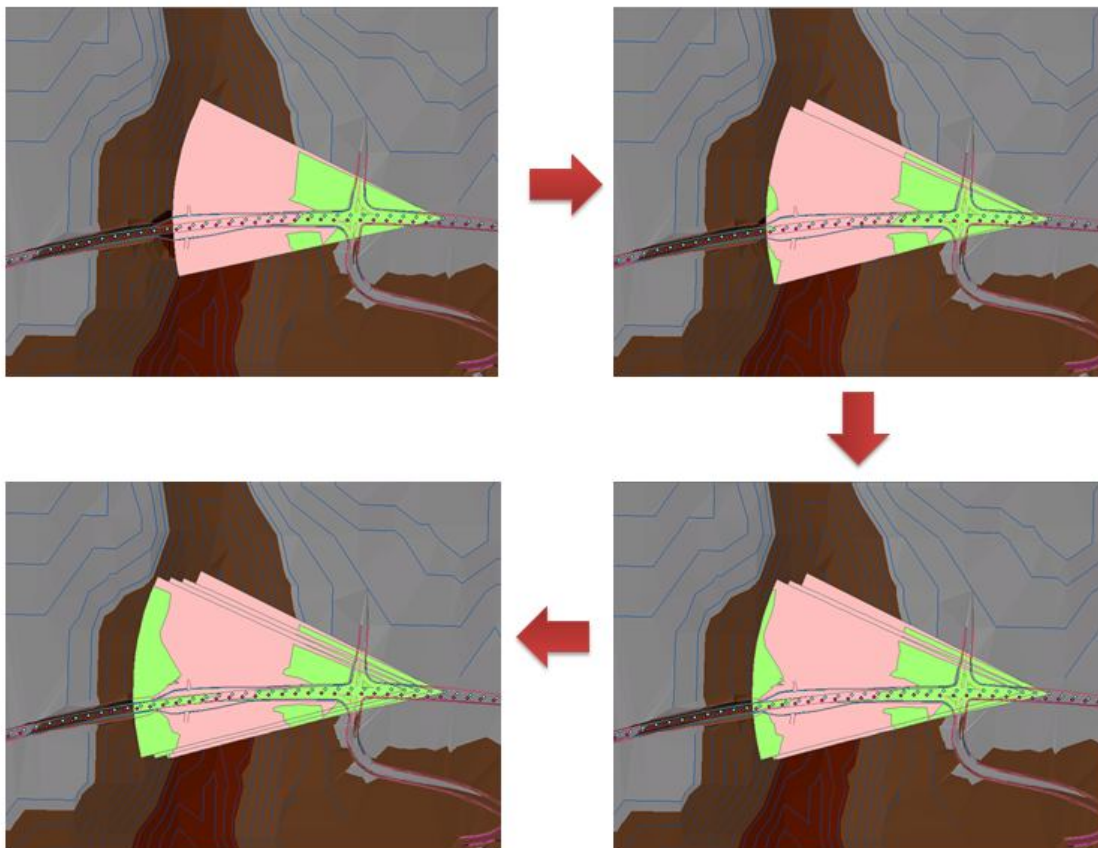
Az ArcGIS képes különböző animációk létrehozására, mellyel egyes folyamatok sokkal szemléletesebben mutathatók be a nagyközönség számára. Sok online videó lelhető fel az animációkkal támogatott demonstrálásokról, melyek az esetek többségében mindig a megértést segítik. Vegyük például egy toxikus anyagokat tartalmazó gáz terjedését egy lakott zónában. Az elemzések során kimutatható, hogy a gáz mely órában, hogyan terjedt szét. Ha azonban van a területről egy térképünk, továbbá ismerjük, hogy a toxikus felhő mely időpontban mekkora kiterjedésű volt, akkor egy ArcGIS-es animálás segítségével bemutathatjuk a terjedés folyamatát mozgóképekben is. A vizuális értelmezés szempontjából tehát egy animáció kiemelt jelentőséggel bírhat.

Az út láthatósági vizsgálata során hasonlóképpen jártam el, mint az előbb bemutatott szennyezőanyag terjedéses példában. Egy mozgó animáció létrehozásához minden egyes esetben kell egy idő adatsorral rendelkezünk, ami alapján meg tudjuk határozni, hogy az adott időpillanatban milyen adat kerüljön megjelenítésre. Egy-egy időadat attribútum táblához való csatolása egyszerűen megoldható egy új, üres oszlop létrehozásával. Az oszlop meglétét követően lehetőségünk van megadni különböző függvényeket, mellyel az oszlop celláit feltöltjük. A függvényadásnál – hasonlóan az Excel-hez – a matematikai függvényektől kezdve egészen a dátum és idő függvényekig bezárólag választhatunk. Mivel időpontot szeretnénk tárolni, ezért a legcélszerűbb a „Now”, azaz „Most” függvény. A ciklus lefutása során ilyenkor a „Time” oszlop az adat létrehozásának pillanatával fog tárolódni. Bár elegendő lenne, magának az időpontnak a megadása, a program mégis megköveteli egy másik oszlop a „Track ID” jelenlétét. Ez igazából csak másodlagos azonosításra szolgál.

A két oszlop megléte után a ciklus már csak törli a feleslegessé vált adatokat és kezdődik előlről a következő ponttal. Miután az utolsó pontra is lefutott, az egyes végeredményeket össze kell fűznünk egyetlen réteggé. Ez nem csak az animáció, de a kiértékelés céljait is szolgálja. Következtetéseket tehát, az egész területre vonatkozóan csak ezután vonhatunk le. Egy ilyen művelet elvégzéséhez az ArcGIS-ben rendelkezésünkre áll egy „Append” nevű parancs. Segítségével több, különálló rétegben található objektumot tudunk egy réteggé egyesíteni. Elő követelményként azonban kell

egy minta réteget gyártanunk. Fontos, hogy a minta meghatározásánál a réteg azon oszlopokat tartalmazza, melyekre szükségünk van a további számoláshoz. A lefutáshoz tehát mindösszesen egy mintára és a ciklusfüggvény segítségével előállított végleges láthatóságokra van szükség. A művelet végeztével az egyesített adatok a mintarétegben kerülnek megjelenítésre.

Az animáció létrehozására a „Tracking Layer” nevű művelet szolgál. Itt beállítható az időpontot tartalmazó oszlop neve és az időpont formátuma, valamint a követés egymásutániságáért felelős Track ID”. A program a művelet során létrehoz számunkra egy *.lyr fájlt, mely az ArcGIS által támogatott külön kiterjesztés. Ebben az *.shp fájllal (az ESRI saját formátuma) ellentétben különböző mozgatási műveleteket is tárolhatunk. Az animáció megalkotásával szemléletesebben értékelhetjük az út egyes pontjaiban a látási viszonyokat. Az animációra példát a 8-3. ábra mutat.



8-3. ábra Vizuális animáció a látótávolságok változásáról [a szerző munkája]

8.2.4 Adatok kivitele Excel-táblázatba

Természetesen a ciklus függvény és az egyesítés lefutását követően az eredményül kapott látótávolságokat értékelni kell. Az értékelés végrehajtása már a Microsoft Excel segítségével történt. Azért, hogy egy *.shp fájl adatait ki tudjuk vinni olyan formátumba – melyet az Excel is kezel –, nem kell különösebben keresgálnunk. Minden réteg esetében lehetőség van a benne található attribútumok megtekintésére. Ekkor egy táblázatos megjelenítést kapunk az összes objektum valamennyi információjáról. E megjelenítés már exportálható *.dbf formátumba, ezt az Excel már meg tudja nyitni.

9 Excel vizsgálat

Az említett *.dbf formátum az Excel programba minden gond nélkül behívható, így az ArcGIS által számolt különböző látótávolságok egy helyen értékelhetők. Minden egyes látótávolság (légvonal-, útvonal mentén értelmezett) egyetlen Excel-fájlban került feldolgozásra. Az összes rekord tartalmazza a következőket:

- Objektumazonosító;
- Kezdőpont szelvényszáma;
- Végpont szelvényszáma.

Légvonalaknál további attribútumok a vonal hossza, a láthatóság kódja, valamint a vonal azonosítója, melyből a program képezte a végleges láthatóságot. Útvonalak esetén pedig a kezdőpont X, Y, és Z koordinátája.

Ezek ismeretében létrehoztam egy négyzetes mátrixot, mely a láthatóság értékelésének alapját képezte. A mátrix elve a 9-1. táblázatban látható:

9-1. táblázat Példa a négyzetes mátrix végeredményére [a szerző munkája]

<--	Kezdőpont	Vég-pont →												
			55620	55640	55660	55680	55700	55720	55740	55760	55780	55800	55820	55840
	55620	1	1	1	1	1	1	1	1	1	1	0	0	0
	55640	1	1	1	1	1	1	1	1	1	1	0	0	0
	55660	1	1	1	1	1	1	1	1	1	1	0	0	0
	55680	1	1	1	1	1	1	1	1	1	1	0	0	0
	55700	1	1	1	1	1	1	1	1	1	1	0	0	0
	55720	0	0	0	0	1	1	1	1	1	1	0	0	0
	55740	1	1	1	1	1	1	1	1	1	1	1	1	0
	55760	1	1	1	1	1	1	1	1	1	1	1	1	0
	55780	0	1	1	1	1	1	1	1	1	1	1	1	1
	55800	0	0	0	0	0	1	1	1	1	1	1	1	1
	55820	0	0	0	0	0	0	0	1	1	1	1	1	1
	55840	0	0	0	0	0	0	0	1	1	1	1	1	1
	55860	0	0	0	0	0	0	0	1	1	1	1	1	1

A táblázat mind az oda, mind pedig a visszalátás viszonyait tartalmazza. Ha például oda irányban haladunk, és jelenleg az 55620 szelvényszámú ponton tartózkodunk, akkor előre látunk egészen az 55800-ig. Azaz itt a látótávolság értéke 180 m. Visszafelé irányban hasonló a helyzet, azonban, ha azt vesszük, hogy ismételten az 55620-as

pontban vagyunk, akkor a táblázat szerint csak önmagunkat látjuk. Ez azért lehetséges, mert ez a legelső pont. Ha megnézzük az 55760-as szelvényszámot, akkor látható, hogy ebből egészen 55620-ig előrelátunk. Tétélezzük fel, hogy önmagunkat minden esetben látjuk, ezek a pontok azok, melyeknél megegyezik a kezdő-, és végpont szelvény száma, így a jobb átláthatóság érdekében lila háttérszínezéssel vannak jelölve. A lila színű cellák, tehát a négyzetes mátrix átlójában helyezkednek el. Mindaz, ami a mátrix átlójának jobb oldalán található, az oda irányú láthatóságról nyújt információt, mindaz pedig, ami a bal oldalán, a vissza irányúrol.

