



Széchenyi István Egyetem
Műszaki Tudományi Kar

Másodrendű főutak külterületi szakaszának láthatósági vizsgálata

TDK dolgozat

Konzulens:

Dr. Koren Csaba

egyetemi tanár

Készítette:

Horváth Tamás

Nappali tagozat

Infrastruktúra építőmérnök Msc
Közlekedésépítés Szakirány

Győr

2013

Tartalomjegyzék

1	Bevezetés	4
2	Közúti biztonság és a megállási látótávolság	6
3	Digitális domborzat és az elérhető adatok	8
3.1	Digitális domborzatmodellezés	8
3.2	Digitális modellek Magyarországon	10
3.2.1	FÖMI DDM	10
3.2.2	Katonai DDM	11
3.2.3	Google SRTM	11
4	Modellezés GIS adatok felhasználásával	13
4.1	ESRI ArcGIS 10	13
4.2	Előkészítő lépések	13
4.3	Modellezési lehetőségek	15
5	Légvonalas láthatósági vizsgálat	17
5.1	Folyamatábra a Model Builderben	17
5.2	A légvonalas vizsgálat folyamatábrája	18
5.2.1	Adatok beolvasása	19
5.2.2	A domborzatmodellről a látóvonalakig	19
5.2.3	Láthatósági vonalak	22
6	Láthatósági vizsgálat az útpálya mentén	24
6.1	Bevezető lépések	24
6.2	Python programozás	26
6.2.1	A Python programozásról általában	26
6.2.2	FOR ciklus a vizsgálatnál	27
6.2.3	Láthatósági animáció	29
7	Excel vizsgálat	31
8	Eredmények	37

9	Irodalomjegyzék.....	38
10	Ábrajegyzék.....	40
11	Táblázatjegyzék	40

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 közlekedő számára. Nem csupán egy, még a tervezés folyamatában lévő útnál igaz e megállapítás, hanem egy már használatban lévő útszakasz esetén is. A megfelelő látótávolságok kialakítása ennek okán elengedhetetlen. Az ideális látótávolságok hiányában egy-egy balesetnek a bekövetkezése nagyobb valószínűséggel történik meg, hiszen nincsenek biztosítva az előzésre-, vagy a megállásra vonatkozó szabályok.

Az útszakaszok tervezése során nagy gondot kell fordítanunk a korrekt 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 tett meglévő útszakaszoknál természetesen a felülvizsgálat és a későbbi korrekció jelenti.

Dolgozatomban jelenleg már egy meglévő főút kiválasztott mintaszakaszait 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 adat érhető el. Az egészen pontos FÖMI¹ DDM-től² kezdve a Google Earth program digitális domborzatmodelljéig bezárólag. 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.

A kérdés pontosan az, hogy ezek az adatok mennyire használhatók egy megállási látótávolság értékeléséhez. A látótávolságok elemzésénél többféle módszert

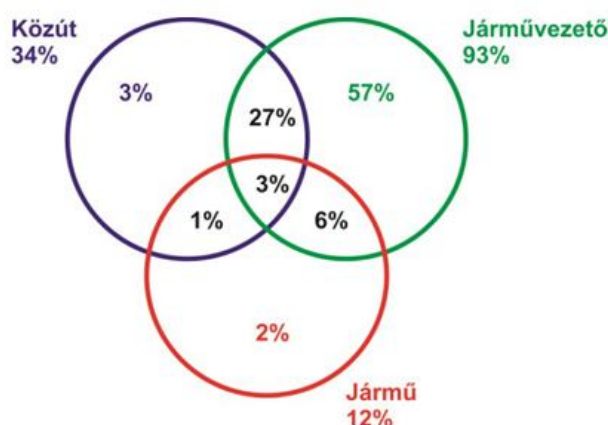
¹ Földmérési és Távérzékelési Intézet

² Digitális Domborzat Modell

alkalmazok az egyes mintaterületeken, melyek alapján értékelem az adatok pontosságát. Az elemzések modellezése minden esetben GIS (földrajzi) adatok felhasználásával történik. Annak érdekében, hogy ne kelljen minden területnél műveletek sokaságát elvégezni az egyes szakaszokra külön-külön, az automatizálás lehetősége tűnt a legjobb megoldásnak. Az automatizálása így szintén a kutatás tárgyát képezi.

2 Közúti biztonság és a megállási látótávolság

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 különböző 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]

Jól látható, hogy 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 tolerancia bevezetését, vagy a pontrendszer teljes körű változtatását. A szankciók az egyes vétségekkel szemben is sokkal komolyabbak lettek. Persze nem minden baleset miatt hibáztatható egyszerűen csak a járművezető, így ennek okán, az ábrán nem különálló, hanem egymással metsződő helyzetű diagramokat találunk. [Koren, 2012]

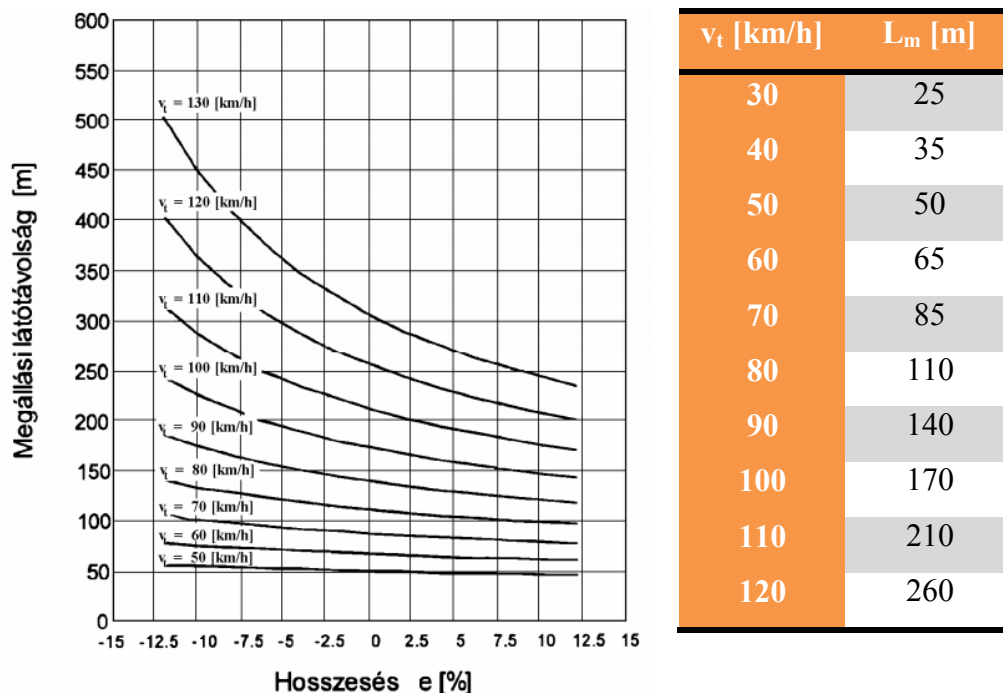
A balesetek harmada az út műszaki adatai és állapota miatt következik be. Megfelelő útkonstrukcióval és felülvizsgálattal ezek azonban csökkenthetők. Az európai országok többsége nem csupán a járművezetők felelősségre vonásával és szigorúbb szabályokkal kívánja megoldani a halálozási ráta csökkenését, hanem helytálló 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 koncentrálniuk kell a körülöttük lévő világra. 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]

A legmegfelelőbb biztonsági megoldások érdekében az úton nem csak az autósok, de minden közlekedő számára ideális láthatósági viszonyokat kell teremtenünk. Ideális viszonyok esetén, a résztvevők képesek időben észrevenni és reagálni az úton közlekedő másik járműre, kerékpárosra, vagy gyalogosra. Ahhoz tehát, hogy minden egy balesetet el tudjunk kerülni, korrekt megállási látótávolságot kell biztosítani.

Megállási látótávolság

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 hosszesítés függvényében [MAÚT, 2008; BME 2010]

3 Digitális domborzat és az elérhető adatok

Láthatósági vizsgálatok végzéséhez feltétlenül szükségünk van valamilyen formában 3 dimenziós adatokra. 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 valamilyen tereptárgyak a látóvonalunkba, vagy az út úrszelvényébe, szükségünk van egy domborzatmodell létrehozására. A következő alfejezetekben bemutatom mit is nevezünk domborzatmodellnek.