9.1 A Microsoft Excel határai

A Microsoft Excel bár temérdek előre megírt függvénnyel rendelkezik, melyeket az egyszerű felhasználók egy kis olvasást követően képesek elsajátítani, de ezek alkalmazási tartományainak is megvannak a határai. Ha például szeretnénk kikeresni egy, a táblázatunkban lévő információt – mondjuk egy baleseti adatbázisból a 2009-ben bekövetkezett karambolok számát – akkor ez több függvény egymásba ágyazásával oldható meg. Ekkor még csak egy feltétel alapján kívántuk keresni. Mi történik, ha két, előre meghatározott értékünk van, melyek közül mindkettőnek eleget kell tennünk? Ekkor az előzőhöz hasonló módon megpróbálhatunk egymás után illeszteni különböző függvényeket. Ez azonban nem biztos, hogy helyes eredményt fog kiadni. Helyes eredményt akkor kapunk egy két feltételes keresésnél, ha e kettő közül legalább az egyik csak egyszer fordul elő az adatbázisban. Ha mindkét kritérium többször is megtalálható, akkor az összetett függvény végeredményként az első kritérium első előfordulási helyét fogja megadni. Erre példát a következő táblázat szolgáltat:

9-2. táblázat Mintapépda a kettős keresés adatbázisára [a szerző munkája]

OID	PSZ	Y	X	Z	Szelvényszám	Szelvényszám
1	1001	482683,49	172542,36	179,50	55620	55640
2	1001	482683,49	172542,36	179,50	55620	55660
3	1001	482683,49	172542,36	179,50	55620	55680
4	1001	482683,49	172542,36	179,50	55620	55700
5	1001	482683,49	172542,36	179,50	55620	55720
6	1001	482683,49	172542,36	179,50	55620	55740
7	1001	482683,49	172542,36	179,50	55620	55760
8	1001	482683,49	172542,36	179,50	55620	55780
9	1002	482674,15	172560,05	179,50	55640	55660
10	1002	482674,15	172560,05	179,50	55640	55680

A 9-2. táblázatban található oszlopok rendre a következők: Objektumazonosító, kiinduló pont száma, X, Y, Z koordinátája, a kezdő-, és a vég szelvénytérkép száma. Tehát, ha a kezdő szelvénytérkép alapján szeretnénk egy-egy értéket kikeresni, akkor ez nem fog megoldásra vezetni, hiszen, mint látható többször is szerepel ugyanazon pont. Ugyanez vonatkozik a végszelvénytérkép alapján indított keresésre is. A két kritérium szerinti függvényes keresésnél a különböző funkciójú függvények egymásba építése komplikált és szintén nem megfelelő. Egy két érték alapján kereső függvényt mutat be a következő sor:

„HA(ÉS(NEM(HIBÁS(HOL.VAN(\$B\$2;\$A:\$A;0)));NEM(HIBÁS(HOL.VAN(\$C\$2;\$B:\$B;0))));HA(HOL.VAN(\$B\$2;\$A:\$A;0)=HOL.VAN(\$C\$2;\$B:\$B;0);INDEX(\$A:\$C;HOL.VAN(\$B\$2;\$A:\$A;0);3);HA(INDEX(\$A:\$C;HOL.VAN(\$B\$2;\$A:\$A;0);2)=\$C\$2;INDEX(\$A:\$C;HOL.VAN(\$B\$2;\$A:\$A;0);3);HA(INDEX(\$A:\$C;HOL.VAN(\$C\$2;\$B:\$B;0);1)=\$B\$2;INDEX(\$A:\$C;HOL.VAN(\$C\$2;\$B:\$B;0);3);”Nincs találat”));”Nincs találat”))” [Nagy, 2011]

A példa alapján érthető, hogy a kettős keresés nem egyszerű, egymásba épített függvényekkel sem. Mint említettem ezek is csak akkor vezetnek helyes eredményre, ha legalább az egyik keresési érték csak egyszer szerepel az információhalmazban. Ha példának vesszük az 9-2. táblázatot és szeretnénk kikeresni az 55620 kezdőszelvénytérkép és 55680 végszelvénytérképhez tartozó információt, akkor akadályokba ütközünk. A kérdés egyszerűen csak annyi, hogy egyáltalán létezik-e olyan sor a táblázatban, ami ezeket az értékeket egyszerre tartalmazza. Az akadály ez esetben, hogy mindkettő többször is előfordul, így a függvény nem fog helyes megoldást találni.

A példából jól látható a megállapítás, miszerint az Excel előreprogramozott ismeretei is végesek. Ahhoz, hogy két információ alapján képesek legyünk keresni, saját magunknak kell egy „Makró” írunk.

9.2 Excel programozás

Az ArcGIS-hez hasonlóan az Excel-hez is tartozik egy fejlesztői környezet. E fejlesztői környezet esetében ismételt programozási tudásunkat kell kamatoztatnunk. A korábban említett python programozástól eltérően itt VBA¹² nyelv szerint írjuk meg az egyes utasításokat. Az előző alfejezetben említett makró tulajdonképpen egymás után

¹² Visual Basic for Applications

épített számítógépes utasítások sorozata. Minden olyan esetben, amikor túlságosan összetett munkát kellene végeznünk a beépített függvényekkel, vagy ugyanazon lépésekből álló feladatot kell újra és újra megismételnünk, célszerű egy makrót írni.

A makró írásnak kétféle lehetséges esete van. Az egyik, mikor előre megírunk bizonyos függvényekből álló utasítássorozatot, mely a táblázat megfelelő elemeire vonatkozik. Ezt követően egy makró-rögzítő segítségével az Excel maga végzi el a kódolást. Tehát ennek folyamán bekapcsolva a rögzítés funkciót, megmutatjuk, hogy mely értékeket mely cellák szerint számoljon. Ilyenkor a program maga állítja elő a futáshoz szükséges utasítássorozatot. [Aitken, 2001]

A másik lehetőség az Excelben található Visual Basic Editort használata. Itt már a makró elejétől saját magunk határozzuk meg, mit tartalmazzon a folyamatábra. Természetesen a két lehetőség egymással kombinálható is, így lehet olyan része az utasításoknak, melyet rögzítő segítségével veszünk fel, a továbbiakat pedig saját ismereteink szerint egészítjük ki. A diplomamunka során én a második lehetőség szerint dolgoztam. A továbbiakban egy kettős keresési funkcióval ellátott és egy diagramkészítést segítő makrót kívánok ismertetni.

9.2.1 Kettős keresés

Az egyes makrók logikailag egymást követő utasítások sorozatának összességéként tekinthetők. A korábbi megállapítások figyelembevételével mellett – mely szerint a függvények csak bizonyos tartományokban vezetnek eredményre – a kettős keresés elvégzéséhez nekem is egy saját makrót kellett írnom. A makróban található utasítássorozatok egymás után ellenőrzik az ArcGIS-el számolt adatokat. A keresett információ megléte esetén az eredménymátrixba 1-esek, az összes többi esetben pedig 0-k kerülnek. A kereső egyes részei egymás után kerülnek a továbbiakban bemutatásra.

1) Ismeretlenek meghatározása

A legelső sorban található **Sub** kifejezéssel kezdődik a VBA Makró. A kifejezés után meg kell határoznunk a makró nevét, mely vagy egyetlen szó, vagy alsóvonallal elválasztott szavak egymásutánja. A következő sorokba írhatók a programra jellemző információk. Mint minden programozási nyelvben, így a VBA-ban is jegyzetek fűzhetők az utasítások közé, mellyel nem csak a felhasználók, de maga a programozó is könnyebben el tud igazodni. Célszerű a makró működésének elvét saját magunknak is

lejegyzetelni a későbbi jobb áttekinthetőség érdekében. Hasonlóan a HTML nyelvben alkalmazott `<!-- -->` karakterek közé írt utasításhoz, a megjegyzés itt sem vesz részt a folyamatban.

Minden programnyelv esetében definiálnunk kell a használt ismeretleneket. VBA-nál a meghatározás a **Dim** szóval oldható meg, ezt követően írjuk be a keresett ismeretlent, valamint, hogy milyen értékeket vehet fel. Az értékek többfélék lehetnek, attól függően, hogy szöveges vagy szám formátumú eredményre törekszünk. Szám esetén sem mindegy, hogy a különböző intervallumok közé sorolható értéknagyságok közül melyiket választjuk saját elemzésünkhöz. Segítséget az egyes ismeretlenek megfelelő besorolásához a *Makro Mester* nevű internetes oldal nyújt. [MM, 2013]

Az ismeretlenek definiálása után a következő dolgunk a célmunkalap meghatározása, melyen a munkát végezzük. Ez a munkalap a fejezet elején található négyzetes mátrix munkalapja. A következő művelet „n” értékének a megadása, mely a vizsgált pontok darabszámát jelenti. A cellák meghatározásánál fontos tudnunk, hogy itt az oszlopok jelölése is számmal történik, tehát ennek figyelembevétele mellett kell meghatároznunk, hogy a program melyik oszlopból kérje be az információt. Cellaadásnál a „Cells” szó után zárójelbe írt első érték a sor, a második pedig az oszlop jelölése. Mivel az előző lépés folyamán meghatároztuk a cél munkalapunkat, így, ha a „Cells” szót íránk csak be, akkor a céllapról várná az értékeket. Ez így azonban nem jó, hiszen ez egy eredménylap, tehát a pontos munkafüzet meghívásához a „Worksheets” szót kell használnunk. Ez után zárójelben, idézőjelek közé írva meghatározhatjuk a pontos munkalapot. Megemlíteném, hogy a VBA programozás során figyelemmel kell lennünk a pontos névadásra is, mert a nyelv külön kezeli a kis-, és nagybetűket.

2) Többszörös ciklusfüggvény

Azért, hogy az ArcGIS-ből exportált látótávolságokból meg tudjak határozni egy összehasonlító diagramot, első lépésként elő kellett állítanom a korábban említett négyzetes mátrixot. Ehhez ismernem kellett, hogy a kinyert több ezer adat mindegyike mely ponthoz tartozik. A kettős keresés ezt a problémát hivatott megoldani. Az előző pontban ismertetésre került, hogy egy VBA programozás során, hogyan kell meghatározni az ismeretleneket és megválasztani az eredmény munkafüzetet. A továbbiakban a kereső ciklus lényegét szeretném bemutatni.

Ahhoz, hogy kinyerjük az értékes információt egy nagy halmazból, végig kell futtatnunk egy ciklust az adatok sokaságán. A ciklus azért érdekes, mert nem egyetlen adatot kívánunk megkeresni, hanem mindig az aktuális értéknek megfelelőt. A ciklusfüggvény nem egyetlen oda-vissza ugrást tartalmaz, hanem négy különbözőt. A négy ciklusból kettő egy „For”, kettő pedig egy „Do Loop” utasítás. A „For” a korábban említettekhez híven egy számlálós utasítás. Ez egészen addig fut, amíg az előre beállított értéket el nem éri a program. Ezzel szemben a „Do Loop” egy hátul tesztelési ciklus, melynél a „Do” utasítás után meg kell határoznunk egy feltételt. A hátul tesztelés miatt a folyamat egészen addig ismétlődik, amíg a megadott feltétel igazsá nem válik.

A „For” ciklusnál „a” és „b” változók az eredménymátrix sorát és oszlopát jelölik. E sor és oszlop határozza meg, melyek azok a szelvényszámok, amiknek kíváncsiak vagyunk a meglétére. A „Do Loop” ciklus esetén „i” és „j” a meglévő látótávolságok kezdő-, és végszelvényének számai. Fontos a ciklusok helyes lefutása miatt, hogy az ArcGIS-ből exportált információt ABC sorrendben olvassuk be az Excelbe. A jobb értelmezhetőség kedvéért egy példán vezetem végig az elmondottakat.

Kezdetnek vegyünk egy oda-, és vissza irányban 3000-3000 rekordot tartalmazó adatbázist és egy 56000-es és 58000-es szelvényszámok közötti útszakaszt. Az adatbázisban található információk a két szelvény közötti, 20 m-enként elhelyezkedő pontok látótávolságai. Érdeklődésünk tárgya, hogy látható-e az 56000-es szelvényből a tőle 300 m-re lévő. Az ArcGIS a számolási részt már elvégezte, már csak választ kell adnunk a feltett kérdésre. Mivel a programozott állomány egy „For” ciklussal kezdődik, így első lépésként „a” kerül behelyettesítésre. Az „a” értéke mindig az aktuális kezdő szelvényszám sora, itt most 3. Ennek ismeretében kerül behelyettesítésre „b” értéke is, mely az aktív végszelvény oszlopa. Megjegyezném, hogy az adatbázisban nem található olyan érték, melynél a kezdő-, és végszelvény egyezik, azonban tudvalevő, hogy saját magunkat látjuk. Erre a megállapításra visszavezetve az első „ha” függvény ezt határozza meg. Ha tehát a kezdő és végpont szelvény száma egyezik, az eredménymátrix cellája 1-et vesz fel. Mivel mi jelen esetben az 56000-es kezdőszelvényhez tartozó 56300-as végszelvényt keressük, ezért a program folyamatosan 20 m-enként lépkedve az előtte lévő szelvények létezését is megvizsgálja. Viszont, ha úgy vesszük fel „b” értékét, hogy ez már indulásként az 56300-as oszlop legyen, akkor az előtte lévők nem kerülnek kiértékelésre. A két megoldás nem különbözik lényegesen egymástól, így általánosságban írhatók a továbbiak.

Az „a” és „b” értékek alapján a program kiválasztja a megfelelő sorban és oszlopban található szelvényszámot, ez a KP (kezdőpont) és VP (végpont) érték. Ezután következik az említett „ha” függvény. Jelen esetben, mivel $KP=56000$, $VP=56300$, így az állítás nem igaz, ennek alapján a program az „Else” (Különben) sorra ugrik. Itt indul a hátul tesztelés „Do Loop” ciklus. Feltétel, hogy a keresett cellának végeredményben valamilyen értéket tartalmaznia kell. A folyamat egészen addig ismétlődik, amíg valamely – az intervallumon belüli – megállapítás igazzá nem válik. E ciklusban ismételten találunk ismeretlenek, ezek az LK1 és LV1. Ezen paraméterek a kiexportált adatmunkalapra vonatkoznak, ami az oda látótávolságokat tartalmazza. LK1 és LV1 értéke a folyamat előrehaladásával folyamatosan változik, ismételten addig, míg az utasításoknál az egyik állítás igaz nem lesz. Ha igaz lesz, akkor LK1 és LV1 értéke újra az első látótávolságtól indul és a program lép a következő végszelvényre (VP).

A „Do Loop” cikluson belül találunk egy hasonló, szintén „Do Loop” utasítást. Itt csak úgy, mint az előző esetben, van két ismeretlen. Ezek az LK2 és LV2 értékek, ezek azonban már a vissza látótávolságokra vonatkoznak. Az alapok tisztázását követően tehát a program működési folyamata a következő:

Adatok:

- $KP=56000$
- $VP=56300$

Lépések:

- If $KP = VP \rightarrow$ Nem teljesül $\rightarrow$ Else
- If $KP > VP \rightarrow$ Nem teljesül $\rightarrow$ Elseif
- Elseif $LK1=KP$ And $LV1=VP$
 - tételezzük fel, hogy az első adatsor $LK1=56000$ és $LV1=56020$
 - Tehát nem teljesül $\rightarrow$ Elseif
 - Ha teljesülne, akkor a cellába 1 kerül beírásra
- Elseif $LK1 < KP$ Or $(LK1=KP$ And $LV1 < VP)$
 - Ha ez igaz, a következő „i”, vagy „j” értékre ugrik
 - Ha nincs a keresésnek megfelelő elem, akkor az eredmény 0

Mint minden függvényi utasítást, a ciklusokat is le kell zárunk. Erre szolgál a „Do Loop” esetén a Loop parancs, mely előtt meg kell határoznunk, milyen ráta szerint

növekedjen a halmazban keresett sor száma. Erre hivatott az „i” érték szerinti futásnál a Loop sor előtti $i = i + 1$ egyenlőség. A „For” ciklus lezárása a Next szócska segítségével oldható meg, így ez tényleg csak addig ismétlődik, amíg a számláló el nem éri a maximálisan meghatározott értéket.

A bemutatott VBA programozást teljes terjedelmében a 4. számú melléklet tartalmazza.

9.2.2 Maximális látótávolság meghatározása

Bár a kettős keresés végeredményeként kapott négyzetes mátrix jó alapot nyújt az elemzések értékeléséhez, nem határozható meg belőle egyértelműen az egyes távolságok eltérései. Az eltérések érzékeltetésének szemléletesebb módja egy új táblázat létrehozása. A következőkben ismételten egy VBA makrót szeretnék bemutatni, mely a távolságok számolásához és értékeléséhez nyújt segítséget.

9-3. táblázat Az elkészült négyzetes mátrix első néhány sora [a szerző munkája]

↓ Kezdőpont	Vég- pont -->	Négyzetes mátrixszámolás									
		55620	55640	55660	55680	55700	55720	55740	55760	55780	55800
	55620	1	1	1	1	1	0	0	0	0	0
	55640	1	1	1	1	1	0	0	0	0	0
	55660	1	1	1	1	1	1	0	0	0	0

E makró lényege, a megfelelő adatok kikeresése a négyzetes mátrixból. A megfelelő szó, itt arra utal, hogy bár minden egyes pontról rendelkezhetünk adatokkal, azonban ezek többsége továbbra is felesleges. Mind a megállási-, mind pedig az előzési látótávolság meghatározásához a korábbiakban említett 600 m sugarú környezetben található adatok értékesek számunkra. A négyzetes mátrix nem csak az aktuális ponttól 600 m-re lévő információkat tartalmazza, hanem a teljes útvonal adatait is, függetlenül attól, hogy bizonyos távolság megtétele után valószínűleg nem látszik semmi. A makró tehát kikeresi a mátrixból azokat a pontokat, melyek távolsága az aktuális helyzetünktől kisebb, vagy egyenlő, mint 600 m. Nem csak a távolsági követelménynek, de a láthatóságnak is eleget kell tenni, így a mátrix pillanatnyi cellájában 1-es kell, hogy

legyen. A diagramkészítés előfeltételének számító táblázat előállítását ismételten egy példán keresztül szeretném bemutatni.

Jelen esetben vegyük alapul a rendelkezésünkre álló, immáron teljesen kitöltött négyzetes mátrixot, melynek első néhány sorát mutatja a 9-3. táblázat.

A táblázatból látható, hogy az 55620-as szelvényszám esetén 80 m-re látunk előre zavartalanul. A 100 m-re lévő pontot nem látjuk, majd a 120 m-re lévő ismét érzékeljük. Ebben az esetben tehát, akadályok belógása nélkül maximálisan csak 80 m a látóvonalunk hossza. A makró a maximálisan, akadálymentes környezetben való előrelátást határozza meg. A kettős keresésnél említett ciklusok e programozásban is megtalálhatók, hiszen többféle munkalapról kell meghatározni az aktuális pontból látható legnagyobb távolságot. A programban alapadatként most a munkalapok nevei, valamint a kezdő-, és végpont szelvény számai szerepelnek. Első ciklusként meg kell határoznunk, hogy mely munkalapról kezdje kikeresni az információkat és a keresés melyikig tartson. Ez minden esetben egy-egy cellahivatkozással oldható meg.

A ciklus egyik munkalapot a másik után vizsgálja át. Ezt követően egy „For” utasítás végzi el az egyes munkafüzetekben található adatok értelmezését, és megadja, hogy mely esetekben mekkora a látótávolság. A meghatározáshoz természetesen számolni kell az egyes távolságok értékét, ezt a „Tav” változó tartalmazza. Korábban említettem, hogy nem csak a távolságot kell megadnunk, hanem a láthatóságot is. A láthatóságról a „Latas” változó szolgáltat információkat. A „Latas” mindig az adott szelvény számokhoz tartozó értéket jelenti. A 9-3. táblázat 55620 kezdőponthoz és 55720 végponthoz például 0 tartozik. A „Tav” minden egyes esetben addig nő, amíg a program el nem jut az első olyan celláig, ahol a „Latas” 0 értéket vesz fel, vagy el nem éri a 600 m-es küszöböt. Ha ez megtörténik, akkor lép a következő szelvény számra a keresés.

A bemutatott ismertetést követően az utasítások sorozata a következőképp alakul:

Adatok:

- a= a munkalap nevét tartalmazó cella
- b= az „a” cella tartalma (szöveg)
- KP= a kezdőszelvény száma
- VP= a végszelvény száma

- n = a pontok száma
- i és j az aktuális mátrix sora és oszlópa

Lépések:

- Mátrix munkalapjának kiválasztása
- KP és VP értékek alapján „Tav” meghatározása $\rightarrow \text{Tav} = \text{VP} - \text{KP}$
- „Latas” értékének kiválasztása
- If $\text{Tav} \leq 600$ és $\text{Latas} = 1$
 - Ha igaz, akkor a keresés addig folytatódik, amíg a feltétel teljesül, majd a maximális érték kerül beírásra
 - Ha már nem igaz, akkor a program lép a következő szelvényre

A távolságok meghatározásának a végeredményét a következő táblázat mutatja. Itt minden egyes megoldási módhoz a saját négyzetes mátrixában foglalt információkat rendeli hozzá a program. A 9-4. táblázat részletet mutat az egyes megoldások során fennálló maximális látótávolságokról. A táblázatban látható megoldások, útvonal menti vizsgálat esetén rendre a következők: UVG1m – Google domborzat 1x1 m-es felbontással, UVG32m – Google domborzat 32x32 m-es felbontással, UVGU – Google domborzat javítva a megvalósulási terv adataival, UVF – FÖMI domborzat eredeti felbontással, UVFU – FÖMI domborzat javítva a megvalósulási terv adataival.

9-4. táblázat A keresés végeredményét tartalmazó táblázat egy része 100 m-enkénti bontásban színezve [a szerző munkája]

	ODA					VISSZA				
Útszelvény	UVG1m	UVG32m	UVGU	UVF	UVFU	UVG1m	UVG32m	UVGU	UVF	UVFU
55620	200	0	280	240	280	0	0	0	0	0
55640	180	0	280	220	280	20	0	20	20	20
55660	160	20	280	200	280	40	0	40	40	40
55680	140	0	280	180	280	60	20	60	60	60
55700	140	0	260	160	260	80	80	80	80	80
55720	120	0	260	140	260	100	20	100	100	100
55740	100	0	260	120	260	120	0	120	120	120
55760	100	20	260	100	260	140	0	140	140	140
55780	120	0	280	100	300	160	0	160	160	160