3.1 Digitális domborzatmodellezés

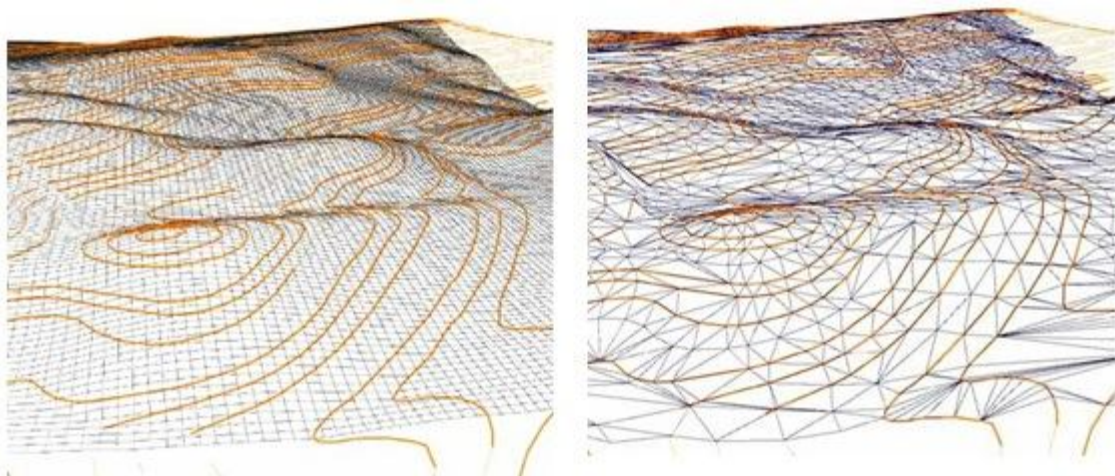
A számítástechnikával segített domborzatmodellezés az utóbbi néhány évtizedben rendkívül sokat fejlődött, ez a fejlődés a mai napig is tart. Térhódítása annak köszönhető, hogy a mérnöki munkák világában egyre nagyobb igények merülnek fel a 3 dimenziós ábrázolásra. Manapság egy munka végrehajtásának szigorú követelménye a pontos magassági felmérés és ábrázolás. A mai világban már léteznek konkrétan 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 domborzatmodell célszerűen változni fog. Minél jobban igazodunk a terep ki-, és bemélyedéseire, annál jobban fog igazodni a modell is a valósághoz. [Márkus, 2006a; Márkus 2006b]

Minden esetben, amikor egy ilyen modell 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 ebben az egyben megegyeznek: mindegyiknek szüksége van a támpontok meglétére. Ha a támpontok 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 magassági adatokkal nem rendelkező helyeket. A kiegészítések elvégzését

követően már 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]

Egy terület felmérése során elsősorban a strukturált modelleket szokás alkalmazni, vagy, ha van elegendő műszer és idő, 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 azonban az úgynevezett TIN⁴-modell, mely során a támpontokat szabálytalan háromszöghálózatban helyezzük el a terepen. Tehát a pontokat megpróbáljuk úgy kiválasztani, hogy azok minél jobban igazodjanak a terep alakulataihoz, ezáltal a később előállított DDM is sokkal jobban fogja tükrözni a valós állapotokat. A TIN-modell egyik leggyakoribb megoldása az egymással, oldalukkal kapcsolódó, a támpontokat csúcspontként kezelő, háromszögekből álló hálózat kialakítása, az ún. Delaunay háromszögelés, amely során igyekszünk a lehető

³ Szabályos négyszöghálózat

⁴ Triangulated Irregular Network – Szabálytalan Háromszög Hálózat

legkisebb háromszögeket kialakítani a támpontokra illeszkedően. A dolgozat végrehajtása során én is TIN-modellt alkalmaztam a domborzatmodellek előállítására. [Horváth, 2011]

3.2 Digitális modellek Magyarországon

A mai világban némely esetben nem szükséges a már említett saját felmérés, minden országban rendelkezésre állnak már digitális adatok a felhasználók különböző igényeihez alkalmazkodva. Egy láthatósági vizsgálat esetében így könnyebben juthatunk adatokhoz, nem szükséges a néha túlzottan körülményes saját felmérés. Annak érdekében, hogy minél jobb eredményt kapjunk, szükségünk van egy digitális domborzatra. Ez a domborzatmodell kiterjedhet egyszerűen csak az út felületére, vagy akár az út felületétől bizonyos távolságra is. A domborzatmodell kiterjedésének 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 pontosság a döntő szempont.

Ha meghatároztuk, hogy milyen információt szeretnénk kinyerni a meglévő 3D-s adatokból, akkor annak alapján már fel tudunk állítani egy követelményi rendszert magunknak, hogy mely források jöhetnek szóba. Fontos azt is eldöntenünk, hogy mekkora a vizsgált terület, és 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, azonban 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.

3.2.2 Katonai DDM

A Magyar Katonai Térképészet a másik nagy szereppel rendelkező intézmény a hazai gyakorlatban. A katonai térképészetet 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 ugyanis levezetésre a modell 10x10 m-es terepi felbontással. A modell rendszerét tekintve, a felhasználó igényeinek megfelelően többféle vetületi és szelvényrendszerben is kérhető, így például a mérnöki gyakorlatban használt EOVR⁶ rendszerben is, EOTR szelvényezéssel. A katonai modell 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.

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

⁵ Egységes Országos Térkép Rendszer

⁶ Egységes Országos Vetületi Rendszer

⁷ Shuttle Radar Topography Mission

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 modell legnagyobb előnye az ingyenessége, ezért rendkívül sok felhasználó alkalmazza akár eredeti, akár javított formában. A javított forma azt jelenti, hogy a letöltött adatot a felhasználó saját egyéb forrásból elérhető adatokkal pontosítja. Ez a fajta pontosítás a dolgozatoknak is tárgyát képezi, ezért a későbbiekben több szót fogok róla ejteni.

4 Modellezés GIS adatok felhasználásával

A GIS adatok felhasználása azért tűnt sokkal előnyösebbnek, mint az AutoCAD kínálta lehetőségek, mert ebben az esetben a felhasználó képes különböző attribútum adatokkal is dolgozni, melyek minden egyes réteg esetén egy úgynevezett attribútum táblában vannak eltárolva. A spanyol TRIVIUM program, hasonlóan az AutoCAD Civil 3D-hez képes láthatósági vizsgálatok elvégzésére [*Castro et al., 2009*], de nem ötvözhetők a különböző elemzések. Egy térinformatikai programban azonban több, más jellegű feltárást is végrehajthatunk ugyanazokkal az információkkal.

4.1 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 adatok kezelésétől a 3D adatkezelésen át egészen egy SRTM alapú földmodellig bezárólag minden megtalálható.

A legfontosabb, amit az ArcGIS működéséről tudni kell, hogy mivel vektor alapú térinformatikai program, így az AutoCAD-hez hasonlóan itt is megvan a rétegmegfeleltetés. A rétegek itt azonban nem különböző típusú vonalakat jelölnek. Definíálásuk során a felhasználónak el kell dönteni, 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.

4.2 Előkészítő lépések

Egy elemzésnél nem ugorhatunk csak úgy egyből az információk tengerébe. Lehetőség szerint végig kell gondolni, hogy hogyan is hajtsuk végre, és ehhez milyen adatok definiálására, milyen rétegek létrehozására van elsődlegesen szükség. Mivel a dolgozatban arra a kérdésre keresem a választ, mennyire használhatók ingyenesen elérhető Google adatok egy ilyen elemzésre, ezért először meg kellett vizsgálnom, hogy miképp egyesíthető egy Google 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 programmal tudjuk megtenni. További hibái ennek a megoldásnak akkor jelentkeznek, amikor egy nagyobb területrészről adatokat hívunk le. A dolgozat készítése során megpróbáltam egészben lekérni a zalaegerszegi 761-es elkerülő utat és annak tágabb környezetét. Itt lépett fel egy olyan hiba, mely során a letöltött adatok nem csak vízszintes x,y irányú torzulást szenvedtek, de egy elcsavarodás is fellépett. Természetesen nem ezek az egyetlen hibák, melyekkel találkozunk, ha ezt a megoldást választjuk, hiszen az AutoCAD nem képes a pontos helyre letölteni az adatokat, ezért ezeket utólag saját magunknak kell illeszteniünk a helyes koordinátákra.