9.2.3 Dinamikus diagram készítése

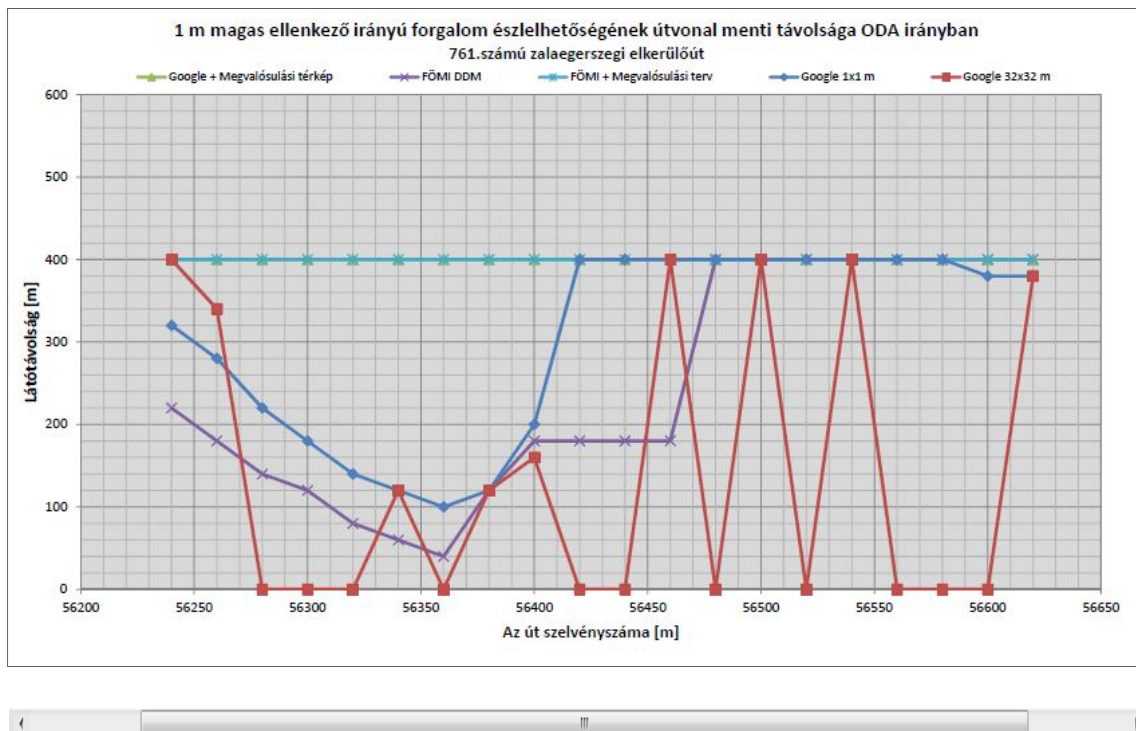
Az előzőekben bemutatott táblázatból következő lépésként diagramokat szerettem volna előállítani. Annak érdekében, hogy egy akár több km hosszú útszakaszról ne kelljen minden egyes km-szelvény adatait külön-külön grafikonon megjeleníteni, ezért egy mozgó adatsorral dolgozó műveletet gondoltam ki.

9-5. táblázat Dinamikus adatok csúsztható sávval [a szerző munkája]

		Google 1x1 m	Google 32x32 m	Google + Megvalósulási térkép	FÖMI DDM	FÖMI + Megvalósulási terv
1		ODA				
	Útszelvény	UVG1m	UVG32m	UVGU	UVF	UVFU
1	55620	200	0	280	240	280
2	55640	180	0	280	220	280
3	55660	160	20	280	200	280
4	55680	140	0	280	180	280
5	55700	140	0	260	160	260
6	55720	120	0	260	140	260
7	55740	100	0	260	120	260
8	55760	100	20	260	100	260
9	55780	120	0	280	100	300
10	55800	400	220	320	80	320
11	55820	400	0	400	60	400
12	55840	400	0	400	60	400
13	55860	400	0	400	400	400
14	55880	360	20	400	400	400
15	55900	320	140	400	400	400
16	55920	280	0	400	400	400
17	55940	240	0	400	240	400
18	55960	220	0	400	180	400
19	55980	180	0	400	140	400
20	56000	140	0	400	120	400

A 9.4 táblázatban részletezett eredmények dinamikus adatsor létrehozására nem alkalmasak, ezért ezeket egy új táblázatba kellett „eltolnom”. Az Excelben található egy „Eltolás” nevű függvény, ez megfelelő alapul szolgálhat egy mozgó grafikon előállítására. A függvélynél bemenő információként hivatkoznunk kell az egyes cellák helyeire, valamint – jelen esetben – az új 9-5. táblázat bal felső sarkára. E sarokcellához fogjuk majd a későbbiekben a mozgatósávot hozzárendelni. A következő táblázat az újonnan létrehozott, csökkentett adatsort mutatja be.

A 9.5 táblázatból jól látható, hogy csak 20 sort tartalmaz. Egy-egy diagram megjelenítési tartományához a 20 sor éppen elegendő. Az eltolás függvény egyszerű alkalmazásával még nem oldottuk meg a mozgás problémáját, ehhez egy csúszka is szükséges. Az Excel fejlesztői környezete nem csak makrók írására szolgál, de a megírt makrókat hozzá is lehet rendelni megfelelő gombokhoz. Egy gomb alkalmazására már láthattunk példát a 9.3 táblázatban (*Négyzetes mátrix számolása*). Ilyenkor ez Excelben található, előre definiált gombokból választhatunk, de létrehozhatunk újakat is. Az egyik előre megalkotott gombfajta a csúsztatható sáv, melyet akár vízszintesen, akár függőlegesen is felvehetünk. A csúszkát ismételten valamelyik cellához kell kötnünk. Ez a cella az eltolás során alkalmazott bal felső sarok. A mozgató sávnak különböző tulajdonságait tudjuk állítani, így többek között a mozgástartományt is. Ez az intervallum fog megjelenni a sarokcellában. Mivel az eltolás függvénnyel is hivatkoztunk erre a cellára, így az első adatsort leszámítva a továbbiak változtatást igényelnek. A cellahivatkozásnál hozzá kell adnunk a csúszkaléptetés egységét is – mely a 9.4 táblázat megadott sorára fog vonatkozni –, így minden egyes mozgatsávnál a 9.5 táblázatban változni fognak a látótávolságok.



9-1. ábra Megállási látótávolságok változása mozgatósávval [a szerző munkája]

A dinamikus táblázat meglétét követően már lehetőség nyílik különböző mozgó grafikonok kidolgozására. A grafikonon való mozgáshoz elméletileg elég a 9.5 táblázat mellett található mozgatósáv, azonban az egyszerűség kedvéért célszerű a diagramokhoz is beszúrni egy-egy csúszkát. A 9-1. ábra egy dinamikus grafikonot mutat be, melynek vízszintes tengelyén az út szelvényszámait, függőleges tengelyén pedig a számolt távolságértékeket találjuk. A grafikonon az út nyomvonala mentén történő vizsgálat eredményei vannak feltüntetve, a különböző megoldási módok eloszlásában ODA irányban. Szembetűnő lehet, hogy a piros vonal teljesen szabálytalan távolságoként 0 értékeket vesz fel. Ez a vonal a 32x32 m-es felbontású, eredeti Google domborzatból számolt előzési látótávolságokat mutatja be. A diagramból is megállapítható, hogy a javítatlan Google információk többségében nem alkalmasak a látótávolságok értékelésére, ugyanis felbontásuk nem elegendő a megfelelő következtetések levonására, de erről az *Eredmények értékelése* című fejezetben bővebben lesz szó.

Dinamikus mozgatóval többféle diagramot hoztam létre az előzési-, és a megállási látótávolságok kiértékelési eredményeinek bemutatására. A létrehozott grafikonok a következők:

- Vonal-;
- Oszlop-;
- Terület diagram.

A különböző kiértékelések (légvonalas-, és útvonal menti módszer) során kapott eredmények összehasonlítását nem tartottam relevánsnak, hiszen a két eljárás nem ugyanazon elvek szerint dolgozik. Az egyes módszereken belüli eltérő adatokból történt számolások eredmény grafikonjait azonban megalkottam. Az eredmények felvázolásánál az ismertetett 5 féle megoldásból általában 3 került ábrázolásra, melyek a következők:

- Google SRTM 1x1 m-es felbontással
- Google SRTM pontos vonalvezetéssel javítva
- FÖMI DDM pontos vonalvezetéssel javítva

Ahogy a 9-1. ábrán bemutatott grafikonból is látható a 3 féle megoldás közül a javított modellek nem térnek el lényegesen egymástól.

9.3 AutoCAD hossz-szelvény Excel adatokból

Dinamikus mozgatas segítségével bár az egyes látótávolságok változásai szemléletesebbek, azonban még így sem kapunk tökéletes képet arról, hogy ezek miként alakulnak a vízszintes-, és a magassági vonalvezetés függvényében. A vonalvezetést legegyszerűbben egy hossz-szelvény szerkesztésével tudjuk bemutatni. Egy meglévő főútnál, ahhoz, hogy a hossz-szelvényt tökéletesen, a jelenlegi állapotnak megfelelően megrajzoljuk, ismerni kell a kitűzési adatokat, az átmenti-, és tisztakörívek helyét és nagyságát, továbbá a magassági lekerekítések pontos adatait.

A zalaegerszegi 761. számú elkerülő esetén sajnos csak a megvalósulási térkép állt rendelkezésemre, így a pontos ívviszonyokat nem ismertem, csak közelíteni tudtam. Az említett okokból kifolyólag az irányviszonyokat bemutató, a hossz-szelvényen található ábra is csak egyenesekből és tiszta körívekből áll, valamint a magassági lekerekítő ívek helyei sincsenek berajzolva.

Hossz-szelvényi megjelenítést a légvonalas eredmények értékelésére nem alkalmaztam, mert a végső következtetések levonása során megállapítható volt, hogy az út nyomvonala mentén történő elemzés sokkal pontosabb értékeket adott eredményül. Az átnézeti hossz-szelvény esetén is az előző alfejezetben említett 3 féle adatsor került feltüntetésre oda-, és vissza irányban külön-külön, előzési-, és megállási látótávolságok szerinti bontásban. A látótávolságok értékeivel ellátott, a vízszintes-, és magassági vonalvezetéseket is bemutató hossz-szelvényeket teljes terjedelmükben a *8. és 9. számú mellékletek* tartalmazzák.

9.4 A vonalvezetés és a látótávolságok összehasonlítása

A hossz-szelvények segítségével, mind a megállási-, mind pedig az előzési látótávolságokkal kapcsolatban több megállapítást is tehetünk a kiválasztott útszakaszra vonatkozóan. Általánosságban elmondható, hogy a Google domborzat még a lineárisan interpolált adatokkal is rosszabb látási viszonyokat eredményez, mint a tervezési térkép információival javított megoldások.

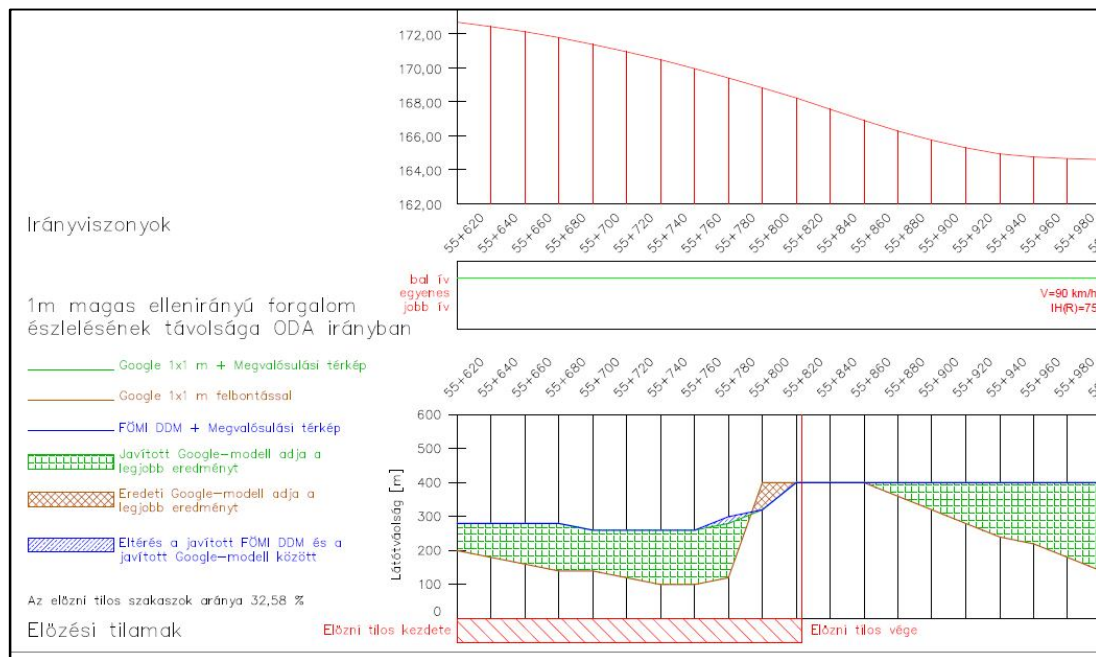
Hossz-szelvényi nézetben az úton haladva folyamatosan elemezhetjük, hogy az ábrázolt 3 féle megoldás közül, az adott szelvényben melyik szolgáltatja a leghosszabb távolságértékeket. Elöljáróban kijelenthetjük, hogy minden bizonnyal nem az 1x1 m-es felbontású Google adatsor lesz a legmegbízhatóbb érték, még akkor sem, ha helyenként esetleg a javított modellekből számolt végeredményeknél nagyobb látótávolság adódik. A

következőkben részletesen bemutatom, miként alakulnak a vonalvezetés függvényében az egyes vizsgálatok végeredményei.

1) Előzési látótávolságok alakulása oda-, és vissza irányban

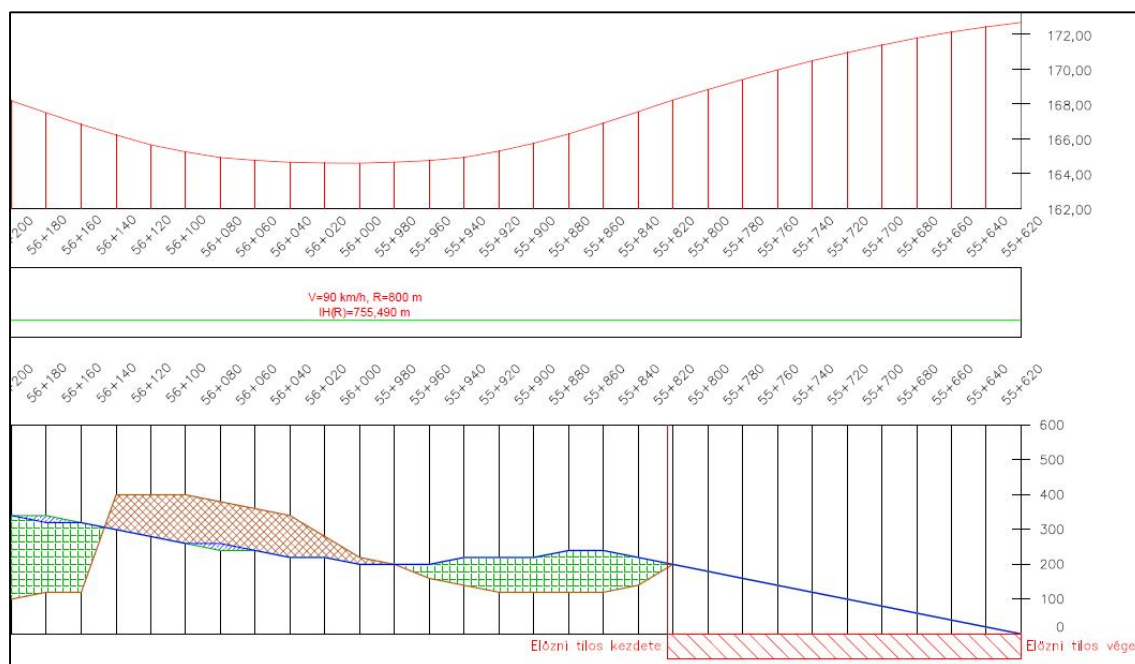
Az előzési látótávolságokról általánosságban elmondható, hogy egyetlen alkalommal sem érik el a *Közutak tervezése ÚT 2-1.201 műszaki előírásban* meghatározott, a 90 km/h tervezési sebességhez tartozó 560 m-es értéket. Ez több okra is visszavezethető. Az egyik, hogy az elkerülő lehatárolt szakaszának egy harmadában előzési tilalom van érvényben. A másik ok, hogy a további, kétharmadnyi részben is sűrűn változó domborzati egységen haladunk keresztül, így maximálisan 400-420 m-t látunk előre a nyomvonalpontokból.

Oda irányban az 55+620-as kezdőszelvénytől egy, a haladási irány szerinti, közelítőleg 800 m sugarú bal ívből indulunk. Az első 200 m-es szakaszra előzési tilalom vonatkozik. Jelen esetben az előzési tilalmat indokolja, hogy ívben haladunk lejtésiránynak lefelé, így a vonalvezetéssel nehéz volt kialakítani olyan helyzetet, melynél tartható az előírt 560, vagy akár a továbbiakban megjelenő 400 m-es előrelátás. Természetesen vissza irányban itt teljesen más helyzet áll elő, hiszen ellenkező haladásnál az 55+860-as szelvénytől kezdve folyamatosan csökken a látótávolságunk. Ez egyszerűen azzal magyarázható, hogy a mintaútszakasz elemzése nem terjed tovább a korábban említett 55+620-as szelvénynél.

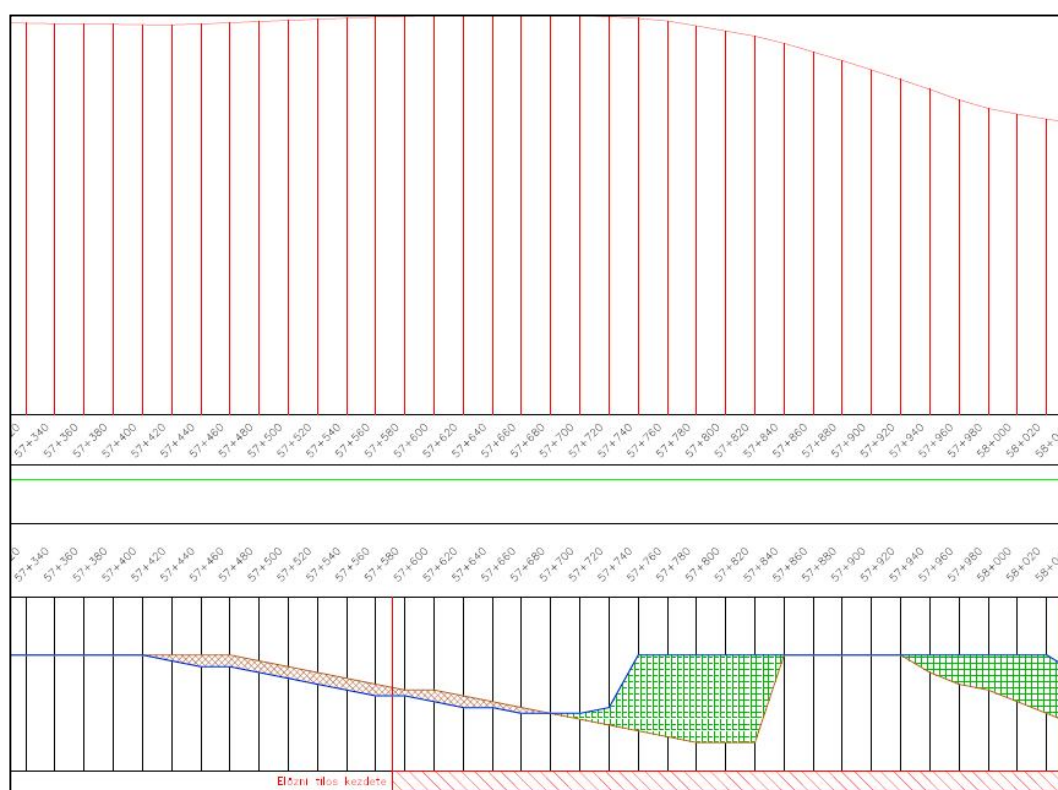


9-2. ábra Előzési látótávolság oda irányban az 55+620 kezdőszelvénytől [a szerző munkája]

Az előző 9-2. és a következő 9-3. ábrák a bemutatott szakasz hossz-szelvénnnyel összekapcsolt ábrázolását mutatják a két haladási iránynak megfelelően.

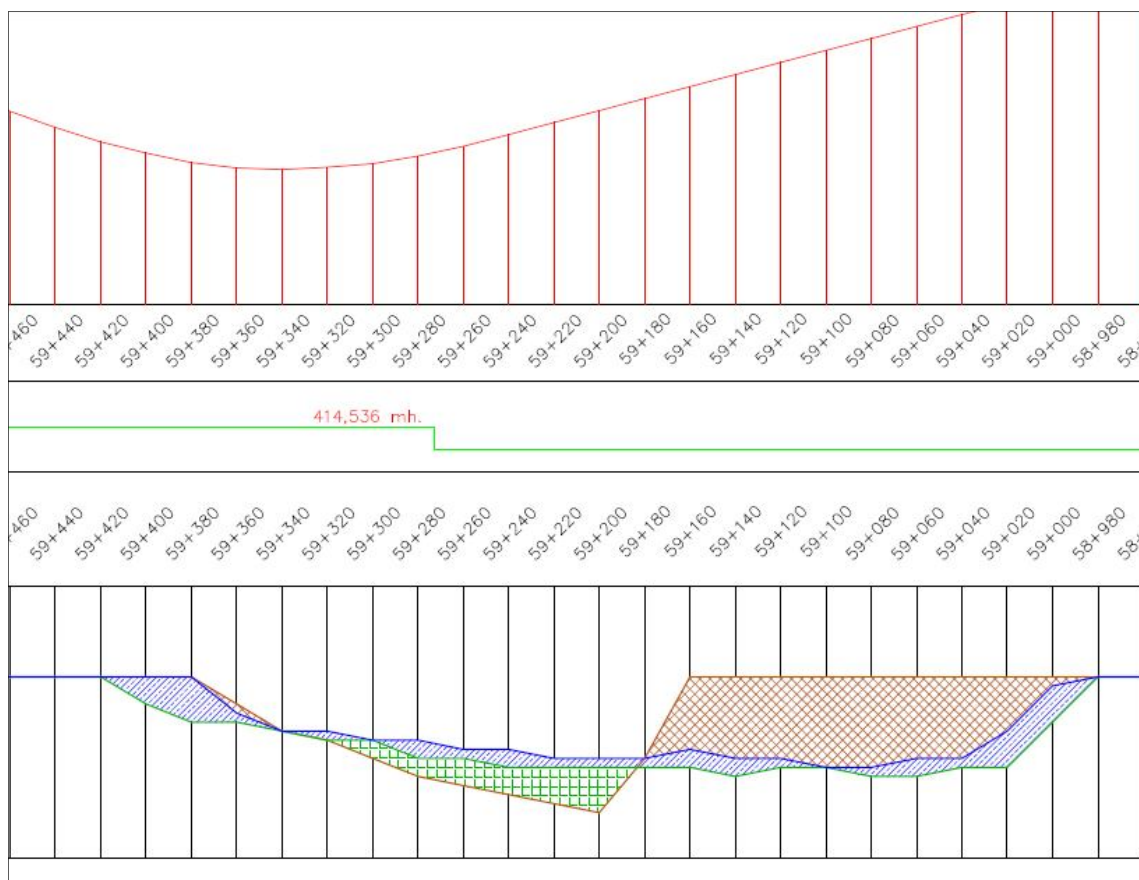


9-4. ábra Előzési látótávolság vissza irányban az 55+620 kezdőszelvénytől [a szerző munkája]



9-3. ábra Előzési látótávolság oda irányban az 57+600-as szelvénytől [a szerző munkája]

Az ábrákról jól látható, hogy helyeként a barna vonal, vagyis az 1x1 m-es felbontású Google adatsor a javított modelleknél jobb láthatóságot mutat. Ez azonban nem jelenti azt, hogy az eredmény teljesen helyes, hiszen ez az ingyenes adat eredetileg 32x32 m-es felbontással készült, éppen ezért a lineáris interpolációja nem tökéletes. A 9-2. ábráról továbbá az is észrevehető, hogy az előzési tilalom elhagyását követően rögtön felugrik a látótávolság értéke 400 m-re. Ez abból adódhat, hogy csökken az út magassági lejtése és, bár a hossz-szelvényen nem látható, helyismeretem révén elmondhatom, hogy már az átmeneti ívben haladunk, ahol egyszerűbben biztosíthatók a nagyobb előrelátás feltételei.