A másik megoldási lehetőség az adatok letöltésére, hogy egy ingyenes forráskódú szoftvert a Sketch' Up-ot használjuk. A Sketch' Up, korábban – a Google Földhöz hasonlóan – szintén a Google fejlesztésében állt, azonban 2012-ben a Trimble cég 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. Itt képesek vagyunk közvetlenül letölteni nem csak a domborzatot, de ezzel együtt a domborzat részét képező műholdas felvételeket is. További előnyként könyvelhető el, hogy ha egyszerre több, egymáshoz kapcsolódó területet szeretnénk megvizsgálni, úgy nem kell a két, vagy több modell egymáshoz illesztésével bajlódni, 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. Az útfelület lehatárolása nagyon egyszerű módon történt, mivel mint említettem a Sketch' Up-ba a műholdas felvételek is letöltődnek a domborzattal együtt. A műholdas 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 vettem volna fel a, akkor túlságosan sok adatom lett volna, ami számítási időben is nagyon megnövelné a

futási időt. 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.

4.3 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.

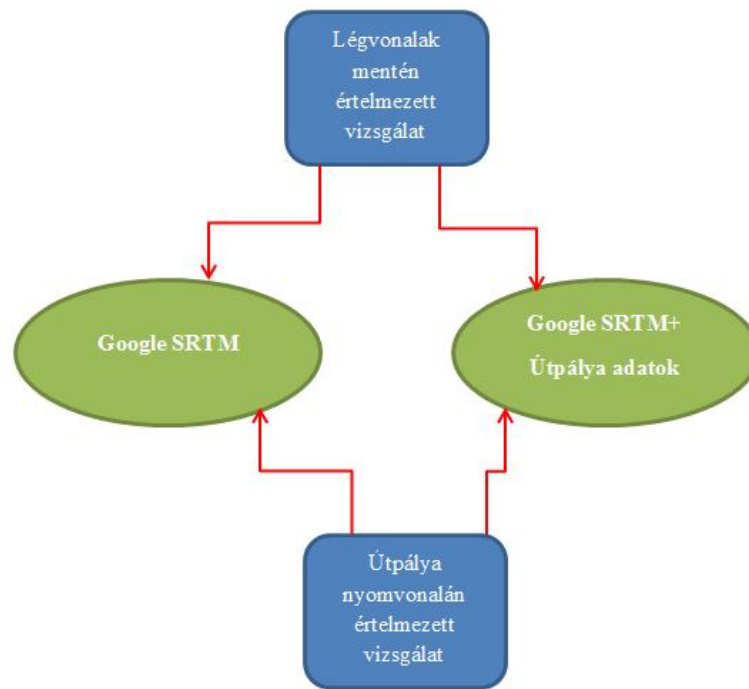
Korábbi tanulmányaim során már volt szerencsém használni az ArcGIS-t, így működése nem volt ismeretlen számomra. A programban egy-egy művelet elvégzéséhez több, előre megírt lekérdezésből választhatunk, de lehetőségünk van arra is, hogy valamilyen szisztéma szerint megadhatjuk, mely lekérdezések, milyen adatokkal és milyen sorrendben hajtódjanak végre. Ezt a lehetőséget reprezentálja a Model Builder nevű alkalmazás. [ESRI, 2013a]

A Model Builder úgy képzelhető el, mint egy folyamatábra szerkesztője, ahová a meglévő 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 vonalakat, megfogalmaztam kétféle megoldási lehetőséget, melyek bemutatgatják, 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.

Az egyes eredmények értékelése miatt, a kétféle vizsgálati lehetőségből megoldást dolgoztam ki, melyek a következő ábra ismerteti:



4-2. ábra A megoldások variációi [a szerző munkája]



4-1. ábra A vizsgált útszakasz térképe [Google Earth]

5 Légvonalas láthatósági vizsgálat

Egy beláthatósági elemzés során fontos, hogy több szempontból kezdjünk neki a vizsgálódásnak. Az elemzés nem biztos, hogy első körben megfelelő eredményt fog adni. Első nekifutásra egy légvonalakban történő vizsgálat jutott az eszembe. A lényege, hogy a pontok között meghatározásra kerül, hogy a domborzati viszonyok függvényében melyek azok látóvonalak, amik teljes egészében a felület felett futnak. A légvonalak létrehozásának lépéseit a továbbiakban ismertetem.

5.1 Folyamatábra a Model Builderben

Az ArcGIS különböző kiegészítő, úgynevezett „toolbox⁸”-okkal rendelkezik, melyekben a korábban említett funkciók, 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 neve „3D Analyst Tool”. [ESRI, 2013b]

A 3D Analyst Tool-ban találunk majdnem minden olyan lekérdezési és művelet végrehajtási megoldást, mely nevéhez híven a 3D-s adatokkal dolgozik. Ebben van egy olyan műveletet, mellyel meghatározott rétegek között légvonalakat tudunk létrehozni. [ESRI, 2013c] Ahhoz azonban, hogy ezek a légvonalak létrehozhatók legyenek meg kell első lépésben határoznunk, hogy 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 például az adatok beolvasása – a Model Builderben elkészíteni.

A légvonalas láthatóság létrehozásához sok műveleten kell végigmennie a programnak, míg a keresett adatok megfelelőek lesznek az eredmények értékeléséhez. 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 addig szükségesek, amíg a következő funkció, amihez szervesen kapcsolódik, végrehajtódik. A végrehajtást követően persze már a továbbiakban nincs szükség ezekre az adatokra, így akár törölhetők is.

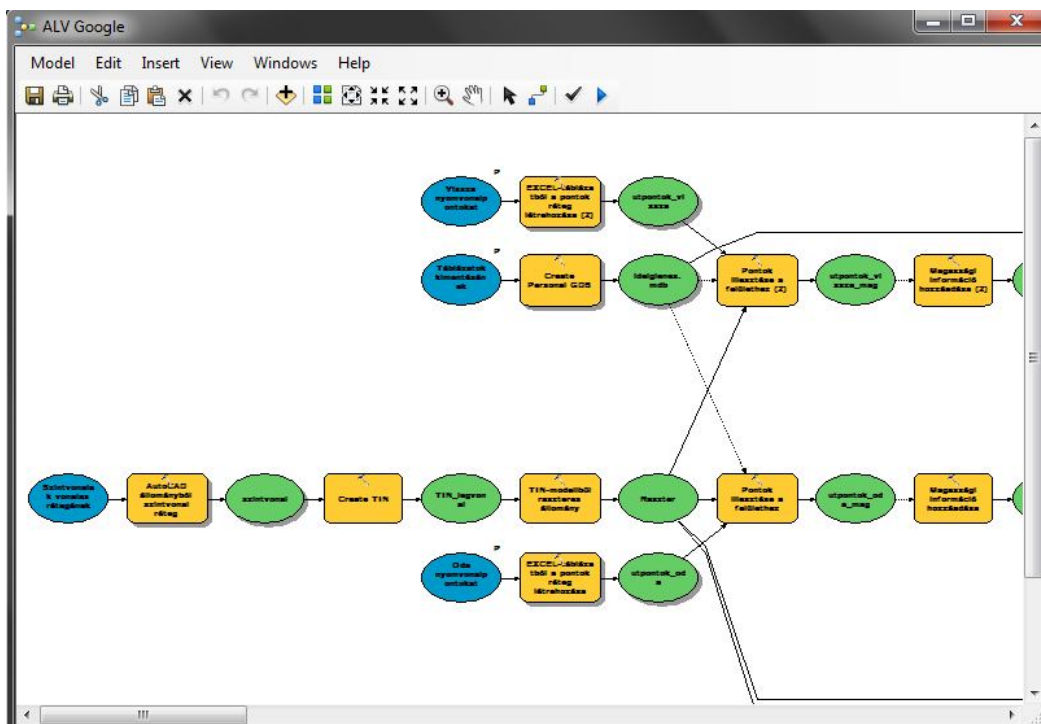
⁸ Az ESRI „Toolbox”-nak nevezi a különböző funkciók végrehajtására irányuló egyes eszköztárakat az ArcGIS programban

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. Fontos tudnunk, hogy melyek azok a funkciók, amiknél valamilyen külső adatforrásból importálunk, valamint melyek azok, amelyek a tényleges végkimenetelt tartalmazzák. Az ilyen kritikus utak megjelölését szolgálja a paraméterezés. Egy paraméter sok minden lehet, azonban beállításukat gondosan kell elvégezni, hogy minden megfeleljen az ArcGIS követelményeinek.