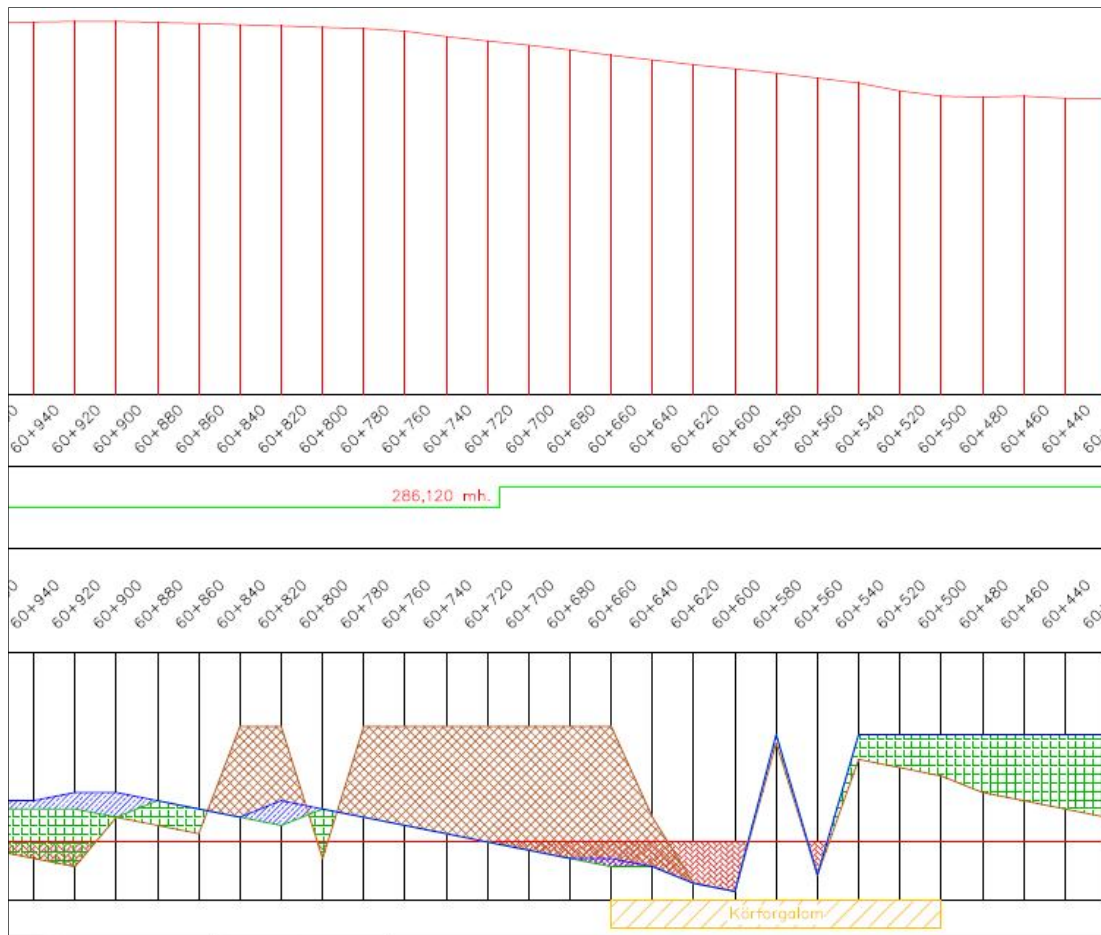
9-5. ábra Előzési látótávolság vissza irányban az 59+000 szelvénytől [a szerző munkája]

Az előző megállapítások hasonlóan elmondhatók a 9-4. ábráról is, ahol az 57+600-as szelvényenél kezdődő előzési tilalmat láthatjuk. Ahogy az előbbi esetekben, itt is észrevehető, hogy a tilalom kezdetéhez közeledve egyre csökkennek a látótávolság értéke. Észrevehető azonban, hogy ez az érték a tilos szakaszban is felugrik 400 m-re, ami a lejtésben haladásnak köszönhető, viszont éppen ezen okból kifolyólag előzni tilos van érvényben.

A 9-5. ábráról látható, hogy nem csupán az előzni tilos szakaszokban csökken 400 m alá a látótávolság értéke, hanem olyan helyeken is, ahol dombnak felfelé haladunk egy homorú lekerekítő ívet követően. Ezen a szakaszon, amennyiben a számítás helyes eredményeket mutat, célszerűen lenne egy előzni tilos jelzést elhelyezni.

2) Megállási látótávolságok oda-, és vissza irányban

A megállási látótávolságok hossz-szelvény menti értékelése hasonló módon történt, mint az előzésinél elmondottak. Itt is megállapítható, hogy a javított modellek esetén vannak olyan helyek, ahol nem teljesül a szabványban meghatározott 140 m-es küszöbérték. Túlnyomó részben a megállási látótávolságok jóval nagyobbra adódtak, mint 140 m. Azon szakaszrészeken, melyekre az előző megállapítás nem igaz, általánosságban elmondható, hogy az eltérések ekkor sem relevánsak.

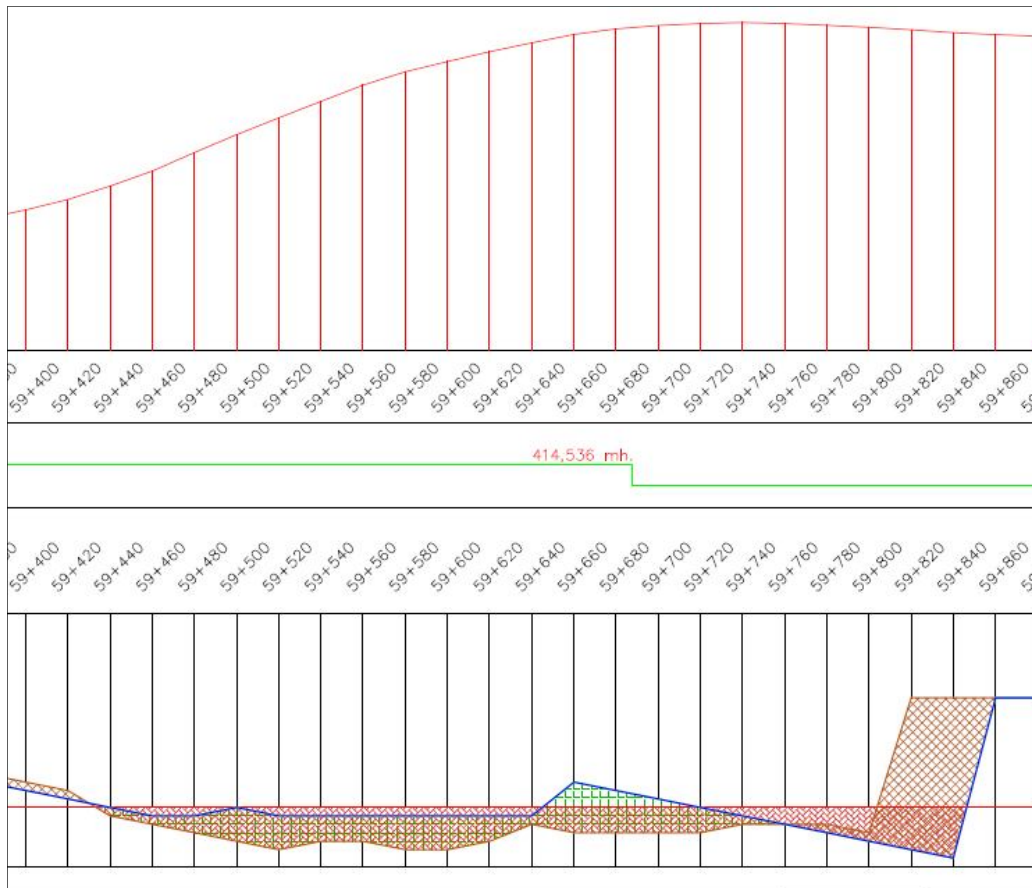


9-6. ábra Körforgalmi kialakítás miatt lecsökkent láthatóság vissza irányban [a szerző munkája]

Ha a vissza irányú haladásnál a 61+000 szelvénytől indulunk, és eljutunk a 60+640-eshez, akkor látható, hogy a látótávolság értéke erősen lecsökken. Ez ugyanúgy elmondható ebben a szakaszban az előzési adatokról is. Az értékek drasztikus

csökkenését itt a körforgalmi kialakítás okozza, hiszen e helyen célszerűen úgy vettem fel a nyomvonalpontokat, hogy a forgalomi helyzethez igazodjanak. Ebből adódóan a kezdeti lépések során meghatározott irányszög számításnál a takarási ábrát nem a körforgalmon túlmutató irány szerint számolja a program, hanem mindig a következő ponthoz viszonyítva. Ezt a helyzetet szemlélteti a 9-6. ábra.

Ahogy az ábráról is látható, kiérve a körforgalomból a látótávolság értéke ismét jelentősen növekszik.



9-7. ábra Domborzati hibahely oda irányban [a szerző munkája]

A 9-7. ábráról látszik, hogy még a pontos információkkal javított modellek sem tökéletesek minden esetben, ugyanis az 59+840-es szelvényben a látótávolság értéke körül-belül 20 m-re csökken. A szelvényt megvizsgálva a vonalvezetések függvényében, látható, hogy semmi sem indokolja a csökkenést, így valószínűleg egy domborzati hiba található az adott helyen. Ez a domborzati hiba lehet egy hirtelen kiugrás, esetleg egy két, vagy több magassági adattal rendelkező pont, melynél a program nem tudja eldönteni, hogy melyik alapján számolja ki a TIN-hálózatot.

A látótávolságok hossz-szelvényi nézetben történő bemutatását követően ismertetném a dolgozat során megszerzett tapasztalatokat.

10 Tapasztalatok, jövőbeni tervek

Az előző fejezetben bemutatott értékelési eljárások segítségével különböző következtetéseket tudtam levonni az egyes eredményeket illetően. A következőkben az értékelésnél tapasztalt megállapításokat szeretném részletezni.