5.2 A légvonalas vizsgálat folyamatábrája

Az előző alfejezetben említettem, hogy a légvonalas vizsgálat folyamatábrája hozzávetőleg 50 összekapcsolódó műveletet tartalmaz. E folyamatábrát a *2. számú melléklet* tartalmazza. A műveletek között vannak ismétlődések is, így feleslegesnek tartom lépésről lépésre leírni a folyamatábra elemeit. A jobb megértés érdekében azonban a fontosabb részeket célszerű elemezni, hogy a vizsgálat logikája megfelelően követhető legyen.



5-1. ábra Folyamatábra felépítése a Model Builderben [a szerző munkája]

5.2.1 Adatok beolvasása

Az adatokat különböző formátumuk révén más és más módon kell beolvasni a programba. Egy Excel-táblázat beolvasására a legmegfelelőbb megoldás a „Make XY Event Layer” funkció. A funkció lényege, hogy megadjuk, melyik az a táblázatos adat, 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 az útvonalbeli adata. A beolvasásnál meg kell adnunk, hogy 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 koordináta-rendszer van, így az x és y koordinátákat felcserélve kell beolvasatni, máskülönben egy 90°-kal elforgatott nyomvonalat kapunk.

Az AutoCAD-állomány beolvasása már sokkal egyszerűbb, hiszen az ArcGIS képes kezelni a *.dwg és a *.dxf kiterjesztésű fájlokat. A Model Builderben egy szintvonalas adat importálását így megtehetjük a „Feature To Line” modul segítségével. Mind a pontok, mind pedig a vonalak beolvasásánál lehetőség van még egy további beállításra, melyre a későbbiekben néhány funkciónak szüksége lehet. Ez pedig a koordináta-rendszer meghatározása. Az ArcGIS rengeteg koordináta-rendszert ismer, így többek között a Magyarországon alkalmazott EOVT és EOMA⁹-t is.

5.2.2 A domborzatmodelltől a látóvonalakig

A beolvasást követően rendelkezéseimre 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 domborzatmodell. Egy 3D adatokkal dolgozó modell a már említett „3D Analyst Tool” segítségével állítható elő [ESRI, 2013b]. Itt találunk egy „Create TIN” funkciót, ami a TIN-modell létrehozását támogatja. A funkció beállítási lehetőségei között itt is szerepel a koordináta-rendszer meghatározása, továbbá egy kiegészítő lehetőséggel is élhetünk, mellyel a létrehozott háromszögek előnyösebben illeszkednek a terep alakulatához.

⁹ Egységes Országos Magassági Alappont Hálózat

A láthatósági vonalak létrehozásához nem egyszerűen csak egy terepet követelünk meg, hanem azt, hogy ez a terep egy raszter legyen. Egy raszter előállításához a kialakított TIN-modellt át kell konvertálnunk egy raszteres felületté. A raszter konverzió során fontos, hogy megfelelően járjunk el, mert a program automatikusan felajánl nekünk egy cellaméretet, ami alapján kialakításra kerül a raszterkép. 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 a nagyságát. Egy raszterképnél a modell magassági információi a raszter celláiban kerülnek eltárolásra, ezáltal a folyamat végeredményeként egy GeoTiff állományt kapunk. A GeoTiff egy olyan képformátum, melyben a pixelek már rendelkeznek vízszintes és magassági koordinátákkal.

A GeoTiff megléte után már illeszthetjük az út nyomvonalpontjait. Ha a pontok magassági adatai is rendelkezésünkre állnak, akkor megnézhetjük közöttük a légvonalakat. Ezeket a légvonalakat tekintjük egyszerűen látóvonalaknak, hiszen ezeknél még nincsenek meghatározva, hogy metszik-e a terepet vagy sem. A látóvonalak létrehozásáért a „Construct Sight Lines” funkció a felelős. [ESRI, 2013c] Megnyitásakor meg kell határozni, mik azok a pontok, vonalak, vagy felületek, amelyeket meg szeretnénk nézni, hogy látszódnak-e. 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. 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. 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 javítást több lépésben végeztem el, mert egyetlen funkció általában csak egyetlen műveletet hajt végre. A javítás lépései a következők voltak:

- 0 hosszúságú vonalak törlése;

- 600 m-nél rövidebb vonalak leválogatása;
- a redundancia csökkentése.

Az elvégzett javításoknak köszönhetően az adatok mennyisége lecsökkent, így a felesleges adatokkal már nem kellett foglalkoznom. A látóvonalak létrehozása során a program alapértelmezetten ad a vonalaknak attribútum adatokat. Mivel a vonalak attribútumai és a pontok attribútumai nem egyeznek, nehéz eldönteni szemrevételezés nélkül, hogy egy vonal mely pontból indul és melyikben végződik. Az eldöntést és a későbbi elemzést segítő, fontos volt, hogy összekapcsoljam az eltérő rétegek adattábláit. Kapcsolatot többféle módon is lehetőségünk van kialakítani, azonban mindegyik 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;

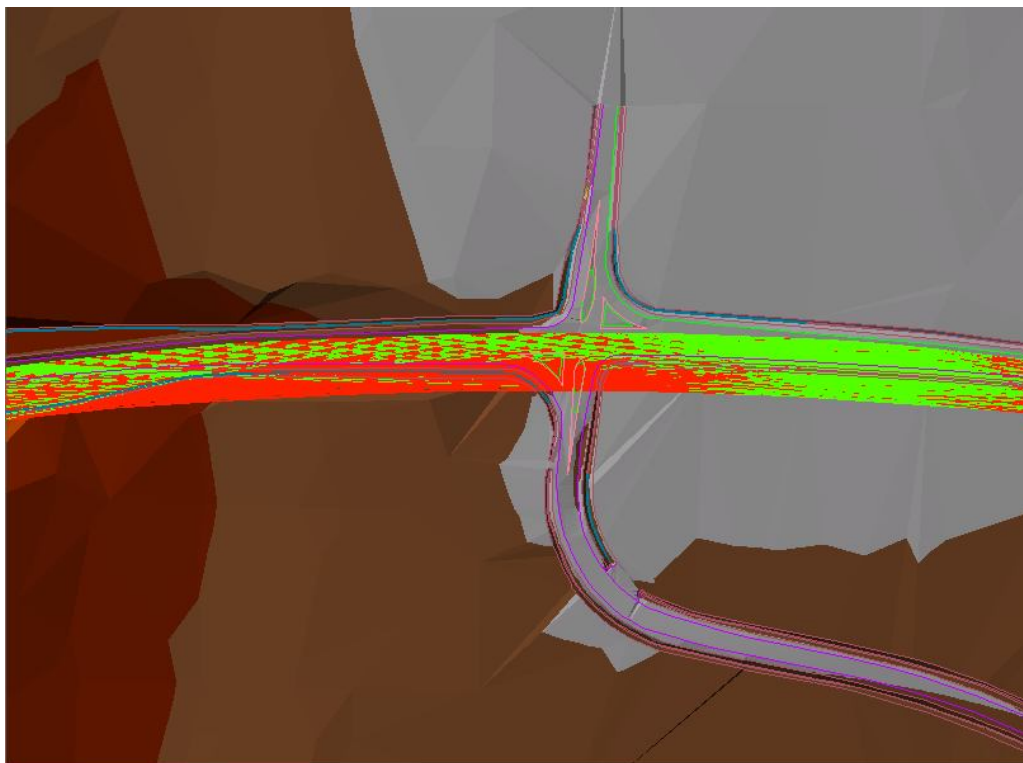
relációk alapján. Hogy egy „Special Join” típusú műveletet végrehajtsunk, a látóvonalakból elő kell állítani egy, vagy több pont típusú objektumot. Erre szintén találunk lehetséges megoldást, mellyel ki tudjuk külön rétegbe íratni a vonalak kezdő-, és végpontját. A kezdő-, és végpontokat külön rétegekben 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 nyomvonalpontnak 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” segítségével olyan táblázatokat tudunk összeilleszteni, melyekben van legalább egy közös oszlop, mint a látóvonalak és a látóvonalakból számolt pontok esetén az egyes vonalak azonosítói. Ha már megvannak a látóvonalak attribútum

tábláiban a kezdő-, és végszelvény számok, akkor következhet a láthatósági vonalak képzése.