10.1 Eredmények értékelése

Az Excelben minden egyes vizsgálatfajtánál végeredményként előállítottam a négyzetes mátrixot, és a bemutatott dinamikus grafikonokat. A különböző értékek összehasonlításánál megállapítható volt, hogy a simán Google adatokkal számolt vizsgálatok és a javított értékekkel számolt látótávolságok között esetenként nagyobb eltérések is lehetnek. Az eltérések több okra vezethetők vissza, melyek a következők:

a) Ingyenes modellek kiértékelésének pontatlansága

Az ingyenes domborzatmodell műholdas felvételek sztereoszkópikus kiértékelésével kerül előállításra, erre vetítik rá a korrigált műholdképeket. A folyamat végeredményként egy közelítőleg helyes domborzatot kapnak, melyen az út felület csak egy rávetített kép, ezáltal akár kisebb, nagyobb bukkanak is lehetnek rajta. Ezek természetesen befolyásolják a látótávolság kiértékelését, ezért a Google adatsor pontosítása mindenképp szükséges.

b) Légvonalas módszer hibái

A javított adatokkal való munka esetén, a légvonalas látótávolság számolásának végeredményei helyenként lényegesen kisebb értékeket adtak, mint az eredeti modellnél. Ez arra vezethető vissza, hogy a magassági adatok, az ingyenes modellek esetén nem illeszthetők tökéletesen, így például az utak bevágásánál a különböző pontok nem lesznek összeláthatók. Természetesen a légvonalas vizsgálat pontatlanabb, mint az útvonal mentén történő.

c) Illesztési pontatlanság

Az előző pontban említett magassági illesztés jelenti a 3. kritériumot, mely a végeredményt befolyásolja. A különböző információkat a modell javításának érdekében egyesítenünk kell (ingyenes domborzat, pontos útdatok). Ez csak hozzávetőlegesen oldható meg. Tehát a kiértékelés eredménye attól függően fog változni, hogy milyen pontosan sikerült az illesztési kapcsolatot megoldanunk.

d) *Fizetős adatokkal való munka*

Az eredményekből jól látszik, hogy a fizetős adatok esetén is szükségünk van a domborzat pontosítására, hiszen a kiértékelés során itt sincsenek figyelembe véve az út felületi adatai. A pontosításnál azonban elmondható, hogy az útvonal menti vizsgálatnál a Google és a FÖMI adatok közelítőleg hasonló eredményre vezettek, így voltaképp mindegy melyiket alkalmazzuk. A döntést gazdasági okok határozhatják meg.

e) *Fizetős adatok illesztése*

A fizetős adatok legnagyobb előnye az ingyenesekkel szemben, hogy ezek illesztéséhez nem szükséges a közös pontok ismerete, mivel a modell már mind vízszintesen-, mind pedig magasságilag georeferálva van.

A következőkben szeretném ismertetni a vizsgálatok kiterjesztésének, átalakításának jövőbeni lehetséges megoldásait.

10.2 Jövőbeni vizsgálati lehetőségek

Az előzőekben leírt értékelések figyelembe vételével véleményem szerint további megoldások dolgozhatók ki árnyaltabb, részletesebb vizsgálatok elvégzésére.

10.2.1 Szemmagasság változtatásának lehetősége

Az első fejezetekben említett *Közutak tervezése ÚT 2-1.201 Útügyi Műszaki Előírás* 1 m-ben határozza meg a vezető szemmagasságát. Természetesen a szemmagasság az autók fajtájától függően változik, az 1 m csak egy közelítő érték. Az autó típusának függvényében változó észlelési pont magasságának eltéréseiről szól például *Francesco Saverio Capaldo Driver Eye Height* című írása. Kutatásában a szemmagasság eltérő voltát vizsgálata országok és nemek szerinti eloszlásban.

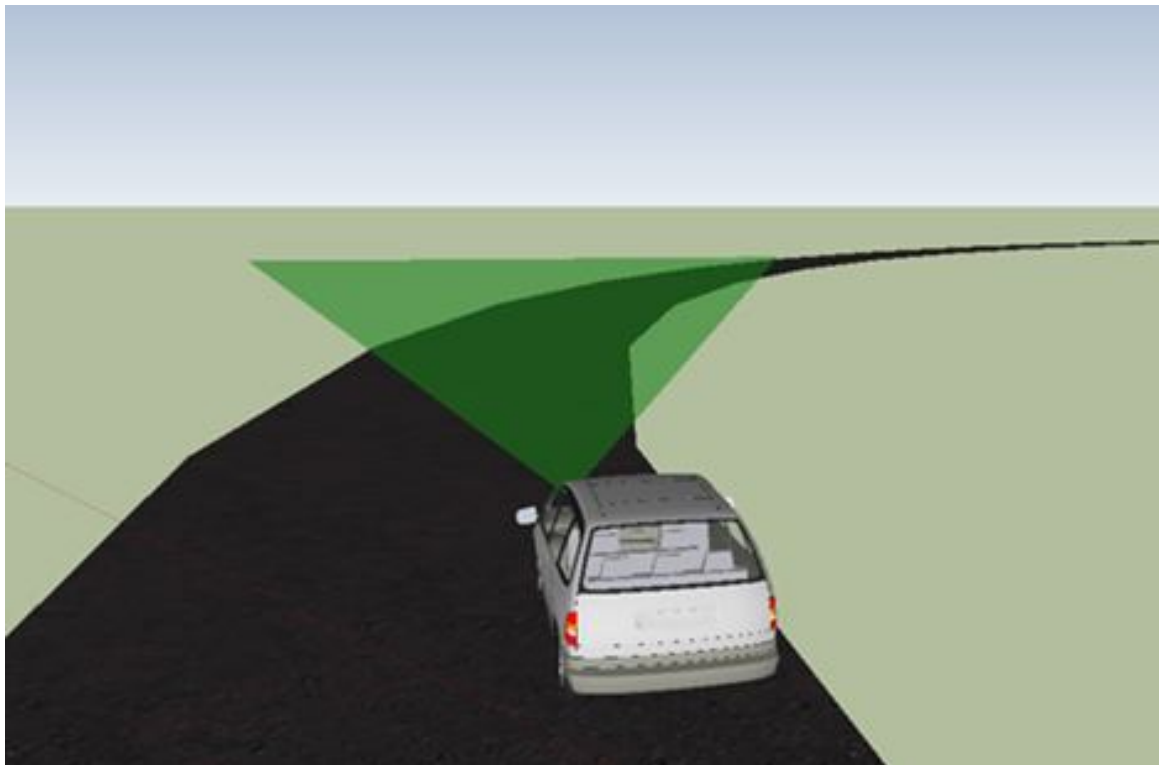
A szemmagasság változtatásának lehetősége egy további vizsgálati módszert jelenthetne a jövőben pontosított meghatározás érdekében. Érdekes lenne megfigyelni, hogy mondjuk 20 cm-el megnövelt szemmagasság mennyiben befolyásolja a végeredményeket.

10.2.2 3D objektumok figyelembe vétele

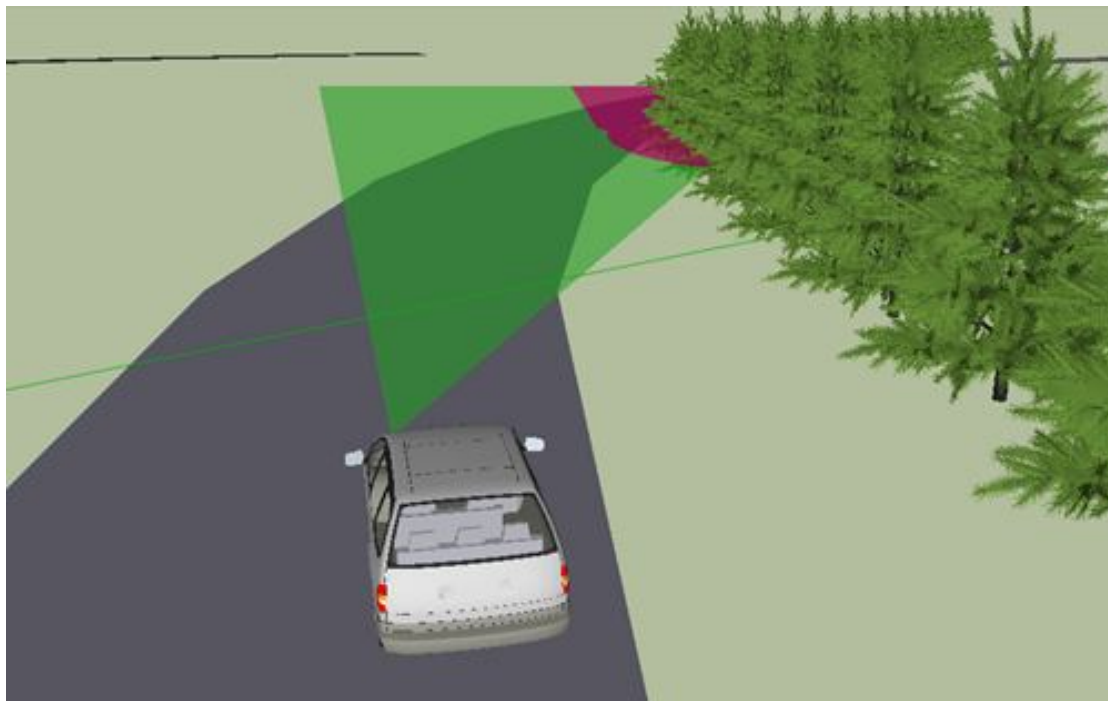
Részletesebb kutatási lehetőségeket nyújthatna továbbá egy kiterjesztettebb elemzés, melynél a számolásba már különböző tereptárgyak és útbútorok 3D kiterjedése

is bevonható lenne. A 3D objektumok bevonása további, fontos megállapításokhoz vezethetne, hiszen nem mindegy, hogy például egy kanyarban mennyit látunk előre egyik, vagy a másik lehetőséggel.

Vegyünk például egy ívben haladó autót, erre mintát a *10-1. és 10-2. ábrák* mutatnak. Abban az esetben, ha nem vesszük bele a számolásba a 3D objektumokat, mint a fák, a szalagkorlát, vagy éppen az előttünk bizonyos távolságra haladó másik gépjármű, akkor a *10-1. ábra* szerint alakul a látótávolság értéke. Ha azonban az útbútorokat és az erdőket is bevonjuk a folyamatba, akkor a *10-2. ábráról* jól látható, hogy a látótávolság akár jelentősen meg is változhat. Ez persze mindenképpen szoros összefüggésben áll az út vonalvezetésének kialakításával és a közvetlen környezettel.



10-1. ábra Láthatósági vizsgálat 3D objektumok nélkül [*a szerző munkája*]



10-2. ábra 3D fák bevétele a vizsgálatba *[a szerző munkája]*

A távlati terveket tekintve elmondható, hogy egy láthatósági vizsgálat rendkívül sokrétű lehet. Természetesen az eredmények, mint láttuk szoros kapcsolatban állnak a számításba bevont információkkal, melynek okán számos megoldás dolgozható ki a továbbiakban a láthatósági felülvizsgálati eljárás javítására, pontosítására.

11 Összegzés

Vezetés során a biztonságérzet az egyik legfontosabb komforttényező, hiszen ennek tudatában választunk útirányt magunk és családunk számára. A biztonságos haladást, így minden útszakaszon meg kell követelnünk. Dolgozatomban éppen ezért egy külterületi főútvonal vizsgálatát végeztem el. Ez az útszakasz a zalaegerszegi 761. számú elkerülő út volt.

Az elemzésnél a fő kérdéskört az ingyenesen elérhető domborzati adatok felhasználhatósága alkotta. A kérdés pontosan az volt, hogy egy, már meglévő útszakasznál elegendő-e megállási-, és előzési látótávolságok értékelésére egy díjmentes adatsor, vagy esetleg meghatározott összegű információt kell igénybe vennünk. Az ingyenes modell értékeléséhez pontosabb, fizetős adatokat is bevontam a számításba. Mind a díjtalan, mind pedig a díjköteles információknál is többféle megoldás szerint dolgoztam, azért, hogy a lehető legalaposabb képet tudjam megalkotni az eredmények értékeléséhez.

Az első bemutatott megoldás egy légvonalakban számoló eljárás volt, melynél a program az út nyomvonalpontjait összekötve vizsgálta meg a látótávolságok értékeit. Mivel e megoldás légvonalakat számolt az egyes pontok között, így nem bizonyult a legmegfelelőbbnek a domborzati információk értékeléséhez, így egy másik metódus is kidolgozásra került.