5.2.3 Láthatósági vonalak

A látóvonalakból láthatósági vonalak előállításához szükségünk van egy raszter felületre. A raszter felület megléte esetén csak annyi a dolgunk, hogy meg kell adni a programnak, melyek a bemenő adatok. Jelen esetben a látóvonalak, valamint a felület képi adatai. 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. Például 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 ennek alapján kerül osztályzásra. Ez az osztályzást azt jelenti, hogy a „Line of Sight” funkció a bemenő vonalakat vagy egészben hagyja, vagy több egymáshoz kapcsolódó darabra töri, attól függően, hogy a teljes felülete látszik, vagy sem. Ha a két pont összelátható teljes egészében, akkor az eredmény egy teljes hosszban zöld vonal lesz, 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

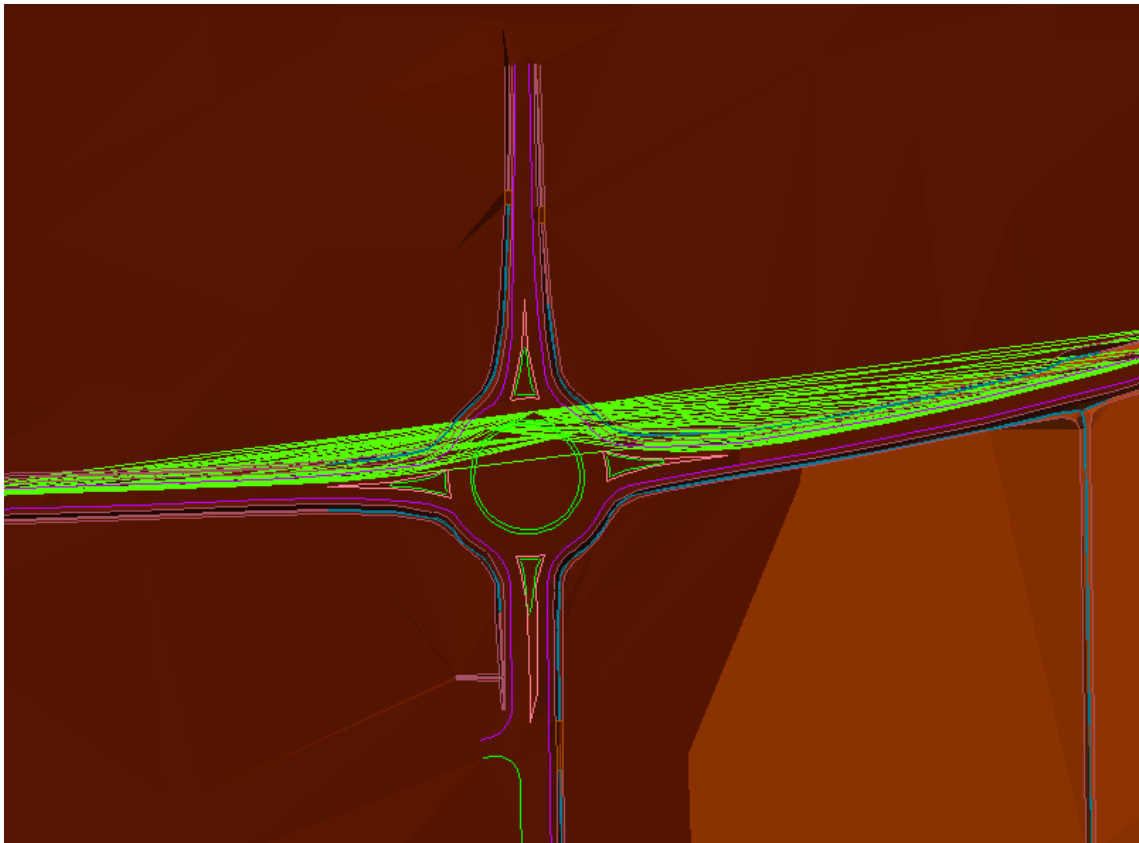


5-2. ábra Láthatóság alapján tördelt látóvonalak [a szerző munkája]

sorolja. A 0 osztály az adott pontról nem látszik, és piros színezést kap, az 1 pedig a látható, ezáltal zöld színt vesz fel.

A kérdéses adatok végkifejletéig azonban még mindig nem jutottunk el, hiszen azok a vonalak, melyek több részből állnak, a további elemzésekhez nem szükségesek. A vonalakat éppen ezért valamilyen rendszer szerint összegeznem kellett. Erre megfelelőnek bizonyult a „Frequency” eszköz. Működésének lényeg, hogy képes valamilyen oszlop szerint csoportosítani az adattáblában lévő adatokat, és ebből a csoportosításból már leválogathatók a ténylegesen keresett értékek.

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



5-3. ábra Ténylegesen látható vonalak [*a szerző munkája*]

6 Láthatósági vizsgálat az útpálya mentén

A légvonalas láthatósági vizsgálat inkább olyan célokat szolgál, hogy megnézzük, mennyire takarnak be a domborzati elemek egy útszakasz különböző pontjainak összelátásába. Ezzel szemben az út nyomvonala mentén haladó vizsgálat azt hívatott elemezni, hogy a domborzati adatok mennyire alkalmazkodnak az úthálózathoz. Tehát egy Google domborzat mennyire alkalmazható arra, hogy az útvonal beláthatóságát illetően következtetéseket vonjunk le. A leghelyesebb következtetések levonásáért azonban nem csak szimplán egy Google domborzat került górcső alá, hanem annak javított, az útvonal pontos adataival kiegészített megoldása is.

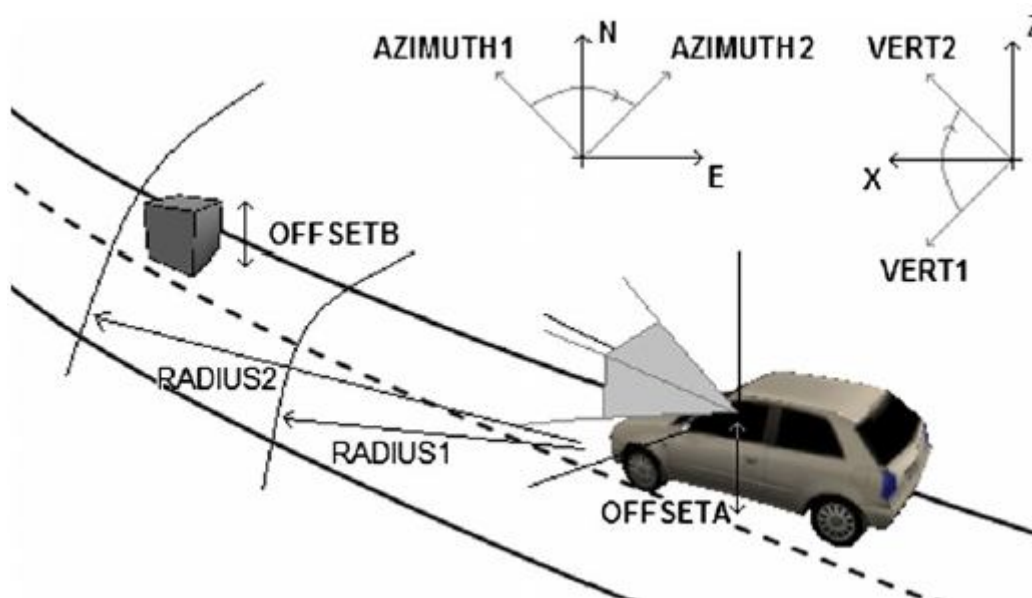
Az útvonal menti láthatósági vizsgálat a légvonalassal szemben annyiban jobb a domborzati adatok értékelésére, hogy konkrétan a haladás szerint vagyunk képesek értékelni a megállási látótávolságot.

6.1 Bevezető lépések

A legelső lépés itt is az adatok beolvasása, mely teljesen ugyanazon elvek szerint történik, mint a korábban leírtak, azonban ezt követően a folyamatára persze megváltozik. Az útvonal menti vizsgálat folyamatábráját a *3. számú melléklet* tartalmazza. A további lépésekben az adatok elsősorban nem leválogatásra kerülnek, hanem az adattáblákat egészítem ki újabb oszlopokkal, hogy a későbbi funkciók korrekt módon fussanak le. Az ArcGIS „3D Analyst Tool” tartalmaz egy láthatósági ábrakészítő megoldást, ez a „Viewshed” [ESRI, 2013c]. Lényege, hogy egy-egy kiválasztott pontban a program a terephez viszonyítva megnézi mi látható, és mi nem az. A láthatóság végeredményét az alkalmazás egy raster 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 a programmal végig tudjunk futtatni, nem kell mást megadnunk, mint a terepmodellt tartalmazó raster állományt, valamint egy pontot, amelyre számoljuk az elemzést. Ha azonban különböző kikötésekkel kívánunk élni, melyeket a későbbiekben állítani is szeretnénk, akkor a beolvasni kívánt pont adattáblájába többféle új oszlopot írhatunk bele, melyek a következő adatokat tartalmazzák [ESRI, 2013e; Castro et al., 2009]:

- SPOT: a magassági adat definiálása;

- OFFSETA: a kiinduló pont látómagassága, jelen esetben a sofőr szemmagasság, mely alapértelmezésként 1 m;
- OFFSETB: a szemlélési végpont magassági, itt az útfelületen található 0,1 m magas tereptárgy, vagy akadály;
- AZIMUTH1: A kezdő irányszög az óra számlapját nézve az óramutatónak megfelelő haladási irányban;
- AZIMUTH2: A végső irányszög értéke;
- VERT1: a horizonthoz képesti magassági szög felfelé;
- VERT2: a horizonthoz képesti magassági szög lefelé;
- RADIUS1: a belső kör sugara;
- RADIUS2: a külső kör sugara, itt a vizsgált látótávolság értéke.



6-1. ábra Beállítási lehetőségek értékeinek szemléletes megjelenítése [Castro et al., 2009]

Alapértelmezésben a program a következő táblázat második sorában található értékek szerint számol, melyek a vizsgálatomnál a 3. sor szerint alakulnak:

6-1. táblázat Viewshed alapértelmezett és a vizsgálat során használt beállításai [a szerző munkája]

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

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 körcikk kerül megvizsgálásra.

Az előbb látott adatoszlopok meghatározását követően természetesen valamilyen módon ki kellett számoltatnom a programmal az egyes pontokból kiinduló takarási ábrákat. 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 kerülnek létrehozásra. Két pont esetén a 0 és 1 értékeken kívül találni fogunk az adatsorban egy 2-t is, mely a két pontból közösen látható területet jelenti. Éppen ezen okokból kifolyólag nem célravezető egy láthatósági vizsgálat esetén több pont bevitele egyszerre. A műveletet azonban mégis meg lehet oldani, ekkor természetesen a futási idő exponenciálisan nőni fog. A több pontra való takarási ábra számolásához már szükség van egy kis programozási tudásra.

6.2 Python programozás

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

6.2.1 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.

Az előző fejezettrészben említett takarási ábra létrehozásánál azért volt szükség python programozásra, mert egyfajta ciklikusságot kellett megoldanom. A „Viewshed” bár képes egyszerre több pontra láthatóságot számolni ez mégsem célorientált számunkra. A ciklikus futás megoldható a többi programnyelvben is megtalálható „for”

utasítással. Egy ciklus futása mindösszesen néhány percig tart, ez azonban akár több órát is jelenthet egy több km hosszú szakasz esetén. Mivel nem csak egy műveletet tartalmaz, ezért ez teljesen reális. Jelen esetben az ismétlődés körülbelül 25 egymás utáni funkció folyamatoként valósul meg.

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

Ha készen vagyunk a programozással az ArcGIS-be azt valahogyan vissza kell olvasnunk. Az ArcCatalog programrészben 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 kell megadnunk, hogy hol található az előzőleg megírt *.py kiterjesztésű alkalmazás. Mind a ki-, mind pedig a bemenet esetében különböző típusú adatok használhatók, a könyvtárak helyétől kezdve a vektoros adatokon keresztül egészen a raszterekig bezárólag.

A vizsgálat során alkalmazott ciklust a továbbiakban részletesebben kívánom ismertetni.

6.2.2 FOR ciklus a vizsgálatnál

A ciklus első lépésében lényeges, hogy valamilyen argumentum alapján kiválasztásra kerüljön először az első, majd – a benne foglalt utasítássorozat teljesülését követően – az azt követő pontok. Ez az attribútum a pontszám.

Az első lépés tehát egy kiválasztó funkció, ahol minden egyes esetben az útpontok rétegből leválasztásra kerül az aktuális pont. A program ez alapján egészen addig ismétli a cikluson belüli folyamatokat, amíg van az adatbázisban adatsor.

A keresett pont leválogatását követően már csak ezzel az eggyel dolgozik tovább a folyamat többi művelete. Miután tehát a leválogatás megtörtént meg kellett határozni a programnak egy 40° szélességű 600 m sugarú meghatározott irányba mutató körcíkkel, mely később a raszteres adatok vágását segítette elő. Ilyen művelet nem létezik az alap konstrukcióban, azonban az interneten található egy kiegészítő, mely

hasznos elven végez szintén láthatósági vizsgálatokat. Az internetes kiegészítő esetében a programozók írtak egy olyan python programot, mely az említett területet képes meghatározni, így ez egyfajta könnyebbséget jelent egy saját python program írásával szemben. Természetesen ez a programozási fájl is beolvasható az általunk írt adatok mellé, így könnyedén tudjuk alkalmazni a ciklus során.

A műveletsorozat következő lépése már lehet a korábban említett takarási ábra készítése, immáron egyetlen pontra vonatkoztatva. A takarási ábra esetén, bár megadtuk előre a pontokból számolandó irányszög és távolság értékeket, 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, saját értékek helyett, akkor a láthatósági ábra a teljes területre kiértékelésre kerül, tehát az egész domborzatról képet kapunk. Saját adatok használata során a látható részeket csak arra a szűk sávra határozza meg, melyet a bevitt értékekkel lehatároltunk, a felület többi részét a pontból nem láthatónak tekinti. A saját értékek a *6.1 táblázat* tartalmazza.

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. Egy raszter vágását legegyszerűbben egy maszkolási művelettel oldhatjuk meg. Segítségével egy bizonyos területre vonatkozó raszter adatok nyerhetők ki az eredeti képárból. E maszkolási művelet miatt volt szükség a felvázolt internetes kiegészítőre.

A kivágási művelet végrehajtásával persze a továbbiakban is csak képi információink vannak, melyekből vektoros adatokat kell létrehoznunk, annak érdekében, hogy meghatározzuk mi az a távolság, ameddig egy-egy megfigyelőállásból ellátunk. Korábban volt már szó vektor raszter konverzióról, itt viszont a fordítottjára van szükségünk. A két átalakítás hasonló elven működik, csak, míg az előbbinél pontokról, vonalokról és felületekről térünk át képi információra, addig az utóbbinál egy meglévő képet szeretnénk felületekké alakítani. Egy képi adatból egy vektoros létrehozása során a program minden egyes képpontot megvizsgál, és az összetartozókat egy felületté alakítja.

Az átalakítás után már rendelkezésre áll egy vektoros adathalmaz a program által számolt – az adott pontból látható – minden egyes részletről. A létrehozott állomány 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 a pontszám képezte. Mihelyst megvan mindkét adattáblában a közös pontszám 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 valamilyen módon 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 számolt adatok 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.