A második elemzett lehetőség már az út nyomvonala mentén értékelte a láthatóságot, így jobb alapot nyújtott helytállóbb következtetések levonásához. Mindkét esetben az előzési-, és megállási látótávolság minősítése is 5-5 variációban történt. Az egyes megoldások számolását követően a pontos távolságok számítása már az Excelben került végrehajtásra.

Véleményem szerint a dolgozatban alkalmazott, az út nyomvonala mentén dolgozó térinformatikai elemzés továbbfejlesztve, jó alapot nyújthat a már meglévő főutak látótávolságainak minősítésére. A használt módszertan segítségével a főútvonalak látótávolsági felülvizsgálatát lehetne elvégezni. A továbbfejlesztett elemzési funkciókkal megfelelő, a nagyobb biztonsági szint elérését célzó javaslatokat lehetne tenni.

Dolgozatom végén szeretnék köszönetet mondani mindazoknak, akik valamilyen módon segítettek a kutatásban.

Ezúton szeretném megköszönni konzulenseimnek **Koren Csaba**, és **Busics György** tanáruraknak a folyamatos útmutatást és támogatást. **Pődör Andrea** tanárnőnek pedig az ArcGIS használatában nyújtott segítségét.

Köszönöm továbbá **Horváth Zsoltnak**, a Közlekedésépítési Tanszék munkatársának, és **Faragó Gábornak**, szobatársamnak a különböző programozási nyelvek megértéséhez adott instrukcióikat.

Végül, de nem utolsó sorban külön köszönet illeti húgomat, **Horváth Esztert**, és édesanyámat, **Horváthné Fábián Zsuzsannát**, mert minden egyes alkalommal ők volt azok, akik végigolvasták a leírt részeket és véleményükkel, észrevételeikkel segítettek a további folytatásban.

12 Irodalomjegyzék

- Aithken, P. G. (2001). *Programozás Visual Basic 6 nyelven*. Budapest: Kiskapu Kiadó.
- Capaldo, F. S. (2012). Driver Eye Height: Experimental Determination and Implications on Sight Distances. *8th International Conference on Traffic and Transportation Studies*. Kína, Changsha.
- Castro, M., Iglesias, L., Sánchez, J. A., & Ambrosio, L. (2009. December). Sight distance analysis of highways using GIS tools. *Transportation Research Part C: Emerging Technologies*, p. 997-1005.
- Eötvös Lóránt Tudomány Egyetem. (2001). *SRTM – Shuttle Radar Topography Mission*.
Letöltés dátuma: 2013. november 03, forrás: ELTE Honlap:
http://sas2.elte.hu/mg/foldkutatas_v3/11radar3srtm.htm
- ESRI. (2012a). *What is Model Builder?* Letöltés dátuma: 2013. október 20., forrás:
ArcGIS Help 10.0:
<http://resources.arcgis.com/en/help/main/10.1/index.html#//002w00000001000000>
- ESRI. (2012b). *What is the ArcGIS 3D Analyst extention.* Letöltés dátuma: 2013. október 20.,
forrás: ArcGIS Help 10.0:
<http://resources.arcgis.com/en/help/main/10.1/index.html#//00q80000000wv0000000>
- ESRI. (2012c). *Understanding visibility analysis.* Letöltés dátuma: 2013. október 23.,
forrás: ArcGIS Help 10.0:
http://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Understanding_visibility_analysis/00q8000000p2000000/
- ESRI. (2012d). *Visibility and Range Template.* Letöltés dátuma: 2013. október 25, forrás:
ArcGIS Help 10.0: <http://blogs.esri.com/esri/arcgis/2011/07/06/visibility-and-range-template-available-for-download/>
- ESRI. (2012e). *Using Viewsheed and Observer Points for Visibility Analysis.* Letöltés dátuma: 2013. október 26, forrás: ArcGIS Help 10.0:

- http://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Using_Viewshed_and_Observer_Points_for_visibility_analysis/00q90000008n000000/
- GPS Magazin. (2004). *SRTM mindenkinek*. Letöltés dátuma: 2013. november 03., forrás: Kütyü Magazin: <http://kutyu.hu/cikk/86290>
- Graham, R., & Holland, L. (2012). *Mastering AutoCAD Civil 3D 2012. Official Training Guide*. Autodesk.
- Horváth, T. (2011a). A 3D modellezés lehetőségei egy Zala-megyei település példáján. *TDK dolgozat*. Székesfehérvár: Nyugat-magyarországi Egyetem Geoinformatikai Kar.
- Horváth, T. (2011b). Nagykutas területének 3D modellezése. *Szakedolgozat*. Székesfehérvár: Nyugat-magyarországi Egyetem Geoinformatikai Kar.
- Khattak, A., & Shamayleh, H. (2005). Highway safety assessment through geographic information system-based data visualization. *Journal of Computing in Civil Engineering* 19 (4), p. 407–411.
- Koren, C. (2012). A közúti infrastruktúra biztonsága. *Főiskolai Jegyzet*, p. 350. Győr: Széchenyi István Egyetem.
- Koren, C. (2013). *Közlekedésépítés IV. - Közúti biztonság*. Koren Csaba, Győr, Széchenyi István Egyetem.
- Közlekedésfejlesztési Koordinációs Központ (KKK). (2011). *UTADAT.HU*. Letöltés dátuma: 2013. október 15., forrás: Interaktív portál: www.utadat.hu
- Központi Statisztikai Hivatal (KSH). (2010). *Személyi sérüléssel közúti közlekedési balesetek*. Letöltés dátuma: 2013. november 02., forrás: <http://www.ksh.hu/docs/hun/xftp/gyor/ser/ser21012.pdf>
- Magyar Útügyi Társaság (MAÚT). (2008). *Közutak tervezése Útügyi Műszaki Előírás. ÚT 2-1.201*.
- Makró Mester (MM). (2013). *A VBA nyelvi elemei*. Letöltés dátuma: 2013. november 08., forrás: Makró Mester Honlapja: <http://www.konyveles-budapest-bpc.hu/excel-makro/leirasok/vba-nyelvi-elemei/valtozok-adattipusok-konstansok>

- Márkus, B. (2006). Térinformatika II. *Tankönyv*, p. 341. Budapest: FVM KSZI.
- Márkus, B. (2006). Térinformatikai I. *Tankönyv*, p. 341. Budapest: FVM KSZI.
- Nagy, T. (2011). *Olyan, mint az FKERES*. Letöltés dátuma: 2013. november 08, forrás: Nagy Tamás Honlapja: <http://nagytamas.wordpress.com/2011/01/13/olyan-mint-az-fkeres/>
- Országos Rendőr Főkapitányság (ORFK), Rendészeti Főigazgatóság, & Közlekedérendészeti Főosztály. (2011). *A személyi sérüléses közúti közlekedési balesetek alakulása*. Letöltés dátuma: 2013. november 02, forrás: http://www.baleset-megelozes.eu/pics/baleseti_statisztika_2011_egesz_ev.pdf
- Országos Rendőr Főkapitányság (ORFK), Rendészeti Főigazgatóság, & Közlekedérendészeti Főosztály. (2012). *A személyi sérüléses közúti közlekedési balesetek alakulása*. Letöltés dátuma: 2013. november 02, forrás: http://www.baleset-megelozes.eu/pics/2012_junius_balstat.pdf
- Schuchmann, G. (2010). *Látótávolságok, előzési szakaszok, kapaszkodósáv, túlelemelés, pályaszélesítés*. Schuchmann Gábor, Budapest, Magyarország.
- Szikszai, C. (2003). *Digitális domborzatmodelljeink*. Letöltés dátuma: 2013. november 02., forrás: Természet Világa: <http://www.termeszetvilaga.hu/szamok/tv2003/tv0305/szikszai.html>

13 Ábrajegyzék

2-1. ábra Kiváltó okok részvétele a balesetekben [Koren, 2013].....	6
2-2. ábra Megállási látótávolságok a sebesség és hosszesés függvényében [MAÚT, 2008; Schuchmann 2010].....	8
2-3. ábra Előzési látótávolság adatai [Schuchmann, 2010].....	9
3-1. ábra Egy terület GRID-, (bal oldalt) és TIN-modellje (jobb oldalt) [Szikszai, 2003]	11
5-1. ábra Láthatósági vizsgálat az AutoCAD Civil 3D-nél [a szerző munkája]	20
6-1. ábra Láthatósági vizsgálat megoldásai [a szerző munkája]	24
6-2. ábra A főbb utak helyzete Zalaegerszeg körül [KKK, 2013]	25
6-3. ábra Zalaegerszeg a Google Földben [Google Earth]	26
7-1. ábra A légvonalas vizsgálat folyamatábrájának részletes [a szerző munkája].....	29
7-2. ábra Javításokkal ellátott és leválogatott látóvonalak [a szerző munkája]	33
7-3. ábra Látóvonalak a nyompontok között [a szerző munkája]	33
7-4. ábra Láthatóság alapján tördelt egyenesek [a szerző munkája]	34
7-5. ábra Teljes egészében látható egyenesek [a szerző munkája].....	35
8-1. ábra Az útvonala vizsgálat folyama ábrájának részlete [a szerző munkája].....	36
8-2. ábra Beállítási lehetőségek szemléltetése [Castro et al., 2009]	38
8-3. ábra Vizuális animáció a látótávolságok változásáról [a szerző munkája].....	45
9-1. ábra Megállási látótávolságok változása mozgatósávval [a szerző munkája]	58
9-2. ábra Előzési látótávolság oda irányban az 55+620 kezdőszelvénytől [a szerző munkája].....	61
9-3. ábra Előzési látótávolság oda irányban az 57+600-as szelvényénél [a szerző munkája]	62
9-4. ábra Előzési látótávolság vissza irányban az 55+620 kezdőszelvénytől [a szerző munkája].....	62
9-5. ábra Előzési látótávolság vissza irányban az 59+000 szelvénytől [a szerző munkája]	63
9-6. ábra Körforgalmi kialakítás miatt lecsökkent láthatóság vissza irányban [a szerző munkája].....	64
9-7. ábra Domborzati hibahely oda irányban [a szerző munkája].....	65
10-1. ábra Láthatósági vizsgálat 3D objektumok nélkül [a szerző munkája]	68
10-2. ábra 3D fák bevétele a vizsgálatba [a szerző munkája]	69

14 Táblázatjegyzék

2-1. táblázat Az előzési látótávolság értékei [Schuchmann, 2010]	9
8-1. táblázat Viewshed beállítások [a szerző munkája]	38
9-1. táblázat Példa a négyzetes mátrix végeredményére [a szerző munkája]	47
9-2. táblázat Mintapélda a kettős keresés adatbázisára [a szerző munkája]	48
9-3. táblázat Az elkészült négyzetes mátrix első néhány sora [a szerző munkája]	54
9-4. táblázat A keresés végeredményét tartalmazó táblázat egy része 100 m-enkénti bontásban színezve [a szerző munkája]	56
9-5. táblázat Dinamikus adatok csúsztható sávval [a szerző munkája]	57

15 Mellékletek

1. Légvonalas vizsgálat folyamatábrája
2. Útvonalas vizsgálat folyamatábrája
3. Python programozás
4. Excel - Kettőskereső makró
5. Részlet a négyzetes mátrixból
6. Excel - Diagramkészítést segítő makró
7. Dinamikus diagramok
8. Átnézeti hossz-szelvény az 1 m magas ellenirányú forgalom észlelésének távolságaival
9. Átnézeti hossz-szelvény a 10 cm magas akadály észlelésének távolságaival