6.2.3 Láthatósági animáció

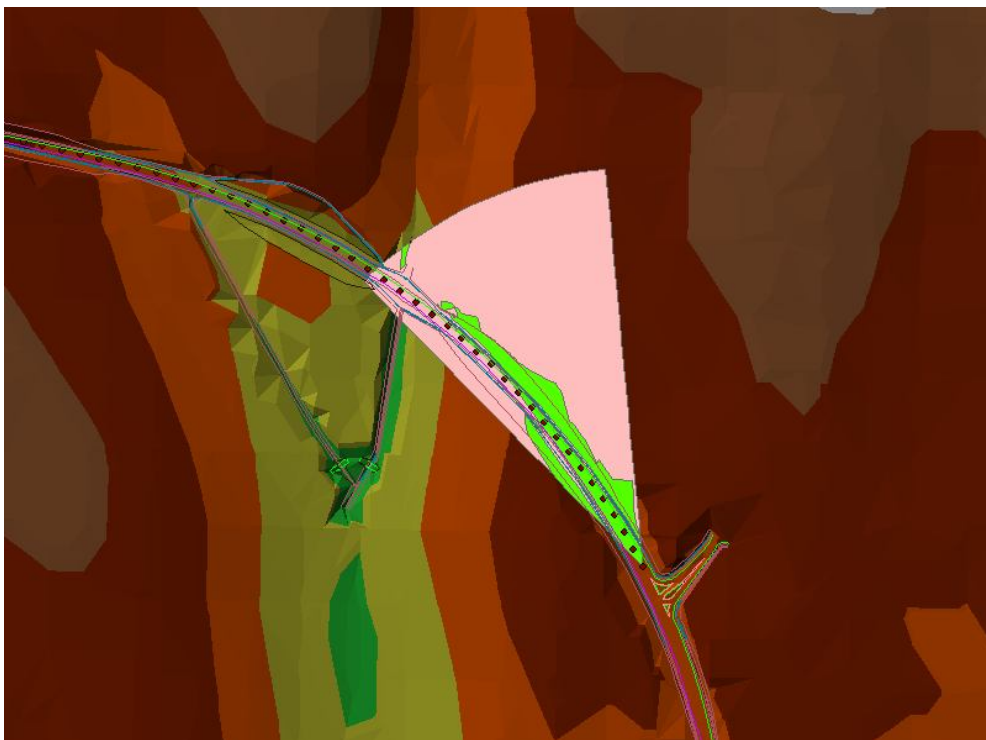
Bár a láthatósági animáció létrehozása is a „For” ciklus részét képezi, mégis úgy találtam, hogy külön alfejezetet érdemel. Az ArcGIS képes különböző animációk létrehozására, mellyel egyes folyamatok sokkal szemléletesebben mutathatók be. Sok online videó lelhető fel az animációkkal támogatott demonstrálásokról, melyek az esetek többségében a jobb megértést segítik.

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 kitöltésére legegyszerűbb esetben a „Now”, azaz „Most” függvényt alkalmazhatjuk. A ciklus lefutása során ilyenkor az oszlop a létrehozás pillanatával lesz kitöltve. Bár elegendő lenne az időpontnak a megadása, a program mégis megköveteli egy másik oszlop a „Track ID” jelenlétét. Ez igazából csak egy másodlagos azonosításra szolgál.

A két oszlop megléte után a ciklus már csak törli a feleslegessé vált adatokat és kezdődik előről a következő ponttal. Miután a ciklus az utolsó pontra is lefutott, az

egyes számolt láthatóságokat 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 funkció. Segítségével több, különálló rétegben található objektumot tudunk eggyé egyesíteni. Elő követelményként kell egy minta réteget gyártanunk. Fontos, hogy a minta meghatározása során a réteg azon adatszlopokat tartalmazza, melyekre szükségünk van a további számoláshoz. A lefutásához 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” művelet szolgál. Itt kell beállítani az időpontot tartalmazó oszlop nevét és az időpont formátumát, valamint a követés egymásutániságáért felelős Track ID”-t. A program a művelet során létrehoz számunkra egy *.lyr nevű fájlt, mely az ArcGIS által támogatott külön kiterjesztés. Ebben az *.shp fájllal (az ESRI saját formátuma) ellentétben különböző mozgatási műveleteket is megjeleníthetünk. Az animáció megalkotásával szemléletesebben értékelhetjük az út egyes pontjaiban a látási viszonyokat. Az animációra példát a 6-2. ábra mutat.



6-2. ábra Az útvonal mentén futó láthatósági vizsgálat végeredménye [a szerző munkája]

7 Excel vizsgálat

Az előző vizsgálatok során számolt értékeket egy *.dbf formátumban ki tudjuk menteni. Ez Excelbe 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 szelvénszáma;
- Végpont szelvénszáma.

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 következő ábrán látható:

7-1. táblázat Példa a négyzetes mátrix végeredményére [a szerző munkája]

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

Az ábra mind az oda, mind pedig a visszalátás viszonyait tartalmazza. Ha az ábra szerint oda irányban haladunk és jelenleg az 55620 szelvénszámú ponton tartózkodunk, akkor előre látunk egészen az 55800-ig. Azaz ebben a pontban a látótávolság értéke 180 m. Visszafelé irányban hasonló a helyzet, azonban, ha azt

vesszük, hogy ismételtén az 55620-as pontban vagyunk, akkor a táblázat szerint csak önmagunkat látjuk. Ez azért lehetséges, mert ennél a vizsgálatnál ez a legelső pont. Ha megnézzük az 55760-as szelvényszámot, akkor látható, hogy ebből egészen 55620-ig előrelátunk. Tételezzük fel, hogy önmagunkat minden esetben látjuk, ezek a pontok azok, melyeknél megegyezik a kezdő-, és végpont szelvényszáma, 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, ami a bal oldalán, az pedig a vissza irányról.

A Microsoft Excel bár temérdek előre megírt függvénnyel rendelkezik, melyeket az egyszerű felhasználók egy kis olvasást követően képesek elsajátítani, azonban ezek alkalmazási tartományainak is megvannak a határai. Ha például szeretnénk kikeresni egy a táblázatunkban lévő információt, 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 keresés folyamán, 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 találatot fogja megadni.

Az ArcGIS-hez hasonlóan az Excel-hez is tartozik egy fejlesztői környezet. A korábban említett python programozástól eltérően itt VBA¹⁰ nyelv szerint kell megírni az egyes utasításokat. A programozás egy „Makróban” tárolható, mely tulajdonképpen egymás után 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 makrórögzítő segítségével az Excel maga végzi el a

¹⁰ Visual Basic for Applications

kódolást. A másik lehetőség során az Excelben található Visual Basic Editort használjuk. Itt már a Makró elejétől saját magunk határozzuk meg, hogy mit tartalmazzon a folyamat. [Aitken, 2001] 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 dolgozat elkészítése során én a második lehetőséggel éltem.

Kettős kereső makró

1) Ismeretlenek meghatározása

Az első sor egy „Sub” kifejezéssel kezdődik a VBA Makró esetén. A kifejezés után kell meghatároznunk a makró nevét. A következő sorokba írhatók a programra jellemző információk (készítő neve, készítés ideje, stb.), ezek beírásához, azért, hogy a futásban ne vegyenek részt egy felsőindexben található vesszőt kell a sorok elé írni.

Minden programnyelv esetében valamilyen módon definiálnunk kell a használt ismeretleneket. Ez VBA-nál a „Dim” szóval oldható meg, ezt követően írjuk be a megjelölt ismeretlent, valamint, hogy milyen értékeket vehet fel. Az értékek többfélék lehetnek, attól függően, hogy szöveges vagy szám formátumú eredményre törekszünk. Segítséget az egyes ismeretlenek megfelelő besorolásához a *Makro Mester* nevű internetes oldal nyújt.

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 darabszáma. 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. A cellák bekérésénél 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 értéket. Ez így azonban nem jó, hiszen ez egy eredménylap lesz, tehát a pontos munkafüzet meghívásához a „Worksheets” szót kell használnunk. Ez után zárójelben, idézőjelek közé írva meghatározhatjuk a pontos munkalapot.

2) Többszörös ciklusfüggvény

Azért, hogy az ArcGIS-ből exportált összehasonlíthassam, elő kellett állítanom a korábban említett négyzetes mátrixot. Ehhez természetesen szükséges volt tudnom, hogy a kinyert több ezer adat mindegyike mely ponthoz tartozik. A kettős keresés ezt a problémát hivatott megoldani.

Ahhoz, hogy kinyerjük a számunkra 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 azonban nem egyetlen ciklusugrást 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 tesztelős ciklus, melynél a „Do” utasítás után meg kell határozni egy feltételt. Hátul tesztelős ciklus lévén a folyamat egészen addig ismétlődik, amíg a megadott feltétel igazzá nem válik.

A „For” ciklusnál „a” és „b” változók az eredménymátrix sorát és oszlopát jelölik. E sor és oszlop határozza meg, hogy melyek azok a szelvénytérképek, amiknek kíváncsiak vagyunk a meglétére. A „Do Loop” ciklus esetén az „i” és „j” változók a meglevő 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 végigvezetem az elmondottakat.

Kezdetnek vegyünk egy oda-, és vissza irányban 3000-3000 rekordot tartalmazó adatbázist és egy 56000-es és 58000-es szelvénytérképek 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 pont. Az ArcGIS a számolási részt már elvégezte, tehát 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 az első futásnál mindig a kezdő szelvénytérkép sora, itt most 3. Ennek ismeretében kerül behelyettesítésre „b” értéke is, mely a végszelvény oszlopa. Megjegyezném, hogy az adatbázisban nem található olyan érték, melynél a kezdő-, és végszelvény egyezik,

azonban tudvalevő, hogy saját magunkat látjuk. Ha tehát a kezdő és végpont szelvény száma egyezik, az eredmény mátrix cellája 1-et vesz fel.

Tehát „a” és „b” értékek alapján a program kiválasztja a megfelelő sorban és oszlopban található szelvény számot, ez a KP (kezdőpont) és VP (végpont) érték. Ezután következik egy „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. Ezért 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, mely az oda látótávolságokat tartalmazza. LK1 és LV1 értéke természetesen a folyamat előrehaladásával folyamatosan változik, ismételten addig, míg az egyik állítás igaz nem lesz. Ha igaz lesz, akkor LK1 és LV1 értéke újra az első látótávolságtól indul.

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 a vissza látótávolságokra vonatkoznak. Az alapok tisztázását követően tehát elindulhatunk a program működésének folyamatán.

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, valamilyen módon a ciklusokat is le kell zárunk. Erre szolgál a „Do Loop” esetén a Loop parancs, mely előtt meg kell határoznunk, hogy 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 programozás teljes terjedelmében az *5. számú mellékletben* található.

8 Eredmények

Az Excelben mindegy egyes vizsgálatfajta esetében végeredményként előállítottam a négyzetes mátrixot. A különböző mátrixok összehasonlításával megállapítható, hogy az 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:

- Az ingyenes domborzatmodell műholdas felvételek sztereoszkópikus kiértékelésével kerül előállításra, erre vetítik rá a korrigált műholdképeket. A folyamat végeredményeként egy közelítőleg helyes domborzatot kapnak, melyen az út felület csak egy rávetített kép, ezáltal akár kisebb, nagyobb bukkánók is lehetnek rajta. Ezek természetesen befolyásolják a látótávolság kiértékelését, ezért a Google adatsor pontosítása mindenképp szükséges.
- A javított adatokkal való munka esetén, valószínű lesz, hogy a légvonal látótávolsági számolás végeredményei lényegesen kevesebb értéket fognak tartalmazni, mint az eredeti modellnél. Ez arra vezethető vissza, hogy a magassági adatok nem illeszthetők tökéletesen, így az utaknál bevágás esetén a különböző pontok nem lesznek összeláthatók. Természetesen a légvonalas vizsgálat pontatlanabb, mint az útvonal mentén történő.
- Az előző pontban említett magassági illesztés jelenti a 3. kritériumot, amelyek a végeredményt befolyásolják. A különböző információkat a modell javításának érdekében egyesíteniünk 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.

Végeredményben, az útnyomvonal mentén történő vizsgálat a javított adatokkal véleményem szerint alkalmazható egy láthatósági értékelésre, ugyanis az előírt megállási látótávolságok az esetek nagy részében teljesülnek.

9 Irodalomjegyzék

- BME (2010): Látótávolságok, előzési szakaszok, kapaszkodósáv, túlemelés, pályaszélesítés, http://www.epito.bme.hu/uvt/oktatas/feltoltesek/BMEEOUVASE1/bscutterv2010_ea04_schg_latotav_kapeltulem.pdf, Előadás, 2013. november 11-i állapot
- ESRI (2013a): What is Model Builder?, <http://resources.arcgis.com/en/help/main/10.1/index.html#//002w00000001000000>, 2013. november 4-i állapot
- ESRI (2013b): What is the ArcGIS 3D Analyst extention?, <http://resources.arcgis.com/en/help/main/10.1/index.html#//00q8000000wv000000>, 2013. november 4-i állapot
- ESRI (2013c): Understanding visibility analysis, http://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Understanding_visibility_analysis/00q8000000p2000000/, 2013. november 4-i állapot
- ESRI (2013d): Visibility and Range Template, <http://blogs.esri.com/esri/arcgis/2011/07/06/visibility-and-range-template-available-for-download/>, 2013. november 5-i állapot
- ESRI (2013e): Using Viewshed and Observer Points for Visibility Analysis, http://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Using_Viewshed_and_Observer_Points_for_visibility_analysis/00q900000008n000000/, 2013. november 5-i állapot
- Horváth Tamás (2011): Nagykutas területének 3D modellezése, Szakdolgozat, Nyugat-magyarországi Egyetem Geoinformatikai Kar
- Khattak, A.J., Shamyaleh, H., 2005. Highway safety assessment through geographic information system-based data visualization. *Journal of Computing in Civil Engineering* 19 (4), p. 407–411
- Koren Csaba (2012): A közúti infrastruktúra biztonsága, Győr, Széchenyi István Egyetem, Főiskolai jegyzet, p. 350
- Koren Csaba (2013): Közlekedésepítés IV. – Közúti Biztonság, Győr, Széchenyi István Egyetem, Előadás
- Központi Statisztikai Hivatal (KSH), 2010: Személyi sérüléssel közúti közlekedési balesetek, <http://www.ksh.hu/docs/hun/xftp/gyor/ser/ser21012.pdf>, letöltés: 2013. november 2.

- Makro Mester (MM) (2013): A VBA nyelvi elemei, <http://www.konyveles-budapest-bpc.hu/excel-makro/leirasok/vba-nyelvi-elemei/valtozok-adattipusok-konstansok>, 2013. november 9-i állapot
- Maria Castro, Luis Iglesias, José A. Sánchez, Luis Ambrosio (2009): Sight distance analysis of highways using GIS tools, Transportation Research Part C: Emerging Technologies, Volume 19, Issue 6, December 2011, p. 997-1005
- Márkus Béla (2006a): Térinformatika I., Budapest, FVM KSZI tankönyv, p. 341
- Márkus Béla (2006b): Térinformatika II., Budapest, FVM KSZI tankönyv, p. 341
- Magyar Útügyi Társaság (MAÚT) (2008): Közutak Tervezése Útügyi Műszaki Előírás, ÚT 2-1.201
- Országos Rendőr-Főkapitányság Rendészeti Főigazgatóság Közlekedésrendészeti Főosztály (ORFK) 2011, A személyi sérüléses közúti közlekedési balesetek alakulása, http://www.baleset-megelozes.eu/pics/baleseti_statiztika_2011_egesz_ev.pdf, letöltés: 2013. november 2.
- Országos Rendőr-Főkapitányság Rendészeti Főigazgatóság Közlekedésrendészeti Főosztály (ORFK) 2012, A személyi sérüléses közúti közlekedési balesetek alakulása, http://www.baleset-megelozes.eu/pics/2012_junius_balstat.pdf, letöltés: 2013. november 2.
- Peter G. Aitken (2001): Programozás Visual Basic 6 nyelven, Kiskapu Kiadó, p. 601
- Szikszai Csaba (2003): Digitális domborzatmodelljeink, <http://www.termeszetvilaga.hu/szamok/tv2003/tv0305/szikszai.html>, 2013. november 2-i állapot

10 Ábrajegyzék

2-1. ábra Kiváltó okok részvétele a balesetekben [Koren, 2013]	6
2-2. ábra Megállási látótávolságok a sebesség és hosszesés függvényében [MAÚT, 2008; BME 2010]	7
3-1. ábra Egy terület GRID-, (bal oldalt) és TIN-modellje (jobb oldalt) [Szikszai, 2003]9	
4-1. ábra A vizsgált útszakasz térképe [Google Earth]	16
4-2. ábra A megoldások variációi [a szerző munkája]	16
5-1. ábra Folyamatábra felépítése a Model Builderben [a szerző munkája]	18
5-2. ábra Láthatóság alapján tördelt látóvonalak [a szerző munkája]	22
5-3. ábra Ténylegesen látható vonalak [a szerző munkája]	23
6-1. ábra Beállítási lehetőségek értékeinek szemléletes megjelenítése [Castro et al., 2009]	25
6-2. ábra Az útvonal mentén futó láthatósági vizsgálat végeredménye [a szerző munkája]	30

11 Táblázatjegyzék

6-1. táblázat Viewshed alapértelmezett és a vizsgálat során használt beállításai [a szerző munkája]	25
7-1. táblázat Példa a négyzetes mátrix végeredményére [a szerző munkája]	31